

May 24, 2006

**RTP Topologies**  
**draft-wenger-avt-topologies-00.txt>**

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>.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document disucsses multi-endpoint topologies commonly used in RTP based environments. In particular, centralized topologies commonly employed in the video conferencing industry are mapped to the RTP terminology.

TABLE OF CONTENTS

- [1. Introduction.....3](#)
- [2. Definitions.....3](#)
  - [2.1. Glossary.....4](#)
  - [2.2. Terminology.....4](#)
  - [2.3. Topologies.....5](#)
    - [2.3.1. Point to Point.....5](#)
    - [2.3.2. Point to Multi-point using Multicast.....6](#)
    - [2.3.3. Point to Multipoint using the \[RFC 3550\]\(#\) translator.....7](#)
    - [2.3.4. Point to Multipoint using the \[RFC 3550\]\(#\) mixer model.....9](#)
    - [2.3.5. Point to Multipoint using video switching MCU.....11](#)
    - [2.3.6. Point to Multipoint using RTCP-terminating MCU.....12](#)
    - [2.3.7. Combining Topologies.....13](#)
- [3. Acknowledgements.....13](#)
- [4. References.....14](#)
  - [4.1. Normative references.....14](#)
  - [4.2. Informative references.....14](#)
- [5. Authors' Addresses.....14](#)
- [6. List of Changes relative to previous drafts.....15](#)



## 1. Introduction

When working on the Codec Control Messages [[CCM](#)], we noticed a considerable confusion in the community with respect to terms such as MCU, mixer, and translator. In the process of writing, we became increasingly unsure of our own understanding, and therefore added what became the core of this draft to the CCM draft. Later, it was found that this information has its own value, and was 'outsourced' from the CCM draft into the present memo.

It could be argued that this document clarifies and explains sections of the RTP spec [[RFC3550](#)], and is therefore of informational nature. In this case, the present memo may end up as an informational RFC.

When the Audio-Visual Profile with Feedback (AVPF) [[RFC4585](#)] was developed, the main emphasis lied in the efficient support of point-to-point and small multipoint scenarios without centralized multipoint control. However, in practice, many small multipoint conferences operate utilizing devices known as Multipoint Control Units (MCUs). MCUs comprise mixers and translators (in RTP [[RFC3550](#)] terminology), but also signalling support. Long standing experience of the conversational video conferencing industry suggests that there is a need for a few additional feedback messages, to efficiently support MCU-based multipoint conferencing. Some of the messages have applications beyond centralized multipoint, and this is indicated in the description of the message.

Some of the messages defined here are forward only, in that they do not require an explicit acknowledgement. Other messages require acknowledgement, leading to a two way communication model that could suggest to some to be useful for control purposes. It is not the intention of this memo to open up the use of RTCP to generalized control protocol functionality. All mentioned messages have relatively strict real-time constraints and are of transient nature, which make the use of more traditional control protocol means, such as SIP re-invites, undesirable. Furthermore, all messages are of a very simple format that can be easily processed by an RTP/RTCP sender/receiver. Finally, all messages infer only to the RTP stream they are related to, and not to any other property of a communication system.

## 2. Definitions



## [2.1.](#) Glossary

ASM - Asynchronous Multicast  
AVPF - The Extended RTP Profile for RTCP-based Feedback  
MCU - Multipoint Control Unit  
PtM - Point to Multipoint  
PtP - Point to Point

## [2.2.](#) 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 [RFC 2119](#) [[RFC2119](#)].

Message:

Codepoint defined by this specification, of one of the following types:

Request:

Message that requires Acknowledgement

Acknowledgment:

Message that answers a Request

Command:

Message that forces the receiver to an action

Indication:

Message that reports a situation

Notification:

See Indication.

Note that, with the exception of 'Notification', this terminology is in alignment with ITU-T Rec. H.245.

Decoder Refresh Point:

A bit string, packetised in one or more RTP packets, which completely resets the decoder to a known state. Typical examples of Decoder Refresh Points are H.261 Intra pictures and H.264 IDR pictures. However, there are also much more complex decoder refresh points.

Typical examples for "hard" decoder refresh points are Intra pictures in H.261, H.263, MPEG 1, MPEG 2, and MPEG-4 part 2, and IDR pictures in H.264. "Gradual" decoder refresh points may also be used. While both "hard" and "gradual" decoder

refresh points are acceptable in the scope of this

Wenger, et al.

Informational

[Page 4]

specification, in most cases the user experience will benefit from using a "hard" decoder refresh point.

