

Network Working Group
Request for Comments: 4247
Category: Informational

J. Ash
B. Goode
J. Hand
AT&T
R. Zhang
BT Infonet
November 2005

Requirements for Header Compression over MPLS

Status of This Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

Voice over IP (VoIP) typically uses the encapsulation voice/RTP/UDP/IP. When MPLS labels are added, this becomes voice/RTP/UDP/IP/MPLS-labels. For an MPLS VPN, the packet header is typically 48 bytes, while the voice payload is often no more than 30 bytes, for example. Header compression can significantly reduce the overhead through various compression mechanisms, such as enhanced compressed RTP (ECRTP) and robust header compression (ROHC). We consider using MPLS to route compressed packets over an MPLS Label Switched Path (LSP) without compression/decompression cycles at each router. This approach can increase the bandwidth efficiency as well as processing scalability of the maximum number of simultaneous flows that use header compression at each router. In this document, we give a problem statement, goals and requirements, and an example scenario.

Table of Contents

1. Introduction	2
2. Problem Statement	3
2.1. Specification of Requirements	4
3. Goals and Requirements	5
4. Candidate Solution Methods and Needs	6
5. Example Scenario	7
6. Security Considerations	8
7. Normative References	8
8. Informative References	9
9. Acknowledgements	10

1. Introduction

Voice over IP (VoIP) typically uses the encapsulation voice/RTP/UDP/IP. When MPLS labels [MPLS-ARCH] are added, this becomes voice/RTP/UDP/IP/MPLS-labels. For an MPLS Virtual Private Network (VPN) (e.g., [MPLS-VPN]), the packet header is at least 48 bytes, while the voice payload is often no more than 30 bytes, for example. The interest in header compression (HC) is to exploit the possibility of significantly reducing the overhead through various compression mechanisms, such as with enhanced compressed RTP [ECRTP] or robust header compression [ROHC], and also to increase scalability of HC. We consider using MPLS to route compressed packets over an MPLS Label Switched Path (LSP) without compression/decompression cycles at each router. Such an HC over MPLS capability can increase bandwidth efficiency as well as the processing scalability of the maximum number of simultaneous flows that use HC at each router.

To implement HC over MPLS, the ingress router/gateway would have to apply the HC algorithm to the IP packet, the compressed packet routed on an MPLS LSP using MPLS labels, and the compressed header would be decompressed at the egress router/gateway where the HC session terminates. Figure 1 illustrates an HC over MPLS session established on an LSP that crosses several routers, from R1/HC --> R2 --> R3 --> R4/HD, where R1/HC is the ingress router where HC is performed, and R4/HD is the egress router where header decompression (HD) is done. HC of the RTP/UDP/IP header is performed at R1/HC, and the compressed packets are routed using MPLS labels from R1/HC to R2, to R3, and finally to R4/HD, without further decompression/recompression cycles. The RTP/UDP/IP header is decompressed at R4/HD and can be forwarded to other routers, as needed.

