Network Working Group
Request for Comments: 3313
Category: Informational

W. Marshall, Ed.
AT&T

January 2003

Private Session Initiation Protocol (SIP) Extensions for Media Authorization

Status of this Memo

Copyright Notice

Abstract

This document describes the need for Quality of Service (QoS) and media authorization and defines a Session Initiation Protocol (SIP) extension that can be used to integrate QoS admission control with call signaling and help guard against denial of service attacks. The use of this extension is only applicable in administrative domains, or among federations of administrative domains with previously agreed-upon policies, where both the SIP proxy authorizing the QoS, and the policy control of the underlying network providing the QoS, belong to that administrative domain or federation of domains.

Table of Contents

1. Scope of Applicability

This document defines a SIP extension that can be used to integrate QoS admission control with call signaling and help guard against denial of service attacks. The use of this extension is only applicable in administrative domains, or among federations of administrative domains with previously agreed-upon policies, where both the SIP proxy authorizing the QoS, and the policy control of the underlying network providing the QoS, belong to that administrative domain or federation of domains. Furthermore, the mechanism is generally incompatible with end-to-end encryption of message bodies that describe media sessions.

This is in contrast with general Internet principles, which separate data transport from applications. Thus, the solution described in this document is not applicable to the Internet at large. Despite these limitations, there are sufficiently useful specialized deployments that meet the assumptions described above, and can accept the limitations that result, to warrant informational publication of this mechanism. An example deployment would be a closed network,

which emulates a traditional circuit switched telephone network. This document specifies a private header, facilitating use in these specialized configurations.

2. Conventions Used in this Document

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 RFC 2119 [2].

3. Background and Motivation

Current IP telephony systems assume a perfect world in which there is either an unlimited amount of bandwidth, or network layer Quality of Service (QoS) is provided without any kind of policy control. However, the reality is that end-to-end bandwidth is not unlimited and uncontrolled access to QoS, in general, is unlikely. The primary reason for this is that QoS provides preferential treatment of one flow, at the expense of another. Consequently, it is important to have policy control over whether a given flow should have access to QoS. This will not only enable fairness in general, but can also prevent denial of service attacks.

In this document, we are concerned with providing QoS for media streams established via the Session Initiation Protocol (SIP) [3]. We assume an architecture that integrates call signaling with media authorization, as illustrated in the Figure below. The solid lines (A and B) show interfaces, whereas the dotted line (C) illustrates the QoS enabled media flow:

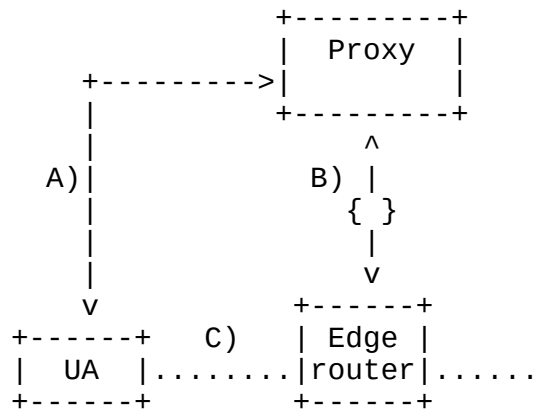

Figure 1 - Basic Architecture

In this architecture, we assume a SIP UA connected to a QoS enabled network with an edge router acting as a Policy Enforcement Point (PEP) [6]. We further assume that a SIP UA that wishes to obtain QoS initiates sessions through a proxy which can interface with the QoS policy control for the data network being used. We will refer to such a proxy as a QoS enabled proxy. We assume that the SIP UA needs to present an authorization token to the network in order to obtain Quality of Service (C). The SIP UA obtains this authorization token via SIP (A) from the QoS enabled proxy by means of an extension SIP header, defined in this document. The proxy, in turn, communicates either directly with the edge router or with a Policy Decision Point (PDP - not shown) [6] in order to obtain a suitable authorization token for the UA.

Examples of access data networks, where such a QoS enabled proxy could be used, include DOCSIS based cable networks and 3rd generation (3G) wireless networks.

