

November 2002

Exclude Routes - Extension to RSVP-TE
<[draft-lee-ccamp-rsvp-te-exclude-route-01.txt](#)>

1. Status of this memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

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."

To view the list Internet-Draft Shadow Directories, see <http://www.ietf.org/shadow.html>.

2. Abstract

The current RSVP-TE specification [[RSVP-TE](#)] and GMPLS extensions [[GMPLS-RSVP-TE](#)] allow abstract nodes and resources to be explicitly included in a path setup, but not to be explicitly excluded.

In some systems where precise explicit paths are not computed at the head end it may be useful to specify and signal abstract nodes and resources that are to be explicitly excluded from routes. These exclusions may apply to the whole of a path, or to parts of a path between two abstract nodes specified in an explicit route.

Shared Risk Link Groups (SRLGs) allow the definition of resources or groups of resources that share the same risk of failure. The knowledge of SRLGs may be used to compute diverse paths that can be used for protection. In systems where it is useful to signal exclusions, it may be useful to signal SRLGs to indicate groups of resources that should be excluded on the whole of a path or between two abstract nodes specified in an explicit path.

This draft specifies ways to communicate route exclusions during path setup using RSVP-TE.

These approaches are equally applicable to other MPLS TE signaling protocols such as CR-LDP.

3. Conventions used in this document

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](#)].

4. Overview

The current RSVP-TE specification [[RSVP-TE](#)] and GMPLS extensions [[GMPLS-RSVP-TE](#)] allow abstract nodes and resources to be explicitly included in a path setup, using the Explicit Route Object (ERO).

In some systems it may be useful to specify and signal abstract nodes and resources that are to be explicitly excluded from routes. This may be because loose hops or abstract nodes need to be prevented from causing a route through a specific resource. This is a special case of path calculation distribution to nodes within the system.

Two types of exclusions are required:

- i) Do not include any of the abstract nodes in a given set anywhere on the path. This set of abstract nodes to exclude is referred to as the Exclude Route list.
- ii) Do not include certain abstract nodes or resources between a specific pair of abstract nodes present in an ERO. Such specific exclusions are referred to as Explicit Exclusion Route.

A new RSVP-TE object is introduced to convey the Exclude Route list. This object is the Exclude Route Object (XRO).

The second type of exclusion is achieved through a modification to the existing ERO. A new subobject type the Explicit Exclude Route Subobject (EXRS) is introduced to indicate an exclusion between a pair of included abstract nodes.

At the same time, it is recognized that SRLGs are a useful means of indicating resources that share the same risk of failure. When establishing protection LSPs they are often required to be node and link diverse from the LSPs that they protect. Further, where SRLGs are known, the protection LSPs are required to not utilize resources in the SRLGs traversed by the protected LSPs.

Expires June 2003

[Page 2]

This draft introduces an ERO subobject to indicate an SRLG to be signaled in either of the two exclusion methods described above. This subobject might also be appropriate for use within Explicit Routes, but that discussion is outside the scope of this draft.

4.1 Scope of Excluded Routes

This draft does not preclude a route exclusion from listing many nodes or network elements to avoid. The intent is, however, to indicate only the minimal number of subobjects to be avoided. For instance it may be necessary to signal only the SRLGs (or Shared Risk Groups) to avoid.

It is envisaged most of the conventional inclusion subobjects are specified in the ERO of signaling only for the area where they pertain. The number of subobjects to be avoided, specified in the ERO in signaling, may be constant throughout the whole path setup or the subobjects to be avoided may be removed from the ERO as they become irrelevant in the subsequent hops of the path setup. For e.g. let the primary path be (Ingress1 A1,A2,AB1,B1,B2,BC1,C1,C2,Egress1) where Xn denotes a node in Area X and XY1 denotes an ABR (Area Border Router) connected to Area X and Y. Since the ERO for nodes to exclude in Area A are already taken into consideration during CSPF at Ingress1, the ERO for the secondary diverse path signaled at Ingress1 would be (A11,A12,AB2,{B1,B2,BC1,C1,C2}) and at AB2 would be (B11,B12,BC2,{C1,C2}) and at BC2 would be (C11,C12).