A decoder refresh point also contains all header information above the picture layer (or equivalent, depending on the video compression standard) that is conveyed in-band. In H.264, for example, a decoder refresh point contains parameter set NAL units that generate parameter sets necessary for the decoding of the following slice/data partition NAL units (and that are not conveyed out of band). To the best of the author's knowledge, the term "Decoder Refresh Point" has been formally defined only in H.264; hence we are referring here to this video compression standard.

**Decoding:**

The operation of reconstructing the media stream.

**Rendering:**

The operation of presenting (parts of) the reconstructed media stream to the user.

**Stream thinning:**

The operation of removing some of the packets from a media stream. Stream thinning, preferably, is performed in a media aware fashion implying that the media packets are removed in the order of their relevance to the reproductive quality. However even when employing media-aware stream thinning, most media streams quickly lose quality when subject to increasing levels of thinning. Media-unaware stream thinning leads to even worse quality degradation.

## **2.3. Topologies**

This subsection defines several basic topologies that are relevant for codec control. The first four relate to the RTP system model utilizing multicast and/or unicast, as envisioned in [RFC 3550](#). The last two topologies, in contrast, describe the widely deployed system model as used in most H.323 video conferences, where both the media streams and the RTCP control traffic terminate at the MCU. More topologies can be constructed by combining any of the models, see [Section 2.3.7](#).

### **2.3.1. Point to Point**

The Point to Point (PtP) topology (Figure 1) consists of two endpoints with unicast capabilities between them. Both RTP and RTCP traffic are conveyed endpoint to endpoint using unicast traffic only





(even if this unicast traffic happens to be conveyed over an IP-multicast address).

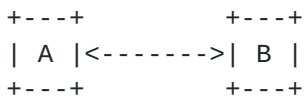


Figure 1 - Point to Point

The main property of this topology is that A sends to B and only B, while B sends to A and only A. This avoids all complexities of handling multiple endpoints and combining the requirements from them. Do note that an endpoint may still use multiple RTP Synchronization Sources (SSRCs) in an RTP session.

**2.3.2. Point to Multi-point using Multicast**

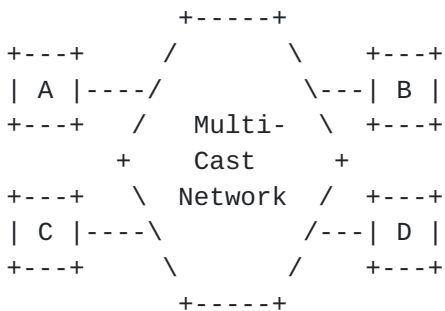


Figure 2 - Point to Multipoint using Multicast

We define Point to Multipoint (PtM) using multicast topology as a transmission model in which traffic from any participant reaches all the other participants, except for cases such as

- o packet loss occurs,
- o a participant participant does not wish to receive the traffic from a certain other participant, and therefore has not subscribed to the IP multicast group in question.

In this sense, "traffic" encompasses both RTP and RTCP traffic. The number of participants can be between one and many -- as RTP and RTCP scales to very large multicast groups (the theoretical limit of RTP is approximately two billion participants).

This draft is primarily interested in the subset of multicast session where the number of participants in the multicast group allows the participants to use early or immediate feedback as defined in AVPF. This document refers to those groups as as "small multicast groups".



2.3.3. Point to Multipoint using the RFC 3550 translator

Two main categories of Translators can be distinguished.

Transport Translators do not modify the media stream itself, but are concerned with transport parameters. Transport parameters, in the sense of this section, comprise the transport addresses to bridge different domains, and the media packetization to allow other transport protocols to be interconnected to a session (gateways).

Media Translators, in contrast, modify the media stream itself. This process is commonly known as transcoding. The modification of the media stream can be as small as removing parts of the stream, and can go all the way to a full transcoding utilizing a different media codec. Media translators are commonly used to connect entities without a common interoperability point.

Stand-alone Media Translators are rare. Most commonly, a combination of Transport and Media Translators are used to translate both the media stream and the transport aspects of a stream between two transport domains (or clouds).

Both Translator types share common attributes that separates them from mixers. For each media stream that the translator receives, it generates an individual stream in the other domain. However, a translator maintains a complete view of all existing participants between both domains. Therefore, the SSRC space is shared across the two domains.

The RTCP translation process can be trivial, for example when Transport translators just need to adjust IP addresses, and can be quite complex in the case of media translators. See section 7.2 of [RFC3550].