4. Overview

A session that needs to obtain QoS for the media streams in accordance with our basic architecture described above goes through the following steps.

The SIP UA sends an INVITE to the QoS enabled proxy, which for each resulting dialog includes one or more media authorization tokens in all unreliable provisional responses (except 100), the first reliable 1xx or 2xx response, and all retransmissions of that reliable response for the dialog. When the UA requests QoS, it includes the media authorization tokens with the resource reservation.

A SIP UA may also receive an INVITE from its QoS enabled proxy which includes one or more media authorization tokens. In that case, when the UA requests QoS, it includes the media authorization tokens with the resource reservation. The resource reservation mechanism is not part of SIP and is not described within the scope of this document.

5. Changes to SIP to Support Media Authorization

This document defines a private SIP header extension to support a media authorization scheme. In this architecture, a QoS enabled SIP proxy supplies the UA with one or more authorization tokens which are to be used in QoS requests. The extension defined allows network QoS resources to be authorized by the QoS enabled SIP proxy.

5.1 SIP Header Extension

A new P-Media-Authorization general header field is defined. The P-Media-Authorization header field contains one or more media authorization tokens which are to be included in subsequent resource reservations for the media flows associated with the session, that is, passed to an independent resource reservation mechanism, which is not specified here. The media authorization tokens are used for authorizing QoS for the media stream(s). The P-Media-Authorization header field is described by the following ABNF [4]:

```
P-Media-Authorization   = "P-Media-Authorization" HCOLON
                           P-Media-Authorization-Token
                           *(COMMA P-Media-Authorization-Token)

P-Media-Authorization-Token = 1*HEXDIG
```

Table 1 below is an extension of tables 2 and 3 in [3] for the new header field defined here. For informational purposes, this table also includes relevant entries for standards track extension methods published at the time this document was published. The INFO, PRACK, UPDATE, and SUBSCRIBE and NOTIFY methods are defined respectively in [11], [9], [12], and [10].

	Where	proxy	ACK	BYE	CAN	INV	OPT	REG
P-Media-Authorization	R	ad	o	-	-	o	-	-
P-Media-Authorization	2xx	ad	-	-	-	o	-	-
P-Media-Authorization	101-199	ad	-	-	-	o	-	-

	Where	proxy	INF	PRA	UPD	SUB	NOT
P-Media-Authorization	R	ad	-	o	o	-	-
P-Media-Authorization	2xx	ad	-	o	o	-	-

Table 1: Summary of header fields.

The P-Media-Authorization header field can be used only in SIP requests or responses that can carry a SIP offer or answer. This naturally keeps the scope of this header field narrow.

5.2 SIP Procedures

This section defines SIP [3] procedures for usage in media authorization compatible systems, from the point of view of the authorizing QoS.

5.2.1 User Agent Client (UAC)

The initial SIP INVITE message, mid-call messages that result in network QoS resource changes, and mid-call changes in call destination should be authorized. These SIP messages are sent through the QoS enabled proxies to receive this authorization. In order to authorize QoS, the QoS enabled SIP proxy MAY need to inspect message bodies that describe the media streams (e.g., SDP). Hence, it is recommended (as may be appropriate within the applicability scope in Section 1 of this document) that such message bodies not be encrypted end-to-end.

The P-Media-Authorization-Token, which is contained in the P-Media-Authorization header, is included for each dialog in all unreliable provisional responses (except 100), the first reliable 1xx or 2xx response, and all retransmissions of that reliable response for the dialog sent by the QoS enabled SIP proxy to the UAC.

The UAC should use all the P-Media-Authorization-Tokens from the most recent request/response that contained the P-Media-Authorization header when requesting QoS for the associated media stream(s). This applies to both initial and subsequent refresh reservation messages (for example, in an RSVP-based reservation system). A reservation function within the UAC should convert each string of hex digits into binary, and utilize each result as a Policy-Element, as defined in RFC 2750 [5] (excluding Length, but including P-Type which is included in each token). These Policy-Elements would typically contain the authorizing entity and credentials, and be used in an RSVP request for media data stream QoS resources.

