                 **Extended Administrative Groups in MPLS-TE**
                 **draft-osborne-mpls-extended-admin-groups-01**

Abstract

   This document provides additional administrative groups (sometimes
   referred to as "link colors") to the IGP extensions for MPLS-TE.

Requirements Language

   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 RFC 2119 [RFC2119].

Table of Contents

## 1.  Introduction

MPLS-TE advertises 32 administrative groups (commonly referred to as "colors" or "link colors") using the Administrative Group sub-TLV of the Link TLV.  This is defined for OSPFv2 [RFC3630], OSPFv3 [RFC5329]and ISIS [RFC5305].

This document adds a sub-TLV to the IGP TE extensions, "Extended Administrative Group".  This sub-TLV provides for additional administrative groups (link colors) beyond the current limit of 32.

## 2.  Extended Administrative Groups sub-TLV

The Extended Administrative Groups sub-TLV is used in addition to the Administrative Groups when a device wishes to advertise more than 32 colors for a link.  The EAG sub-TLV is optional.

This document uses the term 'colors' as a shorthand to refer to particular bits with an AG or EAG.  The examples in this document use 'red' to represent the least significant bit in the AG (red == 0x1), 'blue' to represent the second bit (blue == 0x2).  To say that a link has a given color or that the specified color is set on the link is to say that the corresponding bit or bits in the link's AG are set to 1.

## 2.1.  Packet Format

The format of the Extended Administrative Groups sub-TLV is the same
for both OSPF and ISIS:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Extended Admin Group                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         ..........                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Extend Admin Group                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Type of the sub-TLV for OSPF and ISIS is TBD.  The Length is the
size of the Extended Admin Group (EAG) value in bytes.  The EAG may
be of any length, but must be a multiple of 4 bytes.

## 2.2.  Admin group numbering

By convention, the existing Administrative Group TLVs are numbered 0
(LSB) to 31 (MSB).  The EAG values are a superset of AG.  That is,
bits 0-31 in the EAG have the same meaning and must have the same
values as an AG flooded for the same link.

[ NOTE to be removed before publication: This is a change from
version -00 of the draft.  Doing it this way allows us to eventually
deprecate AG rather than advertising two sub-TLVs forever.  ]

## 2.3.  Backward compatability

There are two things to consider for backward compatibility with
existing AG implementations - how do AG and EAG coexist, and what
happens if a node has matching criteria for unadvertised EAG bits?

## 2.3.1.  AG and EAG coexistence

If a node advertises EAG it MAY also advertise AG.  If a node
advertises both AG and EAG then the first 32 bits of the EAG MUST be
identical to the advertised AG.  If the AG and EAG advertised for a
link differ, the EAG MUST take priority.  This allows nodes which do
not support EAG to obtain some link color information from the
network, but also allow for an eventual migration away from AG.

**2.3.2**.  **Desire for unadvertised EAG bits**

The existing AG sub-TLV is optional; thus a node may be configured
with a preference to include red or exclude blue, and be faced with a
link that is not advertising a value for either blue or red.  What
does an implementation do in this case?  It shouldn't assume that red
is set, but it is also arguably incorrect to assume that red is NOT
set, as a bit must first exist before it can be set to 0.

Practically speaking this has not been an issue for deployments, as
many implementations always advertise the AG bits, often with a
default value of 0x00000000.  However, this issue may be of more
concern once EAGs are added to the network.  EAGs may exist on some
nodes but not others, and the EAG length may be longer for some links
than for others.

Each implementation is free to choose its own method for handling
this question.  However, to encourage maximum interoperability an
implementation SHOULD treat specified but unadvertised EAG bits as if
they are set to 0.  A node MAY provide other (configurable)
strategies for handling this case.

**3**.  **Attribute filters in RSVP**

In addition to updating the IGP sub-TLV, RSVP needs to be extended to
provide the ability to signal desired resource affinities.  This
section provides that update.

**3.1**.  **EXTENDED_SESSION_ATTRIBUTE_RA**

This section provides the EXTENDED_SESSION_ATTRIBUTE_RA.

RFC3209 defines two types of SESSION_ATTRIBUTE, one with resource
affinities and one without.  The former is C-Type 1 and is referred
to in this document as SESSION_ATTRIBUTE_RA.  The latter is referred
to as SESSION_ATTRIBUTE and is C-Type 7.

The Class and C-Type for EXTENDED_SESSION_ATTRIBUTE_RA are 207 and
TBD, respectively.  The format of the EXTENDED_SESSION_ATTRIBUTE_RA
is:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Attribute filter length                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                         Exclude-any                         //
```

```
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     //                      Include-any                          //
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     //                      Include-all                          //
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |  Setup Prio  | Holding Prio  |     Flags     | Name Length  |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |                                                               |
     //        Session Name     (NULL padded display string)     //
     |                                                               |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Exclude-any, Include-any and Include-all fields are collectively
referred to as the "attribute filter fields".  All three attribute
filter fields MUST be the same length.  All fields in the
EXTENDED_SESSION_ATTRIBUTE_RA MUST be interpreted exactly as they are
in the SESSION_ATTRIBUTE_RA.

The attribute filter length is the sum of the lengths of the three
attribute filter fields, in bytes.  If the user wishes to convey 128
bits of information in each of the fields, the total length of the
attribute filter fields is 3*128 == 384 bits.  The attribute filter
length is thus 384/8 == 48 bytes.

