Network Working Group
Internet Draft
Expiration Date: September 2002
Network Working Group

Kireeti Kompella (Juniper)
Ping Pan (Juniper)
Nischal Sheth (Juniper)
Dave Cooper (Global Crossing)
George Swallow (Cisco)
Sanjay Wadhwa (Unisphere)
Ron Bonica (Worldcom)

Detecting Data Plane Liveliness in MPLS

draft-ietf-mpls-lsp-ping-00.txt

Status of this Memo

Abstract

This document describes a simple and efficient mechanism that can be used to detect data plane failures in MPLS LSPs. There are two parts to this document: information carried in an MPLS "echo request" and "echo reply" for the purposes of fault detection and isolation; and mechanisms for transporting the echo reply.

Sub-IP ID Summary

(See Abstract above.)

RELATED DOCUMENTS

May be found in the "references" section.

WHERE DOES IT FIT IN THE PICTURE OF THE SUB-IP WORK

Fits in the MPLS box.

WHY IS IT TARGETED AT THIS WG

MPLS WG is currently looking at MPLS-specific error detection and recovery mechanisms. The mechanisms proposed here are for packet-based MPLS LSPs, which is why the MPLS WG is targeted.

JUSTIFICATION

The WG should consider this document, as it allows network operators to detect MPLS LSP data plane failures in the network. This type of failures have occurred, and are a source of concern to operators implementing MPLS networks.

1. Introduction

This document describes a simple and efficient mechanism that can be used to detect data plane failures in MPLS LSPs. There are two parts to this document: information carried in an MPLS "echo request" and "echo reply"; and mechanisms for transporting the echo reply. The first part aims at providing enough information to check correct operation of the data plane, as well as a mechanism to verify the data plane against the control plane, and thereby localize faults. The second part suggests two methods of reliable reply channels for the echo request message, for more robust fault isolation.

An important consideration in this design is that MPLS echo requests follow the same data path that normal MPLS packets would traverse. MPLS echo requests are meant primarily to validate the data plane, and secondarily to verify the data plane against the control plane. Mechanisms to check the control plane are valuable, but are not covered in this document.

To avoid potential Denial of Service attacks, it is recommended to regulate the MPLS ping traffic going to the control plane. A rate limiter should be applied to the well-known UDP port defined below.

2. Motivation

When an LSP fails to deliver user traffic, the failure cannot always be detected by the MPLS control plane. There is a need to provide a tool that would enable users to detect such traffic "black holes" or misrouting within a reasonable period of time; and a mechanism to isolate faults.

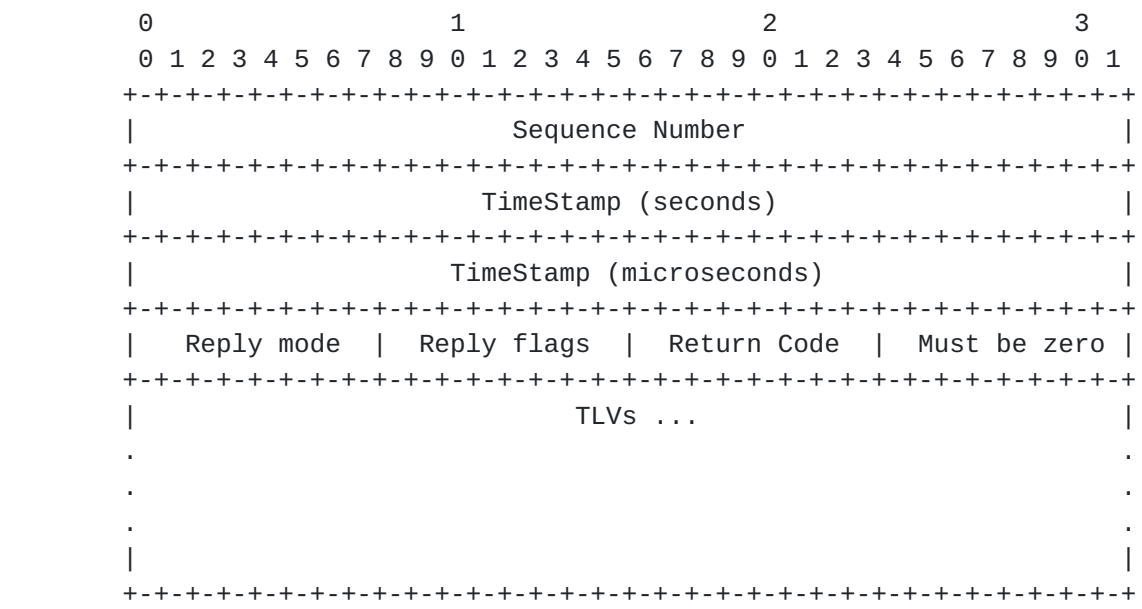
In this document, we describe a mechanism, termed "MPLS ping", that accomplishes these goals. This mechanism is modeled after the ICMP echo request/reply, used by ping and traceroute to detect and localize faults in IP networks. This document also offers some alternative methods for replying to MPLS echo requests.

