

Internet Engineering Task Force
Internet Draft
[draft-rosenberg-sip-3pcc-00.txt](#)
March 10, 2000
Expires: September, 2000

SIP WG
J.Rosenberg, J.Peterson, H.Schulzrinne
dynamicsoft, Level3, Columbia U.

Third Party Call Control in SIP

STATUS OF THIS MEMO

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

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>.

Abstract

This document discusses the usage of SIP for third party call control. Third party call control refers to the ability of one entity to create a call in which communications is actually between other parties. We present a SIP mechanism for accomplishing third party call control that does not require any extensions or changes to SIP.

1 Introduction

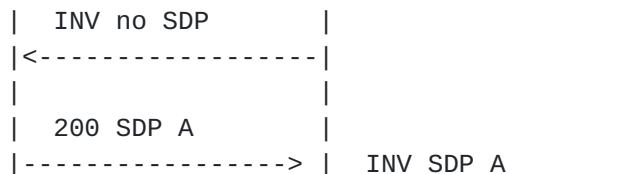
In the traditional telephony context, third party call control refers to the ability of one entity (which we call the controller), to create, modify, or terminate calls between other participants. Third party call control is often used for operator services (where an operator creates a call that connects two participants together), and conferencing.

On the Internet, a wider range of services are enabled through a third party session control mechanism. This is because other IP applications, such as web, email, presence, instant messaging, and chat can now be brought into the picture. An excellent example is click-to-dial. This service allows a user to click on a web page when they wish to speak to a customer service representative. The web server then creates a call between the user and a customer service representative. The call can be between two phones, a phone and an IP host, or two IP hosts.

In order to support third party call control applications, a mechanism is needed that allows a controller to create, modify, and terminate calls with other entities. In this document, we present a mechanism using SIP which allows a controller to execute third party services. The mechanism is not an extension to SIP. It is merely an application of the tools enabled through [RFC 2543](#). A controller can create calls between any entity that contains a normal SIP user agent. After describing the mechanism, we present three third party services which take advantage of this mechanism. One is click-to-dial, the second is a feature that enables a mid-call announcement, and the third is a timed conference bridge initiation.

[2](#) Third Party Control

The basic idea behind the third party mechanism is simple. Consider first the case of just connecting two users in a call. The controller first sends an INVITE to the first user whose phone is to ring. This is a standard INVITE, but it contains no SDP.[\[1\]](#) send an ACK. It generates a second INVITE. This INVITE is addressed to the second user to be connected in the call. This INVITE contains the SDP as received from the 200 OK of the first user. When the 200 OK to this second INVITE arrives, the controller ACK s it, takes the SDP, and includes that in the ACK for the first call. A flow diagram for this mechanism is given in Figure 1.



[\[1\]](#) [RFC 2543](#) does allow for the initial INVITE to not contain a session description

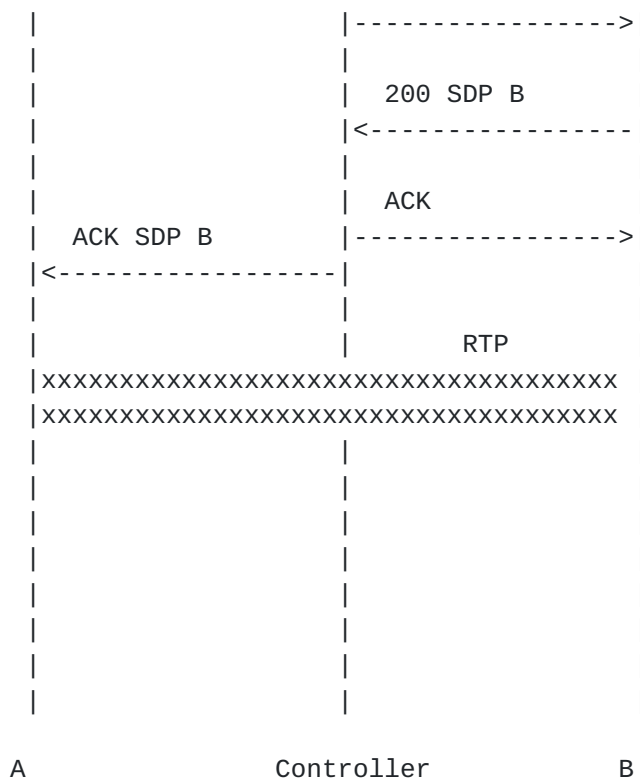
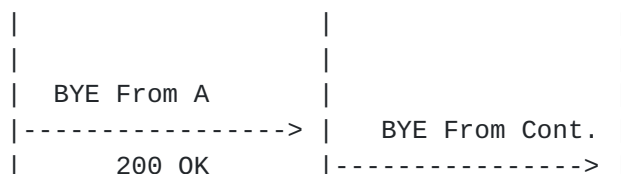