In general, consideration should be given (as with explicit route) to the size of signaled data and the impact on the signaling protocol.

4.2 Relationship to MPLS TE MIB

[MPLS-TE-MIB] defines managed objects for managing and modeling MPLS-based traffic engineering. Included in [MPLS-TE-MIB] is a means to configure explicit routes for use on specific LSPs. This configuration allows the exclusion of certain resources.

In systems where the full explicit path is not computed at the ingress (or at a path computation site for use at the ingress) it may be necessary to signal those exclusions. This draft offers a means of doing this signaling.

Expires June 2003

[Page 3]

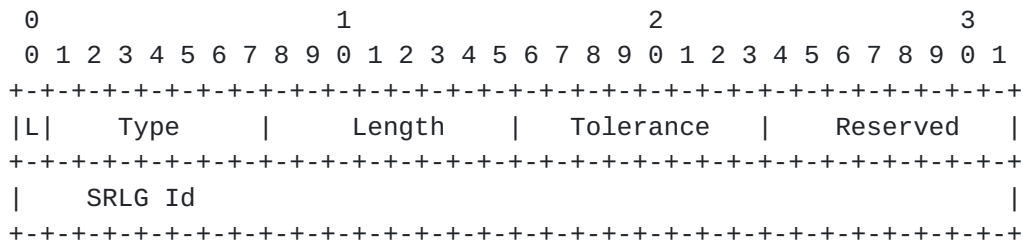
5. Shared Risk Link Groups

The identifier of a SRLG is defined as a 32 bit quantity in [GMPLS-OSPF]. These 32 bits are divided into an 8 bit type field and a 24 bit identifier in [CCAMP-SRLG].

5.1 SRLG ERO Subobject

The format of the ERO and its subobjects are defined in [RSVP-TE].

The SRLG subobject is defined as follows.



L

The L bit is an attribute of the subobject. The L bit is set if the subobject represents a loose hop in the explicit route. If the bit is not set, the subobject represents a strict hop in the explicit route.

For exclusions, the L bit SHOULD be set to zero and ignored.

Type

The type of the subobject [TBD].

Length

The Length contains the total length of the subobject in bytes, including the Type and Length fields. The Length is always 8.

Tolerance

The level to which it is permissible for this SRLG to be included in the path when more than one SRLG is specified. A value of zero indicates that this SRLG MUST be avoided. A tolerance value of $n < m$ indicates that the SRLG MUST be avoided in preference to an SRLG with tolerance value m .

If only one SRLG is present, then a value other than zero indicates the SRLG SHOULD be avoided.

SRLG Id

Expires June 2003

[Page 4]

The 32 bit identifier of the SRLG.

5.2 Exclusion Tolerance Semantics

The Tolerance field in the SRLG subobject indicates the degree to which the SRLG must be avoided. (The degree to which it is permissible to include it.)

If the Tolerance field has the value zero (0), the LSP MUST NOT traverse or use any resource that is a member of the SRLG.

If the value is non-zero, all path computation elements SHOULD attempt to select routes that avoid all resources that are members of the SRLG.

Where more than one SRLG with non-zero Tolerance value is specified for exclusion and no route can be found that avoids both SRLGs, a route SHOULD be chosen that avoids the SRLG with the lower Tolerance value.

6. Exclude Route List

The exclude route identifies a list of abstract nodes that MUST NOT be traversed along the path.

6.1 Exclude Route Object (XRO)

Abstract nodes to be excluded from the path are specified via the EXCLUDE_ROUTE object (XRO). The Exclude Route Class value is [TBD].