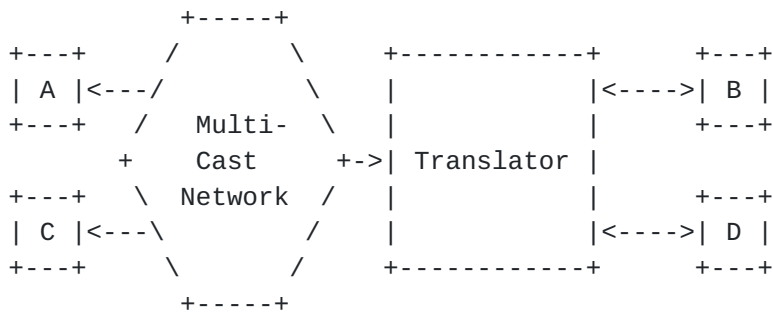


Figure 3 - Point to Multipoint using a Translator

Figure 3 depicts an example of a Transport Translator performing at least IP address translation. It allows the (non multicast capable)



participants B and D to take part in a multicasted session by having the translator forward their unicast traffic to the multicast addresses in use, and vice versa. It must also forward B's traffic to D and vice versa, to provide each of B and D with a complete view of the session.

If B were behind a limited link, the translator may perform media transcoding to allow the traffic received from the other participants to reach B without overloading the link.

When in the example depicted in Figure 3 the translator acts only as a Transport Translator, then the RTCP traffic can simply be forwarded, similar to the media traffic. However, when media translation occurs, the translator's task becomes substantially more complex even with respect to the RTCP traffic. In this case, the translator needs to rewrite B's RTCP receiver report, before forwarding them to D and the multicast network. The rewriting is needed as the stream received by B is not the same stream as the other participants receive. For example, the number of packets transmitted to B may be lower than what D receives, due to the different media format. Therefore, if the receiver reports were forwarded without changes, the extended highest sequence number would indicate that B were substantially behind in reception -- while it most likely it would not be. Therefore, the translator must translate that number to a corresponding sequence number for the stream the translator received. Similar arguments can be made for most other fields in the RTCP receiver reports.

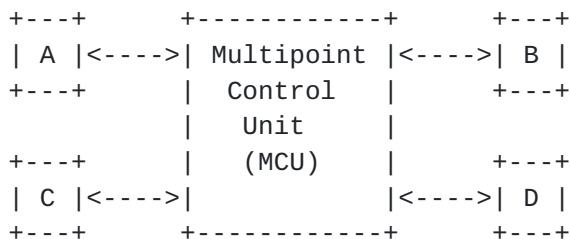


Figure 4 - MCU with RTP Translator (relay) with only unicast links

A common MCU scenario is the one depicted in Figure 4 - MCU with RTP Translator (relay) with only unicast links. Herein, the MCU connects multiple users of a conference through unicast. This can be implemented using a very simple transport translator, which could be called a relay. The relay forwards all traffic it receives, both RTP and RTCP, to all other participants. In doing so, a multicast network is emulated without relying on a multicast capable network structure.



**2.3.4. Point to Multipoint using the [RFC 3550](#) mixer model**

A mixer is a middlebox that aggregates multiple RTP streams that are part of a session, by mixing the media data and generating a new RTP stream. One common application for a mixer is to allow a participant to receive a session with a reduced amount of resources.

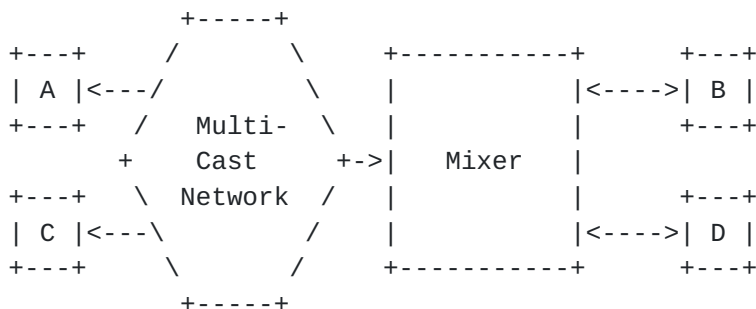


Figure 5 - Point to Multipoint using [RFC 3550](#) mixer model

A mixer can be viewed as a device terminating the media streams received from other session participants. Using the media data from the received media streams, a mixer generates a media stream that is sent to the session participant.

The content that the mixer provides is the mixed aggregate of what the mixer receives from the PtP or PtM links, which are part of the same conference session.