Figure 1

At this point, both participants believe they are in a single point to point call with some control system (assuming the controller identified itself as such in the From field of the INVITE). However, they are exchanging media directly with each other, rather than with the controller. The result is that the controller has set up a call between both participants.

Since the controller is still a central point for signaling, it now has complete control over the call. If it receives a BYE from one of the participants, it can create a new BYE and hang up with the other participant. This is shown in Figure 2.



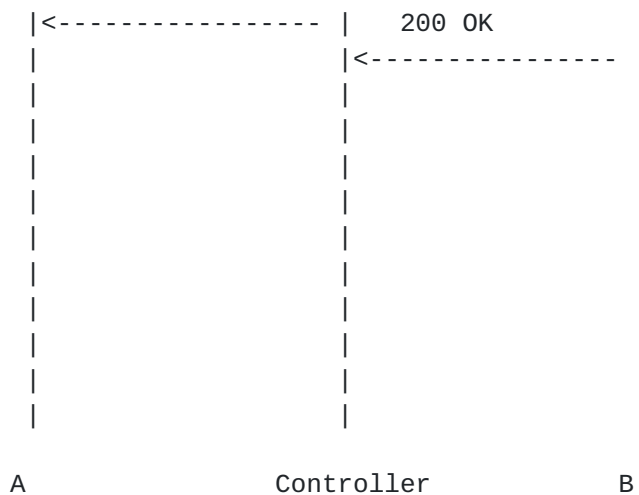
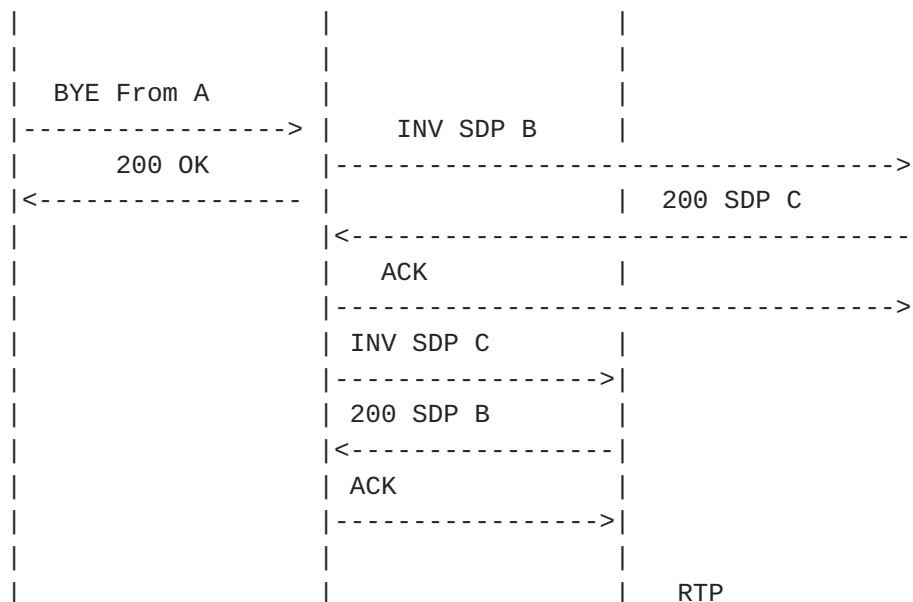


Figure 2

As an alternative, when the controller receives a BYE from A, it can generate a new INVITE to a third party, C, using the SDP from B. When the 200 OK arrives from C, the controller sends a re-INVITE to B, using the SDP from C. If the 200 OK to the re-INVITE contains the same SDP as it used in the INVITE to C, the controller has successfully connected B to C, transparently to B. A call flow for this is shown in Figure 3.



