Network Working Group
Internet Draft
Category: Standards Track
Expires: June 2004

Adrian Farrel (Editor)
Old Dog Consulting

Dimitri Papadimitriou
Alcatel

Jean-Philippe Vasseur
Cisco Systems, Inc.

Arthi Ayyangar
Juniper Networks

December 2003

# Encoding of Attributes for Multiprotocol Label Switching (MPLS) Label Switched Path (LSP) Establishment Using RSVP-TE

draft-ietf-mpls-rsvpte-attributes-01.txt

## Status of this Memo



## Abstract

Multiprotocol Label Switching (MPLS) Label Switched Paths (LSPs) may be established using the Resource Reservation Protocol Traffic Engineering extensions (RSVP-TE). This protocol includes an object (the SESSION_ATTRIBUTE object) which carries a flags field used to indicate options and attributes of the LSP. That flags field has eight bits allowing for eight options to be set.

Recent proposals in many documents that extend RSVP-TE for signaling additional features and function for MPLS LSPs have suggested uses

for each of the previously unused bits.

This document defines a new object for RSVP-TE messages that allows the signaling of further attribute bits and also the carriage of arbitrary attribute parameters to make RSVP-TE easily extensible to support new requirements. Additionally, this document defines a way to record the attributes applied to the LSP on a hop-by-hop basis.





## 0. Change History

This section to be removed before publication.

### 0.1 Changes to 01 Version

- Change Attributes Flags TLV to be variable length so that more bits can easily be added in the future.
- Define default behaviors for bits absent from the TLV and for absence of the TLV.
- Clarify the IANA requirements for tracking Attributes Flags bits.
- Introduce RRO Attibutes Subobject and describe usage.
- Move Fast Reroute reference to informational.
- Update security considerations to handle new RRO subobject
- Remove section that explained the need for this document in advance of any definitive bit definitions.
- Tighten rules for processing LSP_ATTRIBUTES object in cases where TLVs are unknown or unsupported.
- Clarify that LSP Attributes apply to individual LSPs and not to entire sessions.

## 1. Introduction and Problem Statement

Multiprotocol Label Switching (MPLS) Label Switched Paths (LSPs) [RFC3031] may be established using the RSVP-TE signaling protocol [RFC3209]. This protocol uses the Path message to request that a LSP be set up. The Path message includes the SESSION_ATTRIBUTE object which carries a flags field used to indicate desired options and attributes of the LSP.

The flags field in the SESSION_ATTRIBUTE object has eight bits. Just three of those bits are assigned in [RFC3209]. A further two bits are assigned in [FRR] for fast re-reroute functionality leaving only three bits available. Several recent proposals and Internet Drafts have demonstrated that there is a high demand for the use of the other three bits. Some, if not all, of those proposals are likely to go forward as RFCs resulting in depletion or near depletion of the flags field and a consequent difficulty in signaling new options and attributes that may be developed in the future.

This document defines a new object for RSVP-TE messages that allows the signaling of further attributes bits. The new object is constructed from TLVs, and a new TLV is defined to carry a variable number of attributes bits. Because of the nature of the TLV construction the object is flexible and allows the future definition of:

- further sets of bits flags if further, distinct uses are discovered
- arbitrary options and attributes parameters carried as individual TLVs.

Note that the LSP Attributes defined in this document are specifically scoped to an LSP. They may be set differently on separate LSPs within the same Session.

It is noted that that some options and attributes do not need to be acted on by all Label Switched Routers (LSRs) along the path of the LSP. In particular, these options and attributes may apply only to





key LSRs on the path such as the ingress and egress. Special transit LSRs, such as AS Border Routers (ASBRs) may also fall into this category. This means that the new options and attributes should be signaled transparently, and only examined at those points that need to act on them.

On the other hand, other options and attributes may require action at all transit LSRs along the path of the LSP. Inability to support the required attributes by one of those transit LSRs may require the LSR to refuse the establishment of the LSP.

These considerations are particularly important in the context backwards compatibility. In general, it should be possible to provide new MPLS services across a legacy network without upgrading those LSRs that do not need to participate actively in the new services.

RSVP includes a way for unrecognized objects to be forwarded by transit nodes without them refusing the protocol message and with the objects being stripped from the protocol message (see [RFC2205] section 3.10). This extends to RSVP-TE and provides a good way to ensure that only those LSRs that understand a particular object examine it.