5.2.2 User Agent Server (UAS)

The User Agent Server receives the P-Media-Authorization-Token in an INVITE (or other) message from the QoS enabled SIP proxy. If the response contains a message body that describes media streams for which the UA desires QoS, it is recommended (as may be appropriate within the applicability scope in Section 1 of this document) that this message body not be encrypted end-to-end.

The UAS should use all the P-Media-Authorization-Tokens from the most recent request/response that contained the P-Media-Authorization header when requesting QoS for the associated media stream(s). This applies both to initial and subsequent refresh reservation messages (for example, in an RSVP-based reservation system). A reservation function within the UAS should convert each string of hex digits into binary, and utilize each result as a Policy-Element, as defined in RFC 2750 [5] (excluding Length, but including P-Type which is included in each token). These Policy-Elements would typically

contain the authorizing entity and credentials, and be used in an RSVP request for media data stream QoS resources.

5.2.3 Originating Proxy (OP)

When the originating QoS enabled proxy (OP) receives an INVITE (or other) message from the UAC, the proxy authenticates the caller, and verifies that the caller is authorized to receive QoS.

In cooperation with an originating Policy Decision Point (PDP-o), the OP obtains and/or generates one or more media authorization tokens. These contain sufficient information for the UAC to get the authorized QoS for the media streams. Each media authorization token is formatted as a Policy-Element, as defined in RFC 2750 [5] (excluding Length, but including P-Type which is included in each token), and then converted to a string of hex digits to form a P-Media-Authorization-Token. The proxy's resource management function may inspect message bodies that describe the media streams (e.g., SDP), in both requests and responses in order to decide what QoS to authorize.

For each dialog that results from the INVITE (or other) message received from the UAC, the originating proxy must add a P-Media-Authorization header with the P-Media-Authorization-Token in all unreliable provisional responses (except 100), the first reliable 1xx or 2xx response, and all retransmissions of that reliable response the proxy sends to the UAC, if that response may result in network QoS changes. A response with an SDP may result in such changes.

5.2.4 Destination Proxy (DP)

The Destination QoS Enabled Proxy (DP) verifies that the called party is authorized to receive QoS.

In cooperation with a terminating Policy Decision Point (PDP-t), the DP obtains and/or generates a media authorization token that contains sufficient information for the UAS to get the authorized QoS for the media streams. The media authorization token is formatted as a Policy-Element, as defined in RFC 2750 [5] (excluding Length, but including P-Type which is included in each token), and then converted to a string of hex digits to form a P-Media-Authorization-Token. The proxy's resource management function may inspect message bodies that describe the media streams (e.g., SDP), in both requests and responses in order to decide what QoS to authorize.

The Destination Proxy must add the P-Media-Authorization header with the P-Media-Authorization-Token in the INVITE (or other) request that it sends to the UAS if that message may result in network QoS changes. A message with an SDP body may result in such changes.

6. Examples

6.1 Requesting Bandwidth via RSVP Messaging

Below we provide an example of how the P-Media-Authorization header field can be used in conjunction with the Resource Reservation Protocol (RSVP) [7]. The example assumes that an offer arrives in the initial INVITE and an answer arrives in a reliable provisional response [9], which contains an SDP description of the media flow.

6.1.1 User Agent Client Side

Figure 2 presents a high-level overview of a basic call flow with media authorization from the viewpoint of the UAC. Some policy interactions have been omitted for brevity.

When a user goes off-hook and dials a telephone number, the UAC collects the dialed digits and sends the initial (1)INVITE message to the originating SIP proxy.

The originating SIP proxy (OP) authenticates the user/UAC and forwards the (2)INVITE message to the proper SIP proxy.

Assuming the call is not forwarded, the terminating end-point sends a (3)18x response to the initial INVITE via OP. Included in this response is an indication of the negotiated bandwidth requirement for the connection (in the form of an SDP description [8]).