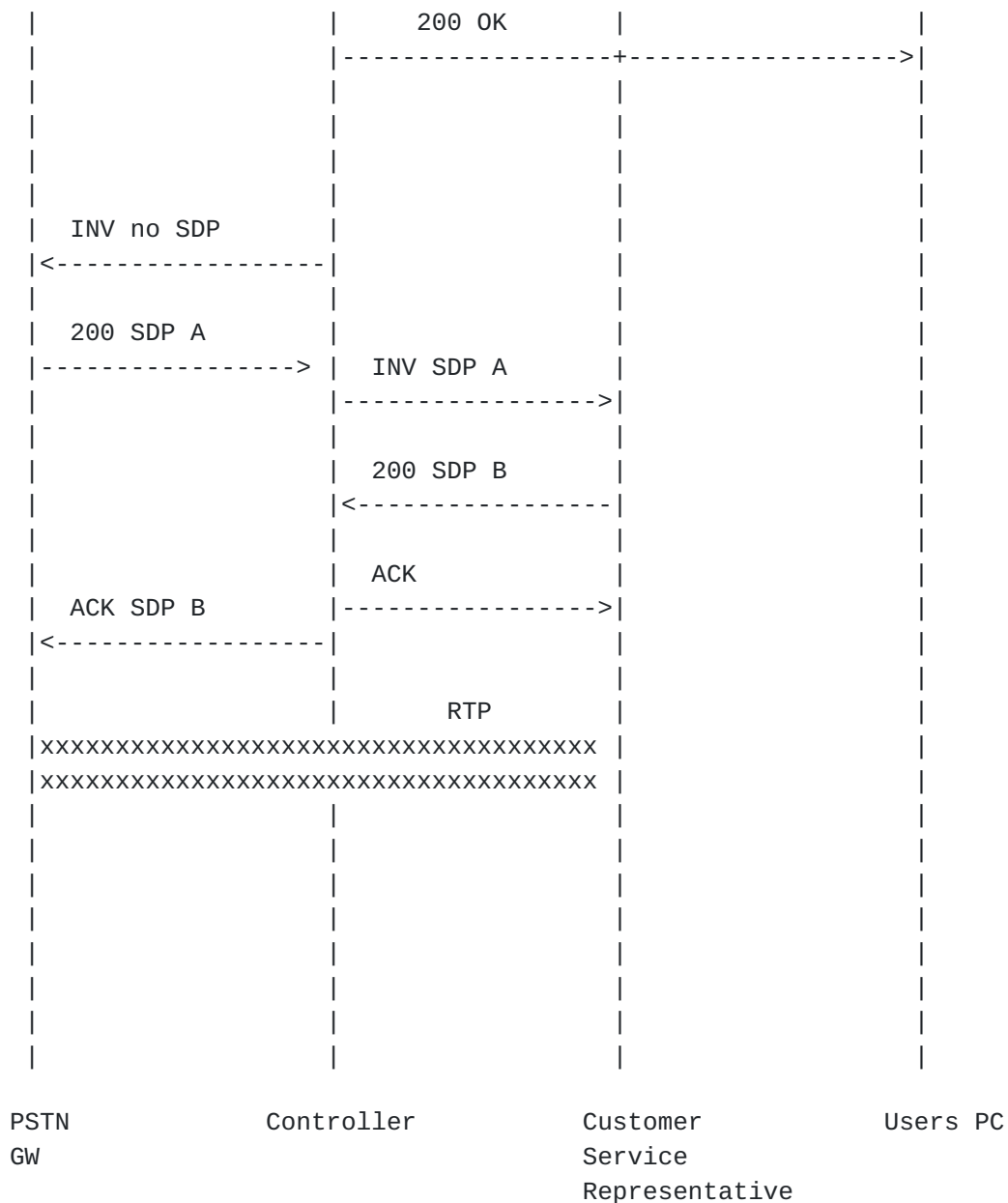


Figure 4

We note that this service can be provided through other mechanisms, namely PINT [\[1\]](#). However, there are numerous differences between the way in which the service is provided by pint, and the way in which it is provided here:

- o The pint solution enables calls only between two PSTN endpoints. The solution described here allows calls between PSTN phones (through SIP enabled gateways) and native IP

phones.

- o When used for calls between two PSTN phones, the solution here may result in a portion of the call being routed over the Internet. In pint, the call is always routed only over the PSTN. This will probably result in better quality calls with the pint solution.
- o The PINT solution requires extensions to SIP (PINT is an extension to SIP), whereas the solution described here is done with baseline SIP.
- o The PINT solution allows the controller (acting as a PINT client) to "step out" once the call is established. The solution described here requires the controller to remain in the picture for the entire duration of the call.

4 Mid-Call Announcement Capability

The third party call control mechanism described here can also be used to enable mid-call announcements. The call is set up by the controller as described above.[2] payphone, in which case the controller determines that the call is to be terminated after some amount of time if the user doesn't add more money to the phone. When this timer expires, the controller initiates places the called party on hold. It then sends an INVITE to the media server which will be collecting digits. It then sends a re-INVITE to the user on the payphone, connecting its media streams with the media server. The media server plays an announcement, and prompts the user to enter a credit card number, for example. After collecting the number and validating the card, if the call can continue, the media server hangs up. The controller takes this as a cue and reconnects the user to the original called party, and takes the original called party off hold.

A call flow for this service is shown in Figure 5

| RTP |

[2] It is actually not necessary for the controller to set up the call. However, if a participant initiates the call, the controller must step in as a virtual UAC/UAS, and act as a termination and re-initiation point

