                  Extensions to PCEP for Temporal LSP
                       draft-chen-pce-tts-00.txt

   Abstract

      This document specifies extensions to PCEP for initiating and
      maintaining a Traffic Engineering (TE) Label Switched Path (LSP) in a
      time interval or a sequence of time intervals, during which the LSP
      carries traffic.

to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   Once an existing multiprotocol label switching (MPLS) traffic
   engineering (TE) label switched path (LSP) is set up, it is assumed
   to carry traffic forever until it is down. When an MPLS TE LSP
   tunnel is up, it is assumed that the LSP consumes its reserved
   network resources forever even though the LSP may only use network
   resources during some period of time. As a result, the network
   resources are not used efficiently. Moreover, a tunnel service can
   not be reserved or booked in advance for a period of time or a
   sequence of time periods.

   This document specifies extensions to PCEP for initiating and
   maintaining an MPLS TE LSP in a period of time called a time interval
   or a sequence of time intervals. It is assumed that the LSP carries
   traffic during this time interval or each of these time intervals.
   Thus the network resources are efficiently used. More importantly,
   some new services can be provided. For example, a consumer can book
   a tunnel service in advance for a given time interval. Tunnel
   services may be scheduled.

2. Terminology

   A Time Interval: a time period from time Ta to time Tb.

   LSP: Label Switched Path. An LSP is a P2P (point-to-point) LSP or a
   P2MP (point-to-multipoiint) LSP.

   LSP in a time interval: LSP that carries traffic in the time
   interval.

   LSP in a sequence of time intervals: LSP that carries traffic in each
   of the time intervals.

   Temporal LSP: LSP in a time interval or LSP in a sequence of time
   intervals.

   TEDB: Traffic Engineering Database.

   This document uses terminologies defined in RFC5440.

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. Operations Overview

   This section briefly describes some operations on a temporal LSP.

4.1. Simple Time Interval

   For a temporal LSP, a user configures it with a time interval or a
   sequence of time intervals. A simple time interval is a time period
   from time Ta to time Tb, which may be represented as [Ta, Tb].

   When an LSP is configured with time interval [Ta, Tb], a path
   satisfying the constraints for the LSP in the time interval is
   computed and the LSP along the path is set up to carry traffic from
   time Ta to time Tb.

   In addition to simple time intervals, there are recurrent time
   intervals and elastic time intervals. Sometimes a simple time
   interval is called a time interval.

4.2. Recurrent Time Interval

   A recurrent time interval represents a series of repeated simple time
   intervals. It has a simple time interval such as [Ta, Tb], a number
   of repeats such as 10 (repeats 10 times), and a repeat cycle/time
   such as a week (repeats every week). The recurrent time interval:
   "[Ta, Tb] repeats n times with repeat cycle C" represents n+1 simple
   time intervals as follows:

     [Ta, Tb], [Ta+C, Tb+C], [Ta+2C, Tb+2C], ..., [Ta+nC, Tb+nC]

   When an LSP is configured with a recurrent time interval such as
   "[Ta, Tb] repeats 10 times with a repeat cycle a week" (representing
   11 simple time intervals), a path satisfying the constraints for the
   LSP in each of the simple time intervals represented by the recurrent
   time interval is computed and the LSP along the path is set up to
   carry traffic in each of the simple time intervals.

4.3. Elastic Time Interval

   An elastic time interval is a time interval with an elastic range,
   which is represented as within -P and Q, where P and Q is an amount
   of time such as 300 seconds. P is called elastic range lower bound
   and Q is called elastic range upper bound.

   For a simple time interval such as [Ta, Tb] with an elastic range,
   elastic time interval: "[Ta, Tb] within -P and Q" means a time period
   from (Ta+X) to (Tb+X), where -P <= X <= Q.

When an LSP is configured with elastic time interval "[Ta, Tb] within
-P and Q", a path is computed such that the path satisfies the
constraints for the LSP in the time period from (Ta+X) to (Tb+X) and
|X| is the minimum value from -P to Q. That is that [Ta+X, Tb+X] is
the time interval closest to time interval [Ta, Tb] within the
elastic range. The LSP along the path is set up to carry traffic in
the time period from (Ta+X) to (Tb+X).

Similarly, for a recurrent time interval with an elastic range,
elastic time interval: "[Ta, Tb] repeats n times with repeat cycle C
within -P and Q" represents n+1 simple elastic time intervals as
follows:

   [Ta+X0, Tb+X0], [Ta+C+X1, Tb+C+X1], ..., [Ta+nC+Xn, Tb+nC+Xn]

where -P <= Xi <= Q, i = 0, 1, 2, ..., n.

If a user wants to keep the same repeat cycle between any two
adjacent time intervals, elastic time interval: "[Ta, Tb] repeats n
times with repeat cycle C within -P and Q SYNC" may be used, which
represents n+1 simple elastic time intervals as follows:

   [Ta+X, Tb+X], [Ta+C+X, Tb+C+X], ..., [Ta+nC+X, Tb+nC+X]

where -P <= X <= Q.

4.4. Changes to Time Interval

   After a temporal LSP is configured, a user may change its parameters
   including some of the time intervals configured. A new time interval
   may be added, an existing time interval may be removed or changed.

   When a new time interval is added to an existing LSP, a path
   satisfying the constraints for the LSP in the time interval is
   computed and the LSP along the path is set up to carry traffic in the
   time interval.

   When an existing time interval is removed from an existing LSP, the
   time interval is deleted from the lifetime of the LSP. If the
   lifetime is over, the LSP is deleted.

   A change to an existing time interval may generate some of four
   possible results: 1) The existing time interval is extended for a
   time period EA after the existing time period; 2) The existing time
   interval is extended for a time period EB before the existing time
   period; 3) The existing time interval is shrunk for a time period SA

from the end of the existing time period; and 4) The existing time
interval is shrunk for a time period SB from the beginning of the
existing time period.

When an existing time interval for an LSP is extended, a path
satisfying the constraints for the LSP in the extended time interval
is computed and the LSP along the path is set up to carry traffic in
the extended time interval. If the LSP is already up to carry
traffic in the existing time interval, the lifetime of the LSP is
extended for time period EA following the existing time interval.

When an existing time interval for an LSP is shrunk, the shrunk time
periods are removed from the lifetime of the LSP.

4.5. Graceful Periods

   For a temporal LSP, a user may want to have some graceful periods for
   each or some of the time intervals for the LSP. Two graceful periods
   may be configured for a time interval. One is the graceful period
   before the time interval, called grace-before, which extends the
   lifetime of the LSP for grace-before (such as 30 seconds) before the
   time interval. The other is the one after the time interval, called
   grace-after, which extends the lifetime of the LSP for grace-after
   (such as 60 seconds) after the time interval.

   When an LSP is configured with a simple time interval such as [Ta,
   Tb] with graceful periods such as grace-before GB and grace-after GA,
   a path is computed such that the path satisfies the constraints for
   the LSP in the time period from Ta to Tb. The LSP along the path is
   set up to carry traffic in the time period from (Ta-GB) to (Tb+GA).
   During graceful periods from (Ta-GB) to Ta and from Tb to (Tb+GA),
   the LSP is up to carry traffic (maybe in best effort).

5. Extensions to PCEP

   This section describes the extensions to PCEP for initiating and
   maintaining temporal LSPs.

5.1. Capability TLV in Existing PCE Discovery Protocol

   There are a couple of options for advertising a PCE capability for
   initiating and maintaining temporal LSPs.

   The first option is to define a new flag in the OSPF and ISIS PCE
   Capability Flags to indicate the capability that a PCE is capable to
   initiate and maintain temporal LSPs. This includes the capability of
   computing both a path for a temporal P2MP LSP and a path for a

temporal P2P LSP.

The second option is to define three new flags. The first new flag
in the OSPF and ISIS PCE Capability Flags indicates the capability
that a PCE is capable to compute a path for a temporal P2MP LSP; the
second new flag indicates the capability that a PCE is capable to
compute a path for a temporal P2P LSP; and the third new flag
indicates the capability that a PCE is capable to initiate and
maintain a temporal LSP.

The format of the PCE-CAP-FLAGS sub-TLV is 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type = 5           |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                    PCE Capability Flags                       ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
   Type:     5
   Length:   Multiple of 4 octets
   Value:    This contains an array of units of 32-bit flags
             numbered from the most significant as bit zero, where
             each bit represents one PCE capability.
```

The following capability bits have been assigned by IANA:

```
   Bit       Capabilities
    0        Path computation with GMPLS link constraints
    1        Bidirectional path computation
    2        Diverse path computation
    3        Load-balanced path computation
    4        Synchronized path computation
    5        Support for multiple objective functions
    6        Support for additive path constraints
             (max hop count, etc.)
    7        Support for request prioritization
    8        Support for multiple requests per message
    9        Global Concurrent Optimization (GCO)
   10        P2MP path computation
   ...
```

Reserved bits SHOULD be set to zero on transmission and MUST be
ignored on receipt.

For the second option, one bit such as bit 16 may be assigned to indicate that a PCE is capable to compute a path for a temporal P2MP LSP; another bit such as bit 17 may be assigned to indicate that a PCE is capable to compute a path for a temporal P2P LSP; and yet another bit such as bit 18 may be assigned to indicate that a PCE is capable to initiate and maintain temporal LSPs.

```
     Bit        Capabilities
      16        Path computation for temporal P2MP LSP
      17        Path computation for temporal P2P LSP
      18        Initiation and maintenance of temporal LSP
     19-31      Reserved for future assignments by IANA.
```

5.2. Open Message Extension

If a PCE does not advertise its capability related to initiation and maintenance of a temporal LSP during discovery, PCEP should be used to allow a PCC to discover, during the Open Message Exchange, which PCEs are capable of supporting initiation and maintenance of a temporal LSP.

To achieve this, we extend the PCEP OPEN object by defining a new optional TLV to indicate the PCE's capability to initiate and maintain a temporal LSP.

We request IANA to allocate a value such as 10 from the "PCEP TLV Type Indicators" subregistry, as documented in Section below ("Temporal LSP Capability TLV"). The description is "temporal LSP capable", and the length value is 2 bytes. The value field is set to indicate the capability of a PCE for initiation and maintenance of a temporal LSP in details.

We can use flag bits in the value field in the same way as the PCE Capability Flags described in the previous section.

The inclusion of this TLV in an OPEN object indicates that the sender can initiate and maintain a temporal LSP.

The capability TLV is meaningful only for a PCE, so it will typically appear only in one of the two Open messages during PCE session establishment. However, in case of PCE cooperation (e.g., inter-domain), when a PCE behaving as a PCC initiates a PCE session it SHOULD also indicate its capabilities.

5.3. RP Object Extension

   The following flags are added into the RP Object:

   A T bit is added in the flag bits field of the RP object to tell a
   receiver of a message that the message is for (initiating and
   maintaining) a temporal LSP.

       o T (Temporal LSP bit - 1 bit):

           0: This indicates that this is not a message
              for a temporal LSP.

           1: This indicates that this is a message
              for a temporal LSP.

   The IANA request is referenced in Section below (Request Parameter
   Bit Flags) of this document.

   This T bit with the N bit defined in RFC6006 can indicate whether the
   message is for a temporal P2P LSP or P2MP LSP.

     o T = 1 and N = 0: This indicates that this is a message
                        for a temporal P2P LSP
     o T = 1 and N = 1: This indicates that this is a message
                        for a temporal P2MP LSP

5.4. TIME INTERVAL Object

   For a TIME-INTERVAL object, its Class is to be assigned by IANA, here
   we use 18, which may be changed late. Its OT is 1, exact number to
   be assigned by IANA. The format of a TIME-INTERVAL object body is
   illustrated below, which comprises a number of time interval TLVs.

      Object-Class: TBD (18),   OT = 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
     +-+
     | (Object Body containing Time Interval TLVs) |
     ~   ~
     +-+

   A time interval TLV may be a relative time interval TLV or an
   absolute time interval TLV, which are two different representations

of a time interval. Their advantages and disadvantages are discussed
below.

5.4.1. Absolute Time Interval TLV

The format of an absolute time interval TLV (Type = 1) for an LSP is
illustrated below. It mainly contains a Start-time and an End-time,
representing time interval [Start-time, End-time]. Both of these two
times are the times that are synchronized among all the elements
involved. Thus the clocks on all the elements MUST be synchronized
if an absolute time interval TLV is used. The time period
represented in an absolute time interval TLV is more accurate.

In addition, it contains an non zero grace-before and grace-after if
graceful periods are configured. It includes an non zero elastic
range lower bound and upper bound if there is an elastic range
configured.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type (1)           |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Start-time                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          End-time                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Reserved (0)                |    GrB    |   GrA    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Elastic-Lower-Bound      |      Elastic-Upper-Bound      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o Start-time: The time LSP starts to carry traffic.

   o End-time: The time LSP ends carrying traffic.

   o GrB (Grace-Before):   The graceful period time length in seconds
     before time interval [Start-time, End-time].

   o GrA (Grace-After):   The graceful period time length in seconds
     after time interval [Start-time, End-time].

   o Elastic-Lower-Bound: The maximum amount of time in seconds that
     time interval [Start-time, End-time] can shift to lower/left.

   o Elastic-Upper-Bound: The maximum amount of time in seconds that
     time interval [Start-time, End-time] can shift to upper/right.

   Discussions: Optionally, we may define three TLVs: 1) an absolute
   time interval TLV containing only a Start-time and an End-time; 2) an
   elastic range TLV containing just an elastic range lower bound and
   upper bound; and 3) a graceful period TLV containing only a grace-
   before and a grace-after. If a time interval is with an elastic
   range, an absolute time interval TLV followed by an elastic range TLV
   is used. If a time interval is with graceful periods, an absolute
   time interval TLV followed by a graceful period TLV is used.

5.4.2. Relative Time Interval TLV

   The format of a relative time interval TLV (Type = 2) for an LSP is
   shown below. It mainly contains a Start-time-length and an End-time-
   length, representing the time interval below for the LSP:

    [current-time + Start-time-length, current-time + End-time-length]

   where current-time is a current local time. When a time interval
   from time Ta to time Tb is configured on a node/element, these two
   time lengths are the time lengths that are computed on the node using
   a current local time as follows.

     Start-time-length = Ta - current-time;
     End-time-length   = Tb - current-time;

   For a relative time interval TLV, the clocks/times on all the
   elements involved can be different. But the time period represented
   in a relative time interval TLV on one element/node may be shifted a
   little bit from another element's point of view since transmitting
   the TLV from one element to another takes a little time, which is
   hard to be considered accurately.

   The TLV also includes an non zero grace-before and grace-after if
   graceful periods are configured. It contains an non zero elastic
   range lower bound and upper bound if there is an elastic range
   configured.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |            Type (2)           |             Length            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Start-time-length                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         End-time-length                       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |            Reserved (0)            |     GrB      |     GrA     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |       Elastic-Lower-Bound      |      Elastic-Upper-Bound     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o Start-time-length: The time length in seconds from a current
     local time to the time LSP starts to carry traffic.

   o End-time-length: The time length in seconds from a current local
     time to the time LSP ends carrying traffic.

   o GrB (Grace-Before):   The graceful period time length in seconds
     before the time interval above for the LSP.

   o GrA (Grace-After):   The graceful period time length in seconds
     after the time interval above for the LSP.

   o Elastic-Lower-Bound: The maximum amount of time in seconds that
     the time interval above for the LSP can shift to lower/left.

   o Elastic-Upper-Bound: The maximum amount of time in seconds that
     the time interval above can shift to upper/right.

5.4.3. Recurrent Absolute Time Interval TLV

   The format of a recurrent absolute time interval TLV (Type = 3) for
   an LSP is illustrated below. It mainly contains a Start-time, an
   End-time, a Repeat-time-length, a Options field and a Number-repeats.

   The Start-time and End-time represents time interval [Start-time,
   End-time]. The Repeat-time-length represents a repeat cycle/time,
   which is valid if the Options field is set to indicate the way to
   repeat is "repeat every Repeat-time-length". The Options field
   indicates a way to repeat. The Number-repeats indicates the number
   of repeats of time interval [Start-time, End-time].

   In addition, the TLV includes an non zero grace-before and grace-
   after if graceful periods are configured. It contains an non zero

elastic range lower bound and upper bound if there is an elastic
range configured.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type (3)           |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Start-time                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            End-time                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Repeat-time-length                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Options |      Number-repeats       |   GrB   |     GrA       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Elastic-Lower-Bound      |      Elastic-Upper-Bound      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o Start-time: The time LSP starts to carry traffic.

   o End-time: The time LSP ends carrying traffic.

   o Repeat-time-length: The time length in seconds after which LSP
     starts to carry traffic again for (End-time - Start-time).

   o Options: Indicates a way to repeat.

        Options = 1: repeat every day;

        Options = 2: repeat every week;

        Options = 3: repeat every month;

        Options = 4: repeat every year;

        Options = 5: repeat every Repeat-time-length.

   o Number-repeats: The number of repeats. In each of repeats, LSP
     carries traffic.

   o GrB (Grace-Before):   The graceful period time length in seconds
     before each of the time intervals represented by the recurrent
     time interval.

   o GrA (Grace-After):   The graceful period time length in seconds
     after each of the time intervals.

   o Elastic-Lower-Bound: The maximum amount of time in seconds that
     each of the time intervals can shift to lower/left.

   o Elastic-Upper-Bound: The maximum amount of time in seconds that
     each of the time intervals can shift to upper/right.

5.4.4. Recurrent Relative Time Interval TLV

   The format of a recurrent relative time interval TLV (Type = 4) for
   an LSP is shown below. It mainly contains a Start-time-length, an
   End-time-length, a Repeat-time-length, a Options field and a Number-
   repeats.

   The Start-time-length and End-time-length represents time interval

      [current-time + Start-time-length, current-time + End-time-length]

   where current-time is a current local time. The Repeat-time-length
   represents a repeat cycle/time, which is valid if the Options field
   is set to indicate the way to repeat is "repeat every Repeat-time-
   length". The Options field indicates a way to repeat. The Number-
   repeats indicates the number of repeats of the time interval above.

   In addition, the TLV includes an non zero grace-before and grace-
   after if graceful periods are configured. It contains an non zero
   elastic range lower bound and upper bound if there is an elastic
   range configured.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           Type (4)            |            Length             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Start-time-length                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         End-time-length                       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Repeat-time-length                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Options |       Number-repeats      |     GrB    |     GrA    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |      Elastic-Lower-Bound     |      Elastic-Upper-Bound       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o Start-time-length: The time length in seconds from a current
     local time to the time LSP starts to carry traffic.

   o End-time-length: The time length in seconds from a current local
     time to the time LSP ends carrying traffic.

   o Repeat-time-length: The time length in seconds after which LSP
     starts to carry traffic again for (End-time-length - Start-time-
     length).

   o Options: Indicates a way to repeat.

        Options = 1: repeat every day;

        Options = 2: repeat every week;

        Options = 3: repeat every month;

        Options = 4: repeat every year;

        Options = 5: repeat every Repeat-time-length.

   o Number-repeats: The number of repeats. In each of repeats, LSP
     carries traffic.

   o GrB (Grace-Before):   The graceful period time length in seconds
     before each of the time intervals represented by the recurrent
     time interval.

   o GrA (Grace-After):   The graceful period time length in seconds
     after each of the time intervals.

   o Elastic-Lower-Bound: The maximum amount of time in seconds that
     each of the time intervals can shift to lower/left.

   o Elastic-Upper-Bound: The maximum amount of time in seconds that
     each of the time intervals can shift to upper/right.

5.5. Messages for Temporal LSP

   This section presents and discusses two classes of messages. One
   class is the messages between a PCE and a PCC on the ingress of a
   temporal LSP for initiating and maintaining the LSP. The other is
   the messages between two PCEs, one of which acts as a PCC.

5.5.1. Messages between PCE and PCC on Ingress

   From function's point of view, there are four groups of messages: 1)
   LSP creation request messages, 2) LSP deletion request messages, 3)
   LSP creation response messages, and 4) LSP deletion response
   messages. A message for an LSP in the first two groups is sent from
   a PCE to the PCC on the ingress of the LSP. A message for an LSP in
   the last two groups is sent from the PCC on the ingress of the LSP to
   a PCE.

   A Path Computation LSP Initiate Request (PCInitiate) message with
   some extensions can be used for a message in the first two groups. A
   Path Computation LSP State Report (PCRpt) message with some
   extensions can be used for a message in the last two groups.

   For an LSP creation request, a PCInitiate message includes objects:
   SRP, LSP, END-POINTS, ERO and TIME-INTERVAL. SRP (Stateful PCE
   Request Parameters) object comprises an SRP-ID-number. LSP object
   comprises PLSP-ID of 0, and SYMBOLIC-PATH-NAME TLV with path name.
   END-POINTS object comprises the source and destination addresses of
   the LSP. ERO object comprise the path (i.e., ERO) for the LSP.
   TIME-INTERVAL object comprises the time intervals for the LSP (the
   path satisfies constraints for the LSP in each of the time
   intervals).

   For an LSP creation response, a PCRpt message includes objects: SRP,
   LSP, ERO and TIME-INTERVAL. SRP object comprises the SRP-ID-number
   in the corresponding LSP creation request message. LSP object
   comprises a PLSP-ID assigned to the LSP by the PCC, SYMBOLIC-PATH-
   NAME TLV with path name, C Flag set to 1 indicates that this LSP is
   created by the LSP creation request. ERO object comprise the path
   (i.e., ERO) for the LSP. TIME-INTERVAL object comprises the time
   intervals for the LSP.

   For an LSP deletion request, a PCInitiate message includes objects:
   SRP, LSP, and TIME-INTERVAL. SRP object comprises an SRP-ID-number
   and R (remove) flag set to 1. LSP object comprises the PLSP-ID for
   the LSP created. TIME-INTERVAL object comprises the time intervals
   for the LSP.

   For an LSP deletion response, a PCRpt message includes objects: SRP,
   LSP, and TIME-INTERVAL. SRP object comprises the SRP-ID-number in
   the corresponding LSP deletion request message. LSP object comprises
   R(Remove) flag set to 1 indicating that the LSP has been removed from
   the PCC, and LSP Identifiers TLV.

   Note: The PCC on the ingress of an LSP does not use any time
   intervals in the TIME-INTERVAL object received for signaling the LSP.

For just creating and deleting LSPs, we do not need to include any
TIME-INTERVAL object in a message if the PCE creates the LSP with a
sequence of time intervals at the beginning of each of the time
intervals and deletes the LSP at the end of each of the time
intervals.

Discussions: For an LSP having a time interval TLV with graceful
periods, we may create the LSP in the time period including the
graceful periods and the LSP has the reserved bandwidth during that
period (including the graceful periods).

Another option is that we create the LSP in the time period including
the graceful periods, but do not reserve any bandwidth for the LSP in
the beginning. The desired bandwidth for the LSP is reserved in the
time period without graceful periods.

After the graceful period before the time interval, the bandwidth for
the LSP is reserved through a update message from the PCE to the PCC
on the ingress of the LSP. After the time interval (i.e., just
before the graceful period after the time interval), the bandwidth
for the LSP is released through another update message from the PCE
to the PCC on the ingress of the LSP.

5.5.2. Messages between two PCEs

Figure below illustrates the format of a request message with a
optional TIME-INTERVAL object:

```
        <PCReq Message>::= <Common Header>
                           [<svec-list>]
                           <request-list>
        <request-list>::=<request>[<request-list>]
        <request>::= <RP> <END-POINTS> [<OF>] [<LSPA>] [<BANDWIDTH>]
                     [<metric-list>] [<RRO>[<BANDWIDTH>]] [<IRO>]
                     [<LOAD-BALANCING>]
                     [<TIME-INTERVAL>]
```

Figure 1: Format for Request Message

Below is the format of a reply message with a optional TIME-INTERVAL
object:

```
          <PCReq Message>::= <Common Header>
                             <response-list>
          <response-list>::=<response>[<response-list>]
          <response>::= <RP>
                        [<NO-PATH>]
                        [<attribute-list>]
                        [<path-list>]
          <path-list>::=<path>[<path-list>]
          <path>::= <ERO><attribute-list>[<TIME-INTERVAL>]
```

                    Figure 2: Format for Reply Message

6. Procedures

   This section focuses on the procedures for creating and deleting a
   temporal LSP. When a PCE receives a request for an LSP with a
   sequence of time intervals from a user or application, it computes a
   path satisfying the constraints for the LSP in each of the time
   intervals and reserved the bandwidth for the LSP along the path in
   each of the time intervals. And then it initiates the creation of
   the LSP in the network to carry traffic in each of the time
   intervals.

   Ther are a couple of ways for a PCE to create an LSP with a sequence
   of time intervals. One way is that the PCE initiates the creation of
   the LSP at the beginning of each of the time intervals. At the end
   of each of the time intervals or when a deletion request for the LSP
   received, the PCE initiates the deletion of the LSP.

6.1. Creating a Temporal LSP

   A procedure for creating a temporal LSP is as follows:

     Step 1:   A PCE receives a request for creating a temporal LSP from
       a user or application.

     Step 2:   The PCE computes a shortest path satisfying constraints
       for the LSP in the time intervals given. It reserves the
       bandwidth in TEDB on each of the links the LSP traverses for each
       of the time intervals and stores the information about the LSP
       into an LSP database.

     Step 3:   At the beginning of each of the time intervals, the PCE
       initiates the setup of the LSP in a network through sending an LSP
       creation request (e.g., a PCInitiate with LSP object with PLSP-
       ID=0) with the path for the LSP to the PCC on the ingress of the

LSP, which triggers RSVP-TE to signal the LSP along the path in
the network (Note that the RSVP-TE is not aware of any time
interval for the LSP and just sets up the LSP in a normal way).
The PCC sends an LSP creation response (e.g., a PCRpt) to the PCE
after the LSP is up.

Step 4:   The PCE receives the LSP creation response (e.g., the
PCRpt) from the PCC corresponding to the request and updates the
status of the LSP accordingly.

6.2. Deleting a Temporal LSP

Suppose that a temporal LSP has been created to carry traffic in a
sequence of time intervals. A procedure for deleting this temporal
LSP is as follows:

Step 1:   A PCE receives a request for deleting the temporal LSP
from an client, or the lifetime for the LSP in a time interval is
over and the LSP needs to be deleted.

Step 2:   The PCE finds the LSP in the LSP database and gets the
information about the LSP.

Step 3:   The PCE initiates the deletion of the LSP in the network
through sending an LSP deletion request (e.g., a PCInitiate with R
flag set and PLSP ID for the LSP) to the PCC on the ingress of the
LSP, which triggers the RSVP-TE to tear down the LSP in the
network (Note that the RSVP-TE is not aware of any time interval
for the LSP and just tears down the LSP in a normal way). The PCC
generates an LSP deletion response (e.g., a PCRpt with R flag set)
and sends it to the PCE after the LSP is torn down.

Step 4:   The PCE receives the LSP deletion response (e.g., the
PCRpt) from the PCC corresponding to the request and updates the
status of the LSP accordingly. For deleting the LSP completely as
requested, it releases the bandwidth reserved for the LSP in TEDB
for each of the time intervals and removes the information about
the LSP from the LSP database after the LSP is deleted.

7. Considerations on TEDB

The existing Traffic Engineering (TE) information in a TEDB
represents an unreserved bandwidth B_i at each of eight priority
levels for a link at one point of time, for example, at the current
time.

```
  Bandwidth
      ^
      |
   Bi |_____
      |
      |
      |
     -+--------------------------------------------------------> Time
      |
```
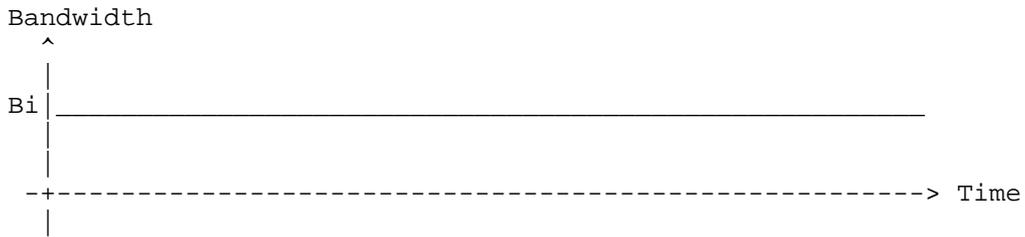
   This means that the link has bandwidth Bi at a priority level from
   now to forever until there is a change to it. Thus, a TE Label
   Switching Path (LSP) tunnel for a given time interval cannot be set
   up in advance using the information in the TEDB and the bandwidth
   cannot be reserved in advance for the LSP in the time interval given.

   TEDB needs to be enhanced for supporting temporal LSPs. Two options
   for enhancing TEDB are presented below.

7.1. TE Representation in Absolute Time

   Suppose that the amount of the unreserved bandwidth at a priority
   level for a link is Bj in a time interval from time Tj to Tk (k =
   j+1), where j = 0, 1, 2, ... The unreserved bandwidth for the link
   can be represented as

      [T0, B0], [T1, B1], [T2, B2], [T3, B3], ...

   This is an absolute time representation of bandwidths for a link.
   Time Tj (j = 0, 1, 2, ...) MUST be a synchronized time among all the
   elements involved.