When OP receives the (3)18x, it has sufficient information regarding the end-points, bandwidth and characteristics of the media exchange. It initiates a Policy-Setup message to PDP-o, (4)AuthProfile.

The PDP-o stores the authorized media description in its local store, generates an authorization token that points to this description, and returns the authorization token to the OP, (5)AuthToken.

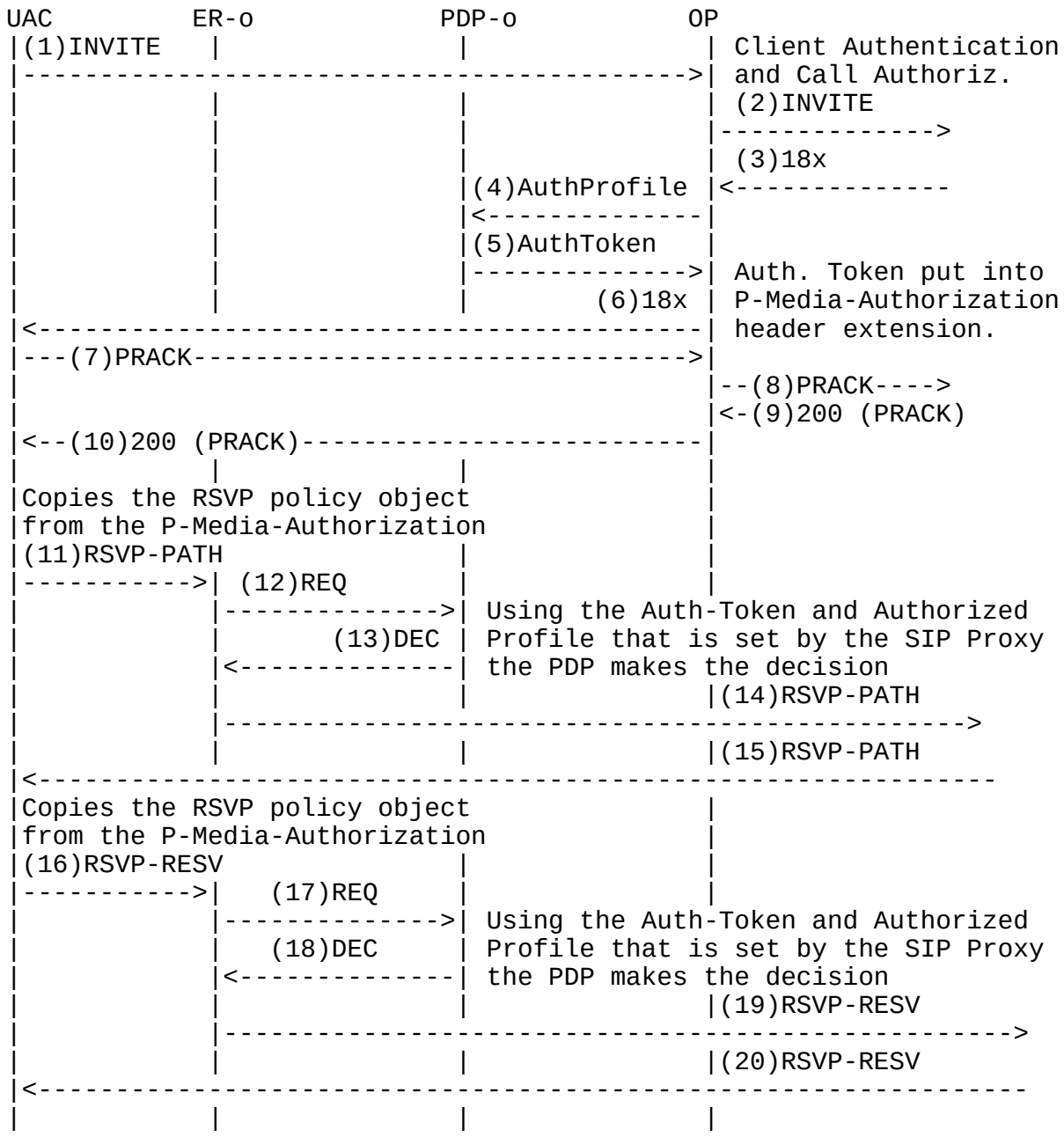

Figure 2 - Media Authorization with RSVP (UAC)

The OP includes the authorization token in the P-Media-Authorization header extension of the (6)18x message.