## 3.2.  Populating the attribute filter fields

Each attribute filter field MUST be the same length.  As with the EAG
sub-TLV, each attribute filter field is a multiple of four bytes in
length.  The length of each field MUST be at least the minimum length
necessary to fully convey the headend's matching criteria, and SHOULD
be no longer than that.  For example, if the headend wishes to
Include-any bits 1 and 17 then all three fields MUST be at least 4
bytes in length and SHOULD be no more than 4 bytes in length.  If the
headend wishes to Include-any bits 1, 17 and 150 then all three
fields MUST be at least 20 bytes (160 bits) in length and SHOULD be
no longer than 20 bytes.

### 3.3.  Formatting a Path message

RFC3209 section 3.1 allows for only a single session attribute
(implicitly, either SESSION_ATTRIBUTE or SESSION_ATTRIBUTE_RA).  In
order to achieve maximum backward compatibility, a node MAY signal
both a SESSION_ATTRIBUTE_RA and an EXTENDED_SESSION_ATTRIBUTE_RA.  A
node MUST NOT signal both a SESSION_ATTRIBUTE and an
EXTENDED_SESSION_ATTRIBUTE_RA.

All fields common to the SESSION_ATTRIBUTE_RA and
EXTENDED_SESSION_ATTRIBUTE_RA (the first 32 bits of Exclude-any,
Include-any and Include-all, as well as the Setup Prio, Holding Prio,
Flags, Name Length and Session Name) MUST be the same in both the
SESSION_ATTRIBUTE_RA and EXTENDED_SESSION_ATTRIBUTE_RA.  If they are
different, the EXTENDED_SESSION_ATTRIBUTE_RA takes priority.  An
implementation SHOULD alert the user if any of these fields differ
between the SESSION_ATTRIBUTE_RA and EXTENDED_SESSION_ATTRIBUTE_RA.

RFC3209 section 3.1 is thus updated with to possible Path message
formats, one which allows for the coexistence of both
SESSION_ATTRIBUTE_RA and EXTENDED_SESSION_ATTRIBUTE_RA and one which
only allows for SESSION_ATTRIBUTE.

[NOTE: is there a clean way to do this in one definition?  Is this
even necessary?  My ASN.1-fu is not so good, and I don't want to
define Path and Path_RA and have to touch everything else in rfc3209]

```
        <Path Message> ::=        <Common Header> [ <INTEGRITY> ]
                                  <SESSION> <RSVP_HOP>
                                  <TIME_VALUES>
                                  [ <EXPLICIT_ROUTE> ]
                                  <LABEL_REQUEST>
                                  [ <SESSION_ATTRIBUTE> ]
```

or

```
        <Path Message> ::=        <Common Header> [ <INTEGRITY> ]
                                  <SESSION> <RSVP_HOP>
                                  <TIME_VALUES>
                                  [ <EXPLICIT_ROUTE> ]
                                  <LABEL_REQUEST>
                                  [ <SESSION_ATTRIBUTE_RA> ]
                                  [ <EXTENDED_SESSION_ATTRIBUTE_RA> ]
```

## 3.4.  Interpreting the attribute filter fields

Since the attribute filter fields are of variable length, it is
possible that an RSVP message may indicate more bits than a given
node has advertised for a link.  It is equally possible that an RSVP
message may indicate fewer bits than a given node has advertised for
a link.  In all cases, the shorter of the two fields (the attribute
filter field or the locally configured link admin group) MUST be
padded with zeros so that both fields are of equal length.

Specifically, length mismatches are to be handled as follows:

The length of any single attribute filter field is A.

The length of the configured link attribute for a given link is C.

If C > A, a node MUST pad the received attribute filter field values
with zeros so that C == A.  A node MUST NOT alter the length of the
signalled attribute filter field; the zero padding is only local to a
given node.

If A > C, a node MUST pad the locally configured link attributes with
zeros so that A == C.  A node SHOULD NOT use this information to
alter the length of the EAG sub-TLV that it floods.

[ NOTE to readers: rfc3209 is unclear about how the attribute filter
fields are to be used.  The intent appears to be that any bits set to
1 in any of the three attribute filter fields must be considered a
match for filtering purposes, and that any bits set to 0 are not used
to match.  In other words, there is no way to say "the following bits
MUST be zero" for any of the attribute filter fields.  A 0 in an
attribute filter field says "I do not care what the value of this bit
is".  I am making this inference largely from the text in Include-any
and Include-all which says "A null set...automatically passes".  If
existing implementations treat these fields differently (e.g.  a 0
MUST be matched as a zero) then I'd like to know that so I can get
the text in this section right.  ]

## 4.  Security Considerations

This extension adds no new security considerations.

## 5.  IANA Considerations

This document requests a sub-TLV allocation in both OSPF and ISIS, as
well as an RSVP C-Type from Class 207.

6.  Acknowledgements

   Thanks to Santiago Alvarez, Rohit Gupta, Liem Nguyen, Tarek Saad, and
   Robert Sawaya for their review and comments.

7.  Normative References

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

   [RFC3630]   Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering
               (TE) Extensions to OSPF Version 2", RFC 3630, September
               2003.

   [RFC5305]   Li, T. and H. Smit, "IS-IS Extensions for Traffic
               Engineering", RFC 5305, October 2008.

   [RFC5329]   Ishiguro, K., Manral, V., Davey, A., and A. Lindem,
               "Traffic Engineering Extensions to OSPF Version 3", RFC
               5329, September 2008.

Author's Address

   Eric Osborne
   Cisco Systems

   Email: eosborne@cisco.com