This document distinguishes between options and attributes that are only required at key LSRs along the path of the LSP, and those that must be acted on by every LSR along the LSP. Two LSP Attributes objects are defined in this document: the first may be passed transparently by LSRs that do not recognize it, the second must cause LSP setup failure with the generation of a PathErr message if an LSR

does not recognize it.

Comments on this document should be made direct to the MPLS mailing list at mpls@uu.net.

### 1.1 Applicability to Generalized MPLS

The RSVP-TE signaling protocol also forms the basis of a signaling protocol for Generalized MPLS (GMPLS) as described in [RFC3471] and [RFC3473]. The extensions described in this document are intended to be equally applicable to MPLS and GMPLS.

### 1.2 A Rejected Alternate Solution

A rejected alternate solution was to define a new C-Type for the existing SESSION_ATTRIBUTE object. This new C-Type could allow a larger Flags field and address the immediate problem.

This solution was rejected because:

- A C-Type is not backward compatible with deployed implementations that expect to see a C-Type of 1 or 7. It is important that any solution be capable of carrying new attributes transparently across legacy LSRs if those LSRs are not required to act on the attributes.



- Support for arbitrary attributes parameters through TLVs would have meant a significant change of substance to the existing object.

## 2. Terminology

This document uses terminology from the MPLS architecture document [RFC3031] and from the RSVP-TE protocol specification [RFC3209] which inherits from the RSVP specification [RFC2205].

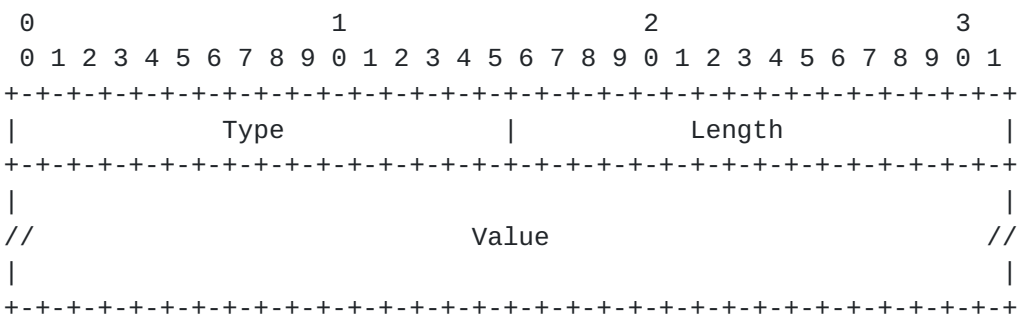
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119 [6].

## 3. Attributes TLVs

Attributes carried by the new objects defined in this document are encoded within TLVs. One or more TLVs may be present in each object. There are no ordering rules for TLVs and no interpretation should be

placed on the order in which TLVs are received.

Each TLV is encoded as follows.




Type

The identifier of the TLV.

Length

The length of the value field in bytes. Thus if no value field is present the length field contains the value zero. Each value field must be zero padded at the end to take it up to a four byte boundary - the padding is not included in the length so that a one byte value would be encoded in an eight byte TLV with length field set to one.

Value

The data for the TLV padded as described above.

## 3.1 Attributes Flags TLV

This document defines only one TLV type value. Type 1 indicates the Attributes Flags TLV. Other TLV types may be defined in future with type values assigned by IANA.



The Attibutes Flags TLV may be present in an LSP_ATTRIBUTES object and/or an LSP_REQUIRED_ATTRIBUTES object. The bits in the TLV represent the same attributes regardeless of which object carries the TLV. Documents that define individual bits MUST specify whether the bit may be set in one object or the other, or both.

The Attributes Flags TLV value field is a variable length array of flags numbered from the MSB as bit zero. The length field for this TLV is always a multiple of 4 bytes, regardless of the number bits

carried.

Unassigned bits are considered as reserved and MUST be set to zero on transmission and ignored on receipt. Bits not contained in the TLV MUST be assumed to be set to zero. If the TLV is absent either because it is not contained in the LSP_ATTRIBUTES or LSP_REQUIRED_ ATTRIBUTES object, or because those objects are themselves absent, all processing MUST be performed as though the bits were present and set to zero.

No bits are defined in this document. The assignment of bits is managed by IANA.

## 4. LSP_ATTRIBUTES Object

The LSP_ATTRIBUTES object is used to signal attributes required in support of an LSP, or to indicate the nature or use of an LSP where that information is not required to be acted on by all transit LSRs. Specifically, if an LSR does not support the object, it forwards it unexamined and unchanged. This facilitates the exchange of attributes across legacy networks that do not support this new object.