Upon receipt of the (6)18x message, the UAC stores the media authorization token from the P-Media-Authorization header. Also, the UAC acknowledges the 18x message by sending a (7)PRACK message, which is responded to with (10) 200.

Before sending any media, the UAC requests QoS by sending an (11)RSVP-PATH message, which includes the previously stored P-Media-Authorization-Token as a Policy-Element.

ER-o, upon receipt of the (11)RSVP-PATH message, checks the authorization through a PDP-o COPS message exchange, (12)REQ. PDP-o checks the authorization using the stored authorized media description that was linked to the authorization token it returned to OP. If authorization is successful, PDP-o returns an "install" Decision, (13)DEC.

ER-o checks the admissibility for the request, and if admission succeeds, it forwards the (14)RSVP-PATH message.

Once UAC receives the (15)RSVP-PATH message from UAS, it sends the (16)RSVP-RESV message to reserve the network resources.

ER-o, upon receiving the (16)RSVP-RESV message checks the authorization through a PDP-o COPS message exchange, (17)REQ. PDP-o checks the authorization using the stored authorized media description that was linked to the authorization token it returned to OP. If authorization is successful, PDP-o returns an "install" Decision, (18)DEC.

ER-o checks the admissibility for the request, and if admission succeeds, it forwards the (19)RSVP-RESV message.

Upon receiving the (20)RSVP-RESV message, network resources have been reserved in both directions.

6.1.2 User Agent Server Side

Figure 3 presents a high-level overview of a call flow with media authorization from the viewpoint of the UAS. Some policy interactions have been omitted for brevity.

Since the destination SIP proxy (DP) has sufficient information regarding the endpoints, bandwidth, and characteristics of the media exchange, it initiates a Policy-Setup message to the terminating Policy Decision Point (PDP-t) on receipt of the (1)INVITE.