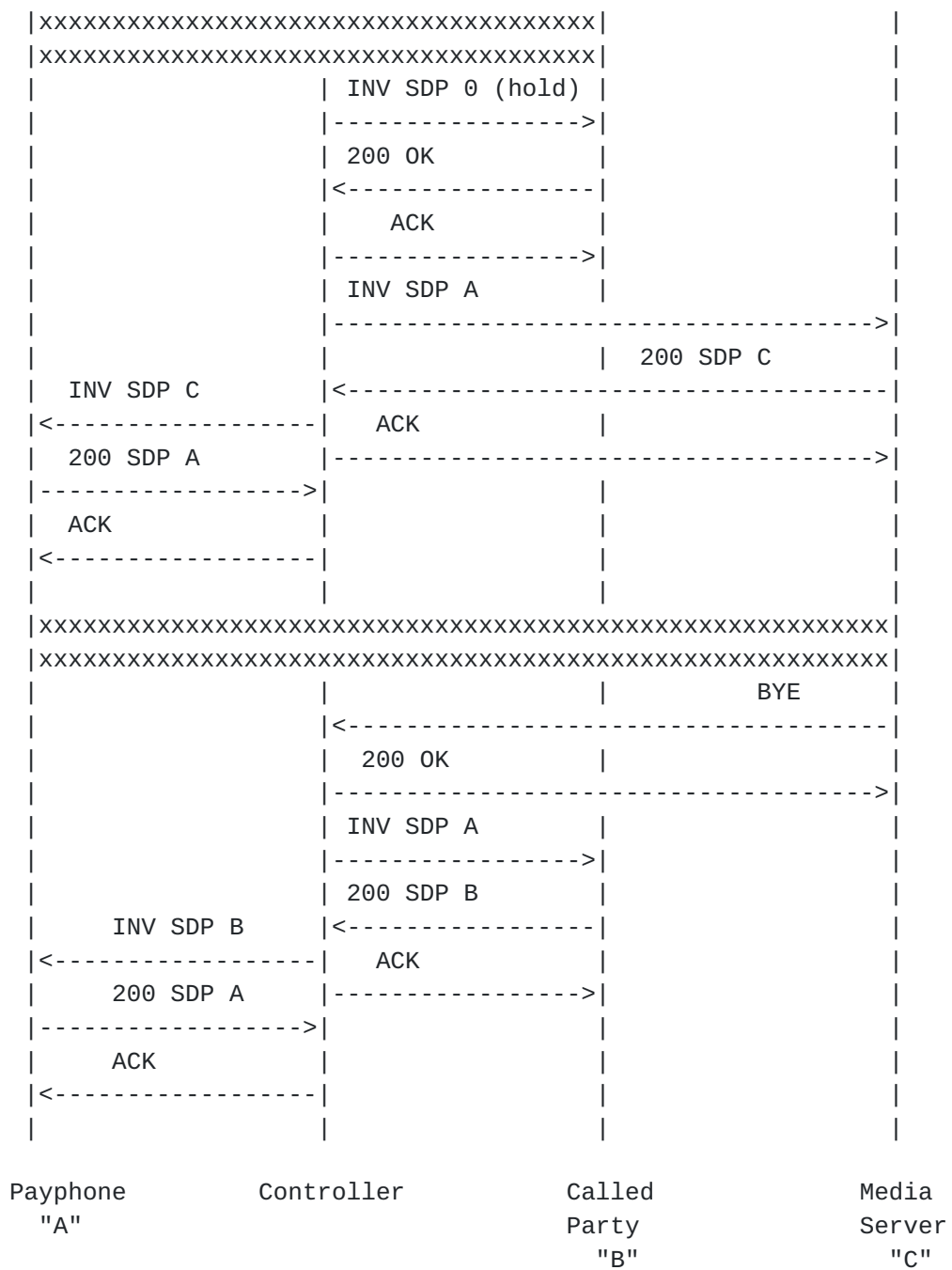


Figure 5

We have assumed that the media server and the controller have agreed, ahead of time, that a hangup implies that the desired service (extending the lifetime of the call) has succeeded. This is effectively allowing a call control interface between the controller