This object effectively extends the flags field in the SESSION_ ATTRIBUTE object and allows for the future inclusion of more complex objects through TLVs.

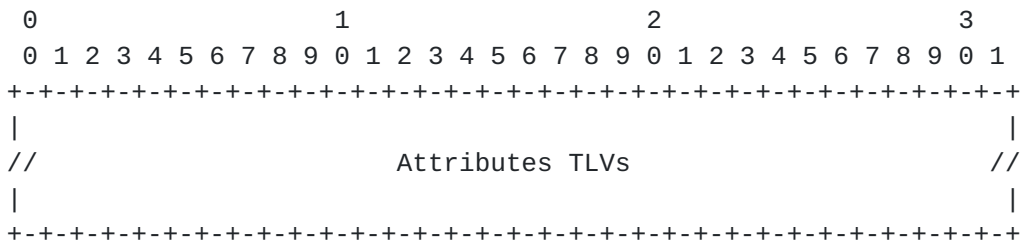
The LSP_ATTRIBUTES object class is TBD of the form 11bbbbbb. This C-Num value (see section 7) ensures that LSRs that do not recognize the object pass it on transparently.

One C-Type is defined, C-Type = 1 for LSP Attributes.

This object is optional and may be placed on Path messages to convey additional information about the desired attributes of the LSP.

### 4.1 Format

LSP_ATTRIBUTES class = TBD, C-Type = 1




The Attributes TLVs are encoded as described in section 3.

### 4.2 Generic Processing Rules

An LSR that does not support this object will pass it on unaltered because of the C-Num.

An LSR that does support this object, but does not recognize a TLV type code carried in this object MUST pass the TLV on un-altered in the LSP_ATTRIBUTES object that it places in the Path message that it sends downstream.

An LSR that does support this object and recognizes a TLV but does not support the attribute defined by the TLV MUST act as specified in the document that defines the TLV.

An LSR that supports the Attributes Flags TLV, but does not recognize a bit set in the Attributes Flags TLV MUST forward the TLV unchanged.

An LSR that supports the Attributes Flags TLV and recognizes a bit that is set but does not support the indicated attribute MUST act as specified in the document that defines the bit.

## 5. LSP_REQUIRED_ATTRIBUTES Object

The LSP_REQUIRED_ATTRIBUTES object is used to signal attributes required in support of a LSP, or to indicate the nature or use of a LSP where that information MUST be inspected at each transit LSR. Specifically, each transit LSR MUST examine the attributes in the LSP_REQUIRED_ATTRIBUTES object and MUST NOT forward the object transparently.

This object effectively extends the flags field in the SESSION_ ATTRIBUTE object and allows for the future inclusion of more complex objects through TLVs. It complements the LSP_ATTRIBUTES object.

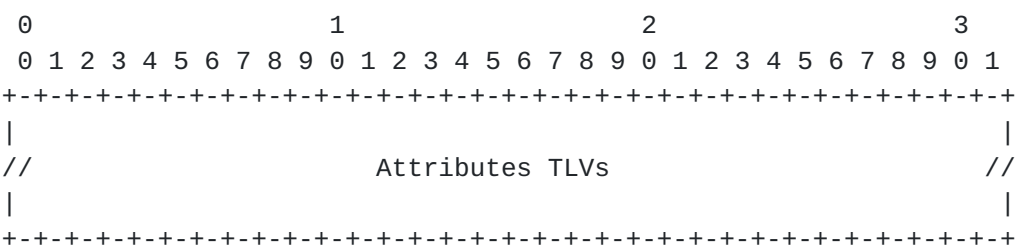
The LSP_REQUIRED_ATTRIBUTES object class is TBD of the form 0bbbbbbb. This C-Num value (see section 7) ensures that LSRs that do not recognize the object reject the LSP setup effectively saying that they do not support the attributes requested. This means that this object should only be used for attributes that require support at some transit LSRs and so require examination at all transit LSRs. See section 4 for how end-to-end and selective attributes are signaled.

One C-Type is defined, C-Type = 1 for LSP Required Attributes.

This object is optional and may be placed on Path messages to convey additional information about the desired attributes of the LSP.

### 5.1 Format

LSP_REQUIRED_ATTRIBUTES class = TBD, C-Type = 1




The Attributes TLVs are encoded as described in section 3.

### 5.2 Generic Processing Rules

An LSR that does not support this object will use a PathErr to reject the Path message based on the C-Num using the error code "Unknown Object Class".

