Network Working Group
Internet Draft
Expiration Date: February 2003

Tom Worster

Yakov Rekhter
Juniper Networks, Inc.

Eric C. Rosen, editor
Cisco Systems, Inc.

August 2002

Encapsulating MPLS in IP or GRE

draft-rosen-mpls-in-ip-or-gre-00.txt

Status of this Memo

Abstract

In various applications of MPLS, label stacks with multiple entries are used. In some cases, it is possible to replace the top label of the stack with an IP-based encapsulation, thereby enabling the application to run over networks which do not have MPLS enabled in their core routers. This draft specifies two IP-based encapsulations, MPLS-in-IP, and MPLS-in-GRE. Each of these is applicable in some circumstances.

Table of Contents

1. Motivation

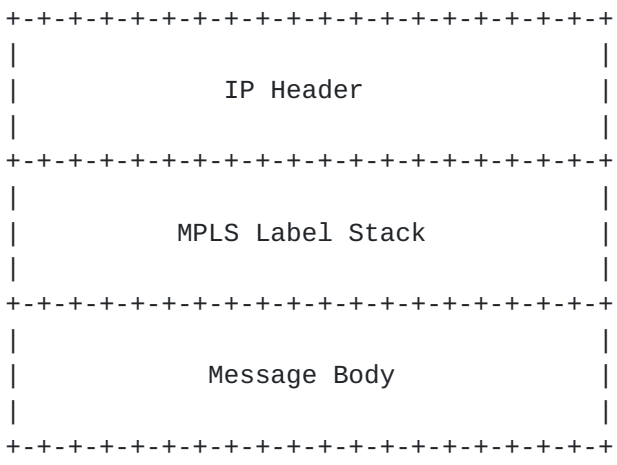
In many applications of MPLS, packets traversing an MPLS backbone carry label stacks with more than one label. As described in [RFC3031], section 3.15, each label represents a Label Switched Path (LSP). For each such LSP, there is a Label Switching Router (LSR) which is the "LSP Ingress", and an LSR which is the "LSP Egress". If LSRs A and B are the Ingress and Egress, respectively, of the LSP corresponding to a packet's top label, then A and B are adjacent LSRs on the LSP corresponding to the packet's second label (i.e., the label immediately beneath the top label)

The purpose (or one of the purposes) of the top label is to get the packet delivered from A to B, so that B can further process the packet based on the second label. In this sense, the top label serves as an encapsulation header for the rest of the packet. In some cases the top label can be replaced, without loss of functionality, by other sorts of encapsulation headers. For example, the top label could be replaced by an IP header or a GRE header. As the encapsulated packet would still be an MPLS packet, the result is an MPLS-in-IP or MPLS-in-GRE encapsulation.

With these encapsulations, it is possible for two LSRs that are adjacent on an LSP to be separated by an IP network, even if that IP network does not provide MPLS.

2. Encapsulation in IP

MPLS-in-IP messages have the following format:



IP Header
: This field contains an IPv4 or an IPv6 datagram header as defined in [RFC791] and [RFC2460] respectively. The source and destination addresses are set to addresses of the encapsulating and decapsulating LSRs respectively.

MPLS Label Stack
: This field contains an MPLS Label Stack as defined in [RFC3032].

Message Body
: This field contains one MPLS message body.

The Protocol Number field in an IPv4 header and the Next Header field in an IPv6 are set as follows:

- X indicates an MPLS unicast packet,

- Y indicates an MPLS multicast packet. (The use of the MPLS-in-IP encapsulation for MPLS multicast packets is for further study.)

Following the IP header is an MPLS packet, as specified in [RFC3032]. This encapsulation causes MPLS packets to be sent through "IP tunnels". When a packet is received by the tunnel's receive endpoint, the receive endpoint decapsulates the MPLS packet by removing the IP header. The packet is then processed as a received MPLS packet whose "incoming label" [RFC3031] is the topmost packet of the decapsulated packet.

3. Encapsulation in GRE

The MPLS-in-GRE encapsulation encapsulates an MPLS packet in GRE [RFC2784]. The packet then consists of an IP header followed by a GRE header followed by an MPLS label stack as specified in [RFC3032]. The protocol type field in the GRE header MUST be set to the Ethertype value for MPLS Unicast (0x8847) or Multicast (0x8848). The optional GRE checksum, key [RFC2890] and sequence number [RFC2890] fields MUST NOT be used.

This encapsulation causes MPLS packets to be sent through "GRE tunnels". When a packet is received by the tunnel's receive endpoint, the receive endpoint decapsulates the MPLS packet by removing the IP header and the GRE header. The packet is then processed as a received MPLS packet whose "incoming label" [RFC3031] is the topmost packet of the decapsulated packet.

4. Common Procedures

Certain procedures are common to both the MPLS-in-IP and the MPLS-in-GRE encapsulations. In the following, the encapsulator, whose address appears in the IP source address field of the encapsulating IP header, is known as the "tunnel head". The decapsulator, whose address appears in the IP destination address field of the decapsulating IP header, is known as the "tunnel tail".