```
    Bandwidth
       ^
       |                                        B3
       |             B1                 _____
       |          _____           |
       |B0       |                      |          B4
       |_____|          B2          |      _____
       |                 _____      |     |
       |                |          |     |     |
      -+----------------+----------+-----+-----+---------------> Time
       |T0       T1       T2              T3        T4
```

   If an LSP is completely deleted at time T and uses bandwidth B, then
   for every time interval/period (after time T) during which bandwidth

B is reserved for the LSP on a link, B is added to the link for that
interval/period.

If an LSP is to be up at time T and uses bandwidth B, then for every
time interval/period (after time T) during which bandwidth B is
reserved for the LSP on a link, B is subtracted from the link for
that interval/period.

7.2. TE Representation in Relative Time

Alternatively, a relative time representation of bandwidths for a
link can be used. For example, the amount of the unreserved
bandwidth at a priority level for a link is Bj during a series of
time intervals/periods can be expressed as

    [P0, B0], [P1, B1], [P2, B2], [P3, B3], ..., where
    Pj = Tk - Tj, k = (j+1) and j = 0, 1, 2, 3, ...

In this representation, every time Tj (j = 0, 1, 2, ...) can be a
local time. A timer may expire for every unit of time (e.g., every
second) and may trigger --P0, which decrements P0. When P0 = 0, P1
becomes P0, P2 becomes P1, and so on.

If an LSP is completely deleted at time T and uses bandwidth B, then
for every time interval/period (after time T) during which bandwidth
B is reserved for the LSP on a link, B is added to the link for that
interval/period.

If an LSP is to be up at time T and uses bandwidth B, then for every
time interval/period (after time T) during which bandwidth B is
reserved for the LSP on a link, B is subtracted from the link for
that interval/period.

An advantage of using relative time representation is that the times
or clocks on all the elements involved can be different.

8. Security Considerations

The mechanism described in this document does not raise any new
security issues for the PCEP protocols.

9. IANA Considerations

This section specifies requests for IANA allocation.

10. Acknowledgement

   The authors would like to thank Xufeng Liu for his valuable comments
   on this draft.

11. References

11.1. Normative References

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

   [RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element
              (PCE) Communication Protocol (PCEP)", RFC 5440,
              March 2009.

   [RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z.,
              and J. Meuric, "Extensions to the Path Computation Element
              Communication Protocol (PCEP) for Point-to-Multipoint
              Traffic Engineering Label Switched Paths", RFC 6006,
              September 2010.

   [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE",
              draft-ietf-pce-stateful-pce-11 (work in progress) ,
              April 2015.

   [I-D.ietf-pce-pce-initiated-lsp]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for PCE-initiated LSP Setup in a Stateful PCE
              Model", draft-ietf-pce-pce-initiated-lsp-04 (work in
              progress) , April 2015.

11.2. Informative References

   [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
              Element (PCE)-Based Architecture", RFC 4655, August 2006.

   [RFC5862] Yasukawa, S. and A. Farrel, "Path Computation Clients
              (PCC) - Path Computation Element (PCE) Requirements for
              Point-to-Multipoint MPLS-TE", RFC 5862, June 2010.

   [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.

Authors' Addresses

    Huaimo Chen
    Huawei Technologies
    Boston, MA
    US

    Email: huaimo.chen@huawei.com

    Mehmet Toy
    Comcast
    1800 Bishops Gate Blvd.
    Mount Laurel, NJ 08054
    USA

    Email: mehmet_toy@cable.comcast.com

    Vic Liu
    China Mobile
    No.32 Xuanwumen West Street, Xicheng District
    Beijing,   100053
    China

    Email: liuzhiheng@chinamobile.com

    Lei Liu
    Fijitsu
    USA

    Email: lliu@us.fujitsu.com

Internet Engineering Task Force   H. Chen
Internet-Draft   Huawei Technologies
Intended status: Standards Track   X. Liu
Expires: July 2, 2017   Ericsson
                                                           M. Toy
                                                          Verizon
                                                           V. Liu
                                                     China Mobile
                                                           L. Liu
                                                          Fujitsu
                                                     K. Pithewan
                                                          Infinera
                                                December 29, 2016

                   Extensions to PCEP for Temporal LSP
                        draft-chen-pce-tts-05.txt

Abstract

   For existing MPLS LSP tunnel services, it is hard for LSP tunnels to
   be booked in advance. In addition, once an LSP tunnel is set up, it
   is assumed to consume a certain amount of resources such as link
   bandwidth forever.

   Temporal LSP tunnel services (TTS) provides an easy way for us to
   book temporal LSP tunnels in advance. More importantly, a temporal
   LSP is an LSP with one or more time intervals and it is assumed to
   consume the resources and carry traffic only in these time intervals.

   This document specifies extensions to PCEP for computing a path for a
   temporal LSP, initiating and maintaining a temporal LSP with a
   sequence of time intervals, during each of which the LSP carries
   traffic.

material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 2, 2017.

Copyright Notice

Table of Contents

1. Introduction

   Once an existing multiprotocol label switching (MPLS) traffic
   engineering (TE) label switched path (LSP) is set up, it is assumed
   to carry traffic forever until it is down. When an MPLS TE LSP
   tunnel is up, it is assumed that the LSP consumes its reserved
   network resources forever even though the LSP may only use network
   resources during some period of time. As a result, the network
   resources are not used efficiently. Moreover, a tunnel service can
   not be reserved or booked in advance for a period of time or a
   sequence of time periods.

   Temporal LSP tunnel services (TTS) provides an easy way for us to
   book temporal LSP tunnels in advance. More importantly, a temporal
   LSP is an LSP with one or more time intervals and it is assumed to
   consume the resources and carry traffic only in each of these time
   intervals.

   This document specifies extensions to PCEP for computing a path for a
   temporal LSP, initiating and maintaining a temporal LSP with a period
   of time called a time interval or a sequence of time intervals. It
   is assumed that the LSP carries traffic during this time interval or
   each of these time intervals. Thus the network resources are
   efficiently used. More importantly, some new services can be
   provided. For example, a consumer can book a tunnel service in
   advance for a given time interval or a sequence of time intervals.
   Tunnel services may be scheduled.

2. Terminology

   A Time Interval: a time period from time Ta to time Tb.

   LSP: Label Switched Path. An LSP is a P2P (point-to-point) LSP or a
   P2MP (point-to-multipoiint) LSP.

   LSP with a time interval: LSP that carries traffic in the time
   interval.

   LSP with a sequence of time intervals: LSP that carries traffic in
   each of the time intervals.

   Temporal LSP: LSP with a time interval or LSP with a sequence of time
   intervals.

   TED: Traffic Engineering Database.

   CSPF: Constrained Shortest Path First.

LER: Label Edge Router.

This document uses terminologies defined in RFC5440.

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. Operations Overview

This section briefly describes some operations on a temporal LSP.

4.1. Simple Time Interval

For a temporal LSP, a user configures it with a time interval or a sequence of time intervals. A simple time interval is a time period from time Ta to time Tb, which may be represented as [Ta, Tb].

When an LSP is configured with time interval [Ta, Tb], a path satisfying the constraints for the LSP in the time interval is computed and the LSP along the path is set up to carry traffic from time Ta to time Tb.

In addition to simple time intervals, there are recurrent time intervals and elastic time intervals. Sometimes a simple time interval is called a time interval.

4.2. Recurrent Time Interval

A recurrent time interval represents a series of repeated simple time intervals. It has a simple time interval such as [Ta, Tb], a number of repeats such as 10 (repeats 10 times), and a repeat cycle/time such as a week (repeats every week). The recurrent time interval: "[Ta, Tb] repeats n times with repeat cycle C" represents n+1 simple time intervals as follows:

   [Ta, Tb], [Ta+C, Tb+C], [Ta+2C, Tb+2C], ..., [Ta+nC, Tb+nC]

When an LSP is configured with a recurrent time interval such as "[Ta, Tb] repeats 10 times with a repeat cycle a week" (representing 11 simple time intervals), a path satisfying the constraints for the LSP in each of the simple time intervals represented by the recurrent time interval is computed and the LSP along the path is set up to

carry traffic in each of the simple time intervals.

4.3. Elastic Time Interval

   An elastic time interval is a time interval with an elastic range,
   which is represented as within -P and Q, where P and Q is an amount
   of time such as 300 seconds. P is called elastic range lower bound
   and Q is called elastic range upper bound.

   For a simple time interval such as [Ta, Tb] with an elastic range,
   elastic time interval: "[Ta, Tb] within -P and Q" means a time period
   from (Ta+X) to (Tb+X), where -P <= X <= Q. Note that both Ta and Tb
   may be shifted the same X.

   When an LSP is configured with elastic time interval "[Ta, Tb] within
   -P and Q", a path is computed such that the path satisfies the
   constraints for the LSP in the time period from (Ta+X) to (Tb+X) and
   |X| is the minimum value from 0 to max(P, Q). That is that [Ta+X,
   Tb+X] is the time interval closest to time interval [Ta, Tb] within
   the elastic range. The LSP along the path is set up to carry traffic
   in the time period from (Ta+X) to (Tb+X).

   Similarly, for a recurrent time interval with an elastic range,
   elastic time interval: "[Ta, Tb] repeats n times with repeat cycle C
   within -P and Q" represents n+1 simple elastic time intervals as
   follows:

      [Ta+X0, Tb+X0], [Ta+C+X1, Tb+C+X1], ..., [Ta+nC+Xn, Tb+nC+Xn]

   where -P <= Xi <= Q, i = 0, 1, 2, ..., n.

   If a user wants to keep the same repeat cycle between any two
   adjacent time intervals, elastic time interval: "[Ta, Tb] repeats n
   times with repeat cycle C within -P and Q SYNC" may be used, which
   represents n+1 simple elastic time intervals as follows:

      [Ta+X, Tb+X], [Ta+C+X, Tb+C+X], ..., [Ta+nC+X, Tb+nC+X]

   where -P <= X <= Q.

4.4. Changes to Time Interval

   After a temporal LSP is configured, a user may change its parameters
   including some of the time intervals configured. A new time interval
   may be added, an existing time interval may be removed or changed.

When a new time interval is added to an existing LSP, a path
satisfying the constraints for the LSP in the time interval is
computed and the LSP along the path is set up to carry traffic in the
time interval.

When an existing time interval is removed from an existing LSP, the
time interval is deleted from the lifetime of the LSP. If the
lifetime is over, the LSP is deleted.

A change to an existing time interval may generate some of four
possible results:

1. The existing time interval is extended for a time period EA after
    the existing time period;

2. The existing time interval is extended for a time period EB
    before the existing time period;

3. The existing time interval is shrunk for a time period SA from
    the end of the existing time period; and

4. The existing time interval is shrunk for a time period SB from
    the beginning of the existing time period.

When an existing time interval for an LSP is extended, a path
satisfying the constraints for the LSP in the extended time interval
is computed and the LSP along the path is set up to carry traffic in
the extended time interval. If the LSP is already up to carry
traffic in the existing time interval, the lifetime of the LSP is
extended for time period EA following the existing time interval.

When an existing time interval for an LSP is shrunk, the shrunk time
periods are removed from the lifetime of the LSP.

4.5. Graceful Periods

For a temporal LSP, a user may want to have some graceful periods for
each or some of the time intervals for the LSP. Two graceful periods
may be configured for a time interval. One is the graceful period
before the time interval, called grace-before, which extends the
lifetime of the LSP for grace-before (such as 30 seconds) before the
time interval. The other is the one after the time interval, called
grace-after, which extends the lifetime of the LSP for grace-after
(such as 60 seconds) after the time interval.

When an LSP is configured with a simple time interval such as [Ta,
Tb] with graceful periods such as grace-before GB and grace-after GA,
a path is computed such that the path satisfies the constraints for

the LSP in the time period from Ta to Tb. The LSP along the path is
set up to carry traffic in the time period from (Ta-GB) to (Tb+GA).
During graceful periods from (Ta-GB) to Ta and from Tb to (Tb+GA),
the LSP is up to carry traffic (maybe in best effort).

5. Extensions to PCEP

   This section describes the extensions to PCEP for computing paths for
   temporal LSP, initiating and maintaining temporal LSPs.

5.1. Capability TLV in Existing PCE Discovery Protocol

   There are a couple of options for advertising a PCE capability for
   computing paths for temporal LSP, initiating and maintaining temporal
   LSPs.

   The first option is to define a new flag in the OSPF and ISIS PCE
   Capability Flags to indicate the capability that a PCE is capable to
   compute paths for temporal LSPs, initiate and maintain temporal LSPs.
   This includes the capability of computing both a path for a temporal
   P2MP LSP and a path for a temporal P2P LSP.

   The second option is to define three new flags. The first new flag
   in the OSPF and ISIS PCE Capability Flags indicates the capability
   that a PCE is capable to compute a path for a temporal P2MP LSP; the
   second new flag indicates the capability that a PCE is capable to
   compute a path for a temporal P2P LSP; and the third new flag
   indicates the capability that a PCE is capable to initiate and
   maintain a temporal LSP.

   The format of the PCE-CAP-FLAGS sub-TLV is 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
      +-+
      | Type = 5 | Length |
      +-+
      | |
      ~   PCE Capability Flags   ~
      +-+

        Type: 5
        Length: Multiple of 4 octets
        Value: This contains an array of units of 32-bit flags
                   numbered from the most significant as bit zero, where
                   each bit represents one PCE capability.

The following capability bits have been assigned by IANA:

```
     Bit         Capabilities
      0          Path computation with GMPLS link constraints
      1          Bidirectional path computation
      2          Diverse path computation
      3          Load-balanced path computation
      4          Synchronized path computation
      5          Support for multiple objective functions
      6          Support for additive path constraints
                 (max hop count, etc.)
      7          Support for request prioritization
      8          Support for multiple requests per message
      9          Global Concurrent Optimization (GCO)
      10         P2MP path computation
      ...
```

Reserved bits SHOULD be set to zero on transmission and MUST be ignored on receipt.

For the first option, one bit such as bit 16 may be assigned to indicate that a PCE is capable to compute paths for temporal LSPs, initiate and maintain temporal LSPs.

```
     Bit         Capabilities
      16         Path computation for temporal LSPs, initiation
                 and maintenance of temporal LSPs
     17-31       Reserved for future assignments by IANA.
```

For the second option, one bit such as bit 16 may be assigned to indicate that a PCE is capable to compute a path for a temporal P2MP LSP; another bit such as bit 17 may be assigned to indicate that a PCE is capable to compute a path for a temporal P2P LSP; and yet another bit such as bit 18 may be assigned to indicate that a PCE is capable to initiate and maintain temporal LSPs.

```
     Bit         Capabilities
      16         Path computation for temporal P2MP LSP
      17         Path computation for temporal P2P LSP
      18         Initiation and maintenance of temporal LSP
     19-31       Reserved for future assignments by IANA.
```

5.2. Open Message Extension

   If a PCE does not advertise its capability related to computation of
   paths for a temporal LSP, initiation and maintenance of a temporal
   LSP during discovery, PCEP should be used to allow a PCC to discover,
   during the Open Message Exchange, which PCEs are capable of
   supporting computation of a path for a temporal LSP, initiation and
   maintenance of a temporal LSP.

   To achieve this, one option is to extend the PCEP OPEN object by
   defining new flag bits in the value field of an existing capability
   TLV such as stateful PCE capability TLV in the same way as the PCE
   Capability Flags described in the previous section. Another option
   is to extend the PCEP OPEN object by defining a new optional TLV to
   indicate the PCE's capability to compute paths for a temporal LSP,
   initiate and maintain a temporal LSP.

   For the second option, we need to request IANA to allocate a value
   such as 10 from the "PCEP TLV Type Indicators" subregistry, as
   documented in Section below ("Temporal LSP Capability TLV"). The
   description is "temporal LSP capable", and the length value is 2
   bytes. The value field is set to indicate the capability of a PCE
   for computation of paths for a temporal LSP, initiation and
   maintenance of a temporal LSP in details. We can use flag bits in
   the value field in the same way as the PCE Capability Flags described
   in the previous section.

   The inclusion of this TLV in an OPEN object indicates that the sender
   can compute paths for a temporal LSP, initiate and maintain a
   temporal LSP.

   The capability TLV is meaningful only for a PCE, so it will typically
   appear only in one of the two Open messages during PCE session
   establishment. However, in case of PCE cooperation (e.g., inter-
   domain), when a PCE behaving as a PCC initiates a PCE session it
   SHOULD also indicate its capabilities.

5.3. RP Object Extension

   The following flags are added into the RP Object:

   A T bit is added in the flag bits field of the RP object to tell a
   receiver of a message that the message is for (computing paths for a
   temporal LSP) a temporal LSP.

            o T (Temporal LSP bit - 1 bit):

                0: This indicates that this is not a message
                   for a temporal LSP.

                1: This indicates that this is a message
                   for a temporal LSP.

    The IANA request is referenced in Section below (Request Parameter
    Bit Flags) of this document.

    This T bit with the N bit defined in RFC6006 can indicate whether the
    message is for a temporal P2P LSP or P2MP LSP.

       o T = 1 and N = 0: This indicates that this is a message
                          for a temporal P2P LSP
       o T = 1 and N = 1: This indicates that this is a message
                          for a temporal P2MP LSP

5.4. TIME INTERVAL Object

    For a TIME-INTERVAL object, its Class is to be assigned by IANA, here
    we use 18, which may be changed late. Its OT is 1, exact number to
    be assigned by IANA. The format of a TIME-INTERVAL object body is
    illustrated below, which comprises a number of time interval TLVs.

       Object-Class: TBD (18),   OT = 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
     +-+
     | (Object Body containing Time Interval TLVs) |
     ~   ~
     +-+

    A time interval TLV may be a relative time interval TLV or an
    absolute time interval TLV, which are two different representations
    of a time interval. Their advantages and disadvantages are discussed
    below.

5.4.1. Absolute Time Interval TLV

    The format of an absolute time interval TLV (Type = 1) for an LSP is
    illustrated below. It mainly contains a Start-time and an End-time,
    representing time interval [Start-time, End-time]. Both of these two

times are the times that are synchronized among all the elements
involved. Thus the clocks on all the elements MUST be synchronized
if an absolute time interval TLV is used. The time period
represented in an absolute time interval TLV is more accurate.

In addition, it contains an non zero grace-before and grace-after if
graceful periods are configured. It includes an non zero elastic
range lower bound and upper bound if there is an elastic range
configured.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type (1)           |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Start-time                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           End-time                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             GrB               |              GrA              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Elastic-Lower-Bound     |      Elastic-Upper-Bound      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

  o Start-time: The time LSP starts to carry traffic.

  o End-time: The time LSP stops carrying traffic.

  o GrB (Grace-Before):   The graceful period time length in seconds
    before time interval [Start-time, End-time].

  o GrA (Grace-After):   The graceful period time length in seconds
    after time interval [Start-time, End-time].

  o Elastic-Lower-Bound: The maximum amount of time in seconds that
    time interval [Start-time, End-time] can shift to lower/left.

  o Elastic-Upper-Bound: The maximum amount of time in seconds that
    time interval [Start-time, End-time] can shift to upper/right.

  Discussions: Optionally, we may define three TLVs below:

  1. an absolute time interval TLV containing only a Start-time and an
      End-time;

  2. an elastic range TLV containing just an elastic range lower bound
      and upper bound; and

   3. a graceful period TLV containing only a grace-before and a grace-
       after.

   If a time interval is with an elastic range, an absolute time
   interval TLV followed by an elastic range TLV is used. If a time
   interval is with graceful periods, an absolute time interval TLV
   followed by a graceful period TLV is used.

5.4.2. Relative Time Interval TLV

   The format of a relative time interval TLV (Type = 2) for an LSP is
   shown below. It mainly contains a Start-time-length and an End-time-
   length, representing the time interval below for the LSP:

    [current-time + Start-time-length, current-time + End-time-length]

   where current-time is a current local time. When a time interval
   from time Ta to time Tb is configured on a node/element, these two
   time lengths are the time lengths that are computed on the node using
   a current local time as follows.

      Start-time-length = Ta - current-time;
      End-time-length   = Tb - current-time;

   For a relative time interval TLV, the clocks/times on all the
   elements involved can be different. But the time period represented
   in a relative time interval TLV on one element/node may be shifted a
   little bit from another element's point of view since transmitting
   the TLV from one element to another takes a little time, which is
   hard to be considered accurately.

   The TLV also includes an non zero grace-before and grace-after if
   graceful periods are configured. It contains an non zero elastic
   range lower bound and upper bound if there is an elastic range
   configured.

```
  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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |            Type (2)           |             Length            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                       Start-time-length                       |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                        End-time-length                        |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |             GrB               |              GrA              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |      Elastic-Lower-Bound      |      Elastic-Upper-Bound      |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o Start-time-length: The time length in seconds from a current
     local time to the time LSP starts to carry traffic.

   o End-time-length: The time length in seconds from a current local
     time to the time LSP ends carrying traffic.

   o GrB (Grace-Before):   The graceful period time length in seconds
     before the time interval above for the LSP.

   o GrA (Grace-After):   The graceful period time length in seconds
     after the time interval above for the LSP.

   o Elastic-Lower-Bound: The maximum amount of time in seconds that
     the time interval above for the LSP can shift to lower/left.

   o Elastic-Upper-Bound: The maximum amount of time in seconds that
     the time interval above can shift to upper/right.

5.4.3. Recurrent Absolute Time Interval TLV

   The format of a recurrent absolute time interval TLV (Type = 3) for
   an LSP is illustrated below. It mainly contains a Start-time, an
   End-time, a Repeat-time-length, a Options field and a Number-repeats.

   The Start-time and End-time represents time interval [Start-time,
   End-time]. The Repeat-time-length represents a repeat cycle/time,
   which is valid if the Options field is set to indicate the way to
   repeat is "repeat every Repeat-time-length". The Options field
   indicates a way to repeat. The Number-repeats indicates the number
   of repeats of time interval [Start-time, End-time].

   In addition, the TLV includes an non zero grace-before and grace-
   after if graceful periods are configured. It contains an non zero

elastic range lower bound and upper bound if there is an elastic
range configured.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type (3)           |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Start-time                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            End-time                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Repeat-time-length                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Options |      Number-repeats     |         Reserved (0)      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             GrB               |              GrA              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Elastic-Lower-Bound      |      Elastic-Upper-Bound      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

o Start-time: The time LSP starts to carry traffic.

o End-time: The time LSP stops carrying traffic.

o Repeat-time-length: The time length in seconds after which LSP
  starts to carry traffic again for (End-time - Start-time).

o Options: Indicates a way to repeat.

      Options = 1: repeat every day;

      Options = 2: repeat every week;

      Options = 3: repeat every month;

      Options = 4: repeat every year;

      Options = 5: repeat every Repeat-time-length.

o Number-repeats: The number of repeats. In each of repeats, LSP
  carries traffic.

o GrB (Grace-Before):   The graceful period time length in seconds
  before each of the time intervals represented by the recurrent
  time interval.

    o GrA (Grace-After):   The graceful period time length in seconds
      after each of the time intervals.

    o Elastic-Lower-Bound: The maximum amount of time in seconds that
      each of the time intervals can shift to lower/left.

    o Elastic-Upper-Bound: The maximum amount of time in seconds that
      each of the time intervals can shift to upper/right.

5.4.4. Recurrent Relative Time Interval TLV

   The format of a recurrent relative time interval TLV (Type = 4) is
   shown below. It mainly contains a Start-time-length, an End-time-
   length, a Repeat-time-length, a Options field and a Number-repeats.

   The Start-time-length and End-time-length represents time interval

     [current-time + Start-time-length, current-time + End-time-length]

   where current-time is a current local time. The Repeat-time-length
   represents a repeat cycle/time, which is valid if the Options field
   is set to indicate the way to repeat is "repeat every Repeat-time-
   length". The Options field indicates a way to repeat. The Number-
   repeats indicates the number of repeats of the time interval above.

   In addition, the TLV includes an non zero grace-before and grace-
   after if graceful periods are configured. It contains an non zero
   elastic range lower bound and upper bound if there is an elastic
   range configured.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type (4)           |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Start-time-length                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        End-time-length                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Repeat-time-length                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Options |       Number-repeats      |        Reserved (0)      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             GrB               |               GrA             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Elastic-Lower-Bound     |       Elastic-Upper-Bound     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   o Start-time-length: The time length in seconds from a current
     local time to the time LSP starts to carry traffic.

   o End-time-length: The time length in seconds from a current local
     time to the time LSP stops carrying traffic.

   o Repeat-time-length: The time length in seconds after which LSP
     starts to carry traffic again for (End-time-length - Start-time-
     length).

   o Options: Indicates a way to repeat.

        Options = 1: repeat every day;

        Options = 2: repeat every week;

        Options = 3: repeat every month;

        Options = 4: repeat every year;

        Options = 5: repeat every Repeat-time-length.

   o Number-repeats: The number of repeats. In each of repeats, LSP
     carries traffic.

   o GrB (Grace-Before):   The graceful period time length in seconds
     before each of the time intervals represented by the recurrent
     time interval.

   o GrA (Grace-After):   The graceful period time length in seconds
     after each of the time intervals.

   o Elastic-Lower-Bound: The maximum amount of time in seconds that
     each of the time intervals can shift to lower/left.

   o Elastic-Upper-Bound: The maximum amount of time in seconds that
     each of the time intervals can shift to upper/right.

5.5. Messages for Temporal LSP

   This section presents and discusses two classes of messages. One
   class is the messages between a PCE and a PCC on the ingress of a
   temporal LSP for initiating and maintaining the LSP. The other is
   the messages between two PCEs, one of which acts as a PCC.

5.5.1. Messages between PCE and PCC on Ingress

   From function's point of view, there are four groups of messages:

   1. LSP creation request messages,

   2. LSP deletion request messages,

   3. LSP creation response messages, and

   4. LSP deletion response messages.

   A message for an LSP in the first two groups is sent from a PCE to
   the PCC on the ingress of the LSP. A message for an LSP in the last
   two groups is sent from the PCC on the ingress of the LSP to a PCE.

   A Path Computation LSP Initiate Request (PCInitiate) message without
   any extensions can be used for a message in the first two groups. A
   Path Computation LSP State Report (PCRpt) message without any
   extensions can be used for a message in the last two groups.

   Alternatively, a PCInitiate message with some optional extensions
   such as TIME-INTERVAL can be used for a message in the first two
   groups. A PCRpt message with some optional extensions such as TIME-
   INTERVAL can be used for a message in the last two groups.

   For an LSP creation request, a PCInitiate message includes objects:
   SRP, LSP, END-POINTS, ERO and optional TIME-INTERVAL. SRP (Stateful
   PCE Request Parameters) object comprises an SRP-ID-number. LSP
   object comprises PLSP-ID of 0, and SYMBOLIC-PATH-NAME TLV with path
   name. END-POINTS object comprises the source and destination
   addresses of the LSP. ERO object comprise the path (i.e., ERO) for
   the LSP. TIME-INTERVAL object comprises the time intervals for the
   LSP (the path satisfies constraints for the LSP in each of the time
   intervals).

   For an LSP creation response, a PCRpt message includes objects: SRP,
   LSP, ERO and optional TIME-INTERVAL. SRP object comprises the SRP-
   ID-number in the corresponding LSP creation request message. LSP
   object comprises a PLSP-ID assigned to the LSP by the PCC, SYMBOLIC-
   PATH-NAME TLV with path name, C Flag set to 1 indicates that this LSP
   is created by the LSP creation request. ERO object comprise the path
   (i.e., ERO) for the LSP. TIME-INTERVAL object comprises the time
   intervals for the LSP.

   For an LSP deletion request, a PCInitiate message includes objects:
   SRP, LSP, and optional TIME-INTERVAL. SRP object comprises an SRP-
   ID-number and R (remove) flag set to 1. LSP object comprises the

PLSP-ID for the LSP created. TIME-INTERVAL object comprises the time intervals for the LSP.

For an LSP deletion response, a PCRpt message includes objects: SRP, LSP, and optional TIME-INTERVAL. SRP object comprises the SRP-ID-number in the corresponding LSP deletion request message. LSP object comprises R(Remove) flag set to 1 indicating that the LSP has been removed from the PCC, and LSP Identifiers TLV.

Note: The PCC on the ingress of an LSP does not use any time intervals in the TIME-INTERVAL object received for signaling the LSP. For just creating and deleting LSPs, we do not need to include any TIME-INTERVAL object in a message if the PCE creates the LSP with a sequence of time intervals at the beginning of each of the time intervals and deletes the LSP at the end of each of the time intervals.

Discussions: For an LSP having a time interval TLV with graceful periods, we may create the LSP in the time period including the graceful periods and the LSP has the reserved bandwidth during that period (including the graceful periods).

Another option is that we create the LSP in the time period including the graceful periods, but do not reserve any bandwidth for the LSP in the beginning. The desired bandwidth for the LSP is reserved in the time period without graceful periods.

After the graceful period before the time interval, the bandwidth for the LSP is reserved through a update message from the PCE to the PCC on the ingress of the LSP. After the time interval (i.e., just before the graceful period after the time interval), the bandwidth for the LSP is released through another update message from the PCE to the PCC on the ingress of the LSP.

5.5.2. Messages between two PCEs

Figure below illustrates the format of a request message with a optional TIME-INTERVAL object for computing paths for a temporal LSP with a sequence of time intervals:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>
<request-list>::=<request>[<request-list>]
<request>::= <RP> <END-POINTS> [<OF>] [<LSPA>] [<BANDWIDTH>]
             [<metric-list>] [<RRO>[<BANDWIDTH>]] [<IRO>]
             [<LOAD-BALANCING>]
             [<TIME-INTERVAL>]
```

Figure 1: Format for Request Message

Below is the format of a reply message with a optional TIME-INTERVAL object:

```
<PCReq Message>::= <Common Header>
                   <response-list>
<response-list>::=<response>[<response-list>]
<response>::= <RP>
              [<NO-PATH>]
              [<attribute-list>]
              [<path-list>]
<path-list>::=<path>[<path-list>]
<path>::= <ERO><attribute-list>[<TIME-INTERVAL>]
```

Figure 2: Format for Reply Message

6. Procedures

   This section focuses on the procedures for creating and deleting a
   temporal LSP. When a PCE receives a request for an LSP with a
   sequence of time intervals from a user or application, it computes a
   path satisfying the constraints for the LSP in each of the time
   intervals and reserved the bandwidth for the LSP along the path in
   each of the time intervals. And then it initiates the creation of
   the LSP in the network to carry traffic in each of the time
   intervals.

   There are a couple of ways for a PCE to create an LSP with a sequence
   of time intervals. One way is that the PCE initiates the creation of
   the LSP at the beginning of each of the time intervals. At the end
   of each of the time intervals or when a deletion request for the LSP
   received, the PCE initiates the deletion of the LSP.

   Another way is that the PCE initiates the creation of the LSP at or
   before the beginning of the first time interval and the deletion of

the LSP at the end of the last time interval. At the start of each
time interval, the PCE initiates the update of the LSP with the
reserved resource such as link bandwidth. At the end of the each
time interval, the PCE initiates the update of the LSP with zero
resource.

We will focus on the first way below.

6.1. Creating a Temporal LSP

A procedure for creating a temporal LSP is as follows:

   Step 1:   A PCE receives a request for creating a temporal LSP from
      a user or application and stores the parameters of the LSP into an
      LSP database (LSPDB) such as LSP State Database. The parameters
      include a number of time intervals for the LSP.

   Step 2:   The PCE computes a shortest path satisfying constraints
      for the LSP in each of the time intervals given. It reserves the
      resources such as the bandwidth in TED on each of the links the
      LSP traverses for each of the time intervals and stores the
      information about the LSP into the LSPDB. The information
      includes the paths computed for the LSP and the resources such as
      link bandwidth reserved for the LSP.

   Step 3:   At the beginning of each of the time intervals, the PCE
      initiates the setup of the LSP in a network through sending an LSP
      creation request (e.g., a PCInitiate with LSP object with PLSP-
      ID=0) with the path for the LSP to the PCC on the ingress of the
      LSP, which triggers RSVP-TE to signal the LSP along the path in
      the network (Note that the RSVP-TE is not aware of any time
      interval for the LSP and just sets up the LSP in a normal way).
      The PCC sends an LSP creation response (e.g., a PCRpt) to the PCE
      after the LSP is up.

   Step 4:   The PCE receives the LSP creation response (e.g., the
      PCRpt) from the PCC corresponding to the request and updates the
      status of the LSP in the LSPDB accordingly.

6.2. Deleting a Temporal LSP

Suppose that a temporal LSP has been created to carry traffic in a
sequence of time intervals. A procedure for deleting this temporal
LSP is as follows:

Step 1:   A PCE receives a request for deleting the temporal LSP
   from an client, or the lifetime for the LSP in a time interval is
   over and the LSP needs to be deleted.

Step 2:   The PCE finds the LSP in the LSPDB and gets the
   information about the LSP.

Step 3:   The PCE initiates the deletion of the LSP in the network
   through sending an LSP deletion request (e.g., a PCInitiate with R
   flag set and PLSP ID for the LSP) to the PCC on the ingress of the
   LSP, which triggers the RSVP-TE to tear down the LSP in the
   network (Note that the RSVP-TE is not aware of any time interval
   for the LSP and just tears down the LSP in a normal way). The PCC
   generates an LSP deletion response (e.g., a PCRpt with R flag set)
   and sends it to the PCE after the LSP is torn down.

Step 4:   The PCE receives the LSP deletion response (e.g., the
   PCRpt) from the PCC corresponding to the request and updates the
   status of the LSP in the LSPDB accordingly. For deleting the LSP
   completely as requested, it releases the resources such as the
   link bandwidth reserved for the LSP in TED for each of the time
   intervals and removes the information about the LSP from the LSPDB
   after the LSP is deleted.

7. Considerations on TED

   The existing Traffic Engineering (TE) information in a TED represents
   an unreserved bandwidth Bi at each of eight priority levels for a
   link at one point of time, for example, at the current time.

```
   Bandwidth
      ^
      |
   Bi |_____
      |
      |
     -+------------------------------------------------------> Time
      |
```
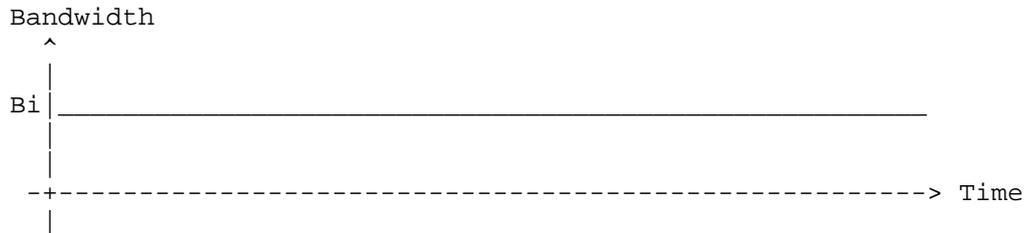
   This means that the link has bandwidth Bi at a priority level from
   now to forever until there is a change to it. Thus, a TE Label
   Switching Path (LSP) tunnel for a given time interval cannot be set
   up in advance using the information in the TED and the bandwidth
   cannot be reserved in advance for the LSP in the time interval given.

   TED needs to be enhanced for supporting temporal LSPs. Two options

for enhancing TED are presented below.

7.1. TE Representation in Absolute Time

   Suppose that the amount of the unreserved bandwidth at a priority
   level for a link is Bj in a time interval from time Tj to Tk (k =
   j+1), where j = 0, 1, 2, ... The unreserved bandwidth for the link
   can be represented as

       [T0, B0], [T1, B1], [T2, B2], [T3, B3], ...

   This is an absolute time representation of bandwidths for a link.
   Time Tj (j = 0, 1, 2, ...) MUST be a synchronized time among all the
   elements involved.