The basic idea is to test that packets that belong to a particular Forwarding Equivalence Class (FEC) actually end their MPLS path on an LSR that is an egress for that FEC. Therefore, an MPLS echo request carries information about the FEC whose MPLS path is being verified. This echo request is forwarded just like any other packet belonging to that FEC. In "ping" mode (basic connectivity check), the packet should reach the end of the path, at which point it is sent to the control plane of the egress LSR, which then verifies that it is indeed an egress for the FEC. In "traceroute" mode (fault isolation), the packet is sent to the control plane of each transit LSR, which performs various checks that it is indeed a transit LSR for this path; this LSR also returns further information that helps check the control plane against the data plane, i.e., that forwarding matches what the routing protocols determined as the path.

One way these tools can be used is to periodically ping a FEC to ensure connectivity. If the ping fails, one can then initiate a traceroute to determine where the fault lies. One can also periodically traceroute FECs to verify that forwarding matches the control plane; however, this places a greater burden on transit LSRs and thus should be used with caution.

3. Packet Format

An MPLS ping packet is a (possibly labelled) UDP packet with the following payload format:



The Sequence Number is assigned by the sender of the MPLS echo request, and can be used to detect missed replies (for example).

The TimeStamp is set to the time of day (in seconds and microseconds) when the MPLS echo request or reply is sent, and may be used to compute delay or round trip time (for example).

The Reply Mode can take one of the following values:

Value	Meaning
1	Reply via an IPv4 UDP packet
2	Reply via an IPv4 UDP packet with Router Alert
3	Reply via the control plane

Reply Flags are a bit vector with bit 0x1 being the Least Significant Bit and bit 0x80 being the Most Significant Bit; the following bits are defined:

Bit	Meaning when set
0x1	Downstream Mappings desired
0x2	Upstream direction pinged

Bit 0x2 is set when the reverse (upstream) direction of a bidirection

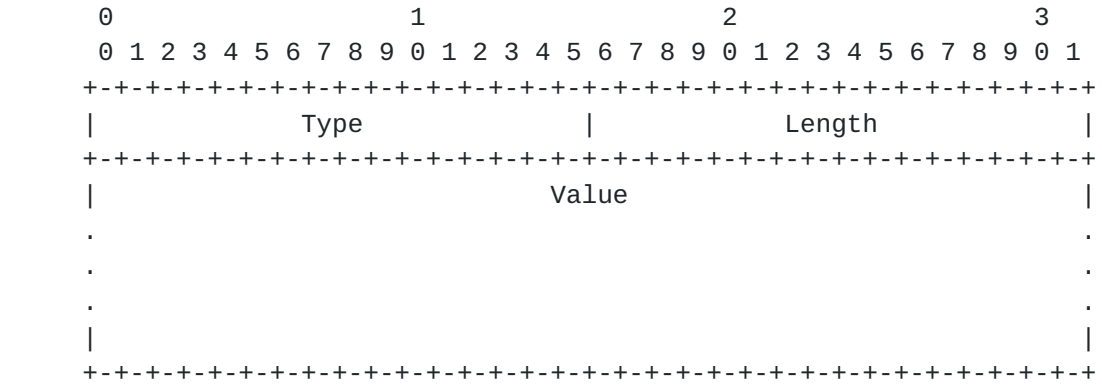
LSP is being tested. The details of the procedures in this case will be given in a later version.