An LSR that does not recognize a TLV type code carried in this object MUST reject the Path message using a PathErr with Error Code "Unknown Attributes TLV" and Error Value set to the value of the unknown TLV type code.

An LSR that does not recognize a bit set in the Attributes Flags TLV MUST reject the Path message using a PathErr with Error Code "Unknown Attributes Bit" and Error Value set to the bit number of the unknown bit in the Attributes Flags.

An LSR that recognizes an attribute, however encoded, but which does not support that attribute MUST act according to the behavior specified in the document that defines that specific attribute.

## 6. Recording Attributes

### 6.1 Requirements

In some circumstances it is useful to determine which of the requested LSP attributes have been applied at which LSRs along the path of the LSP. For example, an attribute may be requested in the

LSP_ATTRIBUTES object such that LSRs that do not support the object are not required to support the attribute or provide the requested function. In this case, it may be useful to the ingress LSR to know which LSRs acted on the request and which ignored it.

Additionally, there may be other qualities that need to be reported on a hop-by-hop basis. These are currently indicated in the Flags field of RRO subobjects. Since there are only eight bits available in this field, and since some are already assigned and there is also likely to an increase in allocations in new documents, there is a need for some other method to report per-hop attributes.

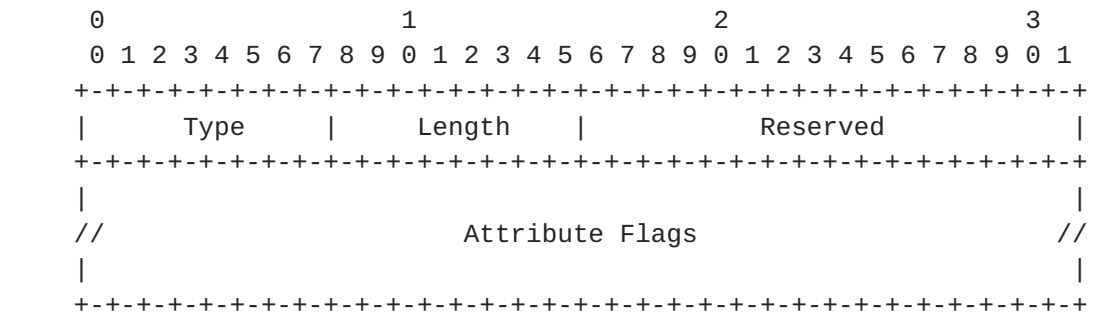


### 6.2 RRO Attributes Subobject

The RRO Attributes Subobject may be carried in the RECORD_ROUTE object if it is present. The subobject uses the standard format of an RRO subobject.

The length is variable as for the Attributes Flags TLV. The content is the same as the Attribute Flags TLV - that is, it is a series of bit flags.

There is a one-to-one correspondance between bits in the Attributes Flags TLV and the RRO Attributes Subobject. If a bit is only required in one of the two places, it is reserved in the other place. See the procedures sections, below, for more information.




Type

0x?? TBD RRO Attribute Subobject

Length

The Length contains the total length of the subobject in bytes, including the Type and Length fields. This length must be a multiple of 4 and must be at least 8.

Attribute Flags

The attribute flags recorded for the specific hop.

## 6.3 Procedures

### 6.3.1 Subobject Presence Rules

The Attributes subobject is pushed onto the RECORD_ROUTE object immediately prior to pushing on the node's IP address. Thus, if label recording is being used, the Attributes subobject SHOULD be pushed onto the RECORD_ROUTE object after the Record Label subobject(s).

A node MUST NOT push on a Attributes subobject without also pushing on an IPv4 or IPv6 subobject.

This means that an Attributes subobject is bound to the LSR identified by the IP address subobject found in the RRO immediately before the Attributes subobject.



If the newly added subobject causes the RRO to be too big to fit in a Path (or Resv) message, the processing SHALL be as described in [RFC3209].

If more than one Attributes subobject is found between a pair of subobjects that identify LSRs, only the first one found SHALL have any meaning within the context of this document. All such subobjects shall be forwarded unmodified by transit LSRs.

### 6.3.2 Reporting Compliance with LSP Attributes

To report compliance with an attribute requested in the Attributes Flags TLV, an LSR MAY set the corresponding bit (see section 7) in the Attributes subobject. To report non-compliance, an LSR MAY clear the corresponding bit in the Attributes subobject.

The requirement to report compliance MUST be specified in the document that defines the usage of any bit. This will reduce to a statement of whether hop-by-hop acknowledgement is required.