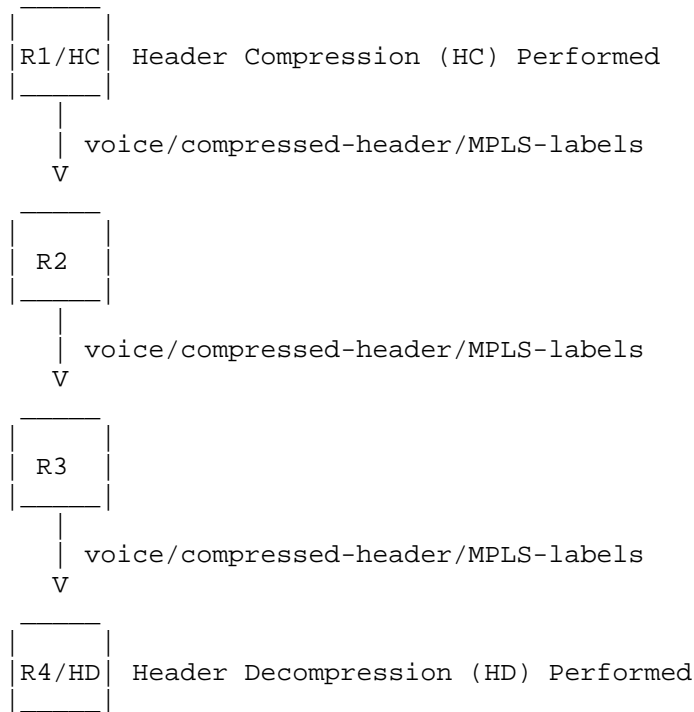


Figure 1. Example of Header Compression over MPLS over Routers R1-->R4

In the example scenario, HC therefore takes place between R1 and R4, and the MPLS path transports voice/compressed-header/MPLS-labels instead of voice/RTP/UDP/IP/MPLS-labels, typically saving 30 octets or more per packet. The MPLS label stack and link-layer headers are not compressed. A signaling method is needed to set up a correspondence between the ingress and egress routers of the HC over MPLS session.

In Section 2 we give a problem statement, in Section 3 we give goals and requirements, and in Section 5 we give an example scenario.

2. Problem Statement

As described in the introduction, HC over MPLS can significantly reduce the header overhead through HC mechanisms. The need for HC may be important on low-speed links where bandwidth is more scarce, but it could also be important on backbone facilities, especially where costs are high (e.g., some global cross-sections). VoIP typically will use voice compression mechanisms (e.g., G.729) on

low-speed and international routes, in order to conserve bandwidth. With HC, significantly more bandwidth could be saved. For example, carrying uncompressed headers for the entire voice load of a large domestic network with 300 million or more calls per day could consume on the order of about 20-40 gigabits per second on the backbone network for headers alone. This overhead could translate into considerable bandwidth capacity.

The claim is often made that once fiber is in place, increasing the bandwidth capacity is inexpensive, nearly 'free'. This may be true in some cases; however, on some international cross-sections, especially, facility/transport costs are very high and saving bandwidth on such backbone links is very worthwhile. Decreasing the backbone bandwidth is needed in some areas of the world where bandwidth is very expensive. It is also important in almost all locations to decrease the bandwidth consumption on low-speed links. So although bandwidth is getting cheaper, the value of compression does not go away. It should be further noted that IPv6 will increase the size of headers, and therefore increase the importance of HC for RTP flows.

Although hop-by-hop HC could be applied to decrease bandwidth requirements, that implies a processing requirement for compression-decompression cycles at every router hop, which does not scale well for large voice traffic loads. The maximum number of compressed RTP (cRTP) flows is about 30-50 for a typical customer premise router, depending upon its uplink speed and processing power, while the need may exceed 300-500 for a high-end case. Therefore, HC over MPLS seems to be a viable alternative to get the compression benefits without introducing costly processing demands on the intermediate nodes. By using HC over MPLS, routers merely forward compressed packets without doing a decompression/recompression cycle, thereby increasing the maximum number of simultaneous compressed flows that routers can handle.

Therefore, the proposal is to use existing HC techniques, together with MPLS labels, to make the transport of the RTP/UDP/IP headers more efficient over an MPLS network. However, at this time, there are no standards for HC over MPLS, and vendors have not implemented such techniques.

2.1. Specification of Requirements

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 [KEY].

3. Goals and Requirements

The goals of HC over MPLS are as follows:

- a. provide more efficient voice transport over MPLS networks,
- b. increase the scalability of HC to a large number of flows,
- c. not significantly increase packet delay, delay variation, or loss probability, and
- d. leverage existing work through use of standard protocols as much as possible.