Currently one C_Type is defined, Type 1 Exclude Route. The EXCLUDE_ROUTE object has the following format:

Class = TBD, C_Type = 1

```

      0               1               2               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                                                    |
//                               (Subobjects)                               //
|                                                                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

Subobjects

The contents of an EXCLUDE_ROUTE object are a series of variable-length data items called subobjects. The subobjects are identical

Expires June 2003

[Page 5]

to those defined in [[RSVP-TE](#)] and [[GMPLS-RSVP-TE](#)] for use in EROs.

The following subobject types are supported.

- 1 IPv4 prefix
- 2 IPv6 prefix
- 32 Autonomous system number
- TBD SRLG

The defined values for Type above are specified in [[RSVP-TE](#)] and in this document.

The concept of loose or strict hops has no meaning in route exclusion. The L bit defined for ERO subobjects in [[RSVP-TE](#)] is re-used to indicate that an abstract node **MUST** be avoided (value 0) or **SHOULD** be avoided (value 1).

An N bit is introduced in subobjects that define IP addresses to distinguish between addresses that identify a node (value 1) and addresses that identify an interface (value 0). In this way whole nodes or specific interfaces can be excluded from the path.

[6.1.1](#) Subobject 1: IPv4 prefix

0																1																2																3															
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9																								
L																Type																Length																IPv4 address (4 bytes)															
IPv4 address (continued)																Prefix Length																N Resvd																															

L

- 0 indicates that the abstract node specified **MUST** be excluded
- 1 indicates that the abstract node specified **SHOULD** be avoided

N

- 0 indicates that the abstract node identifies an interface or set of interfaces that should be excluded or avoided according to the setting of the L bit.
- 1 indicates that the abstract node identifies a node or set of nodes that should be excluded or avoided according to the setting of the L bit.

Resvd

Zero on transmission. Ignored on receipt.

The rest of the fields are as defined in [[RSVP-TE](#)].

Expires June 2003

[Page 6]

6.1.2 Subobject 2: IPv6 Prefix

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|L|   Type   |   Length   | IPv6 address (16 bytes) |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| IPv6 address (continued) |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| IPv6 address (continued) |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| IPv6 address (continued) |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| IPv6 address (continued) | Prefix Length |N|   Resvd   |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

L

- 0 indicates that the abstract node specified MUST be excluded
- 1 indicates that the abstract node specified SHOULD be avoided

N

- 0 indicates that the abstract node identifies an interface or set of interfaces that should be excluded or avoided according to the setting of the L bit.
- 1 indicates that the abstract node identifies a node or set of nodes that should be excluded or avoided according to the setting of the L bit.

Resvd

Zero on transmission. Ignored on receipt.

The rest of the fields are as defined in [[RSVP-TE](#)].

6.1.3 Subobject 32: Autonomous System Number

The L bit of an Autonomous System Number subobject does has meaning in an Exclude Route (contrary to its usage in an Explicit Route defined in [[RSVP-TE](#)]. The meaning is as for other subobjects described above. That is:

- 0 indicates that the abstract node specified MUST be excluded
 - 1 indicates that the abstract node specified SHOULD be avoided
- The rest of the fields are as defined in [[RSVP-TE](#)]. There is no N bit defined.

6.1.4 Subobject TBD: SRLG

The N bit is not present. The rest of the fields are as defined in the "SRLG ERO Subobject" section of this draft.

Expires June 2003

[Page 7]

6.2. Semantics and Processing Rules for the Exclude Route Object (XRO)

The exclude route list is encoded as a series of subobjects contained in an EXCLUDE_ROUTE object. Each subobject identifies an abstract node in the exclude route list.

Each abstract node may be a precisely specified IP address a node, or an IP address with prefix identifying interfaces of a group of nodes or an Autonomous System.

The Explicit Route and routing processing is unchanged from the description in [\[RSVP-TE\]](#) with the following additions:

- a. When a Path message is received at a node, the node must check that it is not a member of any of the abstract nodes in the XRO if it is present in the Path message. If the node is a member of any of the abstract nodes in the XRO it should return a PathErr with the error code "Routing Problem" and error value of "Local node in Exclude Route". If there are SRLGs in the XRO, the node should check that it and the resources it uses are not part of any SRLG that is specified with Tolerance value of zero. If it is, it should return a PathErr with the error code "Routing Problem" and error value of "Local node in Exclude Route". The node may be a member of an SRLG in the XRO that is specified with a non-zero Tolerance value.
- b. When choosing a next hop or expanding an explicit route to include additional subobjects, a node: i) must not introduce an explicit node or an abstract node that equals or is a member of any abstract node that is specified in the Exclude Route Object. ii) must not (or should not, in the case of a non-zero Tolerance value) introduce links, nodes or resources identified by the SRLG ID specified in the SRLG subobjects(s). If these rules preclude further forwarding of the Path message, the node should return a PathErr with the error code "Routing Problem" and error value of "Route blocked by Exclude Route".
- c. The subobjects in the ERO and XRO SHOULD not contradict each other. If they do contradict, the subobjects with the L bit not set, strict or MUST be excluded, respectively, in the ERO or XRO MUST take precedence. If there is still a conflict, the subobjects in the ERO MUST take precedence.

The XRO Class-Num is of the form 11bbbbbb so that nodes which do not support the XRO will forward it uninspected and will not apply the extensions to ERO processing described above. This makes the XRO a 'best effort' process.

This 'best-effort' approach is chosen to allow route exclusion to

Expires June 2003

[Page 8]

traverse parts of the network that are not capable of parsing or handling the new function. Note that Record Route may be used to allow computing nodes to observe violations of route exclusion and attempt to re-route the LSP accordingly.

7. Explicit Exclude Route

The Explicit Exclude Route defines abstract nodes or resources (such as links, unnumbered interfaces or labels) that must not be used on the path between two inclusive abstract nodes or resources in the explicit route.

7.1. Explicit Exclusion Route Subobject (EXRS)

A new subobject type is defined. The Explicit Exclude Route Subobject (EXRS) has type [TBD].

The EXRS is an ERO subobject.

The format of the EXRS is as follows.