The mixer is the content source, as it mixes the content (often in the uncompressed domain) and then encodes it for transmission to a participant. The CC and CSRC fields in the RTP header are used to indicate the contributors of to the newly generated stream. The SSRCs of the to-be-mixed streams on the mixer input appear as the CSRCs at the mixer output. That output stream uses a new SSRC that identifies the Mixer. The CSRC are forwarded between the two domains to allow for loop detection and identification of sources that are part of the global session.

The mixer is responsible for generating RTCP packets in accordance with its role. It is a receiver and should therefore send reception reports for the media streams it receives. As a media sender itself it should also generate sender report for those media streams sent. The content of the SRs created by the mixer may or may not take into account the situation on its receiving side. Similarly, the content of RRs created by the mixer may or may not be based on the situation on the mixer's sending side. This is left open to the implementation. As specified in [Section 7.3 of RFC 3550](#), a mixer must not forward RTCP unaltered between the two domains.





The mixer depicted in Figure 5 has three domains that needs to be separated; the multicast network, participant B and participant D. The Mixer produces different mixed streams to B and D, as the one to B may contain D and vice versa. However the mixer does only need one SSRC in each domain that is the receiving entity and transmitter of mixed content.

In the multicast domain the mixer does not provide a mixed view of the other domains and only forwards media from B and D into the multicast network using B's and D's SSRC.

The mixer is responsible for receiving the codec control messages and handles them appropriately. The definition of "appropriate" depends on the message itself and the context. In some cases, the reception of a codec control message may result in the generation and transmission of codec control messages by the mixer to the participants in the other domain. In other cases, a message is handled by the mixer itself and therefore not forwarded to any other domains.

It should be noted that this form of mixing technology is not widely deployed. Most multipoint video conferences used today employ one of the models discussed in the next sections.

When replacing the multicast network in Figure 5 (to the left of the mixer) with individual unicast links as depicted in Figure 6, the mixer model is very similar to the one discussed in [section 2.3.6](#) below.

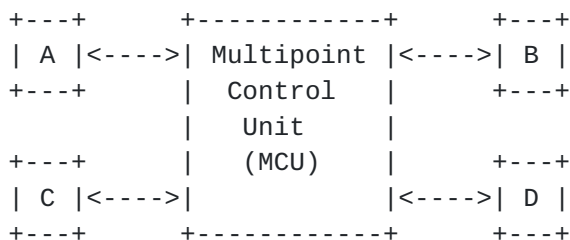


Figure 6 - RTP Mixer with only unicast links



### 2.3.5. Point to Multipoint using video switching MCU

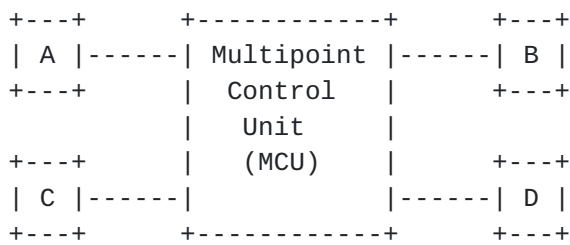


Figure 7 - Point to Multipoint using relaying MCU

This PtM topology is, today, perhaps the most widely deployed one. It reflects today's lack of wide deployment of IP multicast technologies on IP networks and the Internet, as well as the simplicity of content switching when compared to content mixing. The technology is commonly implemented in what is known as 'Video Switching MCUs'.

A video switch MCU forwards to a participant a single media stream, selected from the available streams. The criteria for selection are often based on voice activity in the audio-visual conference, but other conference management mechanisms (like explicit floor control) are known to exist as well.

The video switching MCU may also perform media translation to modify the content in bit-rate, encoding, resolution; however it still indicates the original sender of the content through the SSRC. The values of the CC and CSRC fields are retained.

RTCP Sender Reports are forwarded for the currently selected sender. All RTCP receiver reports are freely forward between the participants. In addition, the MCU may also originate RTCP control traffic in order to control the session and/or report on status from its viewpoint.

The video switching MCU has mostly the attributes of a translator. However its stream selection is a mixing behaviour. This behaviour has some RTP and RTCP issues associated with it. The suppression of all but one media stream results in that most participants see only a subset of the sent media streams at any given time; often a single stream per conference. Therefore, RTCP receiver reports only report on these streams. In consequence, the media senders that are not currently forwarded receive a view of the session that indicates their media streams disappearing somewhere en route. This makes the use of RTCP for congestion control very problematic. To avoid these issues the MCU needs to modify the RTCP RRs.