The rest of the bits are reserved and must be zero.

The Return code can take one of the following values:

Value	Meaning
1	Replying router is an egress for the FEC
2	Replying router has no mapping for the FEC
3	Replying router is not one of the "Downstream Routers".
4	Replying router is one of the "Downstream Routers", and its mapping for this FEC on the received interface is the given label
5	Replying router is one of the "Downstream Routers", but its mapping for this FEC is not the given label

TLVs (Type-Length-Value tuples) have the following format:



Types are defined below; Length is the length of the Value field in octets. The Value field depends on the Type; it is zero padded to align to a four-octet boundary.

Type #	Value Field
1	Target FEC Stack
2	Downstream Mapping

3.1. Target FEC Stack

A Target FEC Stack is a list of sub-TLVs. The number of elements is determined by the looking at the sub-TLV length fields.

Type #	Length	Value Field

1	5	IPv4 prefix
2	17	IPv6 prefix
3	16	RSVP IPv4 Session
4	52	RSVP IPv6 Session
5	6	CR-LDP LSP ID
6	13	VPN IPv4 prefix
7	25	VPN IPv4 prefix
8	??	L2 VPN "prefix"

Other FEC Stack Types will be defined as needed.

Note that this TLV defines a stack of FECs, the first FEC element corresponding to the top of the label stack, etc. However, we will assume for now that the stack consists of just one element. Also, only the formats for FEC Types 1-5 will be described in this version.

3.1.1. IPv4 Prefix

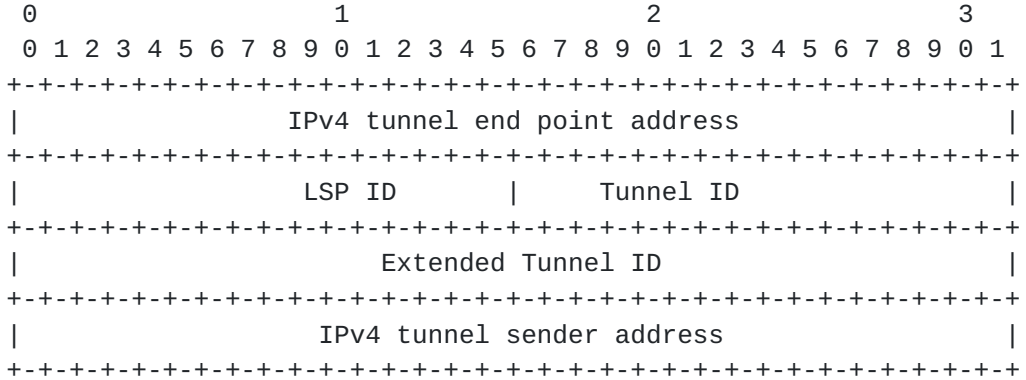
The value consists of four octets of an IPv4 prefix followed by one octet of prefix length in bits. The IPv4 prefix is in network byte order.

3.1.2. IPv6 Prefix

The value consists of sixteen octets of an IPv6 prefix followed by one octet of prefix length in bits. The IPv6 prefix is in network byte order.