Therefore the requirements for HC over MPLS are as follows:

- a. MUST use existing protocols (e.g., [ECRTP], [ROHC]) to compress RTP/UDP/IP headers, in order to provide for efficient transport, tolerance to packet loss, and resistance to loss of session context.
- b. MUST allow HC over an MPLS LSP, and thereby avoid hop-by-hop compression/decompression cycles (e.g., [HC-MPLS-PROTO]).
- c. MUST minimize incremental performance degradation due to increased delay, packet loss, and jitter.
- d. MUST use standard protocols to signal context identification and control information (e.g., [RSVP], [RSVP-TE], [LDP]).
- e. Packet reordering MUST NOT cause incorrectly decompressed packets to be forwarded from the decompressor.

It is necessary that the HC method be able to handle out-of-sequence packets. MPLS [MPLS-ARCH] enables 4-byte labels to be appended to IP packets to allow switching from the ingress Label Switching Router (LSR) to the egress LSP on an LSP through an MPLS network. However, MPLS does not guarantee that packets will arrive in order at the egress LSR, since a number of things could cause packets to be delivered out of sequence. For example, a link failure could cause the LSP routing to change, due perhaps to an MPLS fast reroute taking place, or to the Interior Gateway Protocol (IGP) and Label Distribution Protocol (LDP) converging to another route, among other possible reasons. Other causes could include IGP reroutes due to 'loose hops' in the LSP, or BGP route changes reflecting back into IGP reroutes. HC algorithms may be able to handle reordering magnitudes on the order of about 10 packets, which may make the time required for IGP reconvergence (typically on the order of seconds) untenable for the HC algorithm. On the other hand, MPLS fast reroute may be fast enough (on the order of 50 ms or less) for the HC algorithm to handle packet reordering. The issue of reordering needs to be further considered in the development of the HC over MPLS solution.

Resynchronization and performance also needs to be considered, since HC over MPLS can sometimes have multiple routers in the LSP. Tunneling an HC session over an MPLS LSP with multiple routers in the path will increase the round-trip delay and the chance of packet loss, and HC contexts may be invalidated due to packet loss. The HC error recovery mechanism can compound the problem when long round-trip delays are involved.

4. Candidate Solution Methods and Needs

[cRTP] performs best with very low packet error rates on all hops of the path. When the cRTP decompressor context state gets out of synch with the compressor, it will drop packets associated with the context until the two states are resynchronized. To resynchronize context state at the two ends, the decompressor transmits the CONTEXT_STATE packet to the compressor, and the compressor transmits a FULL_HEADER packet to the decompressor.

[ECRTP] uses mechanisms that make cRTP more tolerant to packet loss, and ECRTP thereby helps to minimize the use of feedback-based error recovery (CONTEXT_STATE packets). ECRTP is therefore a candidate method to make HC over MPLS more tolerant of packet loss and to guard against frequent resynchronizations. ECRTP may need some implementation adaptations to address the reordering requirement in Section 3 (requirement e), since a default implementation will probably not meet the requirement. ECRTP protocol extensions may be required to identify FULL_HEADER, CONTEXT_STATE, and compressed packet types. [cRTP-ENCAP] specifies a separate link-layer packet type defined for HC. Using a separate link-layer packet type avoids the need to add extra bits to the compression header to identify the packet type. However, this approach does not extend well to MPLS encapsulation conventions [MPLS-ENCAP], in which a separate link-layer packet type translates into a separate LSP for each packet type. In order to extend ECRTP to HC over MPLS, each packet type defined in [ECRTP] would need to be identified in an appended packet type field in the ECRTP header.

[ROHC] is also very tolerant of packet loss, and therefore is a candidate method to guard against frequent resynchronizations. ROHC also achieves a somewhat better level of compression as compared to ECRTP. ROHC may need some implementation adaptations to address the reordering requirement in Section 3 (requirement e), since a default implementation will probably not meet the requirement (see [ROHC-REORD]). ROHC already has the capability to identify the packet type in the compression header, so no further extension is needed to identify packet type.