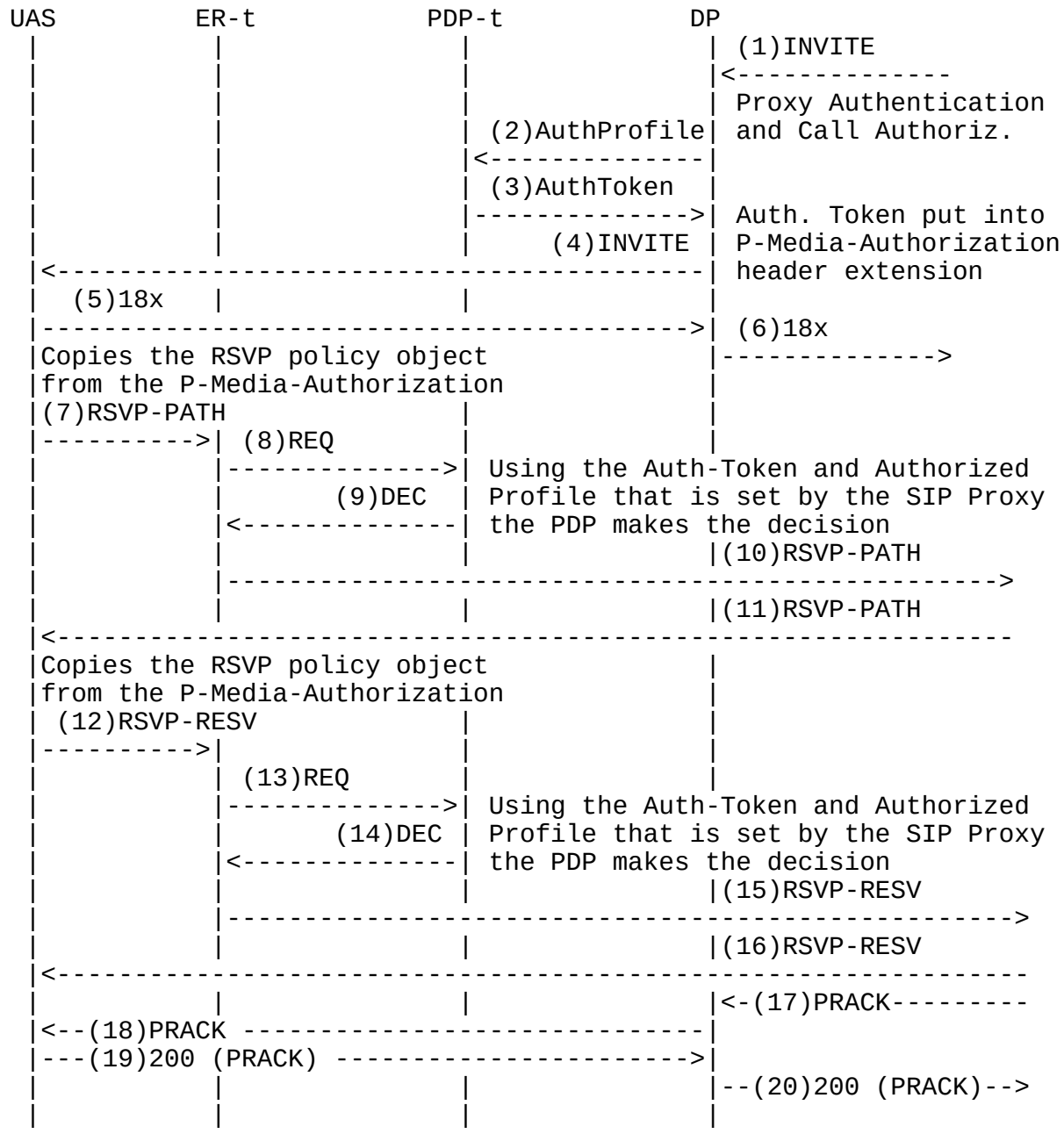

Figure 3 - Media Authorization with RSVP (UAS)

PDP-t stores the authorized media description in its local store, generates an authorization token that points to this description, and returns the authorization token to DP. The token is placed in the (4)INVITE message and forwarded to the UAS.

Assuming that the call is not forwarded, the UAS sends a (5)18x response to the initial INVITE message, which is forwarded back to UAC. At the same time, the UAS sends a (7)RSVP-PATH message which includes the previously stored P-Media-Authorization-Token as a Policy-Element.

ER-t, upon receiving the (7)RSVP-PATH message checks the authorization through a PDP-t COPS message exchange. PDP-t checks the authorization using the stored authorized media description that was linked to the authorization token it returned to DP. If authorization is successful, PDP-t returns an "install" Decision, (9)DEC.

ER-t checks the admissibility for the request, and if admission succeeds, it forwards the (10)RSVP-PATH message.

Once the UAS receives the (11)RSVP-PATH message, it sends the (12)RSVP-RESV message to reserve the network resources.

ER-t, upon reception of the (12)RSVP-RESV message, checks the authorization through a PDP-t COPS message exchange. PDP-t checks the authorization using the stored authorized media description that was linked to the authorization token that it returned to DP. If authorization is successful, PDP-t returns an "install" Decision, (14)DEC.

ER-t checks the admissibility for the request and if admission succeeds, it forwards the (15)RSVP-RESV message.

Upon receiving the (16)RSVP-RESV message, network resources have been reserved in both directions.

For completeness, we show the (17)PRACK message for the (5) 18x response and the resulting (19) 200 response acknowledging the PRACK.

7. Advantages of the Proposed Approach

The use of media authorization makes it possible to control the usage of network resources. In turn, this makes IP Telephony more robust against denial of service attacks and various kinds of service frauds. By using the authorization capability, the number of flows, and the amount of network resources reserved can be controlled, thereby making the IP Telephony system dependable in the presence of scarce resources.