```
    Bandwidth
       ^
       |                                        B3
       |         B1                       _____
       |       _____
       |B0                                                B4
       |_____            B2                     _____
       |                  _____
       |
       |
      -+------------------------------------------------------> Time
       |T0        T1         T2           T3          T4
```

   If an LSP is completely deleted at time T and uses bandwidth B, then
   for every time interval/period (after time T) during which bandwidth
   B is reserved for the LSP on a link, B is added to the link for that
   interval/period.

   If an LSP is to be up at time T and uses bandwidth B, then for every
   time interval/period (after time T) during which bandwidth B is
   reserved for the LSP on a link, B is subtracted from the link for
   that interval/period.

7.2. TE Representation in Relative Time

   Alternatively, a relative time representation of bandwidths for a
   link can be used. For example, the amount of the unreserved
   bandwidth at a priority level for a link is Bj during a series of
   time intervals/periods can be expressed as

       [P0, B0], [P1, B1], [P2, B2], [P3, B3], ..., where
       Pj = Tk - Tj, k = (j+1) and j = 0, 1, 2, 3, ...

In this representation, every time Tj (j = 0, 1, 2, ...) can be a
local time. A timer may expire for every unit of time (e.g., every
second) and may trigger --P0, which decrements P0. When P0 = 0, P1
becomes P0, P2 becomes P1, and so on.

If an LSP is completely deleted at time T and uses bandwidth B, then
for every time interval/period (after time T) during which bandwidth
B is reserved for the LSP on a link, B is added to the link for that
interval/period.

If an LSP is to be up at time T and uses bandwidth B, then for every
time interval/period (after time T) during which bandwidth B is
reserved for the LSP on a link, B is subtracted from the link for
that interval/period.

An advantage of using relative time representation is that the times
or clocks on all the elements involved can be different.

8. Security Considerations

   The mechanism described in this document does not raise any new
   security issues for the PCEP protocols.

9. IANA Considerations

   This section specifies requests for IANA allocation.

10. Acknowledgement

   The authors would like to thank everyone who give his/her valuable
   comments on this draft.

11. References

11.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/
              RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,

                  <http://www.rfc-editor.org/info/rfc5440>.

   [RFC6006] Zhao, Q., Ed., King, D., Ed., Verhaeghe, F., Takeda, T.,
              Ali, Z., and J. Meuric, "Extensions to the Path
              Computation Element Communication Protocol (PCEP) for
              Point-to-Multipoint Traffic Engineering Label Switched
              Paths", RFC 6006, DOI 10.17487/RFC6006, September 2010,
              <http://www.rfc-editor.org/info/rfc6006>.

   [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE",
              draft-ietf-pce-stateful-pce-18 (work in progress),
              December 2016.

   [I-D.ietf-pce-pce-initiated-lsp]
              Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
              Extensions for PCE-initiated LSP Setup in a Stateful PCE
              Model", draft-ietf-pce-pce-initiated-lsp-07 (work in
              progress), July 2016.

11.2. Informative References

   [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
              Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/
              RFC4655, August 2006,
              <http://www.rfc-editor.org/info/rfc4655>.

   [RFC5862] Yasukawa, S. and A. Farrel, "Path Computation Clients
              (PCC) - Path Computation Element (PCE) Requirements for
              Point-to-Multipoint MPLS-TE", RFC 5862, DOI 10.17487/
              RFC5862, June 2010,
              <http://www.rfc-editor.org/info/rfc5862>.

   [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V.,
              and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP
              Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001,
              <http://www.rfc-editor.org/info/rfc3209>.

Authors' Addresses

   Huaimo Chen
   Huawei Technologies
   Boston, MA
   US

   Email: huaimo.chen@huawei.com

   Xufeng Liu
   Ericsson
   USA

   Email: xliu@kuatrotech.com

   Mehmet Toy
   Verizon
   USA

   Email: mehmet.toy@verizon.com

   Vic Liu
   China Mobile
   No.32 Xuanwumen West Street, Xicheng District
   Beijing,   100053
   China

   Email: liu.cmri@gmail.com

   Lei Liu
   Fujitsu
   USA

   Email: lliu@us.fujitsu.com

   Khuzema Pithewan
   Infinera

   Email: kpithewan@infinera.com

PCE Working Group   S. Sivabalan
Internet-Draft   Cisco Systems, Inc.
Intended status: Standards Track   J. Tantsura
Expires: November 5, 2018   Nuage Networks
                                                              I. Minei
                                                          Google, Inc.
                                                              R. Varga
                                              Pantheon Technologies SRO
                                                           J. Hardwick
                                                   Metaswitch Networks
                                                           May 4, 2018

                 Conveying path setup type in PCEP messages
                     draft-ietf-pce-lsp-setup-type-10

Abstract

   A Path Computation Element (PCE) can compute Traffic Engineering (TE)
   paths through a network that are subject to various constraints.
   Currently, TE paths are Label Switched Paths (LSPs) which are set up
   using the RSVP-TE signaling protocol. However, other TE path setup
   methods are possible within the PCE architecture. This document
   proposes an extension to the PCE communication protocol (PCEP) to
   allow support for different path setup methods over a given PCEP
   session.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on November 5, 2018.

Copyright Notice

Table of Contents

1. Introduction

   [RFC5440] describes the Path Computation Element communication
   Protocol (PCEP) for communication between a Path Computation Client
   (PCC) and a Path Computation Element (PCE), or between a PCE and a
   PCE. A PCC requests a path subject to various constraints and
   optimization criteria from a PCE. The PCE responds to the PCC with a
   hop-by-hop path in an Explicit Route Object (ERO). The PCC uses the
   ERO to set up the path in the network.

   [RFC8231] specifies extensions to PCEP that allow a PCC to delegate
   its LSPs to a PCE. The PCE can then update the state of LSPs

delegated to it. In particular, the PCE may modify the path of an
LSP by sending a new ERO. The PCC uses this ERO to re-route the LSP
in a make-before-break fashion. [RFC8281] specifies a mechanism
allowing a PCE to dynamically instantiate an LSP on a PCC by sending
the ERO and the characteristics of the LSP. The PCC creates the LSP
using the ERO and other attributes sent by the PCE.

So far, PCEP and its extensions have assumed that the TE paths are
label switched and are established via the RSVP-TE protocol.
However, other methods of LSP setup are possible in the PCE
architecture (see [RFC4655] and [RFC4657]). This document
generalizes PCEP to allow other LSP setup methods to be used. It
defines two new TLVs and specifies the base procedures to facilitate
this, as follows.

   o The PATH-SETUP-TYPE-CAPABILITY TLV, which allows a PCEP speaker to
      announce which LSP setup methods it supports when the PCEP session
      is established.

   o The PATH-SETUP-TYPE TLV, which allows a PCEP speaker to specify
      which setup method should be used for a given LSP. When multiple
      path setup types are deployed in a network, a given PCEP session
      may have to simultaneously support more than one path setup type.
      A PCEP speaker uses the PATH-SETUP-TYPE TLV to explicitly indicate
      the intended path setup type in the appropriate PCEP messages,
      unless the path setup type is RSVP-TE (which is assumed to be the
      path setup type if no other setup type is indicated). This is so
      that both the PCC and the PCE can take the necessary steps to set
      up the path.

   This document defines a path setup type code for RSVP-TE. When a new
   path setup type (other than RSVP-TE) is introduced for setting up a
   path, a path setup type code and, optionally, a sub-TLV pertaining to
   the new path setup type will be defined by the document that
   specifies the new path setup type.

1.1. Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in BCP
   14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

2. Terminology

    The following terminologies are used in this document:

    ERO: Explicit Route Object.

    LSR: Label Switching Router.

    PCC: Path Computation Client.

    PCE: Path Computation Element.

    PCEP: Path Computation Element Protocol.

    PST: Path Setup Type.

    TLV: Type, Length, and Value.

3. Path Setup Type Capability TLV

    A PCEP speaker indicates which PSTs it supports during the PCEP
    initialization phase, as follows. When the PCEP session is created,
    it sends an Open message with an OPEN object containing the PATH-
    SETUP-TYPE-CAPABILITY TLV. The format of this TLV is 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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |           Type (TBD1)          |            Length             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                    Reserved                    |  Num of PSTs  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     PST#1      |      ...      |     PST#N      |   Padding    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    //              Optional sub-TLVs (variable)                  //
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
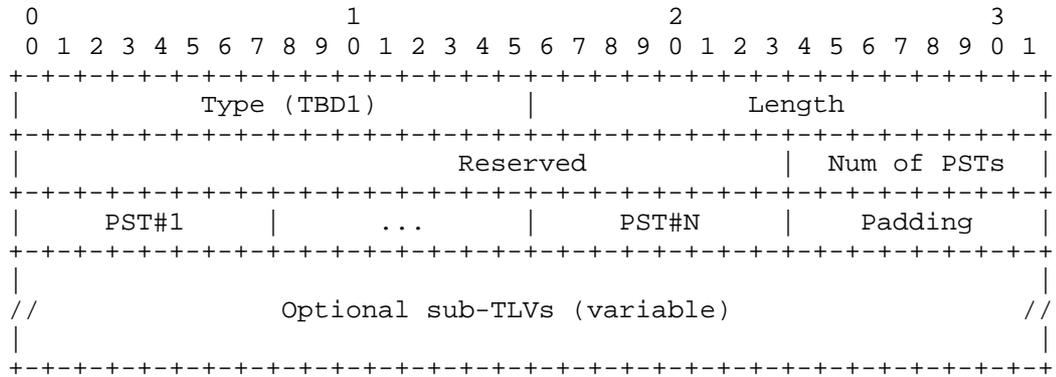
             Figure 1: PATH-SETUP-TYPE-CAPABILITY TLV

    The TLV type is TBD1 (to be assigned by IANA). Its reserved field
    MUST be set to zero by the sender and MUST be ignored by the
    receiver. The other fields in the TLV are as follows.

    Length: The total length in bytes of the remainder of the TLV, that
       is, excluding the Type and Length fields.

Number of PSTs: The number of PSTs in the following list, excluding
   padding.

List of PSTs: A list of the PSTs that the PCEP speaker supports.
   Each PST is a single byte in length. Duplicate entries in this
   list MUST be ignored. The PCEP speaker MUST pad the list with
   zeros so that it is a muliple of four bytes in length. This
   document defines the following PST value:

   * PST = 0: Path is setup using the RSVP-TE signaling protocol.

Optional sub-TLVs: A list of sub-TLVs associated with the supported
   PSTs. Each PST has zero or one sub-TLVs associated with it, and
   each sub-TLV is associated with exactly one PST. Each sub-TLV
   MUST obey the rules for TLV formatting defined in ([RFC5440]).
   That is, each sub-TLV is padded to a four byte alignment, and the
   length field of each sub-TLV does not include the padding bytes.
   This document does not define any sub-TLVs; an example can be
   found in [I-D.ietf-pce-segment-routing].

A PCEP speaker MUST check that this TLV is correctly formatted, as
follows.

o If there are no sub-TLVs, then the TLV length field MUST be equal
   to four bytes plus the size of the PST list, excluding any padding
   bytes.

o If there are sub-TLVs then the TLV Length field MUST be equal to
   four bytes plus the size of the PST list (rounded up to the
   nearest multiple of four) plus the size of the appended sub-TLVs
   excluding any padding bytes in the final sub-TLV.

o The Number of PSTs field MUST be greater than zero.

If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV which
violates these rules, then the PCEP speaker MUST send a PCErr message
with Error-Type = 10 (Reception of an invalid object) and Error-Value
= 11 (Malformed object) and MUST close the PCEP session. The PCEP
speaker MAY include the malformed OPEN object in the PCErr message as
well.

If a PCEP speaker receives an OPEN object with more than one PATH-
SETUP-TYPE-CAPABILITY TLV then it MUST ignore all but the first
instance of this TLV.

The absence of the PATH-SETUP-TYPE-CAPABILITY TLV from the OPEN
object is equivalent to a PATH-SETUP-TYPE-CAPABILITY TLV containing a
single PST of 0 (RSVP-TE signaling protocol) and no sub-TLVs. A PCEP

speaker MAY omit the PATH-SETUP-TYPE-CAPABILITY TLV if the only PST
it supports is RSVP-TE. If a PCEP speaker supports other PSTs
besides RSVP-TE, then it SHOULD include the PATH-SETUP-TYPE-
CAPABILITY TLV in its OPEN object.

If a PCEP speaker does not recognize the PATH-SETUP-TYPE-CAPABILITY
TLV, it will ignore the TLV in accordance with [RFC5440].

4. Path Setup Type TLV

When a PCEP session is used to set up TE paths using different
methods, the corresponding PCE and PCC must be aware of the path
setup method used. That means, a PCE must be able to specify paths
in the correct format and a PCC must be able to take control plane
and forwarding plane actions appropriate to the PST.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type (28)         |           Length (4)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Reserved                  |      PST      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
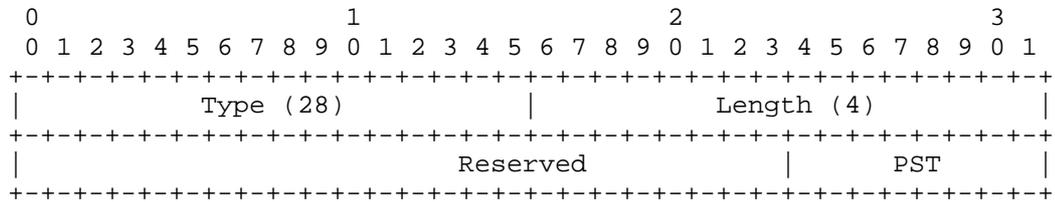
Figure 2: PATH-SETUP-TYPE TLV

PATH-SETUP-TYPE TLV is an optional TLV associated with the RP
([RFC5440]) and the SRP ([RFC8231]) objects. Its format is shown in
the above figure. The TLV type is 28. Its reserved field MUST be
set to zero. The one byte value contains the PST as defined for the
PATH-SETUP-TYPE-CAPABILITY TLV.

The absence of the PATH-SETUP-TYPE TLV is equivalent to a PATH-SETUP-
TYPE TLV with a PST value of 0 (RSVP-TE). A PCEP speaker MAY omit
the TLV if the PST is RSVP-TE. If the RP or SRP object contains more
than one PATH-SETUP-TYPE TLV, only the first TLV MUST be processed
and the rest MUST be ignored.

If a PCEP speaker does not recognize the PATH-SETUP-TYPE TLV, it will
ignore the TLV in accordance with [RFC5440], and will use RSVP-TE to
set up the path.

5. Operation

During the PCEP initialization phase, if a PCEP speaker receives a
PATH-SETUP-TYPE-CAPABILITY TLV from its peer, it MUST assume that the
peer supports only the PSTs listed in the TLV. If the PCEP speaker
and its peer have no PSTs in common, then the PCEP speaker MUST send

a PCErr message with Error-Type = 21 (Invalid traffic engineering
path setup type) and Error-Value = 2 (Mismatched path setup type) and
close the PCEP session.

If the peer has sent no PATH-SETUP-TYPE-CAPABILITY TLV, then the PCEP
speaker MUST infer that the peer supports path setup using at least
RSVP-TE. The PCEP speaker MAY also infer that the peer supports
other path setup types, but the means of inference are outside the
scope of this document.

When a PCC sends a PCReq message to a PCE ([RFC5440]), it MUST
include the PATH-SETUP-TYPE TLV in the RP object, unless the intended
PST is RSVP-TE, in which case it MAY omit the PATH-SETUP-TYPE TLV.
If the PCE is capable of expressing the path in a format appropriate
to the intended PST, it MUST use the appropriate ERO format in the
PCRep message.

When a PCE sends a PCRep message to a PCC ([RFC5440]), it MUST
include the PATH-SETUP-TYPE TLV in the RP object, unless the PST is
RSVP-TE, in which case it MAY omit the PATH-SETUP-TYPE TLV. If the
PCE does not support the intended PST, it MUST send a PCErr message
with Error-Type = 21 (Invalid traffic engineering path setup type)
and Error-Value = 1 (Unsupported path setup type) and close the PCEP
session. If the PSTs corresponding to the PCReq and PCRep messages
do not match, the PCC MUST send a PCErr message with Error-Type = 21
(Invalid traffic engineering path setup type) and Error-Value = 2
(Mismatched path setup type) and close the PCEP session.

When a stateful PCE sends a PCUpd message ([RFC8231]) or a PCInitiate
message ([RFC8281]) to a PCC, it MUST include the PATH-SETUP-TYPE TLV
in the SRP object, unless the intended PST is RSVP-TE, in which case
it MAY omit the PATH-SETUP-TYPE TLV. If the PCC does not support the
PST associated with the PCUpd or PCInitiate message, it MUST send a
PCErr message with Error-Type = 21 (Invalid traffic engineering path
setup type) and Error-Value = 1 (Unsupported path setup type) and
close the PCEP session.

When a PCC sends a PCRpt message to a stateful PCE ([RFC8231]), it
MUST include the PATH-SETUP-TYPE TLV in the SRP object, unless the
PST is RSVP-TE, in which case it MAY omit the PATH-SETUP-TYPE TLV.
The PCC MUST include the SRP object in the PCRpt message if the PST
is not RSVP-TE, even when the SRP-ID-number is the reserved value of
0x00000000. If the PCRpt message is triggered by a PCUpd or
PCInitiate message, then the PST that the PCC indicates in the PCRpt
MUST match the PST that the stateful PCE intended in the PCUpd or
PCInitiate. If it does not, then the PCE MUST send a PCErr message
with Error-Type = 21 (Invalid traffic engineering path setup type)

and Error-Value = 2 (Mismatched path setup type) and close the PCEP
session.

6. Manageability Considerations

   This document generalises PCEP to allow path setup methods other than
   RSVP-TE to be used by the network (but does not define any new path
   setup types, besides RSVP-TE). It is possible that, in a given
   network, multiple path setup methods will be used. It is also
   possible that not all devices will support the same set of path setup
   methods. Managing networks that combine multiple path setup methods
   may therefore raise some challenges from a configuration and
   observability point of view.

   Each document that defines a new Path Setup Type in the Path Setup
   Type Registry (Section 8.2) must include a manageability section.
   The manageability section must explain how operators can manage PCEP
   with the new path setup type. It must address the following
   questions, which are generally applicable when working with multiple
   path setup types in PCEP.

   o What are the criteria for when devices will use the new path setup
      type in PCEP, and how can the operator control this?

   o How can the network be migrated to the new path setup type, and
      are there any backwards compatibility issues that operators need
      to be aware of?

   o Are paths set up using the new path setup type intended to coexist
      with other paths over the long term and, if so, how is this
      situation managed with PCEP?

   o How can operators verify the correct operation of PCEP in the
      network with respect to the new path setup type? Which fault
      conditions must be reported to the operators?

   o Are there any existing management interfaces (such as YANG models)
      that must be extended to model the operation of PCEP in the
      network with respect to the new path setup type?

   See [RFC5706] for further guidance on how to write manageability
   sections in standards-track documents.

7. Security Considerations

   The security considerations described in [RFC5440] and [RFC8281] are
   applicable to this specification. No additional security measure is
   required.

Note that, if the security mechanisms of [RFC5440] and [RFC8281] are not used, then the protocol described by this draft could be attacked in the following new way. An attacker, using a TCP man-in-the-middle attack, could inject error messages into the PCEP session when a particular PST is (or is not) used. By doing so, the attacker could potentially force the use of a specific PST, which may allow them to subsequently attack a weakness in that PST.

8. IANA Considerations

8.1. PCEP TLV Type Indicators

IANA is requested to confirm the early allocation of the following code point in the PCEP TLV Type Indicators registry.

Value	Description	Reference
28	PATH-SETUP-TYPE	This document

IANA is requested to allocate a new code point for the following TLV in the PCEP TLV Type Indicators registry.

Value	Description	Reference
TBD1	PATH-SETUP-TYPE-CAPABILITY	This document

Note to IANA: The above TLV type was not part of the early code point allocation that was done for this draft. It was added to the draft after the early code point allocation had taken place. Please assign a code point from the indicated registry and replace each instance of "TBD1" in this document with the allocated code point.

8.2. New Path Setup Type Registry

IANA is requested to create a new sub-registry within the "Path Computation Element Protocol (PCEP) Numbers" registry called "PCEP Path Setup Types". The allocation policy for this new registry should be by IETF Review. The new registry should contain the following value:

Value	Description	Reference
0	Path is setup using the RSVP-TE signaling protocol.	This document

8.3. PCEP-Error Object

   IANA is requested to confirm the early allocation of the following
   code-points in the PCEP-ERROR Object Error Types and Values registry.

     Error-Type Meaning
        10   Reception of an invalid object

                    Error-value=11: Malformed object

     Error-Type Meaning
        21   Invalid traffic engineering path setup type

                    Error-value=0: Unassigned
                    Error-value=1: Unsupported path setup type
                    Error-value=2: Mismatched path setup type

   Note to IANA: the early allocation for Error-Type=10, Error-value=11
   was originally done by draft-ietf-pce-segment-routing. However, we
   have since moved its definition into this document. Therefore,
   please update the reference for this Error-value in the indicated
   registry to point to RFC.ietf-pce-lsp-setup-type.

9. Contributors

   The following people contributed to this document:

      - Jan Medved
      - Edward Crabbe

10. Acknowledgements

   We like to thank Marek Zavodsky for valuable comments.

11. References

11.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <https://www.rfc-editor.org/info/rfc5440>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for Stateful PCE", RFC 8231,
              DOI 10.17487/RFC8231, September 2017,
              <https://www.rfc-editor.org/info/rfc8231>.

   [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for PCE-Initiated LSP Setup in a Stateful PCE
              Model", RFC 8281, DOI 10.17487/RFC8281, December 2017,
              <https://www.rfc-editor.org/info/rfc8281>.

11.2. Informative References

   [I-D.ietf-pce-segment-routing]
              Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W.,
              and J. Hardwick, "PCEP Extensions for Segment Routing",
              draft-ietf-pce-segment-routing-11 (work in progress),
              November 2017.

   [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
              Element (PCE)-Based Architecture", RFC 4655,
              DOI 10.17487/RFC4655, August 2006,
              <https://www.rfc-editor.org/info/rfc4655>.

   [RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol Generic
              Requirements", RFC 4657, DOI 10.17487/RFC4657, September
              2006, <https://www.rfc-editor.org/info/rfc4657>.

   [RFC5706] Harrington, D., "Guidelines for Considering Operations and
              Management of New Protocols and Protocol Extensions",
              RFC 5706, DOI 10.17487/RFC5706, November 2009,
              <https://www.rfc-editor.org/info/rfc5706>.

Authors' Addresses

   Siva Sivabalan
   Cisco Systems, Inc.
   2000 Innovation Drive
   Kanata, Ontario K2K 3E8
   Canada

   Email: msiva@cisco.com

Jeff Tantsura
Nuage Networks
755 Ravendale Drive
Mountain View, CA 94043
USA

Email: jefftant.ietf@gmail.com

Ina Minei
Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043
USA

Email: inaminei@google.com

Robert Varga
Pantheon Technologies SRO
Mlynske Nivy 56
Bratislava, 821 05
Slovakia

Email: nite@hq.sk

Jon Hardwick
Metaswitch Networks
100 Church Street
Enfield, Middlesex
UK

Email: jonathan.hardwick@metaswitch.com

PCE Working Group   Y. Lee
Internet-Draft   Samsung
Intended status: Standards Track   H. Zheng
Expires: 21 February 2024   Huawei Technologies
                                               O. Gonzalez de Dios
                                                        Telefonica
                                                         V. Lopez
                                                            Nokia
                                                          Z. Ali
                                                           Cisco
                                                  20 August 2023

      Path Computation Element Communication Protocol (PCEP) Extensions for
             Stateful PCE Usage in GMPLS-controlled Networks
                   draft-ietf-pce-pcep-stateful-pce-gmpls-23

Abstract

   The PCE communication Protocol (PCEP) has been extended to support
   stateful PCE functions where the Stateful PCE maintains information
   about paths and resource usage within a network, but these extensions
   do not cover all requirements for GMPLS networks.

   This document provides the extensions required for PCEP so as to
   enable the usage of a stateful PCE capability in GMPLS-controlled
   networks.

Status of This Memo

Copyright Notice

   Copyright (c) 2023 IETF Trust and the persons identified as the
   document authors. All rights reserved.

Table of Contents

1. Introduction

   [RFC4655] presents the architecture of a Path Computation Element
   (PCE)-based model for computing Multiprotocol Label Switching (MPLS)
   and Generalized MPLS (GMPLS) Traffic Engineering Label Switched Paths
   (TE LSPs). To perform such a constrained computation, a PCE stores
   the network topology (i.e., TE links and nodes) and resource
   information (i.e., TE attributes) in its TE Database (TED). A PCE
   that only maintains TED is referred to as a stateless PCE. [RFC5440]
   describes the Path Computation Element Communication Protocol (PCEP)
   for interaction between a Path Computation Client (PCC) and a PCE, or
   between two PCEs, enabling computation of TE LSPs. PCEP is further
   extended to support GMPLS-controlled networks as per [RFC8779].

   Stateful PCEs are shown to be helpful in many application scenarios,
   in both MPLS and GMPLS networks, as illustrated in [RFC8051].
   Further discussion of concept of a stateful PCE can be found in
   [RFC7399]. In order for these applications to able to exploit the
   capability of stateful PCEs, extensions to stateful PCEP for GMPLS
   are required.

   [RFC8051] describes how a stateful PCE can be applicable to solve
   various problems for MPLS-TE and GMPLS networks and the benefits it
   brings to such deployments.

   [RFC8231] specifies a set of extensions to PCEP to enable stateful
   control of TE LSPs where they are configured on the PCC, and control
   over them could be delegated to the PCE. Furthermore, [RFC8281]
   describes the setup and teardown of PCE-initiated LSPs under the
   active stateful PCE model, without the need for local configuration
   on the PCC. However, both documents omit the specification for

   technology-specific objects/TLVs, and do not cover GMPLS-controlled
   networks (e.g., Wavelength Switched Optical Network (WSON), Optical
   Transport Network (OTN), Synchronous Optical Network
   (SONET)/Synchronous Digital Hierarchy (SDH), etc. technologies).

   This document focuses on the extensions that are necessary in order
   for the deployment of stateful PCEs and the requirements for PCE-
   initiated LSPs in GMPLS-controlled networks. Section 3 provides a
   general context of the usage of Stateful PCE and PCEP for GMPLS. The
   various requirements for stateful GMPLS, including PCE-initiation for
   GMPLS LSPs, are provided in Section 4. An overview of the PCEP
   extensions is specified in Section 5, and a solution to address such
   requirements with PCEP object extensions in Section 6.

1.1. Conventions Used in this Document

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in BCP
   14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

2. Terminology

   Terminology used in this document is the same as terminology used in
   [RFC5440], [RFC8231], [RFC8281], and [RFC8779].

3. General Context of Stateful PCE and PCEP for GMPLS

   This section is built on the basis of Stateful PCE specified in
   [RFC8231] and PCEP for GMPLS specified in [RFC8779].

   The operation for Stateful PCE on LSPs can be divided into two types,
   active stateful PCE and passive stateful PCE as described in
   [RFC8051].

   For active stateful PCE, a Path Computation Update Request (PCUpd)
   message is sent from PCE to PCC to update the LSP state for the LSPs
   delegated to the PCE. Any changes to the delegated LSPs generate a
   Path Computation State Report (PCRpt) message from the PCC to PCE to
   convey the changes of the LSPs. Any modifications to the Objects/
   TLVs that are identified in this document to support GMPLS
   technology-specific attributes will be carried in the PCRpt and PCUpd
   messages.

   For passive stateful PCEs, Path Computation Request (PCReq)/ Path
   Computation Reply (PCRep) messages are used to request for path
   computation. GMPLS-technology specific Objects and TLVs are defined

in [RFC8779], this document builds on it and adds the stateful PCE
aspects where applicable. Passive Stateful PCE makes use of PCRpt
messages when reporting LSP State changes sent by PCCs to PCEs. Any
modifications to the Objects/TLVs that are identified in this
document to support GMPLS technology-specific attributes will be
carried in the PCRpt message.

Furthermore, the LSP Initiation function of PCEP is defined in
[RFC8281] to allow the PCE to initiate LSP establishment after the
path is computed. An LSP Initiate Request (PCInitiate) message is
used to trigger the end node to set up the LSP. Any modifications to
the Objects/TLVs that are identified in this document to support
GMPLS technology-specific attributes will be carried in the
PCInitiate messages.

[RFC8779] defines GMPLS-technology specific Objects/TLVs in stateless
PCEP, and this document makes use of these Objects/TLVs without
modifications where applicable. Where these Objects/TLVs require
modifications to incorporate stateful PCE, they are described in this
document. PCE-Initiated LSPs follow the principle specified in
[RFC8281], and the GMPLS-specific extensions are also included in
this document.

4. Main Requirements

This section notes the main functional requirements for PCEP
extensions to support stateful PCE for use in GMPLS-controlled
networks, based on the description in [RFC8051]. Many requirements
are common across a variety of network types (e.g., MPLS-TE networks
and GMPLS networks) and the protocol extensions to meet the
requirements are already described in [RFC8231], such as LSP update,
delegation and state synchronization/report. Protection context
information that describes the GMPLS requirement can also follow the
description in [RFC8745]. This document does not repeat the
description of those protocol extensions. This document presents
protocol extensions for a set of requirements which are specific to
the use of a stateful PCE in a GMPLS-controlled network.

The requirements for GMPLS-specific stateful PCE are as follows:

* Advertisement of the stateful PCE capability. This generic
   requirement is covered in Section 5.4 of [RFC8231]. The GMPLS-
   CAPABILITY TLV specified in section 2.1 of [RFC8779] and its
   extension in this document needs to be advertised as well.

* All the PCEP messages need to be capable of indicating GMPLS-
   specific switching capabilities. GMPLS LSP creation/modification/
   deletion requires knowledge of LSP switching capability (e.g.,

Time-Division Multiplex Capable (TDM), Layer 2 Switch Capable
(L2SC), OTN-TDM, Lambda Switch Capable (LSC), etc.) and the
generalized payload (G-PID) to be used according to [RFC3471],
[RFC3473]. It also requires the specification of data flow
specific traffic parameters (also known as Traffic Specification
(Tspec)), which are technology specific. Such information would
need to be included in various PCEP messages.

* In some technologies, path calculation is tightly coupled with
label selection along the route. For example, path calculation in
a Wavelength Division Multiplexing (WDM) network may include
lambda continuity and/or lambda feasibility constraints and hence
a path computed by the PCE is associated with a specific lambda
(label). Hence, in such networks, the label information needs to
be provided to a PCC in order for a PCE to initiate GMPLS LSPs
under the active stateful PCE model, i.e., explicit label control
may be required.

* Stateful PCEP messages also need to indicate the protection
context information for the LSP specified by GMPLS, as defined in
[RFC4872], [RFC4873].

5. Overview of Stateful PCEP Extensions for GMPLS Networks

5.1. Capability Advertisement for Stateful PCEP in GMPLS

Capability Advertisement has been specified in [RFC8231], and can be
achieved by using the "STATEFUL-PCE-CAPABILITY TLV" in the Open
message. Another GMPLS-CAPABILITY TLV has been defined in [RFC8779].
A subregistry to manage the Flag field of the GMPLS-CAPABILITY TLV is
created by the IANA as requested by [RFC8779]. The following bits
are introduced by this document in the GMPLS-CAPABILITY TLV as flags
to indicate the capability for LSP report, update and initiation in
GMPLS networks: LSP-REPORT-CAPABILITY(TBDa), LSP-UPDATE-CAPABILITY
(TBD1), and LSP-INSTANTIATION-CAPABILITY (TBD2).

5.2. LSP Synchronization

After the session between the PCC and a stateful PCE is initialized,
the PCE must learn the state of a PCC's LSPs (including its
attributes) before it can perform path computations or update LSP
attributes in a PCC. This process is known as LSP state
synchronization. The LSP attributes including bandwidth, associated
route, and protection information etc., are stored by the PCE in the
LSP database (LSP-DB). Note that, as described in [RFC8231], the LSP
state synchronization covers both the bulk reporting of LSPs at
initialization as well the reporting of new or modified LSPs during
normal operation. Incremental LSP-DB synchronization may be desired

in a GMPLS-controlled network and it is specified in [RFC8232].

The format of the PCRpt message is specified in [RFC8231] and
extended in [RFC8623] to include the END-POINTS object. The END-
POINTS object is extended for GMPLS in [RFC8779]. The END-POINTS
object can be carried in the PCRpt message as specified in [RFC8623].
The END-POINTS object type for GMPLS is included in the PCRpt message
as per the same.

The BANDWIDTH, LSP Attributes (LSPA), Include Route Object (IRO) and
Exclude Route Object (XRO) objects are extended for GMPLS in
[RFC8779] and are also used in the PCRpt in the same manner. These
objects are carried in the PCRpt message as specified in [RFC8231]
(as the attribute-list defined in Section 6.5 of [RFC5440] and
extended by many other documents that define PCEP extensions for
specific scenarios).

The SWITCH-LAYER object is defined in [RFC8282]. This object is
carried in PCRpt message as specified in section 3.2 of [RFC8282].

5.3. LSP Delegation and Cleanup

LSP delegation and cleanup procedure specified in [RFC8231] are
equally applicable to GMPLS LSPs and this document does not modify
the associated usage.

5.4. LSP Operations

Both passive and active stateful PCE mechanisms in [RFC8231] are
applicable in GMPLS-controlled networks. Remote LSP Initiation in
[RFC8281] is also applicable in GMPLS-controlled networks.

6. PCEP Object Extensions

6.1. Existing Extensions used for Stateful GMPLS

Existing extensions defined in [RFC8779] can be used in Stateful PCEP
with no or slight changes for GMPLS network control, including the
following:

*   END-POINTS: Generalized END-POINTS was specified in [RFC8779] to
    include GMPLS capabilities. All Stateful PCEP messages MUST
    include the END-POINTS with Generalized Endpoint object type,
    containing the LABEL-REQUEST TLV. Further note that:

-   As per [RFC8779] for stateless GMPLS path computation, the
    Generalized END-POINTS object may contain a LABEL-REQUEST and/
    or LABEL-SET TLV. In this document, only the LABEL-REQUEST TLV
    is used to specify the switching type, encoding type and G-PID
    of the LSP.

-   If unnumbered endpoint addresses are used for the LSP, the
    UNNUMBERED-ENDPOINT TLV [RFC8779] MUST be used to specify the
    unnumbered endpoint addresses.

-   The Generalized END-POINTS MAY contain other TLVs defined in
    [RFC8779].

*   RP: RP object extension, together with the Routing Granularity
    (RG) flag defined in [RFC8779], are applicable in the Stateful
    PCEP for GMPLS networks.

*   BANDWIDTH: Generalized BANDWIDTH was specified in [RFC8779] to
    represent GMPLS features, including asymmetric bandwidth and G-PID
    information.

*   LSPA: LSPA Extensions in Section 2.8 of [RFC8779] is applicable in
    Stateful PCEP for GMPLS networks.

*   IRO: IRO Extensions in Section 2.6 of [RFC8779] is applicable in
    Stateful PCEP for GMPLS networks.

*   XRO: XRO Extensions in Section 2.7 of [RFC8779] is applicable in
    Stateful PCEP for GMPLS networks. A new flag is defined in
    Section 6.2.3 of this document.

*   ERO: The Explicit Route Object (ERO) was not extended in
    [RFC8779], nor is it in this document.

*   SWITCH-LAYER: SWITCHING-LAYER definition in Section 3.2 of
    [RFC8282] is applicable in Stateful PCEP messages for GMPLS
    networks.

6.2. New Extensions

6.2.1. GMPLS-CAPABILITY TLV in OPEN Object

In [RFC8779], IANA has allocated value 45 (GMPLS-CAPABILITY) from the
"PCEP TLV Type Indicators" sub-registry. The specifcation add three
flags to the flag field of this TLV to indicate the Report, Update,
and Initiation capabilities.

R (LSP-REPORT-CAPABILITY(TBDa) -- 1 bit): if set to 1 by a PCC, the R
flag indicates that the PCC is capable of reporting the current state
of a GMPLS LSP, whenever there's a change to the parameters or
operational status of the GMPLS LSP; if set to 1 by a PCE, the R Flag
indicates that the PCE is interested in receiving GMPLS LSP State
Reports whenever there is a parameter or operational status change to
the LSP. The LSP-REPORT-CAPABILITY flag must be advertised by both a
PCC and a PCE for PCRpt messages to be allowed on a PCEP session for
GMPLS LSP.

U (LSP-UPDATE-CAPABILITY(TBD1) -- 1 bit): if set to 1 by a PCC, the U
flag indicates that the PCC allows modification of GMPLS LSP
parameters; if set to 1 by a PCE, the U flag indicates that the PCE
is capable of updating GMPLS LSP parameters. The LSP-UPDATE-
CAPABILITY flag must be advertised by both a PCC and a PCE for PCUpd
messages to be allowed on a PCEP session for GMPLS LSP.

I (LSP-INSTANTIATION-CAPABILITY(TBD2) -- 1 bit): If set to 1 by a
PCC, the I flag indicates that the PCC allows instantiation of a
GMPLS LSP by a PCE. If set to 1 by a PCE, the I flag indicates that
the PCE supports instantiating GMPLS LSPs. The LSP-INSTANTIATION-
CAPABILITY flag must be set by both the PCC and PCE in order to
enable PCE-initiated LSP instantiation.

6.2.2. New LSP Exclusion Sub-object in the XRO

[RFC5521] defines a mechanism for a PCC to request or demand that
specific nodes, links, or other network resources are excluded from
paths computed by a PCE. A PCC may wish to request the computation
of a path that avoids all links and nodes traversed by some other
LSP.

To this end this document defines a new sub-object for use with route
exclusion defined in [RFC5521]. The LSP exclusion sub-object is 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|Type (TBD3)  |     Length    |    Reserved   |     Flags     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                    Symbolic Path Name                       //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
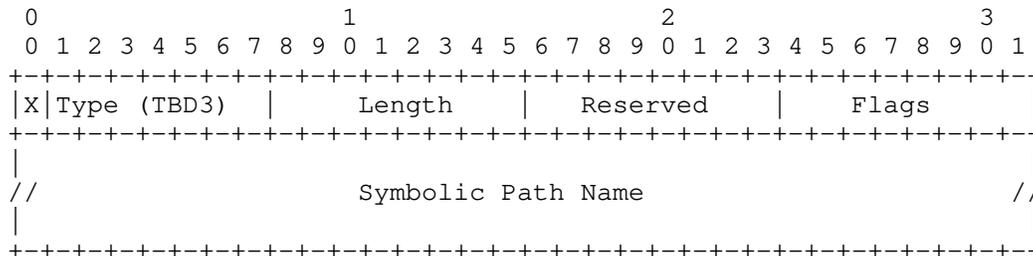
Figure 1: New LSP Exclusion Sub-object Format

X: Same as the X-bit defined in the XRO sub-objects in Section 2.1.1
of [RFC5521] where it says: "The X-bit indicates whether the
exclusion is mandatory or desired. 0 indicates that the resource
specified MUST be excluded from the path computed by the PCE. 1
indicates that the resource specified SHOULD be excluded from the
path computed by the PCE, but MAY be included subject to PCE policy
and the absence of a viable path that meets the other constraints and
excludes the resource.".

Type: Sub-object Type for an LSP exclusion sub-object. Value of
TBD3. To be assigned by IANA.

Length: The Length contains the total length of the sub-object in
bytes, including the Type and Length fields.

Reserved: MUST be set to zero on transmission and ignored on receipt.

Flags: This field may be used to further specify the exclusion
constraint with regard to the LSP. Currently, no flags are defined.

Symbolic Path Name: This is the identifier given to an LSP. Its
syntax and semantics are identical to those of the Symbolic Path Name
field defined in Section 7.3.2 of [RFC8231] where it says: "symbolic
name for the LSP, unique in the PCC. It SHOULD be a string of
printable ASCII characters, without a NULL terminator." The Symbolic
Path Name in the LSP Exclusion Sub-object MUST only vary from being a
string of printable ASCII characters without a NULL terminator when
it is matching the value contained in another subobject. It is worth
noting that given that the Symbolic Path Name is unique in the
context of the headnode, only LSPs that share the same headnode/PCC
could be excluded.

This sub-object MAY be present multiple times in the exclude route
object (XRO) to exclude resources from multiple LSPs. When a
stateful PCE receives a PCReq message carrying this sub-object, it
MUST search for the identified LSP in its LSP-DB and then exclude
from the new path computation all resources used by the identified
LSP.

Note that this XRO Sub-object could also be used by non-GMPLS LSPs.
The description by usage of non-GMPLS LSPs is not in the scope of
this document.

6.2.3. New flags in the LSP-EXTENDED-FLAG TLV in LSP Object

   The LSP Object is defined in Section 7.3 of [RFC8231], and the new
   extended flags TLV is defined in [RFC9357]. This TLV is used in
   PCUpd, PCRpt and PCInitiate messages for GMPLS, with the following
   flags defined in this document.

   * G (GMPLS LSP(TBDb) -- 1 bit) : If set to 1, it indicates the LSP
      is a GMPLS LSP.

   * B (Bidirectional LSP(TBD4) -- 1 bit): If set to 0, it indicates a
      request to create a uni-directional LSP. If set to 1, it
      indicates a request to create a bidirectional co-routed LSP.

   * RG (Routing Granularity(TBDc) -- 2 bits) : RG flag for GMPLS is
      also defined in the LSP-EXTENDED-FLAG TLV. The value are defined
      as per [RFC8779]:

   00: reserved

   01: node

   10: link

   11: label

7. Update to Error Handling

   A PCEP-ERROR object is used to report a PCEP error and is
   characterized by an Error-Type that specifies the type of error and
   an Error-value that provides additional information about the error.
   This section adds additional error handling procedures to those
   specified in Section 3 of [RFC8779]. Please note that all error
   handling specified in Section 3 of [RFC8779] is applicable and MUST
   be supported for a stateful PCE in GMPLS networks.

7.1. Error Handling in PCEP Capabilities Advertisement

   The PCEP extensions described in this document for stateful PCEs with
   GMPLS capability MUST NOT be used if the PCE has not advertised its
   capabilities with GMPLS as per Section 6.2.1.

If the PCC understands the U flag that indicates the stateful LSP-
UPDATE-CAPABILITY, but did not advertise this capability, then upon
receipt of a PCUpd message for GMPLS LSP from the PCE, it SHOULD
generate a PCErr with error-type 19 ("Invalid Operation"), error-
value TBDx ("Attempted LSP Update Request for GMPLS if stateful PCE
capability for GMPLS was not advertised"), and terminate the PCEP
session. Such a PCC MAY decide to utilize the capability even though
it did not advertise support for it.

If the PCE understands the R flag that indicates the stateful LSP-
REPORT-CAPABILITY, but did not advertise this capability, then upon
receipt of a PCRpt message for GMPLS LSP from the PCC, it SHOULD
generate a PCErr with error-type 19 ("Invalid Operation"), error-
value TBDy ("Attempted LSP Report Request for GMPLS if stateful PCE
capability for GMPLS was not advertised"), and terminate the PCEP
session. Such a PCE MAY decide to utilize the capability even though
it did not advertise support for it.

If the PCC understands the I flag that indicates LSP-INSTANTIATION-
CAPABILITY, but did not advertise this capability, then upon receipt
of a PCInitiate message for GMPLS LSP from the PCE, it SHOULD
generate a PCErr with error-type 19 ("Invalid Operation"), error-
value TBDz ("Attempted LSP Instantiation Request for GMPLS if
stateful PCE instantiation capability for GMPLS was not advertised"),
and terminate the PCEP session. Such a PCC MAY decide to utilize the
capability even though it did not advertise support for it.

7.2. Error Handling in LSP Re-optimization

A stateful PCE is expected to perform an LSP re-optimization when
receiving a message with the R bit set in the RP object. If no LSP
state information is available to carry out re-optimization, the
stateful PCE SHOULD report the error "LSP state information
unavailable for the LSP re-optimization" (Error Type = 19, Error
value= TBD6), although such a PCE MAY consider the re-optimization to
have successfully completed. Note that this error message could also
be used by non-GMPLS LSPs.

7.3. Error Handling in Route Exclusion

The LSP exclusion sub-object in XRO is defined in Section 6.2.2 of
this document MAY be present multiple times. When a stateful PCE
receives a PCEP message carrying this sub-object, it searches for the
identified LSP in its LSP-DB and then excludes from the new path
computation all the resources used by the identified LSP. If the
stateful PCE cannot recognize the symbolic path name of the
identified LSP, it SHOULD send an error message PCErr reporting
Error-type = 19 ("Invalid Operation"), Error-value = TBD7 ("The LSP

state information for route exclusion purpose cannot be found").
Optionally, it MAY also provide with the unrecognized symbolic path
name information to the requesting PCC using the error reporting
techniques described in [RFC5440]. An implementation MAY choose to
ignore the requested exclusion when the LSP cannot be found because
it could claim it that it has avoided using all resources associated
with an LSP that doesn't exist.

7.4. Error Handling for generalized END-POINTS

Note that the END-POINTS object in the Stateful PCEP messages was
introduced for P2MP [RFC8623]. Similarly, the END-POINTS object MUST
be carried for the GMPLS LSP. If the END-POINTS object is missing
and the GMPLS flag in LSP-EXTENDED-FLAG is set, the receiving PCE or
PCC MUST send a PCErr message with Error-type=6 ("Mandatory Object
missing") and Error-value=3 ("END-POINTS object missing") (defined in
[RFC5440]). Similarly, if the END-POINTS object with the Generalized
Endpoint object type is received but if the LSP-EXTENDED-FLAG TLV is
missing in the LSP object or if the G flag in the LSP-EXTENDED-FLAG
TLV is not set, the receiving PCE or PCC MUST send a PCErr message
with Error-type = 19 ("Invalid Operation"), Error-value = TBD9 ("Use
of Generalized Endpoint object type for non-GMPLS LSP").

If the END-POINTS object with Generalized Endpoint Object Type is
missing the LABEL-REQUEST TLV, the receiving PCE or PCC MUST send a
PCErr message with Error-type=6 ("Mandatory Object missing") and
Error-value=TBD8 ("LABEL-REQUEST TLV missing").

8. Implementation

[NOTE TO RFC EDITOR : This whole section and the reference to RFC
7942 is to be removed before publication as an RFC]

This section records the status of known implementations of the
protocol defined by this specification at the time of posting of this
Internet-Draft, and is based on a proposal described in [RFC7942].
The description of implementations in this section is intended to
assist the IETF in its decision processes in progressing drafts to
RFCs. Please note that the listing of any individual implementation
here does not imply endorsement by the IETF. Furthermore, no effort
has been spent to verify the information presented here that was
supplied by IETF contributors. This is not intended as, and must not
be construed to be, a catalog of available implementations or their
features. Readers are advised to note that other implementations may
exist.

According to [RFC7942], "this will allow reviewers and working groups
to assign due consideration to documents that have the benefit of
running code, which may serve as evidence of valuable experimentation
and feedback that have made the implemented protocols more mature.
It is up to the individual working groups to use this information as
they see fit".

8.1. Huawei Technologies

   * Organization: Huawei Technologies, Co. LTD

   * Implementation: Huawei NCE-T

   * Description: PCRpt, PCUpd and PCInitiate messages for GMPLS
      Network

   * Maturity Level: Production

   * Coverage: Full

   * Contact: zhenghaomian@huawei.com

9. IANA Considerations

9.1. title=New Flags in GMPLS-CAPABILITY TLV

   [RFC8779] defines the GMPLS-CAPABILITY TLV; per that RFC, IANA
   created a registry to manage the value of the GMPLS-CAPABILITY TLV's
   Flag field. This document requests IANA to allocate new bits in the
   GMPLS-CAPABILITY TLV Flag Field registry, as follows. IANA is
   requested to make allocations starting from the least significant bit
   (31).

Bit	Description	Reference
TBDa	LSP-REPORT-CAPABILITY (R)	[This.I-D]
TBD1	LSP-UPDATE-CAPABILITY (U)	[This.I-D]
TBD2	LSP-INSTANTIATION-CAPABILITY (I)	[This.I-D]

9.2. New Sub-object for the Exclude Route Object

   IANA maintains the various XRO Subobjects types within the "XRO
   Subobjects" subregistry of the PCEP Numbers registry. IANA is
   requested to allocate a codepoint for another XRO subobject as
   follows:

```
Value   | Description                   | Reference
--------+-------------------------------+------------
TBD3    | LSP                           | [This.I-D]
```

9.3. Flags Field for LSP exclusion Sub-object

   IANA is requested to create a registry named "LSP Exclusion Sub-
   Object Flag Field", within the "Path Computation Element Protocol
   (PCEP) Numbers" group, to manage the Flag field of the LSP Exclusion
   sub-object in the XRO. No Flag is currently defined for this flag
   field in this document.

   Codespace of the Flag field (LSP Exclusion sub-object)

```
Bit    | Description          | Reference
-------+----------------------+------------
0-7    | Unassigned           | [This.I-D]
```

   New values are to be assigned by Standards Action [RFC8126]. Each
   bit should be tracked with the following qualities:

   * Bit number (counting from bit 0 as the most significant bit)

   * Capability description

   * Defining RFC

9.4. New Flags in the LSP-EXTENDED-FLAGS TLV

   [I-D.ietf-pce-lsp-extended-flags] requested IANA to create a
   subregistry, named the "LSP-EXTENDED-FLAG TLV Flag Field", within the
   "Path Computation Element Protocol (PCEP) Numbers" registry, to
   manage the Flag field of the LSP-EXTENDED-FLAG TLV.

   IANA is requested to make assignments from this registry as follows:

```
Bit    | Description                      | Reference
-------+----------------------------------+------------
TBDb   | GMPLS LSP (G)                    | [This.I-D]
TBD4   | Bi-directional co-routed LSP (B) | [This.I-D]
TBDc*  | Routing Granularity Flag (RG)    | [This.I-D]
```

   * - 2 bits need to be allocated

9.5. New PCEP Error Codes

   IANA is requested to make the following allocation in the "PCEP-ERROR
   Object Error Types and Values" registry.

Error-Type	Meaning	Error-value	Reference
6	Mandatory Object missing	TBD8: LABEL-REQUEST TLV missing	This I-D
19	Invalid Operation	TBD6: LSP state info unavailable for the Re-optimization	This I-D
		TBD7: LSP state info for route exclusion not found	This I-D
		TBDx: Attempted LSP Update Request for GMPLS if stateful PCE capability not advertised	This I-D
		TBDy: Attempted LSP State Report for GMPLS if stateful PCE capability not advertised	This I-D
		TBDz: Attempted LSP Instantiation Request for GMPLS if stateful PCE instantiation capability not advertised	This I-D
		TBD9: use of Generalized Endpoint object type for non-GMPLS LSP	This I-D

10. Manageability Considerations

   General PCE management considerations are discussed in [RFC4655] and
   [RFC5440], and GMPLS specific PCEP management considerations are
   described in [RFC8779]. In this document the management
   considerations for stateful PCEP extension in GMPLS are described.

This section follows the guidance of [RFC6123].

10.1. Control of Function through Configuration and Policy

In addition to the parameters already listed in Section 8.1 of
[RFC5440], a PCEP implementation SHOULD allow configuration of the
following PCEP session parameters on a PCC, however, an
implementation MAY choose to make these features available on all
PCEP sessions:

* The ability to send stateful PCEP messages for GMPLS LSPs.

* The ability to use path computation constraints (e.g., XRO).

In addition to the parameters already listed in Section 8.1 of
[RFC5440], a PCEP implementation SHOULD allow configuration of the
following PCEP session parameters on a PCE:

* The ability to compute paths in a stateful manner in GMPLS
   networks.

* A set of GMPLS-specific constraints.

These parameters may be configured as default parameters for any PCEP
session the PCEP speaker participates in, or they may apply to a
specific session with a given PCEP peer or a specific group of
sessions with a specific group of PCEP peers.

10.2. Information and Data Models

The YANG model in [I-D.ietf-pce-pcep-yang] can be used to configure
and monitor PCEP states and messages. To make sure that the YANG
model is useful for the extensions as described in this document, it
would need to include advertised GMPLS stateful capabilities etc. A
future version of [I-D.ietf-pce-pcep-yang] will include this.

As described in [I-D.ietf-teas-yang-path-computation], a YANG-based
interface can be used in some cases to request GMPLS path
computations, instead of PCEP. Refer
[I-D.ietf-teas-yang-path-computation] for details.

10.3. Liveness Detection and Monitoring

This document makes no change to the basic operation of PCEP, so
there are no changes to the requirements for liveness detection and
monitoring in [RFC4657] and Section 8.3 of [RFC5440].

10.4. Verifying Correct Operation

   This document makes no change to the basic operations of PCEP and the
   considerations described in Section 8.4 of [RFC5440]. New errors
   defined by this document should satisfy the requirement to log error
   events.

10.5. Requirements on Other Protocols and Functional Components

   When the detailed route information is included for LSP state
   synchronization (either at the initial stage or during LSP state
   report process), this requires the ingress node of an LSP to carry
   the RRO object in order to enable the collection of such information.

10.6. Impact on Network Operation

   The management considerations concerning the impact on network
   operations described in Section 4.6 of [RFC8779] apply here.

11. Security Considerations

   The security considerations elaborated in [RFC5440] apply to this
   document. The PCEP extensions to support GMPLS-controlled networks
   should be considered under the same security as for MPLS networks, as
   noted in [RFC7025]. So the PCEP extension to support GMPLS specified
   in [RFC8779] is used as the foundation of this document and the
   security considerations in [RFC8779] should also be applicable to
   this document. The secure transport of PCEP specified in [RFC8253]
   allows the usage of Transport Layer Security (TLS). The same can
   also be used by the PCEP extension defined in this document.

   This document provides additional extensions to PCEP so as to
   facilitate stateful PCE usage in GMPLS-controlled networks, on top of
   [RFC8231] and [RFC8281]. Security issues caused by the extension in
   [RFC8231] and [RFC8281] are not altered by the additions in this
   document. The security considerations in [RFC8231] and [RFC8281],
   including both issues and solutions, apply to this document as well.

12. Acknowledgement

   We would like to thank Adrian Farrel, Cyril Margaria, George Swallow,
   Jan Medved, Sue Hares, and John Scudder for the useful comments and
   discussions.

   Thanks to Dhruv Dhody for Shepherding this document and providing
   useful comments.

13. Nomative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <https://www.rfc-editor.org/info/rfc5440>.

   [RFC5521] Oki, E., Takeda, T., and A. Farrel, "Extensions to the
              Path Computation Element Communication Protocol (PCEP) for
              Route Exclusions", RFC 5521, DOI 10.17487/RFC5521, April
              2009, <https://www.rfc-editor.org/info/rfc5521>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for Stateful PCE", RFC 8231,
              DOI 10.17487/RFC8231, September 2017,
              <https://www.rfc-editor.org/info/rfc8231>.

   [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody,
              "PCEPS: Usage of TLS to Provide a Secure Transport for the
              Path Computation Element Communication Protocol (PCEP)",
              RFC 8253, DOI 10.17487/RFC8253, October 2017,
              <https://www.rfc-editor.org/info/rfc8253>.

   [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for PCE-Initiated LSP Setup in a Stateful PCE
              Model", RFC 8281, DOI 10.17487/RFC8281, December 2017,
              <https://www.rfc-editor.org/info/rfc8281>.

   [RFC8779] Margaria, C., Ed., Gonzalez de Dios, O., Ed., and F.
              Zhang, Ed., "Path Computation Element Communication
              Protocol (PCEP) Extensions for GMPLS", RFC 8779,
              DOI 10.17487/RFC8779, July 2020,
              <https://www.rfc-editor.org/info/rfc8779>.

   [RFC9357] Xiong, Q., "Label Switched Path (LSP) Object Flag
              Extension for Stateful PCE", RFC 9357,
              DOI 10.17487/RFC9357, February 2023,
              <https://www.rfc-editor.org/info/rfc9357>.

14. Informative References

   [I-D.ietf-pce-lsp-extended-flags]
              Xiong, Q., "Label Switched Path (LSP) Object Flag
              Extension for Stateful PCE", Work in Progress, Internet-
              Draft, draft-ietf-pce-lsp-extended-flags-09, 23 October
              2022, <https://datatracker.ietf.org/doc/html/draft-ietf-
              pce-lsp-extended-flags-09>.

   [I-D.ietf-pce-pcep-yang]
              Dhody, D., Beeram, V. P., Hardwick, J., and J. Tantsura,
              "A YANG Data Model for Path Computation Element
              Communications Protocol (PCEP)", Work in Progress,
              Internet-Draft, draft-ietf-pce-pcep-yang-21, 6 March 2023,
              <https://datatracker.ietf.org/doc/html/draft-ietf-pce-
              pcep-yang-21>.

   [I-D.ietf-teas-yang-path-computation]
              Busi, I., Belotti, S., de Dios, O. G., Sharma, A., and Y.
              Shi, "A YANG Data Model for requesting path computation",
              Work in Progress, Internet-Draft, draft-ietf-teas-yang-
              path-computation-21, 7 July 2023,
              <https://datatracker.ietf.org/doc/html/draft-ietf-teas-
              yang-path-computation-21>.

   [RFC3471] Berger, L., Ed., "Generalized Multi-Protocol Label
              Switching (GMPLS) Signaling Functional Description",
              RFC 3471, DOI 10.17487/RFC3471, January 2003,
              <https://www.rfc-editor.org/info/rfc3471>.

   [RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label
              Switching (GMPLS) Signaling Resource ReserVation Protocol-
              Traffic Engineering (RSVP-TE) Extensions", RFC 3473,
              DOI 10.17487/RFC3473, January 2003,
              <https://www.rfc-editor.org/info/rfc3473>.

   [RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path
              Computation Element (PCE)-Based Architecture", RFC 4655,
              DOI 10.17487/RFC4655, August 2006,
              <https://www.rfc-editor.org/info/rfc4655>.

   [RFC4657] Ash, J., Ed. and J.L. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol Generic
              Requirements", RFC 4657, DOI 10.17487/RFC4657, September
              2006, <https://www.rfc-editor.org/info/rfc4657>.

   [RFC4872] Lang, J.P., Ed., Rekhter, Y., Ed., and D. Papadimitriou,
              Ed., "RSVP-TE Extensions in Support of End-to-End
              Generalized Multi-Protocol Label Switching (GMPLS)
              Recovery", RFC 4872, DOI 10.17487/RFC4872, May 2007,
              <https://www.rfc-editor.org/info/rfc4872>.

   [RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel,
              "GMPLS Segment Recovery", RFC 4873, DOI 10.17487/RFC4873,
              May 2007, <https://www.rfc-editor.org/info/rfc4873>.

   [RFC5511] Farrel, A., "Routing Backus-Naur Form (RBNF): A Syntax
              Used to Form Encoding Rules in Various Routing Protocol
              Specifications", RFC 5511, DOI 10.17487/RFC5511, April
              2009, <https://www.rfc-editor.org/info/rfc5511>.

   [RFC6123] Farrel, A., "Inclusion of Manageability Sections in Path
              Computation Element (PCE) Working Group Drafts", RFC 6123,
              DOI 10.17487/RFC6123, February 2011,
              <https://www.rfc-editor.org/info/rfc6123>.

   [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C.
              Margaria, "Requirements for GMPLS Applications of PCE",
              RFC 7025, DOI 10.17487/RFC7025, September 2013,
              <https://www.rfc-editor.org/info/rfc7025>.

   [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path
              Computation Element Architecture", RFC 7399,
              DOI 10.17487/RFC7399, October 2014,
              <https://www.rfc-editor.org/info/rfc7399>.

   [RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running
              Code: The Implementation Status Section", BCP 205,
              RFC 7942, DOI 10.17487/RFC7942, July 2016,
              <https://www.rfc-editor.org/info/rfc7942>.

   [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a
              Stateful Path Computation Element (PCE)", RFC 8051,
              DOI 10.17487/RFC8051, January 2017,
              <https://www.rfc-editor.org/info/rfc8051>.

   [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for
              Writing an IANA Considerations Section in RFCs", BCP 26,
              RFC 8126, DOI 10.17487/RFC8126, June 2017,
              <https://www.rfc-editor.org/info/rfc8126>.

   [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X.,
              and D. Dhody, "Optimizations of Label Switched Path State
              Synchronization Procedures for a Stateful PCE", RFC 8232,
              DOI 10.17487/RFC8232, September 2017,
              <https://www.rfc-editor.org/info/rfc8232>.

   [RFC8282] Oki, E., Takeda, T., Farrel, A., and F. Zhang, "Extensions
              to the Path Computation Element Communication Protocol
              (PCEP) for Inter-Layer MPLS and GMPLS Traffic
              Engineering", RFC 8282, DOI 10.17487/RFC8282, December
              2017, <https://www.rfc-editor.org/info/rfc8282>.

   [RFC8623] Palle, U., Dhody, D., Tanaka, Y., and V. Beeram, "Stateful
              Path Computation Element (PCE) Protocol Extensions for
              Usage with Point-to-Multipoint TE Label Switched Paths
              (LSPs)", RFC 8623, DOI 10.17487/RFC8623, June 2019,
              <https://www.rfc-editor.org/info/rfc8623>.

   [RFC8745] Ananthakrishnan, H., Sivabalan, S., Barth, C., Minei, I.,
              and M. Negi, "Path Computation Element Communication
              Protocol (PCEP) Extensions for Associating Working and
              Protection Label Switched Paths (LSPs) with Stateful PCE",
              RFC 8745, DOI 10.17487/RFC8745, March 2020,
              <https://www.rfc-editor.org/info/rfc8745>.

Appendix A. Contributors' Address

Xian Zhang
Huawei Technologies
Email: zhang.xian@huawei.com

Dhruv Dhody
Huawei Technology
India
Email: dhruv.ietf@gmail.com

Yi Lin
Huawei Technologies
Email: yi.lin@huawei.com

Fatai Zhang
Huawei Technologies
Email: zhangfatai@huawei.com

Ramon Casellas
CTTC
Av. Carl Friedrich Gauss n7
Castelldefels, Barcelona 08860
Spain
Email: ramon.casellas@cttc.es

Siva Sivabalan
Cisco Systems
Email: msiva@cisco.com

Clarence Filsfils
Cisco Systems
Email: cfilsfil@cisco.com

Robert Varga
Pantheon Technologies
Email: nite@hq.sk

Appendix B. PCEP Messages

   This section uses the Routing Backus-Naur Form (RBNF) [RFC5511] to
   illustrate the PCEP messages. The RBNF in this section is reproduced
   for informative purposes. It is also expanded to show the GMPLS
   specific objects.

B.1. The PCRpt Message

   According to [RFC8231], the PCRpt Message is used to report the
   current state of an LSP. This document extends the message in
   reporting the status of LSPs with GMPLS characteristics.

   The format of the PCRpt message is as follows:

           <PCRpt Message> ::= <Common Header>
                               <state-report-list>

   Where:

           <state-report-list> ::= <state-report>[<state-report-list>]
           <state-report> ::= [<SRP>]
                              <LSP>
                              [<END-POINTS>]
                              <path>

   Where:

           <path> ::= <intended-path>
                      [<actual-attribute-list><actual-path>]
                      <intended-attribute-list>
           <actual-attribute-list> ::=[<BANDWIDTH>]
                                       [<metric-list>]

   Where:

   * The END-POINTS object MUST be carried in a PCRpt message when the
      G flag is set in the LSP-EXTENDED-FLAG TLV in the LSP object for a
      GMPLS LSP.

   * <intended-path> is represented by the ERO object defined in
      Section 7.9 of [RFC5440], augmented in [RFC8779] with explicit
      label control (ELC) and Path Keys.

   * <actual-attribute-list> consists of the actual computed and
      signaled values of the <BANDWIDTH> and <metric-lists> objects
      defined in [RFC5440].

* <actual-path> is represented by the RRO object defined in
  Section 7.10 of [RFC5440].

* <intended-attribute-list> is the attribute-list defined in
  Section 6.5 of [RFC5440] and extended by many other documents that
  define PCEP extensions for specific scenarios as shown below:

```
<attribute-list> ::= [<of-list>]
                     [<LSPA>]
                     [<BANDWIDTH>]
                     [<metric-list>]
                     [<IRO>][<XRO>]
                     [<INTER-LAYER>]
                     [<SWITCH-LAYER>]
                     [<REQ-ADAP-CAP>]
                     [<SERVER-INDICATION>]
```

B.2. The PCUpd Message

   The format of a PCUpd message is as follows:

```
<PCUpd Message> ::= <Common Header>
                    <update-request-list>
```

   Where:

```
<update-request-list> ::= <update-request>[<update-request-list>]
<update-request> ::= <SRP>
                     <LSP>
                     [<END-POINTS>]
                     <path>
```

   Where:

```
<path> ::= <intended-path><intended-attribute-list>
```

   Where:

* The END-POINTS object MUST be carried in a PCUpd message for the
  GMPLS LSP.

  * <intended-path> is represented by the ERO object defined in
     Section 7.9 of [RFC5440], augmented in [RFC8779] with explicit
     label control (ELC) and Path Keys.

  * <intended-attribute-list> is the attribute-list defined in
     [RFC5440] and extended by many other documents that define PCEP
     extensions for specific scenarios and as shown for PCRpt above.

B.3. The PCInitiate Message

   According to [RFC8281], the PCInitiate Message is used allow LSP
   Initiation. This document extends the message in initiating LSPs
   with GMPLS characteristics. The format of a PCInitiate message is as
   follows:

        <PCInitiate Message> ::= <Common Header>
                                 <PCE-initiated-lsp-list>

   Where:

      <Common Header> is defined in <xref target="RFC5440" />.
      <PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                                   [<PCE-initiated-lsp-list>]
      <PCE-initiated-lsp-request> ::= (<PCE-initiated-lsp-instantiation>|
                                      <PCE-initiated-lsp-deletion>)
      <PCE-initiated-lsp-instantiation> ::= <SRP>
                                            <LSP>
                                            [<END-POINTS>]
                                            <ERO>
                                            [<attribute-list>]
      <PCE-initiated-lsp-deletion> ::= <SRP>
                                       <LSP>

   The format of the PCInitiate message is unchanged from Section 5.1 of
   [RFC8281]. All fields are similar to the PCRpt and the PCUpd
   message.

Authors' Addresses

   Young Lee
   Samsung
   Email: younglee.tx@gmail.com

Haomian Zheng
Huawei Technologies
Email: zhenghaomian@huawei.com

Oscar Gonzalez de Dios
Telefonica
Email: oscar.gonzalezdedios@telefonica.com

Victor Lopez
Nokia
Email: victor.lopez@nokia.com

Zafar Ali
Cisco
Email: zali@cisco.com

                      PCEP Extensions for Segment Routing
                      draft-ietf-pce-segment-routing-16

Abstract

   Segment Routing (SR) enables any head-end node to select any path
   without relying on a hop-by-hop signaling technique (e.g., LDP or
   RSVP-TE). It depends only on "segments" that are advertised by link-
   state Interior Gateway Protocols (IGPs). A Segment Routing Path can
   be derived from a variety of mechanisms, including an IGP Shortest
   Path Tree (SPT), explicit configuration, or a Path Computation
   Element (PCE). This document specifies extensions to the Path
   Computation Element Communication Protocol (PCEP) that allow a
   stateful PCE to compute and initiate Traffic Engineering (TE) paths,
   as well as a PCC to request a path subject to certain constraints and
   optimization criteria in SR networks.

Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in BCP
   14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on September 5, 2019.

Copyright Notice

Table of Contents

1. Introduction

   Segment Routing (SR) leverages the source routing paradigm. Using
   SR, a source node steers a packet through a path without relying on
   hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is
   specified as an ordered list of instructions called "segments". Each
   segment is an instruction to route the packet to a specific place in
   the network, or to perform a function on the packet. A database of
   segments can be distributed through the network using a routing
   protocol (such as IS-IS or OSPF) or by any other means. Several
   types of segment are defined. A node segment uniquely identifies a
   specific node in the SR domain. Each router in the SR domain
   associates a node segment with an ECMP-aware shortest path to the
   node that it identifies. An adjacency segment represents a
   unidirectional adjacency. An adjacency segment is local to the node
   which advertises it. Both node segments and adjacency segments can
   be used for SR.

   [RFC8402] describes the SR architecture. The corresponding IS-IS and
   OSPF extensions are specified in
   [I-D.ietf-isis-segment-routing-extensions] and
   [I-D.ietf-ospf-segment-routing-extensions], respectively.

   The SR architecture can be implemented using either an MPLS
   forwarding plane [I-D.ietf-spring-segment-routing-mpls] or an IPv6
   forwarding plane [I-D.ietf-6man-segment-routing-header]. The MPLS
   forwarding plane can be applied to SR without any change, in which
   case an SR path corresponds to an MPLS Label Switching Path (LSP).
   This document is relevant to the MPLS forwarding plane only. In this

document, "Node-SID" and "Adjacency-SID" denote Node Segment
Identifier and Adjacency Segment Identifier respectively.

A Segment Routing path (SR path) can be derived from an IGP Shortest
Path Tree (SPT). SR-TE paths may not follow an IGP SPT. Such paths
may be chosen by a suitable network planning tool and provisioned on
the ingress node of the SR-TE path.

[RFC5440] describes the Path Computation Element Communication
Protocol (PCEP) for communication between a Path Computation Client
(PCC) and a Path Computation Element (PCE) or between a pair of PCEs.
A PCE computes paths for MPLS Traffic Engineering LSPs (MPLS-TE LSPs)
based on various constraints and optimization criteria. [RFC8231]
specifies extensions to PCEP that allow a stateful PCE to compute and
recommend network paths in compliance with [RFC4657] and defines
objects and TLVs for MPLS-TE LSPs. Stateful PCEP extensions provide
synchronization of LSP state between a PCC and a PCE or between a
pair of PCEs, delegation of LSP control, reporting of LSP state from
a PCC to a PCE, controlling the setup and path routing of an LSP from
a PCE to a PCC. Stateful PCEP extensions are intended for an
operational model in which LSPs are configured on the PCC, and
control over them is delegated to the PCE.

A mechanism to dynamically initiate LSPs on a PCC based on the
requests from a stateful PCE or a controller using stateful PCE is
specified in [RFC8281]. This mechanism is useful in Software Defined
Networking (SDN) applications, such as on-demand engineering, or
bandwidth calendaring [RFC8413].

It is possible to use a stateful PCE for computing one or more SR-TE
paths taking into account various constraints and objective
functions. Once a path is chosen, the stateful PCE can initiate an
SR-TE path on a PCC using PCEP extensions specified in [RFC8281]
using the SR specific PCEP extensions specified in this document.
Additionally, using procedures described in this document, a PCC can
request an SR path from either a stateful or a stateless PCE.

This specification relies on the procedures specified in [RFC8408] to
exchange the segment routing capability and to specify that the path
setup type of an LSP is segment routing. This specification also
updates [RFC8408] to clarify the use of sub-TLVs in the PATH-SETUP-
TYPE-CAPABILITY TLV. See Section 4.1.1 for details.

This specification provides a mechanism for a network controller
(acting as a PCE) to instantiate candidate paths for an SR Policy
onto a head-end node (acting as a PCC) using PCEP. For more
information on the SR Policy Architecture, see
[I-D.ietf-spring-segment-routing-policy].

2. Terminology

   The following terminologies are used in this document:

   ERO: Explicit Route Object

   IGP: Interior Gateway Protocol

   IS-IS: Intermediate System to Intermediate System

   LSR: Label Switching Router

   MSD: Base MPLS Imposition Maximum SID Depth, as defined in [RFC8491]

   NAI: Node or Adjacency Identifier

   OSPF: Open Shortest Path First

   PCC: Path Computation Client

   PCE: Path Computation Element

   PCEP: Path Computation Element Communication Protocol

   RRO: Record Route Object

   SID: Segment Identifier

   SR: Segment Routing

   SR-DB: Segment Routing Database: the collection of SRGBs, SRLBs and
      SIDs and the objects they map to, advertised by a link state IGP

   SRGB: Segment Routing Global Block

   SRLB: Segment Routing Local Block

   SR-TE: Segment Routing Traffic Engineering

3. Overview of PCEP Operation in SR Networks

   In an SR network, the ingress node of an SR path prepends an SR
   header to all outgoing packets. The SR header consists of a list of
   SIDs (or MPLS labels in the context of this document). The header
   has all necessary information so that, in combination with the
   information distributed by the IGP, the packets can be guided from
   the ingress node to the egress node of the path; hence, there is no
   need for any signaling protocol.

In PCEP messages, LSP route information is carried in the Explicit
Route Object (ERO), which consists of a sequence of subobjects. SR-
TE paths computed by a PCE can be represented in an ERO in one of the
following forms:

o An ordered set of IP addresses representing network nodes/links.

o An ordered set of SIDs, with or without the corresponding IP
   addresses.

o An ordered set of MPLS labels, with or without corresponding IP
   address.

The PCC converts these into an MPLS label stack and next hop, as
described in Section 5.2.2.

This document defines a new ERO subobject denoted by "SR-ERO
subobject" capable of carrying a SID as well as the identity of the
node/adjacency represented by the SID. SR-capable PCEP speakers
should be able to generate and/or process such ERO subobject. An ERO
containing SR-ERO subobjects can be included in the PCEP Path
Computation Reply (PCRep) message defined in [RFC5440], the PCEP LSP
Initiate Request message (PCInitiate) defined in [RFC8281], as well
as in the PCEP LSP Update Request (PCUpd) and PCEP LSP State Report
(PCRpt) messages defined in [RFC8231].

When a PCEP session between a PCC and a PCE is established, both PCEP
speakers exchange their capabilities to indicate their ability to
support SR-specific functionality.

A PCE can update an LSP that is initially established via RSVP-TE
signaling to use an SR-TE path, by sending a PCUpd to the PCC that
delegated the LSP to it ([RFC8231]). A PCC can update an undelegated
LSP that is initially established via RSVP-TE signaling to use an SR-
TE path as follows. First, it requests an SR-TE Path from a PCE by
sending a PCReq message. If it receives a suitable path, it
establishes the path in the data plane, and then tears down the
original RSVP-TE path. If the PCE is stateful, then the PCC sends
PCRpt messages indicating that the new path is set up and the old
path is torn down, per [RFC8231].

Similarly, a PCE or PCC can update an LSP initially created with an
SR-TE path to use RSVP-TE signaling, if necessary. This capability
is useful for rolling back a change when a network is migrated from
RSVP-TE to SR-TE technology.

A PCC MAY include an RRO containing the recorded LSP in PCReq and
PCRpt messages as specified in [RFC5440] and [RFC8231], respectively.

This document defines a new RRO subobject for SR networks. The
methods used by a PCC to record the SR-TE LSP are outside the scope
of this document.

In summary, this document:

o Defines a new ERO subobject, a new RRO subobject and new PCEP
   error codes.

o Specifies how two PCEP speakers can establish a PCEP session that
   can carry information about SR-TE paths.

o Specifies processing rules for the ERO subobject.

o Defines a new path setup type to be used in the PATH-SETUP-TYPE
   and PATH-SETUP-TYPE-CAPABILITY TLVs ([RFC8408]).

o Defines a new sub-TLV for the PATH-SETUP-TYPE-CAPABILITY TLV.

The extensions specified in this document complement the existing
PCEP specifications to support SR-TE paths. As such, the PCEP
messages (e.g., Path Computation Request, Path Computation Reply,
Path Computation Report, Path Computation Update, Path Computation
Initiate, etc.,) are formatted according to [RFC5440], [RFC8231],
[RFC8281], and any other applicable PCEP specifications.

4. Object Formats

4.1. The OPEN Object

4.1.1. The Path Setup Type Capability TLV

   [RFC8408] defines the PATH-SETUP-TYPE-CAPABILITY TLV for use in the
   OPEN object. The PATH-SETUP-TYPE-CAPABILITY TLV contains an optional
   list of sub-TLVs which are intended to convey parameters that are
   associated with the path setup types supported by a PCEP speaker.

   This specification updates [RFC8408], as follows. It creates a new
   registry which defines the valid type indicators of the sub-TLVs of
   the PATH-SETUP-TYPE-CAPABILITY TLV (see Section 8.6). A PCEP speaker
   MUST NOT include a sub-TLV in the PATH-SETUP-TYPE-CAPABILITY TLV
   unless it appears in this registry. If a PCEP speaker receives a
   sub-TLV whose type indicator does not match one of those from the
   registry, or else is not recognised by the speaker, then the speaker
   MUST ignore the sub-TLV.

4.1.2. The SR PCE Capability sub-TLV

   This document defines a new Path Setup Type (PST) for SR, as follows:

   o PST = 1: Path is setup using Segment Routing Traffic Engineering.

   A PCEP speaker SHOULD indicate its support of the function described
   in this document by sending a PATH-SETUP-TYPE-CAPABILITY TLV in the
   OPEN object with this new PST included in the PST list.

   This document also defines the SR-PCE-CAPABILITY sub-TLV. PCEP
   speakers use this sub-TLV to exchange information about their SR
   capability. If a PCEP speaker includes PST=1 in the PST List of the
   PATH-SETUP-TYPE-CAPABILITY TLV then it MUST also include the SR-PCE-
   CAPABILITY sub-TLV inside the PATH-SETUP-TYPE-CAPABILITY TLV.

   The format of the SR-PCE-CAPABILITY sub-TLV is shown in the following
   figure:

```
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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=TBD11         |          Length=4             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Reserved            |     Flags     |N|X|    MSD      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
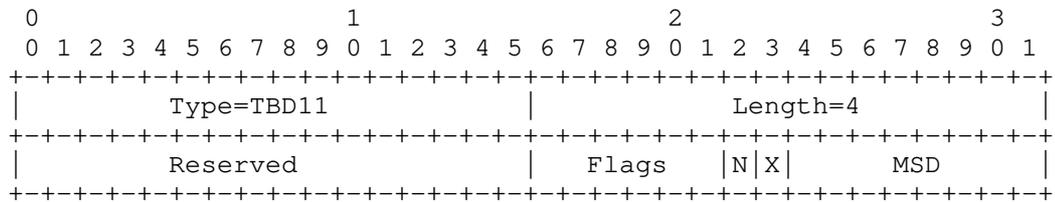
                 Figure 1: SR-PCE-CAPABILITY sub-TLV format

   The code point for the TLV type is TBD11. The TLV length is 4
   octets.

   The 32-bit value is formatted as follows.

   Reserved: MUST be set to zero by the sender and MUST be ignored by
      the receiver.

   Flags: This document defines the following flag bits. The other
      bits MUST be set to zero by the sender and MUST be ignored by the
      receiver.

      * N: A PCC sets this flag bit to 1 to indicate that it is capable
         of resolving a Node or Adjacency Identifier (NAI) to a SID.

      * X: A PCC sets this flag bit to 1 to indicate that it does not
         impose any limit on the MSD.

Maximum SID Depth (MSD): specifies the maximum number of SIDs (MPLS
    label stack depth in the context of this document) that a PCC is
    capable of imposing on a packet. Section 5.1 explains the
    relationship between this field and the X flag.

4.2. The RP/SRP Object

    To set up an SR-TE LSP using SR, the RP (Request Parameters) or SRP
    (Stateful PCE Request Parameters) object MUST include the PATH-SETUP-
    TYPE TLV, specified in [RFC8408], with the PST set to 1 (path setup
    using SR-TE).

    The LSP-IDENTIFIERS TLV MAY be present for the above PST type.

4.3. ERO

    An SR-TE path consists of one or more SIDs where each SID MAY be
    associated with the identifier that represents the node or adjacency
    corresponding to the SID. This identifier is referred to as the
    'Node or Adjacency Identifier' (NAI). As described later, a NAI can
    be represented in various formats (e.g., IPv4 address, IPv6 address,
    etc). Furthermore, a NAI is used for troubleshooting purposes and,
    if necessary, to derive SID value as described below.

    The ERO specified in [RFC5440] is used to carry SR-TE path
    information. In order to carry SID and/or NAI, this document defines
    a new ERO subobject referred to as "SR-ERO subobject" whose format is
    specified in the following section. An ERO carrying an SR-TE path
    consists of one or more ERO subobjects, and MUST carry only SR-ERO
    subobjects. Note that an SR-ERO subobject does not need to have both
    SID and NAI. However, at least one of them MUST be present.

    When building the MPLS label stack from ERO, a PCC MUST assume that
    SR-ERO subobjects are organized as a last-in-first-out stack. The
    first subobject relative to the beginning of ERO contains the
    information about the topmost label. The last subobject contains
    information about the bottommost label.

4.3.1. SR-ERO Subobject

    An SR-ERO subobject is formatted as shown in the following diagram.

```
 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=36   |    Length     |     NT    |     Flags    |F|S|C|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        SID (optional)                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                    NAI (variable, optional)                 //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
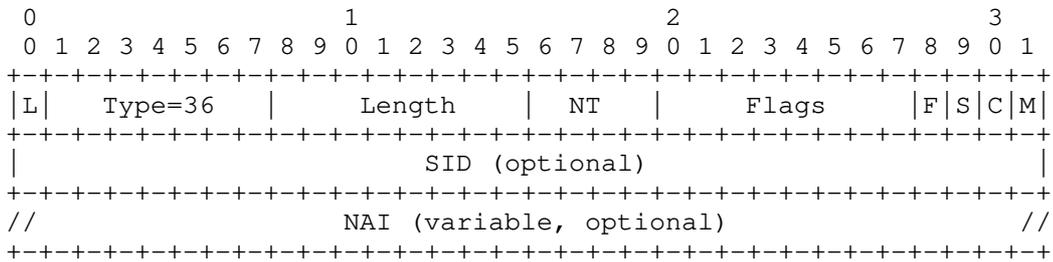
Figure 2: SR-ERO subobject format

The fields in the SR-ERO Subobject are as follows:

The 'L' Flag: Indicates whether the subobject represents a loose-hop
   in the LSP [RFC3209]. If this flag is set to zero, a PCC MUST NOT
   overwrite the SID value present in the SR-ERO subobject.
   Otherwise, a PCC MAY expand or replace one or more SID values in
   the received SR-ERO based on its local policy.

Type: Set to 36.

Length: Contains the total length of the subobject in octets. The
   Length MUST be at least 8, and MUST be a multiple of 4. An SR-ERO
   subobject MUST contain at least one of a SID or an NAI. The flags
   described below indicate whether the SID or NAI fields are absent.

NAI Type (NT): Indicates the type and format of the NAI contained in
   the object body, if any is present. If the F bit is set to zero
   (see below) then the NT field has no meaning and MUST be ignored
   by the receiver. This document describes the following NT values:

   NT=0 The NAI is absent.

   NT=1 The NAI is an IPv4 node ID.

   NT=2 The NAI is an IPv6 node ID.

   NT=3 The NAI is an IPv4 adjacency.

   NT=4 The NAI is an IPv6 adjacency with global IPv6 addresses.

   NT=5 The NAI is an unnumbered adjacency with IPv4 node IDs.

   NT=6 The NAI is an IPv6 adjacency with link-local IPv6 addresses.

Flags: Used to carry additional information pertaining to the SID.
   This document defines the following flag bits. The other bits
   MUST be set to zero by the sender and MUST be ignored by the
   receiver.

   * M: If this bit is set to 1, the SID value represents an MPLS
      label stack entry as specified in [RFC3032]. Otherwise, the
      SID value is an administratively configured value which
      represents an index into an MPLS label space (either SRGB or
      SRLB) per [RFC8402].

   * C: If the M bit and the C bit are both set to 1, then the TC,
      S, and TTL fields in the MPLS label stack entry are specified
      by the PCE. However, a PCC MAY choose to override these values
      according its local policy and MPLS forwarding rules. If the M
      bit is set to 1 but the C bit is set to zero, then the TC, S,
      and TTL fields MUST be ignored by the PCC. The PCC MUST set
      these fields according to its local policy and MPLS forwarding
      rules. If the M bit is set to zero then the C bit MUST be set
      to zero.

   * S: When this bit is set to 1, the SID value in the subobject
      body is absent. In this case, the PCC is responsible for
      choosing the SID value, e.g., by looking up in the SR-DB using
      the NAI which, in this case, MUST be present in the subobject.
      If the S bit is set to 1 then the M and C bits MUST be set to
      zero.

   * F: When this bit is set to 1, the NAI value in the subobject
      body is absent. The F bit MUST be set to 1 if NT=0, and
      otherwise MUST be set to zero. The S and F bits MUST NOT both
      be set to 1.

SID: The Segment Identifier. Depending on the M bit, it contains
   either:

   * A 4 octet index defining the offset into an MPLS label space
      per [RFC8402].

   * A 4 octet MPLS Label Stack Entry, where the 20 most significant
      bits encode the label value per [RFC3032].

NAI: The NAI associated with the SID. The NAI's format depends on
   the value in the NT field, and is described in the following
   section.

At least one of the SID and the NAI MUST be included in the SR-ERO
subobject, and both MAY be included.

4.3.2. NAI Associated with SID

   This document defines the following NAIs:

   'IPv4 Node ID' is specified as an IPv4 address. In this case, the
      NT value is 1 and the NAI field length is 4 octets.

   'IPv6 Node ID' is specified as an IPv6 address. In this case, the
      NT value is 2 and the NAI field length is 16 octets.

   'IPv4 Adjacency' is specified as a pair of IPv4 addresses. In this
      case, the NT value is 3 and the NAI field length is 8 octets. The
      format of the NAI is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Local IPv4 address                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Remote IPv4 address                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
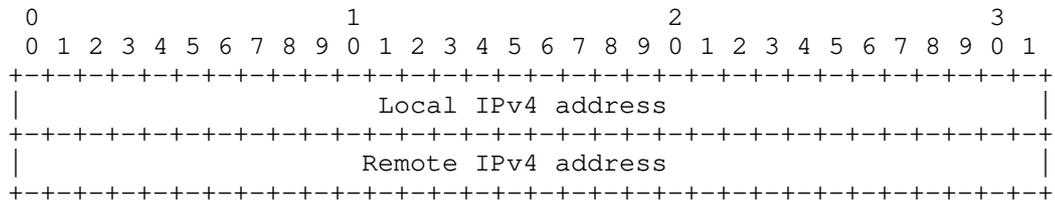
                   Figure 3: NAI for IPv4 adjacency

   'IPv6 Global Adjacency' is specified as a pair of global IPv6
     addresses. It is used to describe an IPv6 adjacency for a link
     that uses global IPv6 addresses. Each global IPv6 address is
     configured on a specific router interface, so together they
     identify an adjacency between a pair of routers. In this case,
     the NT value is 4 and the NAI field length is 32 octets. The
     format of the NAI is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//              Local IPv6 address (16 octets)                //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//              Remote IPv6 address (16 octets)               //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
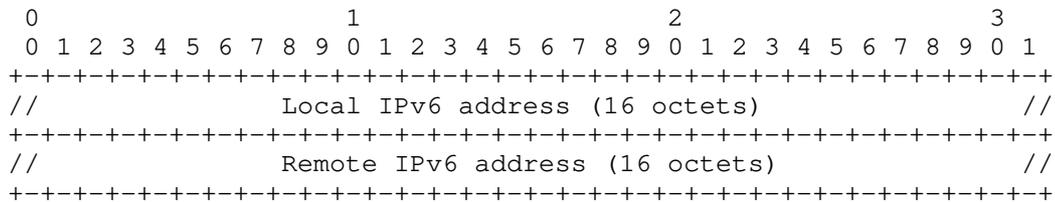
                Figure 4: NAI for IPv6 global adjacency

   'Unnumbered Adjacency with IPv4 NodeIDs' is specified as a pair of
      (node ID, interface ID) tuples. In this case, the NT value is 5
      and the NAI field length is 16 octets. The format of the NAI is
      shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Local Node-ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Local Interface ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Remote Node-ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Remote Interface ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
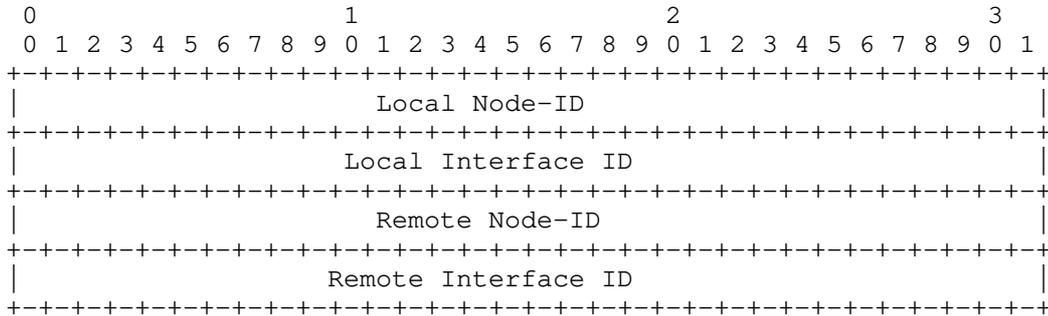
         Figure 5: NAI for Unnumbered adjacency with IPv4 Node IDs

   'IPv6 Link-Local Adjacency' is specified as a pair of (global IPv6
      address, interface ID) tuples. It is used to describe an IPv6
      adjacency for a link that uses only link local IPv6 addresses.
      Each global IPv6 address is configured on a specific router, so
      together they identify a pair of adjacent routers. The interface
      IDs identify the link that the adjacency is formed over. In this
      case, the NT value is 6 and the NAI field length is 40 octets.
      The format of the NAI is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                 Local IPv6 address (16 octets)             //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Local Interface ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                Remote IPv6 address (16 octets)            //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Remote Interface ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
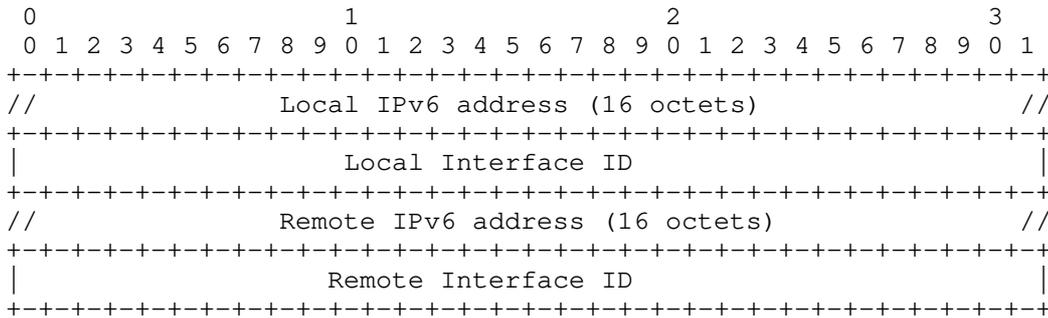
              Figure 6: NAI for IPv6 link-local adjacency

4.4. RRO

   A PCC reports an SR-TE LSP to a PCE by sending a PCRpt message, per
   [RFC8231]. The RRO on this message represents the SID list that was
   applied by the PCC, that is, the actual path taken by the LSP. The
   procedures of [RFC8231] with respect to the RRO apply equally to this
   specification without change.

   An RRO contains one or more subobjects called "SR-RRO subobjects"
   whose format is shown below:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Type=36    |    Length     |   NT  |    Flags      |F|S|C|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             SID                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        NAI (variable)                       //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
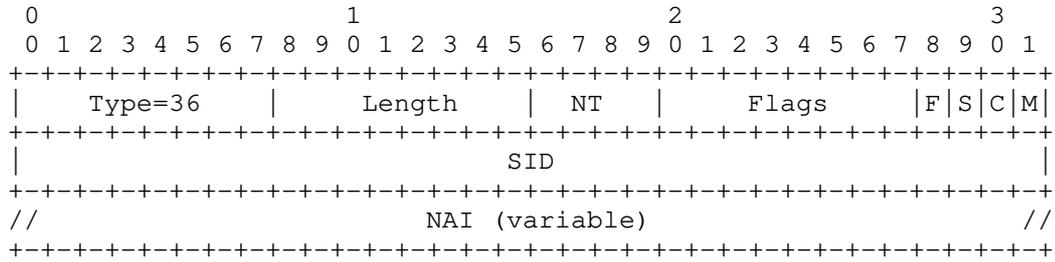
                    Figure 7: SR-RRO Subobject format

   The format of the SR-RRO subobject is the same as that of the SR-ERO
   subobject, but without the L flag.

   A PCC MUST order the SR-RRO subobjects such that the first subobject
   relative to the beginning of the RRO identifies the first segment
   visited by the SR-TE LSP, and the last subobject identifies the final
   segment of the SR-TE LSP, that is, its endpoint.

4.5. METRIC Object

   A PCC MAY request that PCE optimizes an individual path computation
   request to minimize the SID depth of the computed path by using the
   METRIC object defined in [RFC5440]. This document defines a new type
   for the METRIC object to be used for this purpose, as follows:

   o T = 11: Maximum SID Depth of the requested path.

   If the PCC includes a METRIC object of this type on a path
   computation request, then the PCE minimizes the SID depth of the
   computed path. If the B (bound) bit is set to to 1 in the METRIC
   object, then the PCE MUST NOT return a path whose SID depth exceeds
   the given metric-value. If the PCC did not set the X flag in its SR-
   PCE-CAPABILITY TLV, then it MUST set the B bit to 1. If the PCC set
   the X flag in its SR-PCE-CAPABILITY TLV, then it MAY set the B bit to
   1 or zero.

   If a PCEP session is established with a non-zero default MSD value,
   then the PCC MUST NOT send an MSD METRIC object with an MSD greater
   than the session's default MSD. If the PCE receives a path
   computation request with an MSD METRIC object on such a session that
   is greater than the session's default MSD, then it MUST consider the
   request invalid and send a PCErr with Error-Type = 10 ("Reception of
   an invalid object") and Error-Value 9 ("MSD exceeds the default for
   the PCEP session").

5. Procedures

5.1. Exchanging the SR PCE Capability

   A PCC indicates that it is capable of supporting the head-end
   functions for SR-TE LSP by including the SR-PCE-CAPABILITY sub-TLV in
   the Open message that it sends to a PCE. A PCE indicates that it is
   capable of computing SR-TE paths by including the SR-PCE-CAPABILITY
   sub-TLV in the Open message that it sends to a PCC.

   If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a
   PST list containing PST=1, and supports that path setup type, then it
   checks for the presence of the SR-PCE-CAPABILITY sub-TLV. If that
   sub-TLV is absent, then the PCEP speaker MUST send a PCErr message
   with Error-Type 10 (Reception of an invalid object) and Error-Value
   TBD1 (Missing PCE-SR-CAPABILITY sub-TLV) and MUST then close the PCEP
   session. If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV
   with a SR-PCE-CAPABILITY sub-TLV, but the PST list does not contain
   PST=1, then the PCEP speaker MUST ignore the SR-PCE-CAPABILITY sub-
   TLV.

   If a PCC sets the N flag to 1, then the PCE MAY send an SR-ERO
   subobject containing NAI and no SID (see Section 5.2). Otherwise,
   the PCE MUST NOT send an SR-ERO subobject containing NAI and no SID.

   The number of SIDs that can be imposed on a packet depends on the
   PCC's data plane's capability. If a PCC sets the X flag to 1 then
   the MSD is not used and MUST be set to zero. If a PCE receives an
   SR-PCE-CAPABILITY sub-TLV with the X flag set to 1 then it MUST
   ignore the MSD field and assumes that the sender can impose a SID
   stack of any depth. If a PCC sets the X flag to zero, then it sets
   the MSD field to the maximum number of SIDs that it can impose on a
   packet. In this case, the PCC MUST set the MSD to a number greater
   than zero. If a PCE receives an SR-PCE-CAPABILITY sub-TLV with the X
   flag and MSD both set to zero then it MUST send a PCErr message with
   Error-Type 10 (Reception of an invalid object) and Error-Value TBD10
   (Maximum SID depth must be nonzero) and MUST then close the PCEP
   session.

   Note that the MSD value exchanged via the SR-PCE-CAPABILITY sub-TLV
   indicates the SID/label imposition limit for the PCC node. It is
   anticipated that, in many deployments, the PCCs will have network
   interfaces that are homogeneous with respect to MSD (that is, each
   interface has the same MSD). In such cases, having a per-node MSD on
   the PCEP session is sufficient; the PCE SHOULD interpret this to mean
   that all network interfaces on the PCC have the given MSD. However,
   the PCE MAY also learn a per-node MSD and a per-interface MSD from
   the routing protocols, as specified in: [RFC8491]; [RFC8476];

[I-D.ietf-idr-bgp-ls-segment-routing-msd]. If the PCE learns the
per-node MSD of a PCC from a routing protocol, then it MUST ignore
the per-node MSD value in the SR-PCE-CAPABILITY sub-TLV and use the
per-node MSD learned from the routing protocol instead. If the PCE
learns the MSD of a network interface on a PCC from a routing
protocol, then it MUST use the per-interface MSD instead of the MSD
value in the SR-PCE-CAPABILITY sub-TLV when it computes a path that
uses that interface.

Once an SR-capable PCEP session is established with a non-zero MSD
value, the corresponding PCE MUST NOT send SR-TE paths with a number
of SIDs exceeding that MSD value. If a PCC needs to modify the MSD
value, it MUST close the PCEP session and re-establish it with the
new MSD value. If a PCEP session is established with a non-zero MSD
value, and the PCC receives an SR-TE path containing more SIDs than
specified in the MSD value, the PCC MUST send a PCErr message with
Error-Type 10 (Reception of an invalid object) and Error-Value 3
(Unsupported number of Segment ERO subobjects). If a PCEP session is
established with an MSD value of zero, then the PCC MAY specify an
MSD for each path computation request that it sends to the PCE, by
including a "maximum SID depth" metric object on the request, as
defined in Section 4.5.

The N flag, X flag and MSD value inside the SR-PCE-CAPABILITY sub-TLV
are meaningful only in the Open message sent from a PCC to a PCE. As
such, a PCE MUST set the N flag to zero, the X flag to 1 and MSD
value to zero in an outbound message to a PCC. Similarly, a PCC MUST
ignore any MSD value received from a PCE. If a PCE receives multiple
SR-PCE-CAPABILITY sub-TLVs in an Open message, it processes only the
first sub-TLV received.

5.2. ERO Processing

5.2.1. SR-ERO Validation

If a PCC does not support the SR PCE Capability and thus cannot
recognize the SR-ERO or SR-RRO subobjects, it will respond according
to the rules for a malformed object per [RFC5440].

On receiving an SR-ERO, a PCC MUST validate that the Length field,
the S bit, the F bit and the NT field are consistent, as follows.

o If NT=0, the F bit MUST be 1, the S bit MUST be zero and the
   Length MUST be 8.

o If NT=1, the F bit MUST be zero. If the S bit is 1, the Length
   MUST be 8, otherwise the Length MUST be 12.

o If NT=2, the F bit MUST be zero. If the S bit is 1, the Length
   MUST be 20, otherwise the Length MUST be 24.

o If NT=3, the F bit MUST be zero. If the S bit is 1, the Length
   MUST be 12, otherwise the Length MUST be 16.

o If NT=4, the F bit MUST be zero. If the S bit is 1, the Length
   MUST be 36, otherwise the Length MUST be 40.

o If NT=5, the F bit MUST be zero. If the S bit is 1, the Length
   MUST be 20, otherwise the Length MUST be 24.

o If NT=6, the F bit MUST be zero. If the S bit is 1, the Length
   MUST be 44, otherwise the Length MUST be 48.

If a PCC finds that the NT field, Length field, S bit and F bit are
not consistent, it MUST consider the entire ERO invalid and MUST send
a PCErr message with Error-Type = 10 ("Reception of an invalid
object") and Error-Value = 11 ("Malformed object").

If a PCC does not recognise or support the value in the NT field, it
MUST consider the entire ERO invalid and MUST send a PCErr message
with Error-Type = 10 ("Reception of an invalid object") and Error-
Value = TBD2 ("Unsupported NAI Type in Segment ERO subobject").

If a PCC receives an SR-ERO subobject in which the S and F bits are
both set to 1 (that is, both the SID and NAI are absent), it MUST
consider the entire ERO invalid and send a PCErr message with Error-
Type = 10 ("Reception of an invalid object") and Error-Value = 6
("Both SID and NAI are absent in SR-ERO subobject").

If a PCC receives an SR-ERO subobject in which the S bit is set to 1
and the F bit is set to zero (that is, the SID is absent and the NAI
is present), but the PCC does not support NAI resolution, it MUST
consider the entire ERO invalid and send a PCErr message with Error-
Type = 4 ("Not supported object") and Error-Value = 4 ("Unsupported
parameter").

If a PCC receives an SR-ERO subobject in which the S bit is set to 1
and either or both of the M or C bits is set to 1, it MUST consider
the entire ERO invalid and send a PCErr message with Error-Type = 10
("Reception of an invalid object") and Error-Value = 11 ("Malformed
object").

If a PCC receives an SR-ERO subobject in which the S bit is set to
zero and the M bit is set to 1, then the subobject contains an MPLS
label. The PCC MAY choose not to accept a label provided by the PCE,
based on it local policy. The PCC MUST NOT accept MPLS label value 3

(Implicit NULL), but it MAY accept other special purpose MPLS label
values. If the PCC decides not to accept an MPLS label value, it
MUST send a PCErr message with Error-Type = 10 ("Reception of an
invalid object") and Error Value = 2 ("Bad label value").

If both M and C bits of an SR-ERO subobject are set to 1, and if a
PCC finds erroneous setting in one or more of TC, S, and TTL fields,
it MAY overwrite those fields with values chosen according to its own
policy. If the PCC does not overwrite them, it MUST send a PCErr
message with Error-Type = 10 ("Reception of an invalid object") and
Error-Value = 4 ("Bad label format").

If the M bit of an SR-ERO subobject is set to zero but the C bit is
set to 1, then the PCC MUST consider the entire ERO invalid and MUST
send a PCErr message with Error-Type = 10 ("Reception of an invalid
object") and Error-Value = 11 ("Malformed object").

If a PCC receives an SR-ERO subobject in which the S bit is set to
zero and the M bit is set to zero, then the subobject contains a SID
index value. If the SID is an Adjacency-SID then the L flag MUST NOT
be set. If the L flag is set for an Adjacency-SID then the PCC MUST
send a PCErr message with Error-Type = 10 ("Reception of an invalid
object") and Error-Value = 11 ("Malformed object").

If a PCC detects that the subobjects of an ERO are a mixture of SR-
ERO subobjects and subobjects of other types, then it MUST send a
PCErr message with Error-Type = 10 ("Reception of an invalid object")
and Error-Value = 5 ("ERO mixes SR-ERO subobjects with other
subobject types").

The SR-ERO subobjects can be classified according to whether they
contain a SID representing an MPLS label value, a SID representing an
index value, or no SID. If a PCC detects that the SR-ERO subobjects
are a mixture of more than one of these types, then it MUST send a
PCErr message with Error-Type = 10 ("Reception of an invalid object")
and Error-Value = TBD9 ("Inconsistent SIDs in SR-ERO / SR-RRO
subobjects").

If an ERO specifies a new SR-TE path for an existing LSP and the PCC
determines that the ERO contains SR-ERO subobjects that are not
valid, then the PCC MUST NOT update the LSP.

5.2.2. Interpreting the SR-ERO

The SR-ERO contains a sequence of subobjects. Each SR-ERO subobject
in the sequence identifies a segment that the traffic will be
directed to, in the order given. That is, the first subobject

identifies the first segment the traffic will be directed to, the
second subobject represents the second segment, and so on.

The PCC interprets the SR-ERO by converting it to an MPLS label stack
plus a next hop. The PCC sends packets along the segment routed path
by prepending the MPLS label stack onto the packets and sending the
resulting, modified packet to the next hop.

The PCC uses a different procedure to do this conversion, depending
on the information that the PCE has provided in the subobjects.

o If the subobjects contain SID index values, then the PCC converts
   them into the corresponding MPLS labels by following the procedure
   defined in [I-D.ietf-spring-segment-routing-mpls].

o If the subobjects contain NAI only, the PCC first converts each
   NAI into a SID index value and then proceeds as above. To convert
   an NAI to a SID index, the PCC looks for a fully-specified prefix
   or adjacency matching the fields in the NAI. If the PCC finds a
   matching prefix/adjacency, and the matching prefix/adjacency has a
   SID associated with it, then the PCC uses that SID. If the PCC
   cannot find a matching prefix/adjacency, or if the matching
   prefix/adjacency has no SID associated with it, the PCC behaves as
   specified in Section 5.2.2.1.

o If the subobjects contain MPLS labels, then the PCC looks up the
   offset of the first subobject's label in its SRGB or SRLB. This
   gives the first SID. The PCC pushes the labels in any remaining
   subobjects onto the packet (with the final subobject specifying
   the bottom-of-stack label).

For all cases above, after the PCC has imposed the label stack on the
packet, it sends the packet to the segment identified by the first
SID.

5.2.2.1. Handling Errors During SR-ERO Conversion

There are several errors that can occur during the process of
converting an SR-ERO sequence to an MPLS label stack and a next hop.
The PCC deals with them as follows.

o If the PCC cannot find a SID index in the SR-DB, it MUST send a
   PCErr message with Error-Type = 10 ("Reception of an invalid
   object") and Error-Value = TBD3 ("Unknown SID").

o If the PCC cannot find an NAI in the SR-DB, it MUST send a PCErr
   message with Error-Type = 10 ("Reception of an invalid object")
   and Error-Value = TBD4 ("NAI cannot be resolved to a SID").

   o If the PCC needs to convert a SID into an MPLS label value but
      cannot find the corresponding router's SRGB in the SR-DB, it MUST
      send a PCErr message with Error-Type = 10 ("Reception of an
      invalid object") and Error-Value = TBD5 ("Could not find SRGB").

   o If the PCC finds that a router's SRGB is not large enough for a
      SID index value, it MUST send a PCErr message with Error-Type = 10
      ("Reception of an invalid object") and Error-Value = TBD6 ("SID
      index exceeds SRGB size").

   o If the PCC needs to convert a SID into an MPLS label value but
      cannot find the corresponding router's SRLB in the SR-DB, it MUST
      send a PCErr message with Error-Type = 10 ("Reception of an
      invalid object") and Error-Value = TBD7 ("Could not find SRLB").

   o If the PCC finds that a router's SRLB is not large enough for a
      SID index value, it MUST send a PCErr message with Error-Type = 10
      ("Reception of an invalid object") and Error-Value = TBD8 ("SID
      index exceeds SRLB size").

   o If the number of labels in the computed label stack exceeds the
      maximum number of SIDs that the PCC can impose on the packet, it
      MUST send a PCErr message with Error-Type = 10 ("Reception of an
      invalid object") and Error-Value = 3 ("Unsupported number of
      Segment ERO subobjects").

   If an ERO specifies a new SR-TE path for an existing LSP and the PCC
   encounters an error while processing the ERO, then the PCC MUST NOT
   update the LSP.

5.3. RRO Processing

   The syntax checking rules that apply to the SR-RRO subobject are
   identical to those of the SR-ERO subobject, except as noted below.

   If a PCEP speaker receives an SR-RRO subobject in which both SID and
   NAI are absent, it MUST consider the entire RRO invalid and send a
   PCErr message with Error-Type = 10 ("Reception of an invalid object")
   and Error-Value = 7 ("Both SID and NAI are absent in SR-RRO
   subobject").

   If a PCE detects that the subobjects of an RRO are a mixture of SR-
   RRO subobjects and subobjects of other types, then it MUST send a
   PCErr message with Error-Type = 10 ("Reception of an invalid object")
   and Error-Value = 10 ("RRO mixes SR-RRO subobjects with other
   subobject types").

The SR-RRO subobjects can be classified according to whether they
contain a SID representing an MPLS label value or a SID representing
an index value, or no SID. If a PCE detects that the SR-RRO
subobjects are a mixture of more than one of these types, then it
MUST send a PCErr message with Error-Type = 10 ("Reception of an
invalid object") and Error-Value = TBD9 ("Inconsistent SIDs in SR-ERO
/ SR-RRO subobjects").

6. Management Considerations

   This document adds a new path setup type to PCEP to allow LSPs to be
   set up using segment routing techniques. This path setup type may be
   used with PCEP alongside other path setup types, such as RSVP-TE, or
   it may be used exclusively.

6.1. Controlling the Path Setup Type

   The following factors control which path setup type is used for a
   given LSP.

   o The available path setup types are constrained to those that are
      supported by, or enabled on, the PCEP speakers. The PATH-SETUP-
      TYPE-CAPABILITY TLV indicates which path setup types a PCEP
      speaker supports. To use segment routing as a path setup type, it
      is a prerequisite that the PCC and PCE both include PST=1 in the
      list of supported path setup types in this TLV, and also include
      the SR-PCE-CAPABILITY sub-TLV.

   o When a PCE initiates an LSP, it proposes which path setup type to
      use by including it in the PATH-SETUP-TYPE TLV in the SRP object
      of the PCInitiate message. The PCE chooses the path setup type
      based on the capabilities of the network nodes on the path and on
      its local policy. The PCC MAY choose to accept the proposed path
      setup type, or to reject the PCInitiate request, based on its
      local policy.

   o When a PCC requests a path for an LSP, it can nominate a preferred
      path setup type by including it in the PATH-SETUP-TYPE TLV in the
      RP object of the PCReq message. The PCE MAY choose to reply with
      a path of the requested type, or to reply with a path of a
      different type, or to reject the request, based on the
      capabilities of the network nodes on the path and on its local
      policy.

   The operator can influence the path setup type as follows.

   o Implementations MUST allow the operator to enable and disable the
      segment routing path setup type on a PCEP-speaking device.

Implementations MAY also allow the operator to enable and disable the RSVP-TE path setup type.

o PCE implementations MUST allow the operator to specify that an LSP should be instantiated using segment routing or RSVP-TE as the proposed path setup type.

o PCE implementations MAY allow the operator to configure a preference for the PCE to propose paths using segment routing or RSVP-TE in the absence of a specified path setup type.

o PCC implementations MUST allow the operator to specify that a path requested for an LSP nominates segment routing or RSVP-TE as the path setup type.

o PCC implementations MAY allow the operator to configure a preference for the PCC to nominate segment routing or RSVP-TE as the path setup type if none is specified for an LSP.

o PCC implementations SHOULD allow the operator to configure a PCC to refuse to set up an LSP using an undesired path setup type.

6.2. Migrating a Network to Use PCEP Segment Routed Paths

   This section discusses the steps that the operator takes when migrating a network to enable PCEP to set up paths using segment routing as the path setup type.

o The operator enables the segment routing PST on the PCE servers.

o The operator enables the segment routing PST on the PCCs.

o The operator resets each PCEP session. The PCEP sessions come back up with segment routing enabled.

o If the operator detects a problem, they can roll the network back to its initial state by disabling the segment routing PST on the PCEP speakers and resetting the PCEP sessions.

   Note that the data plane is unaffected if a PCEP session is reset. Any LSPs that were set up before the session reset will remain in place and will still be present after the session comes back up.

   An implementation SHOULD allow the operator to manually trigger a PCEP session to be reset.

   An implementation MAY automatically reset a PCEP session when an operator reconfigures the PCEP speaker's capabilities. However, note

that if the capabilities at both ends of the PCEP session are not
reconfigured simultaneously, then the session could be reset twice,
which could lead to unnecessary network traffic. Therefore, such
implementations SHOULD allow the operator to override this behaviour
and wait instead for a manual reset.

Once segment routing is enabled on a PCEP session, it can be used as
the path setup type for future LSPs.

User traffic is not automatically migrated from existing LSPs onto
segment routed LSPs just by enabling the segment routing PST in PCEP.
The migration of user traffic from existing LSPs onto segment routing
LSPs is beyond the scope of this document.

6.3. Verification of Network Operation

The operator needs the following information to verify that PCEP is
operating correctly with respect to the segment routing path setup
type.

o   An implementation SHOULD allow the operator to view whether the
    PCEP speaker sent the segment routing PST capability to its peer.
    If the PCEP speaker is a PCC, then the implementation SHOULD also
    allow the operator to view the values of the L and N flags that
    were sent, and the value of the MSD field that was sent.

o   An implementation SHOULD allow the operator to view whether the
    peer sent the segment routing PST capability. If the peer is a
    PCC, then the implementation SHOULD also allow the operator to
    view the values of the L and N flags and MSD fields that the peer
    sent.

o   An implementation SHOULD allow the operator to view whether the
    segment routing PST is enabled on the PCEP session.

o   If one PCEP speaker advertises the segment routing PST capability,
    but the other does not, then the implementation SHOULD create a
    log to inform the operator of the capability mismatch.

o   An implementation SHOULD allow the operator to view the PST that
    was proposed, or requested, for an LSP, and the PST that was
    actually used.

o   If a PCEP speaker decides to use a different PST to the one that
    was proposed, or requested, for an LSP, then the implementation
    SHOULD create a log to inform the operator that the expected PST
    has not been used. The log SHOULD give the reason for this choice
    (local policy, equipment capability etc.)

o If a PCEP speaker rejects a segment routing path, then it SHOULD
   create a log to inform the operator, giving the reason for the
   decision (local policy, MSD exceeded etc.)

6.4. Relationship to Existing Management Models

   The PCEP YANG module is defined in [I-D.ietf-pce-pcep-yang]. In
   future, this YANG module should be extended or augmented to provide
   the following additional information relating to segment routing:

   o The advertised PST capabilities and MSD per PCEP session.

   o The PST configured for, and used by, each LSP.

   The PCEP MIB [RFC7420] could also be updated to include this
   information.

7. Security Considerations

   The security considerations described in [RFC5440], [RFC8231],
   [RFC8281] and [RFC8408] are applicable to this specification. No
   additional security measure is required.

   Note that this specification enables a network controller to
   instantiate a path in the network without the use of a hop-by-hop
   signaling protocol (such as RSVP-TE). This creates an additional
   vulnerability if the security mechanisms of [RFC5440], [RFC8231] and
   [RFC8281] are not used. If there is no integrity protection on the
   session, then an attacker could create a path which is not subjected
   to the further verification checks that would be performed by the
   signaling protocol.

   Note that this specification adds the MSD field to the OPEN message
   (see Section 4.1.2) which discloses how many MPLS labels the sender
   can push onto packets that it forwards into the network. If the
   security mechanisms of [RFC8231] and [RFC8281] are not used with
   strong encryption, then an attacker could use this new field to gain
   intelligence about the capabilities of the edge devices in the
   network.

8. IANA Considerations

8.1. PCEP ERO and RRO subobjects

   This document defines a new subobject type for the PCEP explicit
   route object (ERO), and a new subobject type for the PCEP record
   route object (RRO). The code points for subobject types of these
   objects is maintained in the RSVP parameters registry, under the

   EXPLICIT_ROUTE and ROUTE_RECORD objects. IANA is requested to
   confirm the early allocation of the following code points in the RSVP
   Parameters registry for each of the new subobject types defined in
   this document.