### 6.3.3 Reporting Per-Hop Attributes

To report a per-hop attribute, an LSR sets the appropriate bit in the Attributes subobject.

The requirement to report a per-hop attribute MUST be specified in the document that defines the usage of the bit.

#### 6.3.4 Default Behavior

By default all bits in an Attibutes subobject SHOULD be set to zero.

If an Attribute subobject is not long enough to include a specific bit, the bit MUST be assumed to be set to zero.

If the RRO subobject is not present for a hop in the LSP, all bits MUST be assumed to be set to zero.

## 7. Summary of Attribute Bit Allocation

This document defines two uses of per-LSP attribute flag bit fields. The bit numbering in the Attributes Flags TLV and the RRO Attributes subobject is identical. That is the same attribute is indicated by the same bit in both places. This means that only a single registry of bits is maintained.

The consequence is a degree of clarity in implementation and registration.

Note, however, that it is not always the case that a bit will be used in both the Attributes Flags TLV and the RRO Attributes subobject. For example, an attribute may be requested using the Attributes Flags TLV, but there is no requirement to report the handling of the attribute on a hop-by-hop basis. Conversely, there may be a requirement to report the attributes of an LSP on a hop-by-hop basis, but there is no corresponding request attribute.



In these cases, a single bit number is still assigned for both the Attributes Flags TLV and the RRO Attributes subobject. The document that defines the usage of the new bit MUST state in which places it is used and MUST handle a default setting of zero.

## 8. Message Formats

The LSP_ATTRIBUTES object and the LSP_REQUIRED_ATTRIBUTES object MAY be carried in a Path message.

The order of objects in RSVP-TE messages is recommended, but

implementations must be capable of receiving the objects in any meaningful order. The LSP_ATTRIBUTES object and LSP_REQUIRED_ ATTRIBUTES objects are RECOMMENDED to be placed immediately after the SESSION_ATTRIBUTE object if it is present, or otherwise immediately after the LABEL_REQUEST object.

If both the LSP_ATTRIBUTES object and the LSP_REQUIRED_ATTRIBUTES object are present, the LSP_REQUIRED_ATTRIBUTES object is RECOMMENDED to be placed first.

LSRs should be prepared to receive these objects in any order in any position within a Path message. Subsequent instances of these objects within a Path message SHOULD be ignored.

## 9. IANA Considerations

### 9.1 New RSVP C-Nums and C-Types

Two new RSVP C-Nums are defined in this document and should be assigned by IANA.

o LSP_ATTRIBUTES object

The C-Num should be of the form 11bbbbbb so that LSRs that do not recognize the object will ignore the object but forward it, unexamined and unmodified, in all messages resulting from this message.

One C-Type is defined for this object and should be assigned by IANA.

o LSP Attributes TLVs

Recommended C-Type value 1.

o LSP_REQUIRED_ATTRIBUTES object

The C-Num should be of the form 0bbbbbbb so that LSRs that do not recognize the object will reject the message that carries it with an "Unknown Object Class" error.



One C-Type is defined for this object and should be assigned by IANA.

o LSP Required Attributes TLVs

Recommended C-Type value 1.

## 9.2 New TLV Space

The two new objects referenced above are constructed from TLVs. Each TLV includes a 16-bit type identifier (the T-field). The same T-field values are applicable to both objects.

IANA is requested to manage TLV type identifiers as follows:

- TLV Type
- TLV Name
- Whether allowed on LSP_ATTRIBUTES object
- Whether allowed on LSP_REQUIRED_ATTRIBUTES object.

This document defines one TLV type as follows:
- TLV Type = 1
- TLV Name = Attributes Flags TLV
- allowed on LSP_ATTRIBUTES object
- allowed on LSP_REQUIRED_ATTRIBUTES object.

## 9.3 Attributes Flags

This document provides new attributes bit flags for use in other documents that specify new RSVP-TE attributes. These flags are present in the Attributes Flags TLV referenced in the previous section.

IANA is requested to manage the space of attributes bit flags numbering them in the usual IETF notation starting at zero and continuing through 2039.

Each bit should be tracked with the following qualities:
- Bit number
- Defining RFC
- Name of bit
- Whether there is meaning in the Attibute Flags TLV (yes/no)
- Whether there is meaning in the RRO Attributes Subobject (yes/no).

Note that this means that all bits in the Attribute Flags TLV and the RRO Attributes Subobject use the same bit number regardless of whether they are used in one or both places. Thus, only one list of bits is required to be maintained.