**2.3.6. Point to Multipoint using RTCP-terminating MCU**

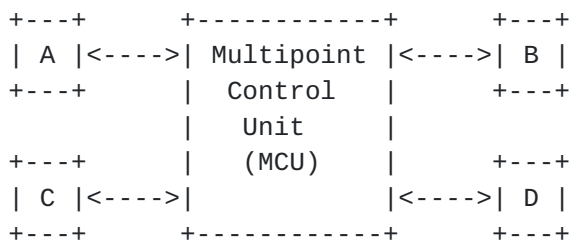


Figure 8 - Point to Multipoint using content modifying MCU

In this PtM scenario, each participant runs an RTP point-to-point session between itself and the MCU. The content that the MCU provides to each participant is either:

- a) A selection of the content received from the other participants, or
- b) The mixed aggregate of what the MCU receives from the other PtP links, which are part of the same conference session.

In case a) the MCU may modify the content in bit-rate, encoding, resolution. No explicit RTP mechanism is used to establish the relationship between the original media sender and the version the MCU sends. In other words, the outgoing session typically uses a different SSRC, and may well use a different PT, even if this different PT happens to be mapped to the same media type. (This is the definition of this topology and distinguishes it from the topologies previously discussed).

In case b) the MCU is the content source as it mixes the content and then encodes it for transmission to a participant. The participant's content that is included in the aggregated content is not indicated through any explicit RTP mechanism. For example, regardless of the number of streams that are aggregated, in the MCU generated streams CC is zero and therefore no CSRC fields are present.

The MCU is responsible for receiving the codec control messages and handle them appropriately. In some cases, the reception of a codec control message may result in the generation and transmission of codec control messages by the MCU to some or all of the other participants.

An MCU may transparently relay some codec control messages and intercept, modify, and (when appropriate) generate codec control messages of its own and transmit them to the media senders.



The main feature that sets this topology apart from what [RFC 3550](#) describes, is the lack of an explicit RTP level indication of all participants. If one were using the mechanisms available in RTP and RTCP to signal this explicitly, the topology would follow the approach of an RTP mixer. The lack of explicit indication has at least the following potential problems:

- 1) Loop detection cannot be performed on the RTP level. When carelessly connecting two misconfigured MCUs, a loop could be generated.
- 2) There is no information about active media senders available in the RTP packet. As this information is missing, receivers cannot use it. It also deprive the participant's clients information about who are actively sending in a machine usable way. Thus preventing clients from doing indication of currently active speakers in user interfaces, etc.

#### **[2.3.7.](#) Combining Topologies**

Topologies can be combined and linked to each other using mixers or translators. Care must however be taken to how the SSRC space is handled, mixers separate the SSRC space into two parts, while translators maintain the space across themselves. Any hybrid, like the video switching MCU, 2.3.5, requires considerable afterthought on how RTCP is dealt with.

### **[3.](#) Security Considerations**

This document does not specify any protocol mechanisms and should not have any security issues

### **[4.](#) IANA considerations**

None

### **[5.](#) Acknowledgements**

The authors would like to thank N.N.





## **6. References**

### **6.1. Normative references**

None.

### **6.2. Informative references**

- [CCM] Wenger, S., Chandra, U., Westerlund, M, Burman, B., 'Codec Control Messages in the Audio-Visual Profile with Feedback (AVPF)', [draft-wenger-avt-avpf-ext-04.txt](#), Work in Progress, May 2006
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, [RFC 3550](#), July 2003.
- [RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., Rey, J., 'Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)', [RFC 4585](#), July 2006

Any 3GPP document can be downloaded from the 3GPP web server, "http://www.3gpp.org/", see specifications.

## **7. Authors' Addresses**

Magnus Westerlund  
Ericsson Research  
Ericsson AB  
SE-164 80 Stockholm, SWEDEN

Phone: +46 8 7190000  
EMail: [magnus.westerlund@ericsson.com](mailto:magnus.westerlund@ericsson.com)

Stephan Wenger  
Nokia Corporation  
P.O. Box 100  
FIN-33721 Tampere  
FINLAND

Phone: +358-50-486-0637  
EMail: [stewe@stewe.org](mailto:stewe@stewe.org)



## **8. List of Changes relative to previous drafts**

### Full 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.

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.

### 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](mailto:ietf-ipr@ietf.org).

### Acknowledgment

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



#### RFC Editor Considerations

The RFC editor is requested to replace all occurrences of XXXX with the RFC number this document receives.