```
   Object                   Subobject                 Subobject Type
   ---------------------    -------------------------  ------------------
   EXPLICIT_ROUTE           SR-ERO (PCEP-specific)    36
   ROUTE_RECORD             SR-RRO (PCEP-specific)    36
```

8.2. New NAI Type Registry

   IANA is requested to create a new sub-registry within the "Path
   Computation Element Protocol (PCEP) Numbers" registry called "PCEP
   SR-ERO NAI Types". The allocation policy for this new registry
   should be by IETF Review. The new registry should contain the
   following values:

```
       Value           Description                 Reference

       0               NAI is absent.              This document
       1               NAI is an IPv4 node ID.     This document
       2               NAI is an IPv6 node ID.     This document
       3               NAI is an IPv4 adjacency.    This document
       4               NAI is an IPv6 adjacency with This document
                       global IPv6 addresses.
       5               NAI is an unnumbered          This document
                       adjacency with IPv4 node IDs.
       6               NAI is an IPv6 adjacency with This document
                       link-local IPv6 addresses.
```

8.3. New SR-ERO Flag Registry

   IANA is requested to create a new sub-registry, named "SR-ERO Flag
   Field", within the "Path Computation Element Protocol (PCEP) Numbers"
   registry to manage the Flag field of the SR-ERO subobject. New
   values are to be assigned by Standards Action [RFC8126]. Each bit
   should be tracked with the following qualities:

   o Bit number (counting from bit 0 as the most significant bit)

   o Capability description

   o Defining RFC

   The following values are defined in this document:

```
                Bit     Description           Reference

                0-7     Unassigned
                8       NAI is absent (F)     This document
                9       SID is absent (S)     This document
                10      SID specifies TC, S   This document
                        and TTL in addition
                        to an MPLS label (C)
                11      SID specifies an MPLS This document
                        label (M)
```

8.4. PCEP-Error Object

   IANA is requested to confirm the early allocation of the code-points
   in the PCEP-ERROR Object Error Types and Values registry for the
   following new error-values:


```
   Error-Type    Meaning
   ----------    -------
   10            Reception of an invalid object.

                 Error-value = 2:                 Bad label value
                 Error-value = 3:                 Unsupported number
                                                  of SR-ERO
                                                  subobjects
                 Error-value = 4:                 Bad label format
                 Error-value = 5:                 ERO mixes SR-ERO
                                                  subobjects with
                                                  other subobject
                                                  types
                 Error-value = 6:                 Both SID and NAI
                                                  are absent in SR-
                                                  ERO subobject
                 Error-value = 7:                 Both SID and NAI
                                                  are absent in SR-
                                                  RRO subobject
                 Error-value = 9:                 MSD exceeds the
                                                  default for the
                                                  PCEP session
                 Error-value = 10:                RRO mixes SR-RRO
                                                  subobjects with
                                                  other subobject
                                                  types
                 Error-value = TBD1:              Missing PCE-SR-
                                                  CAPABILITY sub-TLV
```

```
             Error-value = TBD2:                    Unsupported NAI
                                                    Type in SR-ERO
                                                    subobject
             Error-value = TBD3:                    Unknown SID
             Error-value = TBD4:                    NAI cannot be
                                                    resolved to a SID
             Error-value = TBD5:                    Could not find SRGB
             Error-value = TBD6:                    SID index exceeds
                                                    SRGB size
             Error-value = TBD7:                    Could not find SRLB
             Error-value = TBD8:                    SID index exceeds
                                                    SRLB size
             Error-value = TBD9:                    Inconsistent SIDs
                                                    in SR-ERO / SR-RRO
                                                    subobjects
             Error-value = TBD10:                   MSD must be nonzero
```

   Note to IANA: this draft originally had an early allocation for
   Error-value=11 (Malformed object) in the above list. However, we
   have since moved the definition of that code point to RFC8408.

   Note to IANA: some Error-values in the above list were defined after
   the early allocation took place, and so do not currently have a code
   point assigned. Please assign code points from the indicated
   registry and replace each instance of "TBD1", "TBD2" etc. in this
   document with the respective code points.

   Note to IANA: some of the Error-value descriptive strings above have
   changed since the early allocation. Please refresh the registry.

8.5. PCEP TLV Type Indicators

   IANA is requested to confirm the early allocation of the following
   code point in the PCEP TLV Type Indicators registry. Note that this
   TLV type indicator is deprecated but retained in the registry to
   ensure compatibility with early implementations of this
   specification. See Appendix A for details.

Value	Meaning	Reference
26	SR-PCE-CAPABILITY	This document
	(deprecated)	

8.6. PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators

   IANA is requested to create a new sub-registry, named "PATH-SETUP-
   TYPE-CAPABILITY Sub-TLV Type Indicators", within the "Path
   Computation Element Protocol (PCEP) Numbers" registry to manage the

type indicator space for sub-TLVs of the PATH-SETUP-TYPE-CAPABILITY
TLV. New values are to be assigned by Standards Action [RFC8126].
The valid range of values in the registry is 0-65535. IANA is
requested to initialize the registry with the following values. All
other values in the registry should be marked as "Unassigned".

Value	Meaning	Reference
0	Reserved	This document
TBD11 (recommended 26)	SR-PCE-CAPABILITY	This document

Note to IANA: Please replace each instance of "TBD11" in this
document with the allocated code point. We have recommended that
value 26 be used for consistency with the deprecated value in the
PCEP TLV Type Indicators registry.

8.7. New Path Setup Type

   [RFC8408] created a sub-registry within the "Path Computation Element
   Protocol (PCEP) Numbers" registry called "PCEP Path Setup Types".
   IANA is requested to allocate a new code point within this registry,
   as follows:

Value	Description	Reference
1	Traffic engineering path is setup using Segment Routing.	This document

8.8. New Metric Type

   IANA is requested to confirm the early allocation of the following
   code point in the PCEP METRIC object T field registry:

Value	Description	Reference
11	Segment-ID (SID) Depth.	This document

8.9. SR PCE Capability Flags

   IANA is requested to create a new sub-registry, named "SR Capability
   Flag Field", within the "Path Computation Element Protocol (PCEP)
   Numbers" registry to manage the Flag field of the SR-PCE-CAPABILITY
   TLV. New values are to be assigned by Standards Action [RFC8126].
   Each bit should be tracked with the following qualities:

   o Bit number (counting from bit 0 as the most significant bit)
   o Capability description
   o Defining RFC

The following values are defined in this document:

Bit	Description	Reference
0-5	Unassigned	
6	Node or Adjacency Identifier (NAI) is supported (N)	This document
7	Unlimited Maximum SID Depth (X)	This document

Note to IANA: The name of bit 7 has changed from "Unlimited Maximum
SID Depth (L)" to "Unlimited Maximum SID Depth (X)".

9. Contributors

The following people contributed to this document:

   - Lakshmi Sharma
   - Jan Medved
   - Edward Crabbe
   - Robert Raszuk
   - Victor Lopez

10. Acknowledgements

We thank Ina Minei, George Swallow, Marek Zavodsky, Dhruv Dhody, Ing-
Wher Chen and Tomas Janciga for the valuable comments.

11. References

11.1. Normative References

   [I-D.ietf-spring-segment-routing-mpls]
             Bashandy, A., Filsfils, C., Previdi, S., Decraene, B.,
             Litkowski, S., and R. Shakir, "Segment Routing with MPLS
             data plane", draft-ietf-spring-segment-routing-mpls-18
             (work in progress), December 2018.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
             Requirement Levels", BCP 14, RFC 2119,
             DOI 10.17487/RFC2119, March 1997,
             <https://www.rfc-editor.org/info/rfc2119>.

   [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y.,
             Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack
             Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001,
             <https://www.rfc-editor.org/info/rfc3032>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <https://www.rfc-editor.org/info/rfc5440>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for Stateful PCE", RFC 8231,
              DOI 10.17487/RFC8231, September 2017,
              <https://www.rfc-editor.org/info/rfc8231>.

   [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for PCE-Initiated LSP Setup in a Stateful PCE
              Model", RFC 8281, DOI 10.17487/RFC8281, December 2017,
              <https://www.rfc-editor.org/info/rfc8281>.

   [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L.,
              Decraene, B., Litkowski, S., and R. Shakir, "Segment
              Routing Architecture", RFC 8402, DOI 10.17487/RFC8402,
              July 2018, <https://www.rfc-editor.org/info/rfc8402>.

   [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J.
              Hardwick, "Conveying Path Setup Type in PCE Communication
              Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408,
              July 2018, <https://www.rfc-editor.org/info/rfc8408>.

   [RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg,
              "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491,
              DOI 10.17487/RFC8491, November 2018,
              <https://www.rfc-editor.org/info/rfc8491>.

11.2. Informative References

   [I-D.ietf-6man-segment-routing-header]
              Filsfils, C., Previdi, S., Leddy, J., Matsushima, S., and
              d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header
              (SRH)", draft-ietf-6man-segment-routing-header-16 (work in
              progress), February 2019.

   [I-D.ietf-idr-bgp-ls-segment-routing-msd]
             Tantsura, J., Chunduri, U., Mirsky, G., and S. Sivabalan,
             "Signaling MSD (Maximum SID Depth) using Border Gateway
             Protocol Link-State", draft-ietf-idr-bgp-ls-segment-
             routing-msd-02 (work in progress), August 2018.

   [I-D.ietf-isis-segment-routing-extensions]
             Previdi, S., Ginsberg, L., Filsfils, C., Bashandy, A.,
             Gredler, H., and B. Decraene, "IS-IS Extensions for
             Segment Routing", draft-ietf-isis-segment-routing-
             extensions-22 (work in progress), December 2018.

   [I-D.ietf-ospf-segment-routing-extensions]
             Psenak, P., Previdi, S., Filsfils, C., Gredler, H.,
             Shakir, R., Henderickx, W., and J. Tantsura, "OSPF
             Extensions for Segment Routing", draft-ietf-ospf-segment-
             routing-extensions-27 (work in progress), December 2018.

   [I-D.ietf-pce-pcep-yang]
             Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A
             YANG Data Model for Path Computation Element
             Communications Protocol (PCEP)", draft-ietf-pce-pcep-
             yang-09 (work in progress), October 2018.

   [I-D.ietf-spring-segment-routing-policy]
             Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d.,
             bogdanov@google.com, b., and P. Mattes, "Segment Routing
             Policy Architecture", draft-ietf-spring-segment-routing-
             policy-02 (work in progress), October 2018.

   [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V.,
             and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP
             Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001,
             <https://www.rfc-editor.org/info/rfc3209>.

   [RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation
             Element (PCE) Communication Protocol Generic
             Requirements", RFC 4657, DOI 10.17487/RFC4657, September
             2006, <https://www.rfc-editor.org/info/rfc4657>.

   [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
             Hardwick, "Path Computation Element Communication Protocol
             (PCEP) Management Information Base (MIB) Module",
             RFC 7420, DOI 10.17487/RFC7420, December 2014,
             <https://www.rfc-editor.org/info/rfc7420>.

   [RFC8126]   Cotton, M., Leiba, B., and T. Narten, "Guidelines for
               Writing an IANA Considerations Section in RFCs", BCP 26,
               RFC 8126, DOI 10.17487/RFC8126, June 2017,
               <https://www.rfc-editor.org/info/rfc8126>.

   [RFC8413]   Zhuang, Y., Wu, Q., Chen, H., and A. Farrel, "Framework
               for Scheduled Use of Resources", RFC 8413,
               DOI 10.17487/RFC8413, July 2018,
               <https://www.rfc-editor.org/info/rfc8413>.

   [RFC8476]   Tantsura, J., Chunduri, U., Aldrin, S., and P. Psenak,
               "Signaling Maximum SID Depth (MSD) Using OSPF", RFC 8476,
               DOI 10.17487/RFC8476, December 2018,
               <https://www.rfc-editor.org/info/rfc8476>.

Appendix A. Compatibility with Early Implementations

   An early implementation of this specification will send the SR-
   CAPABILITY-TLV as a top-level TLV in the OPEN object instead of
   sending the PATH-SETUP-TYPE-CAPABILITY TLV in the OPEN object.
   Implementations that wish to interoperate with such early
   implementations should also send the SR-CAPABILITY-TLV as a top-level
   TLV in their OPEN object and should interpret receiving this top-
   level TLV as though the sender had sent a PATH-SETUP-TYPE-CAPABILITY
   TLV with a PST list of (0, 1) (that is, both RSVP-TE and SR-TE PSTs
   are supported) with the SR-CAPABILITY-TLV as a sub-TLV. If a PCEP
   speaker receives an OPEN object in which both the SR-CAPABILITY-TLV
   and PATH-SETUP-TYPE-CAPABILITY TLV appear as top-level TLVs, then it
   should ignore the top-level SR-CAPABILITY-TLV and process only the
   PATH-SETUP-TYPE-CAPABILITY TLV.

Authors' Addresses

   Siva Sivabalan
   Cisco Systems, Inc.
   2000 Innovation Drive
   Kanata, Ontario K2K 3E8
   Canada

   Email: msiva@cisco.com

Clarence Filsfils
Cisco Systems, Inc.
Pegasus Parc
De kleetlaan 6a, DIEGEM BRABANT 1831
BELGIUM

Email: cfilsfil@cisco.com

Jeff Tantsura
Apstra, Inc.
333 Middlefield Rd #200
Menlo Park, CA 94025
USA

Email: jefftant.ietf@gmail.com

Wim Henderickx
Nokia
Copernicuslaan 50
Antwerp 2018, CA 95134
BELGIUM

Email: wim.henderickx@alcatel-lucent.com

Jon Hardwick
Metaswitch Networks
100 Church Street
Enfield, Middlesex
UK

Email: jonathan.hardwick@metaswitch.com

March 1, 2019

PCEP Extension for WSON Routing and Wavelength Assignment

draft-ietf-pce-wson-rwa-ext-17

Abstract

   This document provides the Path Computation Element communication
   Protocol (PCEP) extensions for the support of Routing and Wavelength
   Assignment (RWA) in Wavelength Switched Optical Networks (WSON).
   Path provisioning in WSONs requires a routing and wavelength
   assignment (RWA) process. From a path computation perspective,
   wavelength assignment is the process of determining which wavelength
   can be used on each hop of a path and forms an additional routing
   constraint to optical path computation.

Status of this Memo

The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html.

This Internet-Draft will expire on September 1, 2019.

Copyright Notice

Table of Contents

1. Terminology

   This document uses the terminology defined in [RFC4655], and
   [RFC5440].

2. Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in
   BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

3. Introduction

   [RFC5440] specifies the Path Computation Element (PCE) Communication
   Protocol (PCEP) for communications between a Path Computation Client
   (PCC) and a PCE, or between two PCEs. Such interactions include
   path computation requests and path computation replies as well as
   notifications of specific states related to the use of a PCE in the

context of Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering.

A PCC is said to be any network component that makes such a request and may be, for instance, an Optical Switching Element within a Wavelength Division Multiplexing (WDM) network. The PCE, itself, can be located anywhere within the network, and may be within an optical switching element, a Network Management System (NMS) or Operational Support System (OSS), or may be an independent network server.

This document provides the PCEP extensions for the support of Routing and Wavelength Assignment (RWA) in Wavelength Switched Optical Networks (WSON) based on the requirements specified in [RFC6163] and [RFC7449].

WSON refers to WDM based optical networks in which switching is performed selectively based on the wavelength of an optical signal. The devices used in WSONs that are able to switch signals based on signal wavelength are known as Lambda Switch Capable (LSC). WSONs can be transparent or translucent. A transparent optical network is made up of optical devices that can switch but not convert from one wavelength to another, all within the optical domain. On the other hand, translucent networks include 3R regenerators (Re-amplification, Re-shaping, Re-timing) that are sparsely placed. The main function of the 3R regenerators is to convert one optical wavelength to another.

A Lambda Switch Capable (LSC) Label Switched Path (LSP) may span one or several transparent segments, which are delimited by 3R regenerators typically with electronic regenerator and optional wavelength conversion. Each transparent segment or path in WSON is referred to as an optical path. An optical path may span multiple fiber links and the path should be assigned the same wavelength for each link. In such case, the optical path is said to satisfy the wavelength-continuity constraint. Figure 1 illustrates the relationship between a LSC LSP and transparent segments (optical paths).