```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|L|   Type   |   Length   |   EXRS subobjects   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

L

ignored and must be zero

[Note: The L bit in an ERES subobject is as defined for the XRO subobjects]

Type

The type of the subobject, i.e. EXRS [TBD]

EXRS subobjects

An EXRS subobject indicates the abstract node or resource to be excluded. The format of this field is exactly the format of an XRO subobject and may include an SRLG subobject, both subobjects as described earlier in this draft.

Thus, an EXRO subobject for an IP hop might look as follows:

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|L|   Type   |   Length   |L|   Type   |   Length   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| IPv4 address (4 bytes) |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Prefix Length |N|   Reserved   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

Expires June 2003

[Page 10]

7.2. Semantics and Processing Rules for the EXRS

Each EXRS may carry multiple exclusions. The exclusion is encoded exactly as for XRO subobjects and prefixed by an additional Type and Length.

The scope of the exclusion is the step between the previous ERO subobject that identifies an abstract node, and the subsequent ERO subobject that identifies an abstract node. Multiple exclusions may be present between any pair of abstract nodes.

Exclusions may indicate explicit nodes, abstract nodes or Autonomous Systems that must not be traversed on the path to the next abstract node indicated in the ERO.

Exclusions may also indicate resources (such as unnumbered interfaces, link ids, labels) that must not be used on the path to the next abstract node indicated in the ERO.

SRLGs may also be indicated for exclusion from the path to the next abstract node in the ERO by the inclusion of an EXRO Subobject containing an SRLG subobject. If the Tolerance value in the SRLG subobject is zero, the resources (nodes, links, etc.) identified by the SRLG must not be used on the path to the next abstract node indicated in the ERO. If the Tolerance value is non-zero, the resources identified by the SRLG should be avoided, but may be used in preference to resources associated with another SRLG indicated for exclusion if that SRLG has a (numerically) lower Tolerance value.

The subobjects in the ERO and EXRS SHOULD not contradict each other. If they do contradict, the subobjects with the L bit not set, strict or MUST be excluded, respectively, in the ERO or XRO MUST take precedence. If there is still a conflict, the subobjects in the ERO MUST take precedence.