4.1. Fragmentation, Reassembly, and MTU

If an MPLS-in-IP or MPLS-in-GRE packet were to get fragmented (due to "ordinary" IP fragmentation), it would have to be be reassembled by the tunnel tail before the contained MPLS packet be decapsulated. To avoid the need for the tunnel tail to perform reassembly, the tunnel head MUST set the Don't Fragment flag of the encapsulating IPv4 header.

The tunnel head SHOULD perform Path MTU Discovery [RFC1191] over each MPLS-in-IP and MPLS-in-GRE tunnel.

The tunnel head MUST maintain a Tunnel MTU value for each MPLS-in-IP or MPLS-in-GRE tunnel. This is the minimum of (a) an administratively configured value, and, if known, (b) the discovered Path MTU value minus the encapsulation overhead.

If the tunnel head receives, for encapsulation, an MPLS packet whose size exceeds the Tunnel MTU, that packet MUST be discarded.

In some cases, the tunnel head receives, for encapsulation, an IP packet, which it first encapsulates in MPLS and then encapsulates in MPLS-in-IP or MPLS-in-GRE. If the source of the IP packet is reachable from the tunnel head, and if the result of this encapsulation would be a packet whose size exceeds the Tunnel MTU, then the tunnel head SHOULD use the Tunnel MTU value for the purposes of fragmentation and PMTU discovery outside the tunnel.

4.2. TTL

The tunnel head MAY place the TTL from the MPLS label stack into the encapsulating IP header. The tunnel tail MAY place the TTL from the encapsulating IP header into the MPLS header, but only if that does not cause the TTL value in the MPLS header to become smaller.

Whether such modifications are made, and the details of how they are made, will depend on the configuration of the tunnel tail and the tunnel head.

4.3. EXP and DSCP fields

The tunnel head MAY consider the EXP field of the encapsulated MPLS packet when setting the DSCP field of the encapsulating IP header. The tunnel tail MAY modify the EXP field of the encapsulated MPLS packet, based on consideration of the DSCP field of the encapsulating IP header.

Whether such modifications are made, and the details of how they are made, will depend on the configuration of the tunnel tail and the tunnel head.

5. Applicability

The MPLS-in-IP encapsulation is the more efficient, and would generally be regarded as preferable, other things being equal. There are however some situations in which the MPLS-in-GRE encapsulation may be used:

- Two routers are "adjacent" over a GRE tunnel that exists for some reason that is outside the scope of this document, and those two routers need to send MPLS packets over that adjacency. As all packets sent over this adjacency must have a GRE encapsulation, the MPLS-in-GRE encapsulation is more efficient than the alternative, which would be an MPLS-in-IP encapsulation which is then encapsulated in GRE.

- Implementation considerations may dictate the use of MPLS-in-GRE. For example, some hardware device might only be able to handle GRE encapsulations in its fastpath.

6. Security Considerations

MPLS-in-IP or MPLS-in-GRE tunnels may be secured using IPsec. If they are not secured using IPsec, then some other method should be used to ensure that packets are decapsulated and forwarded by the tunnel tail only if those packets were encapsulated by the tunnel head. This can be done by address filtering at the boundaries of an administrative domain. When the tunnel head and the tunnel tail are not in the same domain, this may become difficult, and it can even become impossible if the packets must traverse the public Internet.

7. Acknowledgments

This draft is a combination of two previous drafts:

- draft-worster-mpls-in-ip, by Tom Worster, Paul Doolan, Yasuhiro Katsube, Tom K. Johnson, Andrew G. Malis, and Rick Wilder

- draft-rekhter-mpls-over-gre, by Yakov Rekhter, Daniel Tappan, and Eric Rosen

The current authors wish to thank all these authors for their contribution.

8. References

[RFC7915] "Internet Protocol," J. Postel, Sep 1981

[RFC2460]"Internet Protocol, Version 6 (IPv6) Specification," S. Deering and R. Hinden, RFC 2460,Dec 1998

[RFC1191] "Path MTU Discovery", J.C. Mogul, S.E. Deering, November 1990

[RFC2784] "Generic Routing Encapsulation (GRE)", D. Farinacci, T. Li, S. Hanks, D. Meyer, P. Traina, March 2000

[RFC2890] "Key and Sequence Number Extensions to GRE", G. Dommety, August 2000

[RFC3031] "Multiprotocol Label Switching Architecture", E. Rosen, A.

Viswanathan, R. Callon, January 2001

[RFC3032] "MPLS Label Stack Encoding", E. Rosen, D. Tappan, G. Fedorkow, Y. Rekhter, D. Farinacci, T. Li, A. Conta. January 2001

9. Author Information

Tom Worster
Email: fsb@thefsb.org

Yakov Rekhter
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
Email: yakov@juniper.net

Eric Rosen
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA, 01824
e-mail: erosen@cisco.com