```
+---+       +-----+       +-----+       +-----+       +-----+
|   |I1     |     |       |     |       |     |       |   I2|     |
|   |o------|     |       |-------[(3R) ]------|     |       |--------o|     |
|   |       |     |       |     |       |     |       |     |     |
+---+       +-----+       +-----+       +-----+       +-----+
  (X   LSC)      (LSC   LSC)      (LSC   LSC)      (LSC   X)
   <------->      <------->        <----->        <------->
   <---------------------><---------------------->
   Transparent  Segment          Transparent  Segment
   <--------------------------------------------------->
                     LSC LSP
```
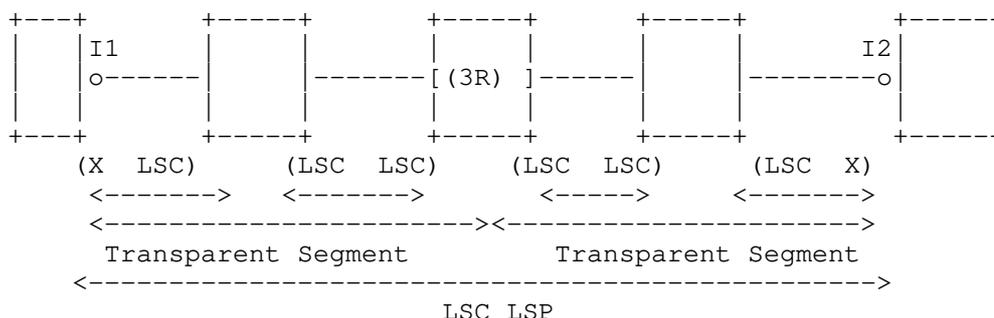
Figure 1 Illustration of a LSC LSP and transparent segments

Note that two transparent segments within a WSON LSP do not need to
operate on the same wavelength (due to the wavelength conversion
capabilities). Two optical channels that share a common fiber link
cannot be assigned the same wavelength; Otherwise, the two signals
would interfere with each other. Note that advanced additional
multiplexing techniques such as polarization based multiplexing are
not addressed in this document since the physical layer aspects are
not currently standardized. Therefore, assigning the proper
wavelength on a path is an essential requirement in the optical path
computation process.

When a switching node has the ability to perform wavelength
conversion, the wavelength-continuity constraint can be relaxed, and
a LSC Label Switched Path (LSP) may use different wavelengths on
different links along its route from origin to destination. It is,
however, to be noted that wavelength converters may be limited due
to their relatively high cost, while the number of WDM channels that
can be supported in a fiber is also limited. As a WSON can be
composed of network nodes that cannot perform wavelength conversion,
nodes with limited wavelength conversion, and nodes with full
wavelength conversion abilities, wavelength assignment is an
additional routing constraint to be considered in all optical path
computation.

For example (see Figure 1), within a translucent WSON, a LSC LSP may
be established between interfaces I1 and I2, spanning 2 transparent
segments (optical paths) where the wavelength continuity constraint
applies (i.e. the same unique wavelength must be assigned to the LSP
at each TE link of the segment). If the LSC LSP induced a Forwarding
Adjacency / TE link, the switching capabilities of the TE link would

be (X X) where X refers to the switching capability of I1 and I2.
For example, X can be Packet Switch Capable (PSC), Time Division
Multiplexing (TDM), etc.

This document aligns with GMPLS extensions for PCEP [PCEP-GMPLS] for
generic properties such as label, label-set and label assignment
noting that wavelength is a type of label. Wavelength restrictions
and constraints are also formulated in terms of labels per
[RFC7579].

The optical modulation properties, which are also referred to as
signal compatibility, are already considered in signaling in
[RFC7581] and [RFC7688]. In order to improve the signal quality and
limit some optical effects several advanced modulation processing
capabilities are used by the mechanisms specified in this document.
These modulation capabilities contribute not only to optical signal
quality checks but also constrain the selection of sender and
receiver, as they should have matching signal processing
capabilities. This document includes signal compatibility
constraints as part of RWA path computation. That is, the signal
processing capabilities (e.g., modulation and Forward Error
Correction (FEC)) indicated by means of optical interface class
(OIC) must be compatible between the sender and the receiver of the
optical path across all optical elements.

This document, however, does not address optical impairments as part
of RWA path computation. See [RFC6566] for the framework for optical
impairments.

4. Encoding of a RWA Path Request

Figure 2 shows one typical PCE based implementation, which is
referred to as the Combined Process (R&WA). With this architecture,
the two processes of routing and wavelength assignment are accessed
via a single PCE. This architecture is the base architecture
specified in [RFC6163] and the PCEP extensions that are specified in
this document are based on this architecture.

```
                  +--------------------------+
   +-----+        |  +-------+     +--+       |
   |     |        |  |Routing|     |WA|       |
   | PCC |<----->|  +-------+     +--+       |
   |     |        |                          |
   +-----+        |           PCE            |
                  +--------------------------+
```
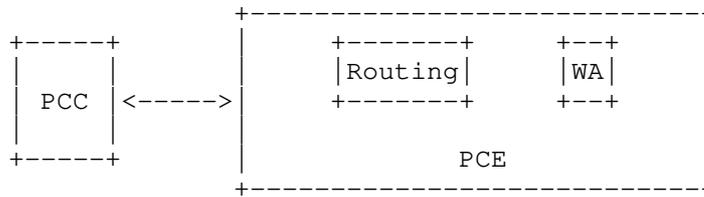
                 Figure 2 Combined Process (R&WA) architecture

4.1. Wavelength Assignment (WA) Object

   Wavelength allocation can be performed by the PCE by different
   means:

   (a) By means of Explicit Label Control [RFC3471] where the PCE
   allocates which label to use for each interface/node along the path.
   The allocated labels MAY appear after an interface route subobject.

   (b) By means of a Label Set where the PCE provides a range of
   potential labels to allocate by each node along the path.

   Option (b) allows distributed label allocation (performed during
   signaling) to complete wavelength assignment.

   Additionally, given a range of potential labels to allocate, a PC
   Request SHOULD convey the heuristic / mechanism used for the
   allocation.

   The format of a PCReq message per [RFC5440] after incorporating the
   Wavelength Assignment (WA) object is as follows:

   <PCReq Message> ::= <Common Header>

                      [<svec-list>]

                      <request-list>

      Where:

          <request-list>::=<request>[<request-list>]

          <request>::= <RP>

                    <END-POINTS>

                    <WA>

                    [other optional objects...]

   If the WA object is present in the request, it MUST be encoded after
   the END-POINTS object as defined in [PCEP-GMPLS]. The WA Object is
   mandatory in this document. Orderings for the other optional objects
   are irrelevant.

   WA Object-Class is (TBD1) (To be assigned by IANA).

   WA Object-Type is 1.

   The format of the WA object body is 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Reserved         |            Flags            |M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                           TLVs                             //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
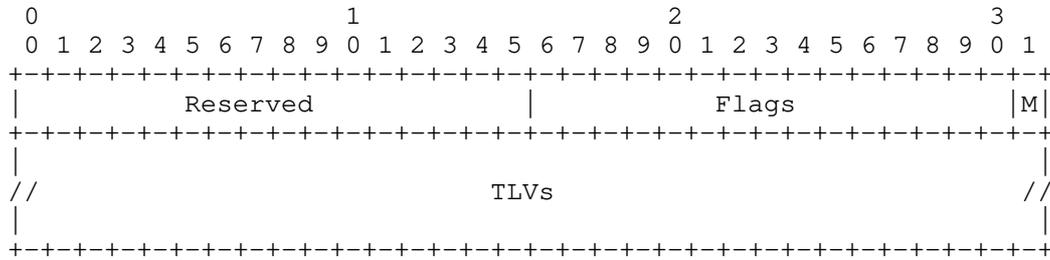
                        Figure 3 WA Object

   o Reserved (16 bits): Reserved for future use and SHOULD be zeroed
      and ignored on receipt.

   o Flags (16 bits)

   One flag bit is allocated as follows:

- M (Mode - 1 bit): M bit is used to indicate the mode of
  wavelength assignment. When M bit is set to 1, this indicates
  that the label assigned by the PCE must be explicit. That is,
  the selected way to convey the allocated wavelength is by means
  of Explicit Label Control for each hop of a computed LSP.
  Otherwise (M bit is set to 0), the label assigned by the PCE
  need not be explicit (i.e., it can be suggested in the form of
  label set objects in the corresponding response, to allow
  distributed WA. If M is 0, the PCE MUST return a Label Set
  Field as described in Section 2.6 of [RFC7579] in the response.
  See Section 5 of this document for the encoding discussion of a
  Label Set Field in a PCRep message.

All unused flags SHOULD be zeroed. IANA is to create a new
registry to manage the Flag field of the WA object.

o TLVs (variable). In the TLVs field, the following two TLVs are
   defined. At least one TLV MUST be present.

   - Wavelength Selection TLV: A TLV of type (TBD2) with fixed
     length of 32 bits indicating the wavelength selection. See
     Section 4.2 for details.

   - Wavelength Restriction Constraint TLV: A TLV of type (TBD3)
     with variable length indicating wavelength restrictions. See
     Section 4.3 for details.

4.2. Wavelength Selection TLV

The Wavelength Selection TLV is used to indicate the wavelength
selection constraint in regard to the order of wavelength assignment
to be returned by the PCE. This TLV is only applied when M bit is
set in the WA Object specified in Section 4.1. This TLV MUST NOT be
used when the M bit is cleared.

The encoding of this TLV is specified as the Wavelength Selection
Sub-TLV in Section 4.2.2 of [RFC7689]. IANA is to allocate a new TLV
type, Wavelength Selection TLV type (TBD2).

4.3. Wavelength Restriction Constraint TLV

For any request that contains a wavelength assignment, the requester
(PCC) MUST specify a restriction on the wavelengths to be used. This
restriction is to be interpreted by the PCE as a constraint on the
tuning ability of the origination laser transmitter or on any other

maintenance related constraints. Note that if the LSP LSC spans
different segments, the PCE must have mechanisms to know the
tunability restrictions of the involved wavelength converters /
regenerators, e.g. by means of the Traffic Engineering Database
(TED) either via IGP or Network Management System (NMS). Even if the
PCE knows the tunability of the transmitter, the PCC must be able to
apply additional constraints to the request.

The format of the Wavelength Restriction Constraint TLV is as
follows:

<Wavelength Restriction Constraint> ::=

        (<Action> <Count> <Reserved>

        <Link Identifiers> <Wavelength Restriction>)...

Where

<Link Identifiers> ::= <Link Identifier> [<Link Identifiers>]

See Section 4.3.1. for the encoding of the Link Identifiers Field.

These fields (i.e., <Action>, <Link Identifiers> and <Wavelength
Restriction>, etc.) MAY appear together more than once to be able to
specify multiple actions and their restrictions.

IANA is to allocate a new TLV type, Wavelength Restriction
Constraint TLV type (TBD3).

The TLV data is defined 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Action        | Count         |          Reserved             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Link Identifiers Field                       |
//                       . . .                                 //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Wavelength Restriction Field                  |
//                       . . . .                               //
```

```
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    ~                            . . . .                           ~
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | Action        |    Count      |          Reserved            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                   Link Identifiers Field                     |
    //                          . . .                            //
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |               Wavelength Restriction Field                   |
    //                          . . . .                          //
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

           Figure 4 Wavelength Restriction Constraint TLV Encoding

   o Action (8 bits):

       o 0 - Inclusive List indicates that one or more link
         identifiers are included in the Link Set. Each identifies a
         separate link that is part of the set.

       o 1 - Inclusive Range indicates that the Link Set defines a
         range of links. It contains two link identifiers. The first
         identifier indicates the start of the range (inclusive). The
         second identifier indicates the end of the range
         (inclusive). All links with numeric values between the
         bounds are considered to be part of the set. A value of zero
         in either position indicates that there is no bound on the
         corresponding portion of the range.

       o 2-255 - For future use

   IANA is to create a new registry to manage the Action values of the
   Wavelength Restriction Constraint TLV.

   If PCE receives an unrecognized Action value, the PCE MUST send a
   PCErr message with a PCEP-ERROR Object (Error-Type=TBD8) and an
   Error-value (Error-value=3). See Section 5.2 for details.

Note that "links" are assumed to be bidirectional.

o Count (8 bits): The number of the link identifiers

Note that a PCC MAY add a Wavelength restriction that applies to all
links by setting the Count field to zero and specifying just a set
of wavelengths.

Note that all link identifiers in the same list MUST be of the same
type.

> o Reserved (16 bits): Reserved for future use and SHOULD be
>   zeroed and ignored on receipt.
>
> o Link Identifiers: Identifies each link ID for which
>   restriction is applied. The length is dependent on the link
>   format and the Count field. See Section 4.3.1. for Link
>   Identifier encoding.
>
> o Wavelength Restriction: See Section 4.3.2. for the Wavelength
>   Restriction Field encoding.

Various encoding errors are possible with this TLV (e.g., not
exactly two link identifiers with the range case, unknown identifier
types, no matching link for a given identifier, etc.). To indicate
errors associated with this encoding, a PCEP speaker MUST send a
PCErr message with Error-Type=TBD8 and Error-value=3. See Section
5.1 for the details.

4.3.1. Link Identifier Field

The link identifier field can be an IPv4 [RFC3630], IPv6 [RFC5329]
or unnumbered interface ID [RFC4203].

<Link Identifier> ::=

           <IPv4 Address> | <IPv6 Address> | <Unnumbered IF ID>

The encoding of each case is as follows:

IPv4 Address Field

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type = 1     |       Reserved  (24 bits)                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv4 address (4 bytes)                                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IPv6 Address Field
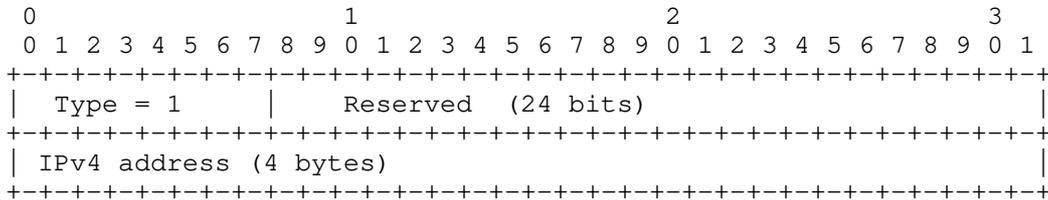
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type = 2     |      Reserved  (24 bits)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address (16 bytes)                                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address (continued)                                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address (continued)                                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address (continued)                                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
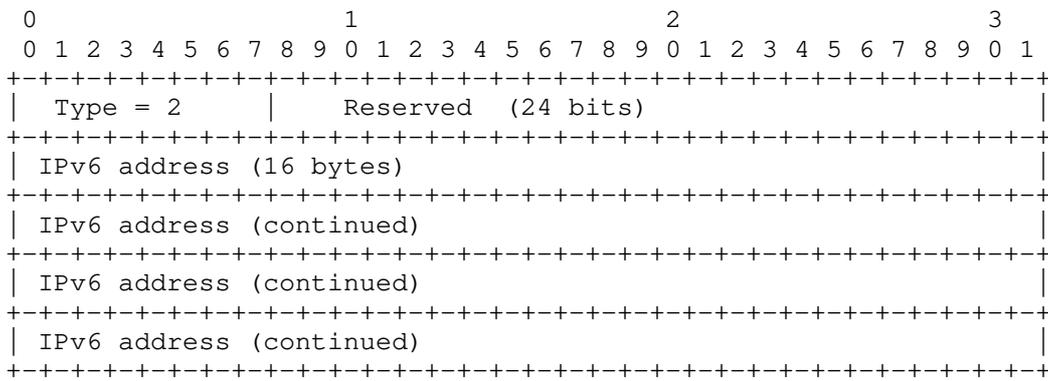
Unnumbered Interface ID Address Field

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type = 3     |     Reserved (24 bits)                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     TE Node ID (32 bits)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Interface ID (32 bits)                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
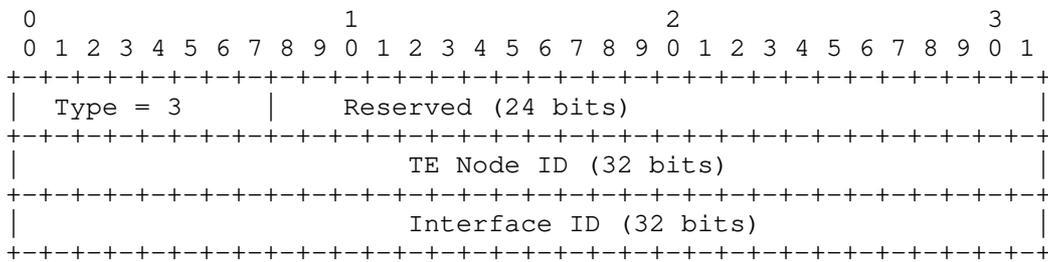
   o Type (8 bits): It indicates the type of the link identifier.

> o Reserved (24 bits): Reserved for future use and SHOULD be
>   zeroed and ignored on receipt.
>
> o Link Identifier: When Type field is 1, 4-bytes IPv4 address
>   is encoded; when Type field is 2, 16-bytes IPv6 address is
>   encoded; when Type field is 3, a tuple of 4-bytes TE node
>   ID and 4-bytes interface ID is encoded.

The Type field is extensible and matches to the IANA registry
created for Link Management Protocol (LMP) [RFC4204] for "TE Link
Object Class Type name space": https://www.iana.org/assignments/lmp-
parameters/lmp-parameters.xhtml#lmp-parameters-15. See Section 8.14
for the request to update the introductory text of the
aforementioned registry to note that the values have additional
usage for the Link Identifier Type field.

4.3.2. Wavelength Restriction Field

The Wavelength Restriction Field of the Wavelength Restriction
Constraint TLV is encoded as a Label Set field as specified in
Section 2.6 in [RFC7579] with base label encoded as a 32 bit LSC
label, defined in [RFC6205]. The Label Set format is repeated here
for convenience, with the base label internal structure included.
See [RFC6205] for a description of Grid, C.S, Identifier and n, as
well as [RFC7579] for the details of each action.

```
 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

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Action|   Num Labels       |            Length               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Grid | C.S |    Identifier   |               n                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Additional fields as necessary per action               |
|                                                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
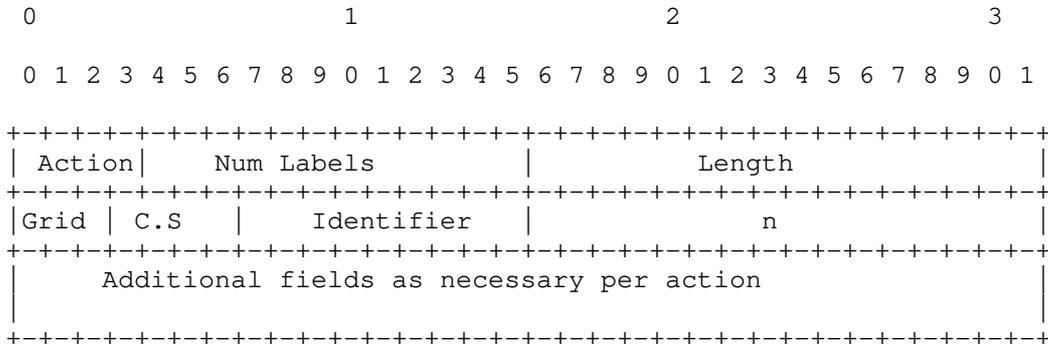
Action (4 bits):

        0 - Inclusive List

        1 - Exclusive List

        2 - Inclusive Range

        3 - Exclusive Range

        4 - Bitmap Set

   Num Labels (12 bits): It is generally the number of labels. It has a
   specific meaning depending on the action value.

   Length (16 bits): It is the length in bytes of the entire Wavelength
   Restriction field.

   Identifier (9 bits): The Identifier is always set to 0. If PCC
   receives the value of the identifier other than 0, it will ignore.

   See Sections 2.6.1 - 2.6.3 of [RFC7579] for details on additional
   field discussion for each action.

4.4. Signal Processing Capability Restrictions

   Path computation for WSON includes checking of signal processing
   capabilities at each interface against requested capability; the PCE
   MUST have mechanisms to know the signal processing capabilities at
   each interface, e.g. by means of the Traffic Engineering Database
   (TED) either via IGP or Network Management System (NMS). Moreover,
   a PCC should be able to indicate additional restrictions to signal
   processing compatibility, either on the endpoint or any given link.

   The supported signal processing capabilities considered in the RWA
   Information Model [RFC7446] are:

      o Optical Interface Class List

      o Bit Rate

      o Client Signal

   The Bit Rate restriction is already expressed in [PCEP-GMPLS] in the
   BANDWIDTH object.

In order to support the Optical Interface Class information and the
Client Signal information new TLVs are introduced as endpoint-
restriction in the END-POINTS type Generalized endpoint:

   o Client Signal TLV

   o Optical Interface Class List TLV

The END-POINTS type generalized endpoint is extended as follows:

<endpoint-restriction> ::=
                      <LABEL-REQUEST> <label-restriction-list>

<label-restriction-list> ::= <label-restriction>
                             [<label-restriction-list>]

<label-restriction> ::= (<LABEL-SET>|
                         [<Wavelength Restriction Constraint>]
                         [<signal-compatibility-restriction>])
Where

<signal-compatibility-restriction> ::=
    [<Optical Interface Class List>] [<Client Signal>]

The Wavelength Restriction Constraint TLV is defined in Section 4.3.

A new TLV for the Optical Interface Class List TLV (TBD5) is
defined, and the encoding of the value part of the Optical Interface
Class List TLV is described in Section 4.1 of [RFC7581].

A new TLV for the Client Signal Information TLV (TBD6) is defined,
and the encoding of the value part of the Client Signal Information
TLV is described in Section 4.2 of [RFC7581].

4.4.1. Signal Processing Exclusion

The PCC/PCE should be able to exclude particular types of signal
processing along the path in order to handle client restriction or
multi-domain path computation. [RFC5440] defines how Exclude Route
Object (XRO) subobject is used. In this draft, we add two new XRO
Signal Processing Exclusion Subobjects.

The first XRO subobject type (TBD9) is the Optical Interface Class List Field defined 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|  Type=TBD9  |    Length     |   Reserved    |  Attribute    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//           Optical Interface Class List                     //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
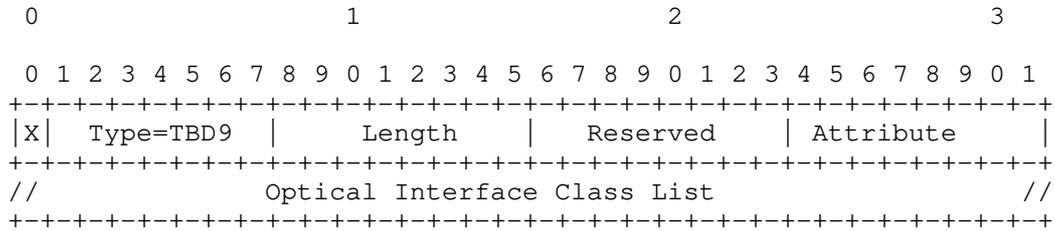
Figure 5 Optical Interface Class List XRO Subobject

Refer to [RFC5521] for the definition of X, Length and Attribute.

Type (7 bits): The Type of the Signaling Processing Exclusion Field. The TLV Type value (TBD9) is to be assigned by the IANA for the Optical Interface Class List XRO Subobject Type.

Reserved bits (8 bits) are for future use and SHOULD be zeroed and ignored on receipt.

The Attribute field (8 bits): [RFC5521] defines several Attribute values; the only permitted Attribute values for this field are 0 (Interface) or 1 (Node).

The Optical Interface Class List is encoded as described in Section 4.1 of [RFC7581].

The second XRO subobject type (TBD10) is the Client Signal Information defined 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|  Type=TBD10 |    Length     |   Reserved    |  Attribute    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//             Client Signal Information                       //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
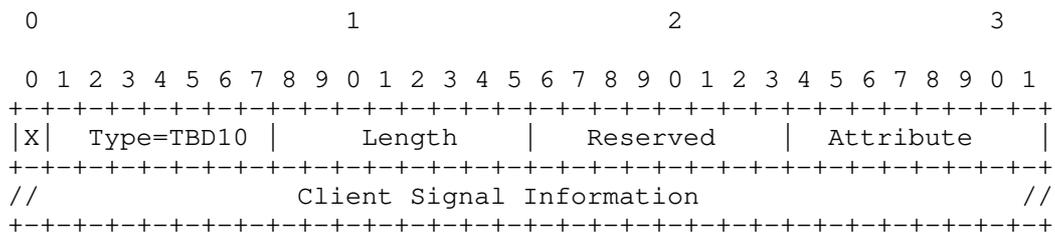
Figure 6 Client Signal Information XRO Subobject

Refer to [RFC5521] for the definition of X, Length and Attribute.

Type (7 bits): The Type of the Signaling Processing Exclusion Field. The TLV Type value (TBD10) is to be assigned by the IANA for the Client Signal Information XRO Subobject Type.

Reserved bits (8 bits) are for future use and SHOULD be zeroed and ignored on receipt.

The Attribute field (8 bits): [RFC5521] defines several Attribute values; the only permitted Attribute values for this field are 0 (Interface) or 1 (Node).

The Client Signal Information is encoded as described in Section 4.2 of [RFC7581].

The XRO needs to support the new Signaling Processing Exclusion XRO Subobject types:

        Type   XRO Subobject Type

        TBD9   Optical Interface Class List

        TBD10   Client Signal Information

4.4.2. Signal Processing Inclusion

Similar to the XRO subobject, the PCC/PCE should be able to include particular types of signal processing along the path in order to handle client restriction or multi-domain path computation. [RFC5440] defines how Include Route Object (IRO) subobject is used. In this draft, we add two new Signal Processing Inclusion Subobjects.

The IRO needs to support the new IRO Subobject types (TBD11 and TBD12) for the PCEP IRO object [RFC5440]:

        Type   IRO Subobject Type

         TBD11   Optical Interface Class List

         TBD12   Client Signal Information

   The encoding of the Signal Processing Inclusion subobjects is
   similar to Section 4.4.1 where the 'X' field is replaced with 'L'
   field, all the other fields remains the same. The 'L' field is
   described in [RFC3209].

5. Encoding of a RWA Path Reply

   This section provides the encoding of a RWA Path Reply for
   wavelength allocation request as discussed in Section 4.

5.1. Wavelength Allocation TLV

   Recall that wavelength allocation can be performed by the PCE by
   different means:

   (a)   By means of Explicit Label Control (ELC) where the PCE
         allocates which label to use for each interface/node along the
         path.
   (b)   By means of a Label Set where the PCE provides a range of
         potential labels to allocate by each node along the path.

   Option (b) allows distributed label allocation (performed during
   signaling) to complete wavelength allocation.

   The Wavelength Allocation TLV type is TBD4 (See Section 8.4). Note
   that this TLV is used for both (a) and (b). The TLV data is defined
   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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |          Reserved           |         Flag               |M|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Link Identifier Field                     |
   //                        . . .                               //
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    Allocated Wavelength(s)                   |
   //                        . . . .                             //
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
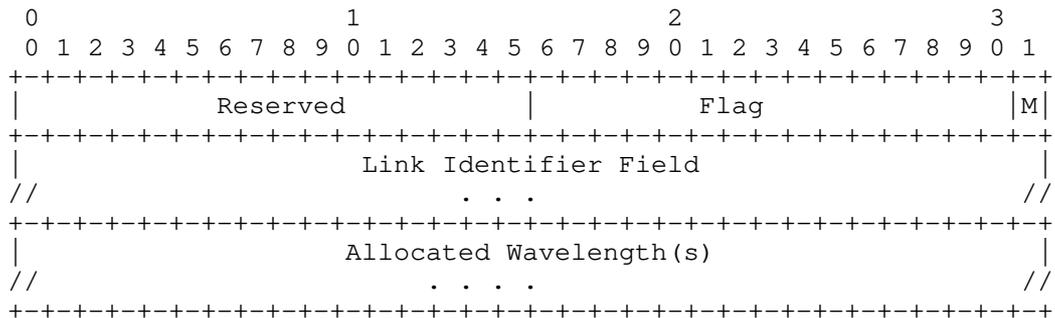
              Figure 7 Wavelength Allocation TLV Encoding

     o Reserved (16 bits): Reserved for future use.

     o Flags (16 bits)

     One flag bit is allocated as follows:

         . M (Mode): 1 bit

         - 0 indicates the allocation is under Explicit Label Control.
         - 1 indicates the allocation is expressed in Label Sets.

     IANA is to create a new registry to manage the Flag field (TBD14) of
     the Wavelength Allocation TLV.

     Note that all link identifiers in the same list must be of the same
     type.

         o Link Identifier: Identifies the interface to which
           assignment wavelength(s) is applied. See Section 4.3.1. for
           Link Identifier encoding.

         o Allocated Wavelength(s): Indicates the allocated
           wavelength(s) to be associated with the Link Identifier. See
           Section 4.3.2 for encoding details.

     This TLV is carried in a PCRep message as an attribute TLV [RFC5420]
     in the Hop Attribute Subobjects [RFC7570] in the ERO [RFC5440].

5.2. Error Indicator

     To indicate errors associated with the RWA request, a new Error Type
     (TBD8) and subsequent error-values are defined as follows for
     inclusion in the PCEP-ERROR Object:

     A new Error-Type (TBD8) and subsequent error-values are defined as
     follows:

o Error-Type=TBD8; Error-value=1: if a PCE receives a RWA request
   and the PCE is not capable of processing the request due to
   insufficient memory, the PCE MUST send a PCErr message with a
   PCEP-ERROR Object (Error-Type=TBD8) and an Error-value (Error-
   value=1). The PCE stops processing the request. The
   corresponding RWA request MUST be cancelled at the PCC.

o Error-Type=TBD8; Error-value=2: if a PCE receives a RWA request
   and the PCE is not capable of RWA computation, the PCE MUST
   send a PCErr message with a PCEP-ERROR Object (Error-Type=TBD8)
   and an Error-value (Error-value=2). The PCE stops processing
   the request. The corresponding RWA computation MUST be
   cancelled at the PCC.

o Error-Type=TBD8; Error-value=3: if a PCE receives a RWA request
   and there are syntactical encoding errors (e.g., not exactly
   two link identifiers with the range case, unknown identifier
   types, no matching link for a given identifier, unknown Action
   value, etc.), the PCE MUST send a PCErr message with a PCEP-
   ERROR Object (Error-Type=TBD8) and an Error-value (Error-
   value=3).

5.3. NO-PATH Indicator

   To communicate the reason(s) for not being able to find RWA for the
   path request, the NO-PATH object can be used in the corresponding
   response. The format of the NO-PATH object body is defined in
   [RFC5440]. The object may contain a NO-PATH-VECTOR TLV to provide
   additional information about why a path computation has failed.

   One new bit flag is defined to be carried in the Flags field in the
   NO-PATH-VECTOR TLV carried in the NO-PATH Object.

      o Bit TBD7: When set, the PCE indicates no feasible route was
         found that meets all the constraints (e.g., wavelength
         restriction, signal compatibility, etc.) associated with RWA.

6. Manageability Considerations

   Manageability of WSON Routing and Wavelength Assignment (RWA) with
   PCE must address the following considerations:

6.1. Control of Function and Policy

   In addition to the parameters already listed in Section 8.1 of
   [RFC5440], a PCEP implementation SHOULD allow configuration of the
   following PCEP session parameters on a PCC:

        o The ability to send a WSON RWA request.

   In addition to the parameters already listed in Section 8.1 of
   [RFC5440], a PCEP implementation SHOULD allow configuration of the
   following PCEP session parameters on a PCE:

        o The support for WSON RWA.

        o A set of WSON RWA specific policies (authorized sender,
           request rate limiter, etc).

   These parameters may be configured as default parameters for any
   PCEP session the PCEP speaker participates in, or may apply to a
   specific session with a given PCEP peer or a specific group of
   sessions with a specific group of PCEP peers.