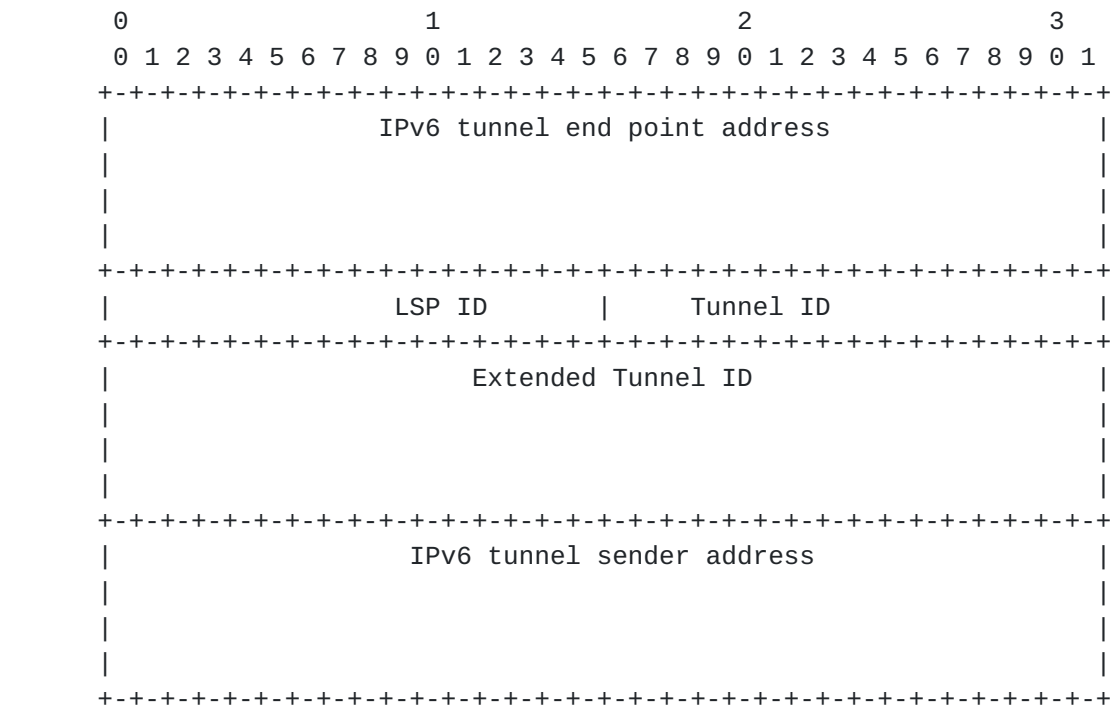
3.1.3. RSVP IPv4 Session

The value has the format below. The value fields are taken from [RFC3209, sections 4.6.1.1 and 4.6.2.1]



3.1.4. RSVP IPv6 Session

The value has the format below. The value fields are taken from [RFC3209, sections 4.6.1.2 and 4.6.2.2]

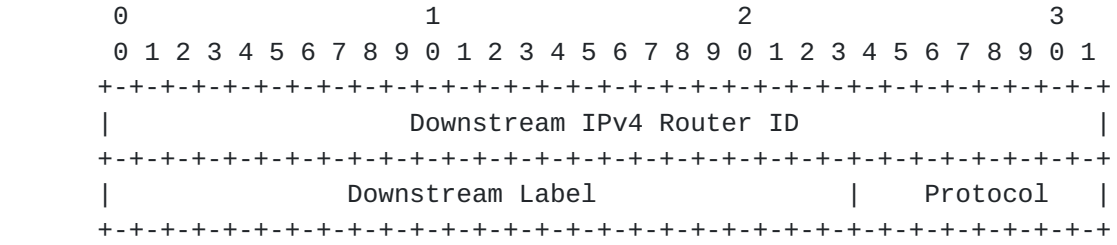


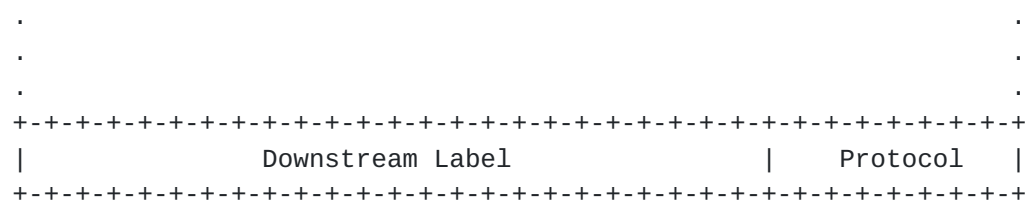
3.1.5. CR-LDP LSP ID

The value consists of the LSPID of the LSP being pinged. An LSPID is a four octet IPv4 address (a local address on the head end LSR) plus a two octet identifier that is unique for each LSP that starts on an LSR.