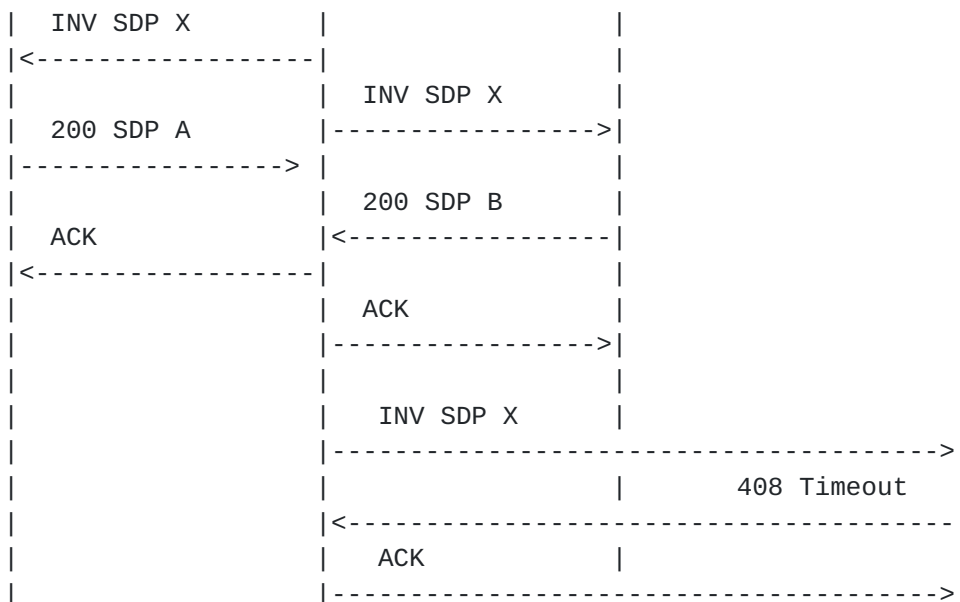
and the media server. Parameters needed between the elements, such as the new expiration of the call, can be passed in the BYE. A separate draft, forthcoming, will discuss call control interfaces to media services in more detail.