Extensions to MPLS signaling may be needed to identify the LSP from HC to HD egress point, negotiate the HC algorithm used and protocol parameters, and negotiate the Session Context IDs (SCIDs) space between the ingress and egress routers on the MPLS LSP. For example, new objects may need to be defined for [RSVP-TE] to signal the SCID spaces between the ingress and egress routers, and the HC algorithm used to determine the context; these HC packets then contain the SCID identified by using the RSVP-TE objects. It is also desirable to signal HC over MPLS tunnels with the Label Distribution Protocol [LDP], since many RFC 2547 VPN [MPLS-VPN] implementations use LDP as the underlying LSP signaling mechanism, and LDP is very scalable. However, extensions to LDP may be needed to signal SCIDs between ingress and egress routers on HC over MPLS LSPs. For example, 'targeted LDP sessions' might be established for signaling SCIDs, or perhaps methods described in [LDP-PWE3] to signal pseudo-wires and multipoint-to-point LSPs might be extended to support signaling of SCIDs for HC over MPLS LSPs. The specific MPLS signaling protocol extensions to support these approved requirements need to be developed as a well-coordinated separate document in the appropriate IETF working groups. The IETF needs to support a coordinated process for the two solution documents, though they are in separate areas.

5. Example Scenario

As illustrated in Figure 2, many VoIP flows are originated from customer sites, which are served by routers R1, R2, and R3, and terminated at several large customer call centers, which are served by R5, R6, and R7. R4 is a service-provider router, and all VoIP flows traverse R4. It is essential that the R4-R5, R4-R6, and R4-R7 low-speed links all use HC to allow a maximum number of simultaneous VoIP flows. To allow processing at R4 to handle the volume of simultaneous VoIP flows, it is desired to use HC over MPLS for these flows. With HC over MPLS, R4 does not need to do HC/HD for the flows to the call centers, enabling more scalability of the number of simultaneous VoIP flows with HC at R4.

- [LDP] Andersson, L., Doolan, P., Feldman, N., Fredette, A., and B. Thomas, "LDP Specification", RFC 3036, January 2001.
- [MPLS-ARCH] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.
- [ROHC] Bormann, C., et al., "RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed ", RFC 3095, July 2001.
- [RSVP] Braden, R., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.
- [RSVP-TE] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

8. Informative References

- [HC-MPLS-PROTO] Ash, G., Goode, B., Hand, J., Jonsson, L-E., Malis, A., and R. Zhang, "Protocol Extensions for Header Compression over MPLS", work in progress.
- [LDP-PWE3] Martini, L., El-Aawar, N., Smith, T., and G. Heron, "Pseudowire Setup and Maintenance using the Label Distribution Protocol", work in progress.
- [MPLS-ENCAP] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, January 2001.
- [MPLS-VPN] Rosen, E. and Y. Rekhter, "BGP/MPLS VPNs", RFC 2547, March 1999.
- [ROHC-REORD] Pelletier, G., Jonsson, L-E., and K. Sandlund, "RObust Header Compression (ROHC): ROHC over Channels that can Reorder Packets", work in progress.

9. Acknowledgements

The authors wish to thank the following people (in alphabetical order) for their helpful comments and suggestions: Loa Andersson, Scott Brim, Thomas Eriksson, Victoria Fineberg, Lars-Erik Jonsson, Allison Mankin, Colin Perkins, Kristofer Sandlund, and Magnus Westerlund.

Authors' Addresses

Jerry Ash
AT&T
Room MT D5-2A01
200 Laurel Avenue
Middletown, NJ 07748, USA
Phone: +1 732-420-4578
EMail: gash@att.com

Bur Goode
AT&T
Phone: + 1 203-341-8705
EMail: bgoode@att.com

Jim Hand
AT&T
Room MT A2-1A03
200 Laurel Avenue
Middletown, NJ 07748, USA
Phone: +1 732-420-3017
EMail: jameshand@att.com

Raymond Zhang
BT Infonet
2160 E. Grand Ave.
El Segundo, CA 90025 USA
EMail: raymond.zhang@bt.infonet.com

Full Copyright Statement

Copyright (C) The Internet Society (2005).

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

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.

Acknowledgement

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