If a node is called upon to process an EXRS and does not support handling of exclusions it will return a PathErr with a "Bad EXPLICIT_ROUTE object" error.

If the presence of EXRO Subobjects precludes further forwarding of the Path message, the node should return a PathErr with the error code "Routing Problem" and error value of "Route blocked by Exclude Route".

8. Security

The new exclude route object poses no security exposures over and above [[RSVP-TE](#)] and [[GMPLS-RSVP-TE](#)]. Note that any security concerns

Expires June 2003

[Page 11]

that exist with Explicit Routes should be considered with regard to route exclusions.

9. IANA Considerations

9.1. New Class Numbers

One new class number is required.

```
EXCLUDE_ROUTE
  Class-Num = 011bbbb
  CType: 1
```

9.2. New Subobject Types

A new subobject type for the Exclude Route Object and Explicit Exclude Route Subobject is required.

SRLG subobject

A new subobject type for the ERO is required.

Explicit Exclude Route subobject

9.3. New Error Codes

New error values are needed for the error code 'Routing Problem'.

Unsupported Exclude Route Subobject Type Local node in Exclude Route Route blocked by Exclude Route

10. Acknowledgments

This draft reuses text from [[RSVP-TE](#)] for the description of EXCLUDE_ROUTE.

The authors would like to express their thanks to Igor Bryskin and Lou Berger their considered opinions on this draft. Also thanks to Yakov Rekhter for reminding us about SRLGs.

11. Normative References

- | | |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------|
| [RFC2119] | Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14 , RFC 2119 , March 1997 |
| [RSVP-TE] | D. Awduche, et al., "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209 , December 2001. |

Expires June 2003

[Page 12]

- [GMPLS-RSVP-TE] P. Ashwood-Smith, et al., "Generalized MPLS Signaling - RSVP-TE Extensions", Internet Draft, [draft-ietf-mpls-generalized-rsvp-te-07.txt](#), April 2002 (work in progress).
- [GMPLS-OSPF] K. Kompela, et al., "OSPF Extensions in Support of Generalized MPLS", Internet Draft, [draft-ietf-ccamp-ospf-gmpls-extensions-07.txt](#), May 2002 (work in progress).
- [CCAMP-SRLG] D. Papadimitriou, et al., "Shared Risk Link Groups Encoding and Processing", Internet Draft, [draft-papadimitriou-ccamp-srlg-processing-01.txt](#), November 2002 (work in progress).
- [MPLS-TE-MIB] C. Srinivasan, et al., "Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base", Internet Draft, [draft-ietf-mpls-te-mib-08.txt](#), January 2002 (work in progress).

12. Informational References

- [MPLS-BUNDLE] Kompella, K., Rekhter, Y., and Berger, L., "Link Bundling in MPLS Traffic Engineering", Internet Draft, [draft-ietf-mpls-bundle-02.txt](#), May 2002, (work in progress).
- [MPLS-UNNUM] Kompella, K., Rekhter, Y., "Signalling Unnumbered Links in RSVP-TE", Internet Draft, [draft-ietf-mpls-rsvp-unnum-06.txt](#), May 2002, (work in progress).
- [GMPLS-SIG] P. Ashwood-Smith, et al, "Generalized MPLS - Signaling Functional Description", [draft-ietf-mpls-generalized-signaling-08.txt](#) April 2002, (work in progress).

13. Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this

Expires June 2003

[Page 13]

document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns. This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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.

14. Authors' Information

Cheng-Yin Lee
Alcatel
600 March Road.
Ottawa, Ontario
Canada K2K 2E6
email: Cheng-Yin.Lee@alcatel.com

Adrian Farrel
Movaz Networks, Inc.
7926 Jones Branch Drive, Suite 615
McLean VA, 22102 USA
Phone: +1-703-847-1867
Email: afarrel@movaz.com

Stefaan De Cnodder
Alcatel
Francis Wellesplein 1
B-2018 Antwerp, Belgium
email: stefaan.de_cnodder@alcatel.be

Expires June 2003

[Page 14]