3.2. Downstream Mapping

The Downstream Mapping is an optional TLV in an echo request. The Length is 4 + 4*N, where N is the number of Downstream Labels. The Value of a Downstream Mapping has the following format:





'Protocol' is taken from the following table:

Protocol #	Signaling Protocol
0	Unknown
1	Static
2	BGP
3	LDP
4	RSVP-TE
5	CR-LDP

The notion of "downstream router" should be explained. Consider an LSR X. If a packet with outermost label L and TTL n>1 arrived at X on interface I, X must be able to compute which LSRs could receive the packet with TTL=n, and what label they would see. (It is outside the scope of this document to specify how this computation may be done.) The set of these LSRs are the downstream routers (and their corresponding labels) for X with respect to L.

The case where X is the LSR originating the echo request is a special case. X needs to figure out what LSRs would receive a labelled packet with TTL=1 when X tries to send a packet to the FEC Stack that is being pinged.

4. Theory of Operation

4.1. MPLS Echo Request

An MPLS echo request is a labeled UDP packet sent to the well-known port for MPLS ping [UDP port # 3503 assigned by IANA], with destination IP address set to the ALL-ROUTERS multicast address (224.0.0.2). The source IP address is set to a routable address of the sender; the source port identifies the sending process. The IP TTL in the UDP packet is set to 1.

An MPLS echo request MUST have a FEC Stack TLV. Also, the Reply Mode must be set to the desired reply mode; the Return Code is set to zero and ignored on receipt.

In "ping" mode (end-to-end connectivity check), the TTL in the

outermost label is set to 255.

In "traceroute" mode (fault isolation mode), the TTL is set successively to 1, 2, ...

In the "traceroute" mode, the echo request SHOULD contain one or more Downstream Mapping TLVs. For TTL=1, all the downstream routers (and corresponding labels) for the sender with respect to the FEC Stack being pinged SHOULD be sent in the echo request. For n>1, the Downstream Mapping TLVs from the echo reply for TTL=(n-1) are copied to the echo request with TTL=n.

4.2. MPLS Echo Reply

An MPLS echo reply is a UDP packet. It MUST ONLY be sent in response to an MPLS echo request. The source IP address is the Router ID of the replier; the source port is the well-known UDP port for MPLS ping. The destination IP address and UDP port are copied from the source IP address and UDP port of the echo request. The IP TTL is set to 255. If the Reply Mode in the echo request is "Reply via an IPv4 UDP packet with Router Alert", then the IP header MUST contain the Router Alert IP option.

The format of the echo reply is the same as the echo request. The Sequence Number is copied from the echo request; the TimeStamp is set to the time-of-day that the echo request is received (note that this information is most useful if the time-of-day clocks on the requestor and the replier are synchronized). The FEC Stack TLV from the echo request is copied to the reply.

The replier MUST fill in the Return Code. This is set based on whether the replier has a mapping for the FEC, and whether it is an egress for that FEC. Note that 'having a mapping' for an RSVP FEC means that the replier is a transit LSR for the RSVP LSP defined by the FEC.

If the echo request contains a Downstream Mapping TLV, the replier MUST further check whether its Router ID matches one of the Downstream IPv4 Router IDs; and if so, whether the given Downstream Label is in fact the label that the replier sent as its mapping for the FEC. For an RSVP FEC, the downstream label is the label that the replier sent in its Resv message. The result of these checks are captured in the Return Code.

If the flag requesting Downstream Mapping TLVs is set in the Reply Flags, the replier SHOULD compute its downstream routers and corresponding labels for the incoming label, and add Downstream

Mapping TLVs for each one to the echo reply it sends back.