8. Security Considerations

In order to control access to QoS, a QoS enabled proxy should authenticate the UA before providing it with a media authorization token. Both the method and policy associated with such authentication are outside the scope of this document, however it could, for example, be done by using standard SIP authentication mechanisms, as described in [3].

Media authorization tokens sent in the P-Media-Authorization header from a QoS enabled proxy to a UA MUST be protected from eavesdropping and tampering. This can, for example, be done through a mechanism such as IPSec or TLS. However, this will only provide hop-by-hop security. If there is one or more intermediaries (e.g., proxies), between the UA and the QoS enabled proxy, these intermediaries will have access to the P-Media-Authorization header field value, thereby compromising confidentiality and integrity. This will enable both theft-of-service and denial-of-service attacks against the UA. Consequently, the P-Media-Authorization header field MUST NOT be available to any untrusted intermediary in the clear or without integrity protection. There is currently no mechanism defined in SIP that would satisfy these requirements. Until such a mechanism exists, proxies MUST NOT send P-Media-Authorization headers through untrusted intermediaries, which might reveal or modify the contents of this header. (Note that S/MIME-based encryption in SIP is not available to proxy servers, as proxies are not allowed to add message bodies.)

QoS enabled proxies may need to inspect message bodies describing media streams (e.g., SDP). Consequently, such message bodies should not be encrypted. In turn, this will prevent end-to-end confidentiality of the said message bodies, which lowers the overall security possible.

9. IANA Considerations

This document defines a new private SIP header for media authorization, "P-Media-Authorization". This header has been registered by the IANA in the SIP header registry, using the RFC number of this document as its reference.

10. Notice Regarding Intellectual Property Rights

The IETF has been notified of intellectual property rights claimed in regard to some or all of the specification contained in this document. For more information consult the online list of claimed rights.

11. Normative References

[1] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

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

[3] 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.

[4] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

[5] Herzog, S., "RSVP Extensions for Policy Control", RFC 2750, January 2000.

12. Informative References

[6] Yavatkar, R., Pendarakis, D. and R. Guerin, "A Framework for Policy-based Admission Control", RFC 2753, January 2000.

[7] Braden, R., Zhang, L., Berson, S., Herzog, S. and S. Jamin, "Resource Reservation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.

[8] Handley, M. and V. Jacobson, "SDP: Session Description Protocol", RFC 2327, April 1998.

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

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

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

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

13. Contributors

The following people contributed significantly and were co-authors on earlier versions of this document:

Bill Marshall (AT&T), K. K. Ramakrishnan (AT&T), Ed Miller (Terayon), Glenn Russell (CableLabs), Burcak Beser (Juniper Networks), Mike Mannette (3Com), Kurt Steinbrenner (3Com), Dave Oran (Cisco), Flemming Andreasen (Cisco), John Pickens (Com21), Poornima Lalwaney (Nokia), Jon Fellows (Copper Mountain Networks), Doc Evans (Arris), and Keith Kelly (NetSpeak).

14. Acknowledgments

The Distributed Call Signaling work in the PacketCable project is the work of a large number of people, representing many different companies. The contributors would like to recognize and thank the following for their assistance: John Wheeler, Motorola; David Boardman, Daniel Paul, Arris Interactive; Bill Blum, Jay Strater, Jeff Ollis, Clive Holborow, Motorola; Doug Newlin, Guido Schuster, Ikhlaq Sidhu, 3Com; Jiri Matousek, Bay Networks; Farzi Khazai, Nortel; John Chapman, Bill Guckel, Michael Ramalho, Cisco; Chuck Kalmanek, Doug Nortz, John Lawser, James Cheng, Tung-Hai Hsiao, Partho Mishra, AT&T; Telcordia Technologies; and Lucent Cable Communications. Dean Willis and Rohan Mahy provided valuable feedback as well.

15. Editor’s Address

Bill Marshall
AT&T
Florham Park, NJ 07932

EMail: wtm@research.att.com

16. Full Copyright Statement

Acknowledgement

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