5 Timed Conference Intitation

In this service, a conference bridge is booked for some number of participants. In order to make sure the conference begins on time, the conference bridge will call each participant at the time of the call. If a participant doesn't answer, the bridge tries to contact them again (unless they call in) five minutes later.

In the call flow described here, we assume that the controller acts as the media bridge. This is not strictly necessary; some kind of control interface could be used to separate the media function from the controller.

The call flow, shown in Figure 6, is, not surprisingly, remarkably like that of Figure 1. The only difference is that the SDP listed in the INVITE s generated by the controller always contain SDP that points to the conference bridge, rather than one of the other participants. In the call flow diagram, user 1 is invited first, then user 2, and then user 3. User 3 is not available, but is called again five minutes later.



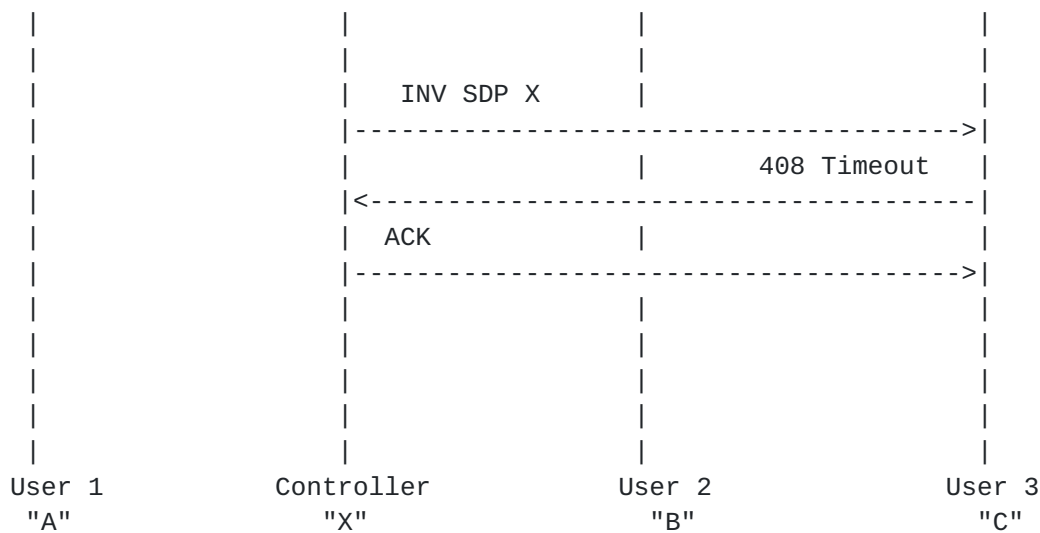


Figure 6

6 Future Work

We plan on considering other mechanisms which might be used for third party call control, and discuss the pros and cons for each for providing numerous services.

7 Conclusions

We have presented a basic third party call control mechanism that uses SIP. This mechanism does not require any extensions to SIP and is completely backwards compatible.

8 Authors Addresses

Jonathan Rosenberg
dynamicsoft
200 Executive Drive
Suite 120
West Orange, NJ 07052
email: jdrosen@dynamicsoft.com

Jon Peterson
Level 3 Communications

Henning Schulzrinne
Columbia University

M/S 0401
1214 Amsterdam Ave.
New York, NY 10027-7003
email: schulzrinne@cs.columbia.edu

9 Bibliography

[1] S. Petrack and L. Conroy, "The PINT service protocol: Extensions to SIP and SDP for IP access to telephone call services," Internet Draft, Internet Engineering Task Force, June 1999. Work in progress.

Full Copyright Statement

Copyright (c) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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.