4.3. Non-compliant Routers

If the egress for the FEC Stack being pinged does not support MPLS ping, then no reply will be sent, resulting in possible "false negatives". If in "traceroute" mode, a transit LSR does not support MPLS ping, then no reply will be forthcoming from that LSR for some TTL, say n. The LSR originating the echo request SHOULD try sending the echo request with TTL=n+1, n+2, ..., n+k in the hope that some transit LSR further downstream may support MPLS echo requests and reply. In such a case, the echo request for TTL>n MUST NOT have Downstream Mapping TLVs, until a reply is received with a Downstream Mapping.

5. Reliable Reply Path

One of the issues that are faced with MPLS ping is to distinguish between a failure in the forward path (the MPLS path being 'pinged') and a failure in the return path. Note that this problem exists with vanilla IP ping as well. In the case of MPLS ping, it is assumed that the IP control and data planes are reliable. However, it could be that the forwarding in the return path is via an MPLS LSP.

In this specification, we give two solutions for this problem. One is to set the Router Alert option in the MPLS echo reply. When a router sees this option, it MUST forward the packet as an IP packet. Note that this may not work if some transit LSR does not support MPLS ping.

Another option is to send the echo reply via the control plane. At present, this is defined only for RSVP-TE LSPs, and described below.

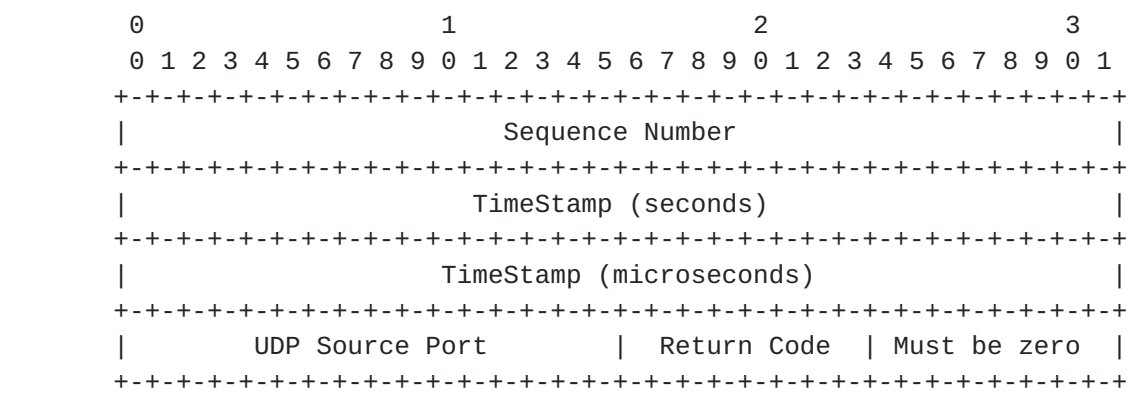
These options are controlled by the ingress LSR, using the Reply Mode in the MPLS echo request packet.

5.1. RSVP-TE Extension

To test an LSP's liveliness, an ingress LSR sends MPLS echo requests over the LSP being tested. When an egress LSR receives the message, it needs to acknowledge the ingress LSR by sending an LSP_ECHO object in a RSVP Resv message. The object has the following format:

Class = LSP_ECHO (use form 11bbbbbb for compatibility)

C-Type = 1



The Sequence Number is copied from the Sequence Number of the echo request. The TimeStamp is set to the time the echo request is received. The UDP Source Port is copied from the UDP source port of the MPLS echo request. The FEC is implied by the Session and the Sender Template Objects.

5.2. Operation

For the sake of brevity in the context of this document by "the control plane" we mean "the RSVP-TE component of the control plane".

Consider an LSP between an ingress LSR and an egress LSR spanning multiple LSR hops.

5.3. Procedures at the ingress LSR

One must ensure before setting the Reply Mode to "reply via the control plane" that the egress LSR supports this feature.