## 9.4 SESSION_ATTRIBUTE Flags Field

This document does not make any alterations to the definition of the existing SESSION_ATTRIBUTE object nor to the definition of meanings assigned to the flags in the Flags field of that object. These flags

are assigned meanings in various other RFCs and Internet Drafts.



It is suggested that IANA manage the allocation of meaning to the bits in the Flags field of the SESSION_ATTRIBUTE object to prevent accidental double allocation of any one bit.

### 9.5 New Error Codes

This document defines the following new error codes and error values. Numeric values should be assigned by IANA.

| Error Code | Error Value |
|---|---|
| "Unknown Attributes TLV" | Identifies the unknown TLV type code. |
| "Unknown Attributes Bit" | Identifies the unknown Attribute Bit. |

### 9.6 New Record Route Subobject Identifier

A new subobject is defined for inclusion in the RECORD_ROUTE object.

The RRO Attributes subobject is identified by a Type value of TBD.

## 10. Security Considerations

This document adds two new objects to the RSVP Path message as used in MPLS and GMPLS signaling, and a new subobject to the RECORD_ROUTE object carried on may RSVP messages. It does not introduce any new direct security issues and the reader is referred to the security considerations expressed in [RFC2205], [RFC3209] and [RFC3473].

It is of passing note that any signaling request that indicates the functional preferences or attributes of an MPLS LSP may provide anyone with unauthorized access to the contents of the message with information about the LSP that an administrator may wish to keep secret. Although this document adds new objects for signaling desired LSP attributes, it does not contribute to this issue which can only be satisfactorily handled by encrypting the content of the signaling message.

Similarly, the addition of attribute recording information to the RRO may reveal information about the status of the LSP and the capabilities of individual LSRs that operators wish to keep secret. The same strategy that applies to other RRO subobjects also applies here. Note, however, that there is a tension between notifying the head end of the LSP status at transit LSRs, and hiding the existence or identity of the transit LSRs.

## 11. Acknowledgements

Credit to the OSPF Working Group for inspiration from their solution to a similar problem.

Thanks to Rahul Aggarwal for his careful review and support of this work. Thanks also to Raymond Zhang, Kireeti Kompella and Philip Matthews for their input.



## 12. Intellectual Property Consideration



## 13. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2205] Braden, R. (Ed.), Zhang, L., Berson, S., Herzog, S. and S. Jamin, "Resource ReserVation Protocol -- Version 1 Functional Specification", RFC 2205, September 1997.

[RFC3209] 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.

[RFC3471] Berger, L. (Editor), "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description", RFC 3471, January 2003.

[RFC3473] Berger, L. (Editor), "Generalized MPLS Signaling - RSVP-TE Extensions", RFC 3473 January 2003.

## 14. Informative References

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.

[RFC3031] Rosen, E., Viswanathan, A., and Callon, R., "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.



[INTER-AS] Vasseur, JP., Zhang, R., "Inter-AS MPLS Traffic Engineering", <draft-vasseur-inter-as-te-01.txt>, Internet Draft, work in progress.

[FRR] Pan, P. (Ed.), "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", <draft-ietf-mpls-rsvp-lsp-fastreroute-03.txt>, Internet Draft, work in progress.

[OSPF-CAPS] Lindem, A., Shen, N., Aggarwal, R., Shaffer, S., Vasseur, JP., "Extensions to OSPF for Advertising Optional Router Capabilities", <draft-ietf-ospf-cap-00.txt>, Internet Draft, work in progress.

[REOPT] Vasseur, JP., Ikejiri, Y., "Reoptimization of MPLS Traffic Engineering loosely routed explicit LSP path", <draft-vasseur-mpls-loose-path-reopt-02.txt>, Internet Draft, work in progress.

## 15. Authors' Addresses

Adrian Farrel
Old Dog Consulting
Phone: +44 (0) 1978 860944
EMail: adrian@olddog.co.uk

Dimitri Papadimitriou (Alcatel)
Fr. Wellesplein 1,
B-2018 Antwerpen, Belgium
Phone:  +32 3 240-8491
EMail:  dimitri.papadimitriou@alcatel.be

Jean Philippe Vasseur
Cisco Systems, Inc.
300 Apollo Drive
Chelmsford, MA 01824
300 Beaver Brook Road
Boxborough , MA - 01719
USA
EMail: jpv@cisco.com

Arthi Ayyangar
Juniper Networks, Inc.
1194 N.Mathilda Ave
Sunnyvale, CA 94089
USA
EMail: arthi@juniper.net

## 16. Full Copyright Statement