The ingress LSR, say X, selects a unique UDP source port for its MPLS ping. X also sets the FEC Stack TLV Type to RSVP IPv4 (IPv6), and copies the SESSION and SENDER_TEMPLATE into the appropriate fields of the value field. Finally, X sets the Reply Mode to "reply via the control plane".

If X does not receive an Resv message from the egress LSR that contains an LSP_ECHO object within some period of time, it declares the LSP as "down". At this point, the ingress LSR may apply the necessary procedures to fix the LSP. These may include generating a message to network management, tearing-down and re-building the LSP, and/or rerouting user traffic to a backup LSP.

To test an LSP that carries non-IP traffic, before injecting ICMP and MPLS ping messages into the LSP, the IPv4 Explicit NULL label should be prepended to such messages. The ingress and egress LSR's must follow the procedures defined in [LABEL-STACKING].

5.4. Procedures at the egress LSR

When the egress LSR receives an MPLS ping message, it follows the procedures given above. If the Reply Mode is set to "Reply via the control plane", the LSR can, based on the RSVP SESSION and SENDER_TEMPLATE objects carried in the MPLS ping message, find the corresponding LSP in its RSVP-TE database. The LSR then checks to see if the Resv message for this LSP contains an LSP_ECHO object with the same source UDP port value. If not, the LSR adds or updates the LSP_ECHO object and refreshes the Resv message.

5.5. Procedures for the intermediate LSR's

At intermediate LSRs, normal RSVP processing procedures will cause the LSP_ECHO object to be forwarded as RSVP messages are refreshed.

At the LSR's that support MPLS ping the Resv messages that carry the LSP_ECHO object MUST be delivered upstream immediately.

Note that an intermediate LSR using RSVP refresh reduction [RSVP-REFRESH], the new or changed LSP_ECHO object will cause the LSR to classify the RSVP message as a trigger message.

6. Security Considerations

The security considerations pertaining to the original RSVP protocol remain relevant.

7. Intellectual Property Considerations

Juniper Networks, Inc. is seeking patent protection on technology described in this Internet-Draft. If technology in this Internet-Draft is adopted as a standard, Juniper Networks agrees to license, on reasonable and non-discriminatory terms, any patent rights it obtains covering such technology to the extent necessary to comply with the standard.

8. Acknowledgments

This is the outcome of many discussions among many people, that also include Manoj Leelanivas, Paul Traina, Yakov Rekhter, Der-Hwa Gan, Brook Bailey and Eric Rosen.

9. References

[ICMP] J. Postel, "Internet Control Message Protocol", RFC792.

[RSVP] R. Braden, Ed., et al, "Resource ReSerVation protocol (RSVP) -- version 1 functional specification," RFC2205.

[RSVP-TE] D. Awduche, et al, "RSVP-TE: Extensions to RSVP for LSP tunnels" Internet Draft.

[LABEL-STACKING] E. Rosen, et al, "MPLS Label Stack Encoding", RFC3032.

[RSVP-REFRESH] L. Berger, et al, "RSVP Refresh Overhead Reduction Extensions", RFC2961.

[RFC-IANA] T. Narten and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 2434.

10. Author Information

Kireeti Kompella
Ping Pan
Nischal Sheth
Juniper Networks
1194 N.Mathilda Ave
Sunnyvale, CA 94089
e-mail: kireeti@juniper.net
e-mail: pingpan@juniper.net
e-mail: nsheth@juniper.net
phone: 408.745.2000

Dave Cooper
Global Crossing
960 Hamlin Court
Sunnyvale, CA 94089
email: dcooper@gblx.net
phone: 916.415.0437

George Swallow
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
e-mail: swallow@cisco.com
phone: 978.244.8143

Sanjay Wadhwa
Unisphere Networks, Inc.
10 Technology Park Drive
Westford, MA 01886-3146
email: swadhwa@unispherenetworks.com
phone: 978.589.0697

Ronald P. Bonica
WorldCom
22001 Loudoun County Pkwy
Ashburn, Virginia, 20147
email: ronald.p.bonica@wcom.com
phone: 703.886.1681