6.2. Liveness Detection and Monitoring

   Mechanisms defined in this document do not imply any new liveness
   detection and monitoring requirements in addition to those already
   listed in section 8.3 of [RFC5440].

6.3. Verifying Correct Operation

   Mechanisms defined in this document do not imply any new
   verification requirements in addition to those already listed in
   section 8.4 of [RFC5440]

6.4. Requirements on Other Protocols and Functional Components

   The PCEP Link-State mechanism [PCEP-LS] may be used to advertise
   WSON RWA path computation capabilities to PCCs.

6.5. Impact on Network Operation

   Mechanisms defined in this document do not imply any new network
   operation requirements in addition to those already listed in
   section 8.6 of [RFC5440].

7. Security Considerations

   The security considerations discussed in [RFC5440] are relevant for
   this document, this document does not introduce any new security
   issues. If an operator wishes to keep private the information
   distributed by WSON, PCEPS [RFC8253] SHOULD be used.

8. IANA Considerations

   IANA maintains a registry of PCEP parameters. IANA has made
   allocations from the sub-registries as described in the following
   sections.

8.1. New PCEP Object: Wavelength Assignment Object

   As described in Section 4.1, a new PCEP Object is defined to carry
   wavelength assignment related constraints. IANA is to allocate the
   following from "PCEP Objects" sub-registry
   (http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-objects):

Object Class Value	Name	Object Type	Reference
TBD1	WA	1: Wavelength Assignment	[This.I-D]

8.2. WA Object Flag Field

   As described in Section 4.1, IANA is to create a registry to manage
   the Flag field of the WA object. New values are to be assigned by
   Standards Action [RFC8126]. Each bit should be tracked with the
   following qualities:

o Bit number (counting from bit 0 as the most significant bit)

o Capability description

o Defining RFC

The following values are defined in this document:

One bit is defined for the WA Object flag in this document:

Codespace of the Flag field (WA Object)

```
Bit      Description                    Reference
-------------------------------------------------
0-14     Unassigned                     [This.I-D]

15       Explicit Label Control         [This.I-D]
```

8.3. New PCEP TLV: Wavelength Selection TLV

As described in Sections 4.2, a new PCEP TLV is defined to indicate
wavelength selection constraints. IANA is to allocate this new TLV
from the "PCEP TLV Type Indicators" subregistry
(http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-tlv-type-
indicators).

```
Value            Description               Reference
----------------------------------------------------------
TBD2             Wavelength Selection      [This.I-D]
```

8.4. New PCEP TLV: Wavelength Restriction Constraint TLV

As described in Sections 4.3, a new PCEP TLV is defined to indicate
wavelength restriction constraints. IANA is to allocate this new TLV
from the "PCEP TLV Type Indicators" subregistry
(http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-tlv-type-
indicators).

```
Value            Description               Reference
----------------------------------------------------------
TBD3             Wavelength Restriction    [This.I-D]
```

Constraint

8.5. Wavelength Restriction Constraint TLV Action Values

   As described in Section 4.3, IANA is to allocate a new registry to
   manage the Action values of the Action field in the Wavelength
   Restriction Constraint TLV. New values are assigned by Standards
   Action [RFC8126]. Each value should be tracked with the following
   qualities: value, meaning, and defining RFC. The following values
   are defined in this document:

   Value   Meaning   Reference

   ----

   0   Inclusive List   [This.I-D]

   1   Inclusive Range   [This.I-D]

   2-255   Reserved   [This.I-D]

8.6. New PCEP TLV: Wavelength Allocation TLV

   As described in Section 5.1, a new PCEP TLV is defined to indicate
   the allocation of wavelength(s) by the PCE in response to a request
   by the PCC. IANA is to allocate this new TLV from the "PCEP TLV Type
   Indicators" subregistry
   (http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-tlv-type-
   indicators).

   Value   Description   Reference

   TBD4   Wavelength Allocation   [This.I-D]

8.7. Wavelength Allocation TLV Flag Field

   As described in Section 5.1, IANA is to allocate a registry to
   manage the Flag field of the Wavelength Allocation TLV. New values
   are to be assigned by Standards Action [RFC8126]. Each bit should
   be tracked with the following qualities:

     o Bit number (counting from bit 0 as the most significant bit)

   o Capability description

   o Defining RFC

   One bit is defined for the Wavelength Allocation flag in this -
   document:

   Codespace of the Flag field (Wavelength Allocation TLV)

   Bit   Description   Reference

   0-14   Unassigned   [This.I-D]

   15   Wavelength Allocation Mode   [This.I-D]

8.8. New PCEP TLV: Optical Interface Class List TLV

   As described in Section 4.4, a new PCEP TLV is defined to indicate
   the optical interface class list. IANA is to allocate this new TLV
   from the "PCEP TLV Type Indicators" subregistry
   (http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-tlv-type-
   indicators).

   Value   Description   Reference

   TBD5   Optical Interface   [This.I-D]
                     Class List

8.9. New PCEP TLV: Client Signal TLV

   As described in Section 4.4, a new PCEP TLV is defined to indicate
   the client signal information. IANA is to allocate this new TLV from
   the "PCEP TLV Type Indicators" subregistry
   (http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-tlv-type-
   indicators).

   Value   Description   Reference

   TBD6   Client Signal Information [This.I-D]

8.10. New No-Path Reasons

   As described in Section 5.3, a new bit flag are defined to be
   carried in the Flags field in the NO-PATH-VECTOR TLV carried in the
   NO-PATH Object. This flag, when set, indicates that no feasible
   route was found that meets all the RWA constraints (e.g., wavelength
   restriction, signal compatibility, etc.) associated with a RWA path
   computation request.

   IANA is to allocate this new bit flag from the "PCEP NO-PATH-VECTOR
   TLV Flag Field" subregistry
   (http://www.iana.org/assignments/pcep/pcep.xhtml#no-path-vector-
   tlv).

   Bit   Description   Reference

   TBD7   No RWA constraints met   [This.I-D]

8.11. New Error-Types and Error-Values

   As described in Section 5.2, new PCEP error codes are defined for
   WSON RWA errors. IANA is to allocate from the ""PCEP-ERROR Object
   Error Types and Values" sub-registry
   (http://www.iana.org/assignments/pcep/pcep.xhtml#pcep-error-object).

   Error-   Meaning   Error-Value   Reference
   Type

   TBD8   WSON RWA Error   0: Unassigned   [This.I-D]

                                 1: Insufficient   [This.I-D]
                                    Memory

                                 2: RWA computation [This.I-D]
                                    Not supported

                              3: Syntactical   [This.I-D]
                                 Encoding error

                              4-255: Unassigned   [This.I-D]

8.12. New Subobjects for the Exclude Route Object

    As described in Section 4.4.1, the "PCEP Parameters" registry
    contains a subregistry "PCEP Objects" with an entry for the Exclude
    Route Object (XRO). IANA is requested to add further subobjects that
    can be carried in the XRO as follows:

        Subobject   Type   Reference

        TBD9   Optical Interface Class List   [This.I-D]

        TBD10   Client Signal Information   [This.I-D]

8.13. New Subobjects for the Include Route Object

    As described in Section 4.4.2, the "PCEP Parameters" registry
    contains a subregistry "PCEP Objects" with an entry for the Include
    Route Object (IRO). IANA is requested to add further subobjects that
    can be carried in the IRO as follows:

        Subobject   Type   Reference

        TBD11   Optical Interface Class List   [This.I-D]

        TBD12   Client Signal Information   [This.I-D]

8.14. Request for Updated Note for LMP TE Link Object Class Type

    As discussed in Section 4.3.1, the registry created for Link
    Management Protocol (LMP) [RFC4204] for "TE Link Object Class Type
    name space": https://www.iana.org/assignments/lmp-parameters/lmp-
    parameters.xhtml#lmp-parameters-15 is requested for the updated

introductory note that the values have additional usage for the Link
Identifier Type field.

9. Acknowledgments

The authors would like to thank Adrian Farrel, Julien Meuric, Dhruv
Dhody and Benjamin Kaduk for many helpful comments that greatly
improved the contents of this draft.

10. References

10.1. Normative References

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

   [RFC3209] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, G.
             Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels",
             RFC 3209, December 2001.

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

   [RFC5329] A. Lindem, Ed., "Traffic Engineering Extensions to OSPF
             Version 3", RFC 5329, September 2008.

   [RFC5440] JP. Vasseur, Ed., JL. Le Roux, Ed., "Path Computation
             Element (PCE) Communication Protocol (PCEP)", RFC 5440,
             March 2009.

   [RFC6205] Tomohiro, O. and D. Li, "Generalized Labels for Lambda-
             Switching Capable Label Switching Routers", RFC 6205,
             January, 2011.

   [RFC7570] C. Margaria, et al., "Label Switched Path (LSP) Attribute
             in the Explicit Route Object (ERO)", RFC 7570, July 2015.

   [RFC7579] G. Bernstein and Y. Lee, "General Network Element
             Constraint Encoding for GMPLS Controlled Networks", RFC
             7579, June 2015.

   [RFC7581] G. Bernstein and Y. Lee, "Routing and Wavelength
             Assignment Information Encoding for Wavelength Switched
             Optical Networks", RFC7581, June 2015.

   [RFC7689] Bernstein et al., "Signaling Extensions for Wavelength
             Switched Optical Networks", RFC 7689, November 2015.

   [RFC7688] Y. Lee, and G. Bernstein, "OSPF Enhancement for Signal and
             Network Element Compatibility for Wavelength Switched
             Optical Networks", RFC 7688, November 2015.

   [RFC8174] B. Leiba, "Ambiguity of Uppercase vs Lowercase in RFC 2119
             Key Words", RFC 8174, May 2017.

   [RFC8253] D. Lopez, O. Gonzalez de Dios, Q. Wu, D. Dhody, "PCEPS:
             Usage of TLS to Provide a Secure Transport for the Path
             Computation Element Communication Protocol (PCEP)", RFC
             8253, October 2017.

   [PCEP-GMPLS] C. Margaria, et al., "PCEP extensions for GMPLS",
             draft-ietf-pce-gmpls-pcep-extensions, work in progress.

10.2. Informative References

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

   [RFC4203] K. Kompella, Ed., Y. Rekhter, Ed., "OSPF Extensions in
             Support of Generalized Multi-Protocol Label Switching
             (GMPLS)", RFC 4203, October 2005.

   [RFC4204] J. Lang, Ed., "Link Management Protocol (LMP)", RFC 4204,
             October 2005.

   [RFC4655] A. Farrel, JP. Vasseur, G. Ash, "A Path Computation
             Element (PCE)-Based Architecture", RFC 4655, August 2006.

   [RFC5420] Farrel, A. "Encoding of Attributes for MPLS LSP
             Establishment Using Resource Reservation Protocol Traffic
             Engineering (RSVP-TE)", RFC5420, February 2009.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
             Element (PCE) communication Protocol", RFC 5440, March
             2009.[RFC5521] Oki, E, T. Takeda, and A. Farrel,
             "Extensions to the Path Computation Element Communication
             Protocol (PCEP) for Route Exclusions", RFC 5521, April
             2009.

   [RFC6163] Lee, Y. and Bernstein, G. (Editors), and W. Imajuku,
             "Framework for GMPLS and PCE Control of Wavelength
             Switched Optical Networks", RFC 6163, March 2011.

   [RFC6566] Lee, Y. and Berstein, G. (Editors), "A Framework for the
             Control of Wavelength Switched Optical Networks (WSONs)
             with Impairments", RFC 6566, March 2012.

   [RFC7446] Y. Lee, G. Bernstein, (Editors), "Routing and Wavelength
             Assignment Information Model for Wavelength Switched
             Optical Networks", RFC 7446, February 2015.

   [RFC7449] Y. Lee, G. Bernstein, (Editors), "Path Computation Element
             Communication Protocol (PCEP) Requirements for Wavelength
             Switched Optical Network (WSON) Routing and Wavelength
             Assignment", RFC 7449, February 2015.

   [PCEP-LS] Y. Lee, et al., "PCEP Extension for Distribution of Link-
             State and TE information for Optical Networks", draft-lee-
             pce-pcep-ls-optical, work in progress.

   [RFC8126] M. Cotton, B. Leiba, T,.Narten, "Guidelines for Writing an
             IANA Considerations Section in RFCs", RFC 8126, June 2017.

11. Contributors

   Fatai Zhang
   Huawei Technologies
   Email: zhangfatai@huawei.com

   Cyril Margaria
   Nokia Siemens Networks
   St Martin Strasse 76
   Munich,   81541
   Germany
   Phone: +49 89 5159 16934
   Email: cyril.margaria@nsn.com

   Oscar Gonzalez de Dios
   Telefonica Investigacion y Desarrollo
   C/ Emilio Vargas 6
   Madrid,   28043
   Spain
   Phone: +34 91 3374013
   Email: ogondio@tid.es

   Greg Bernstein
   Grotto Networking
   Fremont, CA, USA
   Phone: (510) 573-2237
   Email: gregb@grotto-networking.com

Authors' Addresses

   Young Lee, Editor
   Huawei Technologies
   5700 Tennyson Parkway Suite 600
   Plano, TX 75024, USA
   Email: leeyoung@huawei.com

   Ramon Casellas, Editor
   CTTC PMT Ed B4 Av. Carl Friedrich Gauss 7
   08860 Castelldefels (Barcelona)
   Spain
   Phone: (34) 936452916
   Email: ramon.casellas@cttc.es

PCE Working Group   I. Minei
Internet-Draft   Google, Inc.
Intended status: Standards Track   E. Crabbe
Expires: January 7, 2016

                                                            S. Sivabalan
                                                      Cisco Systems, Inc.
                                                       H. Ananthakrishnan
                                                            Packet Design
                                                                X. Zhang
                                                       Huawei Technologies
                                                                Y. Tanaka
                                            NTT Communications Corporation
                                                             July 6, 2015

        PCEP Extensions for Establishing Relationships Between Sets of LSPs
                    draft-minei-pce-association-group-02

Abstract

   This document introduces a generic mechanism to create a grouping of
   LSPs in the context of a PCE. This grouping can then be used to
   define associations between sets of LSPs or between a set of LSPs and
   a set of attributes (such as configuration parameters or behaviors),
   and is equally applicable to the active and passive modes of a
   stateful PCE as well as a stateless PCE.

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 [RFC2119].

Status of This Memo

This Internet-Draft will expire on January 7, 2016.

Table of Contents

1. Introduction

   [RFC5440] describes the Path Computation Element Protocol PCEP. PCEP
   enables the communication between a Path Computation Client (PCC) and
   a Path Control Element (PCE), or between PCE and PCE, for the purpose
   of computation of Multiprotocol Label Switching (MPLS) as well as
   Generalzied MPLS (GMPLS) for Traffic Engineering Label Switched Path
   (TE LSP) characteristics.

   Stateful pce [I-D.ietf-pce-stateful-pce] specifies a set of
   extensions to PCEP to enable stateful control of TE LSPs between and

across PCEP sessions in compliance with [RFC4657] and focuses on a
model where LSPs are configured on the PCC and control over them is
delegated to the PCE. The model of operation where LSPs are
initiated from the PCE is described in
[I-D.ietf-pce-pce-initiated-lsp].

This document introduces a generic mechanism to create a grouping of
LSPs. This grouping can then be used to define associations between
sets of LSPs or between a set of LSPs and a set of attributes (such
as configuration parameters or behaviors), and is equally applicable
to the active and passive modes of a stateful PCE and a stateless
PCE.

2. Terminology

This document uses the following terms defined in [RFC5440]: PCC,
PCE, PCEP Peer.

3. Architectural Overview

3.1. Motivation

Stateful PCE provides the ability to update existing LSPs and to
instantiate new ones. To enable support for PCE-controlled make-
before-break and for protection, there is a need to define
associations between LSPs. For example, the association between the
original and the re-optimized path in the make-before break scenario,
or between the working and protection path in end-to-end protection.
Another use for LSP grouping is for applying a common set of
configuration parameters or behaviors to a set of LSPs.

For a stateless PCE, it might be useful to associate a path
computation request to an association group, thus enabling it to
associate a common set of configuration parameters or behaviors with
the request.

Rather than creating separate mechanisms for each use case, this
draft defines a generic mechanism that can be reused as needed.

3.2. Operation Overview

LSPs are associated with other LSPs with which they interact by
adding them to a common association group. Association groups as
defined in this document can be applied to LSPs originating at the
same head end or different head ends. For LSPs originating at the
same head end, the association can be initiated by either the PCC
(head end) or by a PCE. Only a stateful PCE can initiate an
association for LSPs originating at different head ends. For both

cases, the association is uniquely identified by the combination of
an association identifier and the address of the PCE peer that
created the association.

Multiple types of groups can exist, each with their own identifiers
space. The definition of the different association types and their
behaviors is outside the scope of this document. The establishment
and removal of the association relationship can be done on a per LSP
basis. An LSP may join multiple association groups, of different or
of the same type.

In the case of a stateless PCE, associations are created out of band,
and PCEP peers should be aware of the association and its
significance outside of the protocol.

4. ASSOCIATION Object

4.1. Object Definition

Creation of an association group and modifications to its membership
can be initiated by either the PCE or the PCC. Association groups
and their memberships are defined using the ASSOCIATION object for
stateful PCE.

ASSOCIATION Object-Class is to be assigned by IANA (TBD).

ASSOCIATION Object-Type is 1 for IPv4 and its format is shown in
Figure 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Type  |        Flags          |R|           Reserved          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Association ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   IPv4 Association Source                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                      Optional TLVs                          //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: The IPv4 ASSOCIATION Object format

ASSOCIATION Object-Type is 2 for IPv6 and its format is shown in
Figure 2:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type |           Flags           |R|         Reserved        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Association ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                                                               |
|                  IPv6 Association Source                      |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                     Optional TLVs                          //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
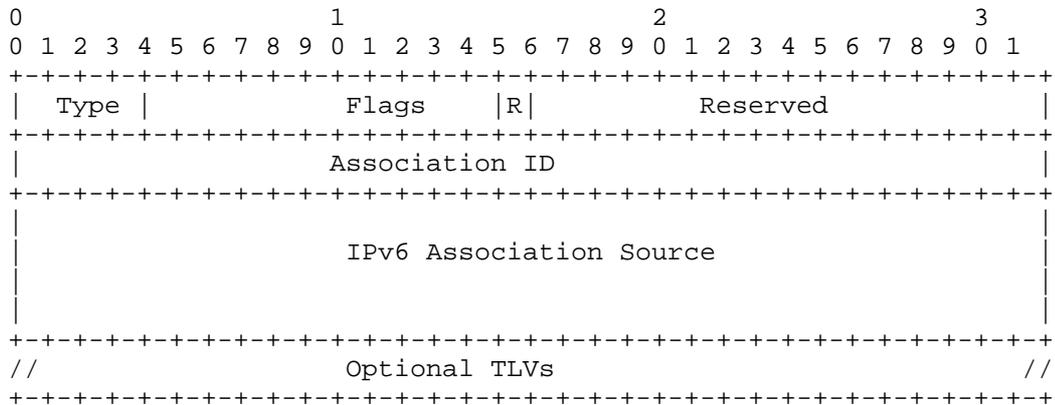
           Figure 2: The IPv6 ASSOCIATION Object format

   Type: 4 bits - the association type (for example protection). The
   association type will be defined in separate documents.

   Flags: 12 bits - The following flags are currently defined:

   R (Removal - 1 bit): when set, the requesting PCE peer requires the
      removal of an LSP from the association group.

   Reserved: MUST be set to 0 and ignored upon receipt.

   Association ID: 32 bits - the identifier of the association group.
   When combined with Type and Association Source, this value uniquely
   identifies an association group. The value 0xffffffff and 0x0 are
   reserved. The value 0xffffffff is used to indicate all association
   groups.

   Association Source: 4 or 16 bytes - An IPv4 or IPv6 address, which is
   associated to the PCE peer that originated the association.

   Optional TLVs: Variable - no TLVs are defined in this document.

4.2. Object Encoding in PCEP messages

   The ASSOCIATION Object is OPTIONAL and MAY be carried in the Path
   Computation Update (PCUpd), Path Computation Report (PCRpt) and Path
   Computation Initiate (PCinit) messages.

   When carried in PCRpt message, it is used to report the association
   group membership information pertaining to a LSP to a stateful PCE.
   It can also be used to remove an LSP from one or more association

groups by setting the R flag to 1. Unless, a PCE wants to delete an
association from an LSP, it does not need to carry the ASSOCIATION
object while updating other LSP attributes using the PCUpd message.

The PCRpt message is defined in [I-D.ietf-pce-stateful-pce] and
updated as below:


```
<PCRpt Message> ::= <Common Header>
                    <state-report-list>
```
Where:

```
   <state-report-list> ::= <state-report>[<state-report-list>]

    <state-report> ::= [<SRP>]
                       <LSP>
                       [<association-list>]
                       <path>
```
Where:

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

When an LSP is delegated to a stateful PCE, the stateful PCE can
initiate a new association group for this LSP, or associate it with
one or more existing association groups. This is done by including
the ASSOCIATION Object in a PCUpd message or in a PCInit message. A
stateful PCE can also remove a delegated LSP from one or more
association groups by setting the R flag to 1.

The PCUpd message is defined in [I-D.ietf-pce-stateful-pce] and
updated as below:

```
<PCUpd Message> ::= <Common Header>
                    <update-request-list>
```
Where:
```
   <update-request-list> ::= <update-request>[<update-request-list>]

   <update-request> ::= <SRP>
                        <LSP>
                        [<association-list>]
                        <path>
```

Where: `<association-list> ::= <ASSOCIATION> [<association-list>]`

The PCInitiate message is defined in [I-D.ietf-pce-pce-initiated-lsp]
and updated as below:

```
   <PCInitiate Message> ::= <Common Header>
                           <PCE-initiated-lsp-list>
   Where:

      <PCE-initiated-lsp-list> ::=
      <PCE-initiated-lsp-request>[<PCE-initiated-lsp-list>]

      <PCE-initiated-lsp-request>::=
      (<PCE-initiated-lsp-instantiation>|<PCE-initiated-lsp-deletion>)

      <PCE-initiated-lsp-instantiation> ::= <SRP>
                                            <LSP>
                                            <END-POINTS>
                                            <ERO>
                                            [<association-list>]
                                            [<attribute-list>]

   Where:
   <association-list> ::= <ASSOCIATION> [<association-list>]
```

   In case of passive stateful or stateless PCE, the ASSOCIATION Object
   is OPTIONAL and MAY be carried in the Path Computation Request
   (PCReq) message.

   When carried in a PCReq message, the ASSOCIATION Object is used to
   associate the path computation request to an association group, the
   association might be further informed via PCRpt message in case of
   passive stateful PCE later or it might be created out of band in case
   of stateless PCE.

   The PCReq message is defined in [RFC5440] and updated in [I-D.ietf-
   pce-stateful-pce], it is further updated below for association:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>

Where:
      <svec-list>::= <SVEC>[<svec-list>]
      <request-list>::= <request>[<request-list>]

      <request>::= <RP>
                   <END-POINTS>
                   [<LSP>]
                   [<LSPA>]
                   [<BANDWIDTH>]
                   [<metric-list>]
                   [<association-list>]
                   [<RRO>[<BANDWIDTH>]]
                   [<IRO>]
                   [<LOAD-BALANCING>]

Where:
      <association-list> ::= <ASSOCIATION> [<association-list>]
```

Note that LSP object MAY be present for the passive stateful PCE.

4.3. Processing Rules

Both a PCC and a PCE can create one or more association groups for an
LSP. But a PCE peer cannot add new members for association group
created by another peer. If a PCC receives a PCUpd or a PCInitiate
message including an ASSOCIATION Object but the sender address does
not match the association source, a PCErr message MUST be sent with
Error-Type = TBD2 (Association Error) and Error-value= 1 (association
source and sender source mismatch in PCUpd). Error handling for
situations such as PCE failures after association groups are created
and other scenarios will be included in future versions of this
draft.

If a PCE peer does not recognize the ASSOCIATION object, it MUST
return a PCErr message with Error-Type "Unknown Object" as described
in [RFC5440]. If a PCE peer is unwilling or unable to process the
ASSOCIATION object, it MUST return a PCErr message with the Error-
Type "Not supported object" and follow the relevant procedures
described in [RFC5440].

5. IANA Considerations

   The "PCEP Parameters" registry contains a subregistry "PCEP Objects".
   This document request IANA to allocate the values from this registry.

   Object-Class Value Name   Reference

         TBD   Association   This document
                       Object-Type
                       1: IPv4
                       2: IPv6

   This document requests IANA to create a subregistry of the "PCEP
   Parameters" for the bits carried in the Flags field of the
   ASSOCIATION object. The subregistry is called "ASSOCIATION Flags
   Field".

   The field contains 12 bits numbered from bit 0 as the most
   significant bit.

         Bit; Name: Description   Reference

         15   R: Removal   This document

   This document defines new Error Type and Error-Value for the
   following new error conditions:

   Error-Type Meaning   Reference

      TBD   Error-Value=1: association source and   This document
               sender source does not match

6. Security Considerations

   The security considerations described in [I-D.ietf-pce-stateful-pce]
   apply to the extensions described in this document. Additional
   considerations related to a malicious PCE are introduced, as the PCE
   may now create additional state on the PCC through the creation of
   association groups.

7. Acknowledgements

   We would like to thank Yuji Kamite and Joshua George for their
   contributions to this document. Also Thank Venugopal Reddy and Cyril
   Margaria for their useful comments.

8. Contributors

    Dhruv Dhody
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560037
    India
    Email: dhruv.ietf@gmail.com

9. References

9.1. Normative References

    [I-D.ietf-pce-pce-initiated-lsp]
              Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
              Extensions for PCE-initiated LSP Setup in a Stateful PCE
              Model", draft-ietf-pce-pce-initiated-lsp-04 (work in
              progress), April 2015.

    [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE", draft-ietf-pce-stateful-
              pce-11 (work in progress), April 2015.

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

    [RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element
               (PCE) Communication Protocol (PCEP)", RFC 5440, March
               2009.

9.2. Informative References

    [RFC4657] Ash, J. and J. Le Roux, "Path Computation Element (PCE)
               Communication Protocol Generic Requirements", RFC 4657,
               September 2006.

Authors' Addresses

    Ina Minei
    Google, Inc.
    1600 Amphitheatre Parkway
    Mountain View, CA 94043
    US

    Email: inaminei@google.com

Edward Crabbe

   Email: edward.crabbe@gmail.com

   Siva Sivabalan
   Cisco Systems, Inc.
   170 West Tasman Dr.
   San Jose, CA 95134
   US

   Email: msiva@cisco.com

   Hariharan Ananthakrishnan
   Packet Design

   Email: hari@packetdesign.com

   Xian Zhang
   Huawei Technologies
   F3-5-B R&D Center, Huawei Base Bantian, Longgang District
   Shenzhen, Guangdong 518129
   P.R.China

   Email: zhang.xian@huawei.com

   Yosuke Tanaka
   NTT Communications Corporation
   Granpark Tower 3-4-1 Shibaura, Minato-ku
   Tokyo 108-8118
   Japan

   Email: yosuke.tanaka@ntt.com

PCE Working Group   I. Minei
Internet-Draft   Google, Inc.
Intended status: Standards Track   E. Crabbe
Expires: May 12, 2016

                                                          S. Sivabalan
                                                    Cisco Systems, Inc.
                                                     H. Ananthakrishnan
                                                         Packet Design
                                                              X. Zhang
                                                    Huawei Technologies
                                                              Y. Tanaka
                                         NTT Communications Corporation
                                                      November 9, 2015

       PCEP Extensions for Establishing Relationships Between Sets of LSPs
                   draft-minei-pce-association-group-04

Abstract

   This document introduces a generic mechanism to create a grouping of
   LSPs in the context of a PCE. This grouping can then be used to
   define associations between sets of LSPs or between a set of LSPs and
   a set of attributes (such as configuration parameters or behaviors),
   and is equally applicable to the active and passive modes of a
   stateful PCE as well as a stateless PCE.

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 [RFC2119].

   This Internet-Draft will expire on May 12, 2016.

Copyright Notice

   Copyright (c) 2015 IETF Trust and the persons identified as the
   document authors. All rights reserved.

   This document is subject to BCP 78 and the IETF Trust's Legal
   Provisions Relating to IETF Documents
   (http://trustee.ietf.org/license-info) in effect on the date of
   publication of this document. Please review these documents
   carefully, as they describe your rights and restrictions with respect
   to this document. Code Components extracted from this document must
   include Simplified BSD License text as described in Section 4.e of
   the Trust Legal Provisions and are provided without warranty as
   described in the Simplified BSD License.

Table of Contents

1. Introduction

   [RFC5440] describes the Path Computation Element Protocol PCEP. PCEP
   enables the communication between a Path Computation Client (PCC) and
   a Path Control Element (PCE), or between PCE and PCE, for the purpose
   of computation of Multiprotocol Label Switching (MPLS) as well as
   Generalzied MPLS (GMPLS) for Traffic Engineering Label Switched Path
   (TE LSP) characteristics.

Stateful pce [I-D.ietf-pce-stateful-pce] specifies a set of
extensions to PCEP to enable stateful control of TE LSPs between and
across PCEP sessions in compliance with [RFC4657] and focuses on a
model where LSPs are configured on the PCC and control over them is
delegated to the PCE. The model of operation where LSPs are
initiated from the PCE is described in
[I-D.ietf-pce-pce-initiated-lsp].

This document introduces a generic mechanism to create a grouping of
LSPs. This grouping can then be used to define associations between
sets of LSPs or between a set of LSPs and a set of attributes (such
as configuration parameters or behaviors), and is equally applicable
to the active and passive modes of a stateful PCE and a stateless
PCE.

2. Terminology

   This document uses the following terms defined in [RFC5440]: PCC,
   PCE, PCEP Peer.

   The following term is defined in this document:

   Association Timeout Interval: when a PCEP session is terminated, a
   PCC waits for this time period before deleting associations created
   by the PCEP peer.

3. Architectural Overview

3.1. Motivation

   Stateful PCE provides the ability to update existing LSPs and to
   instantiate new ones. To enable support for PCE-controlled make-
   before-break and for protection, there is a need to define
   associations between LSPs. For example, the association between the
   original and the re-optimized path in the make-before break scenario,
   or between the working and protection path in end-to-end protection.
   Another use for LSP grouping is for applying a common set of
   configuration parameters or behaviors to a set of LSPs.

   For a stateless PCE, it might be useful to associate a path
   computation request to an association group, thus enabling it to
   associate a common set of configuration parameters or behaviors with
   the request.

   Rather than creating separate mechanisms for each use case, this
   draft defines a generic mechanism that can be reused as needed.

3.2. Operation Overview

   LSPs are associated with other LSPs with which they interact by
   adding them to a common association group. Association groups as
   defined in this document can be applied to LSPs originating at the
   same head end or different head ends. For LSPs originating at the
   same head end, the association can be initiated by either the PCC
   (head end) or by a PCE. Only a stateful PCE can initiate an
   association for LSPs originating at different head ends. For both
   cases, the association is uniquely identified by the combination of
   an association identifier and the address of the node that created
   the association.

   Multiple types of groups can exist, each with their own identifiers
   space. The definition of the different association types and their
   behaviors is outside the scope of this document. The establishment
   and removal of the association relationship can be done on a per LSP
   basis. An LSP may join multiple association groups, of different or
   of the same type.

   In the case of a stateless PCE, associations are created out of band,
   and PCEP peers should be aware of the association and its
   significance outside of the protocol.

   Association groups can be created by both PCC and PCE. When a PCC's
   PCEP session with a PCE terminates unexpectedly, the PCC cleans up
   associations (as per the processing rules in this document).

4. ASSOCIATION Object

4.1. Object Definition

   Creation of an association group and modifications to its membership
   can be initiated by either the PCE or the PCC. Association groups
   and their memberships are defined using the ASSOCIATION object for
   stateful PCE.

   ASSOCIATION Object-Class is to be assigned by IANA (TBD).

   ASSOCIATION Object-Type is 1 for IPv4 and its format is shown in
   Figure 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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |             Reserved          |            Flags          |R|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |        Association type       |        Association ID         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                  IPv4 Association Source                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   //                    Optional TLVs                          //
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

                Figure 1: The IPv4 ASSOCIATION Object format

   ASSOCIATION Object-Type is 2 for IPv6 and its format is shown in
   Figure 2:

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |             Reserved          |            Flags          |R|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |        Association type       |        Association ID         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                              |
   |                  IPv6 Association Source                      |
   |                                                              |
   |                                                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   //                    Optional TLVs                          //
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
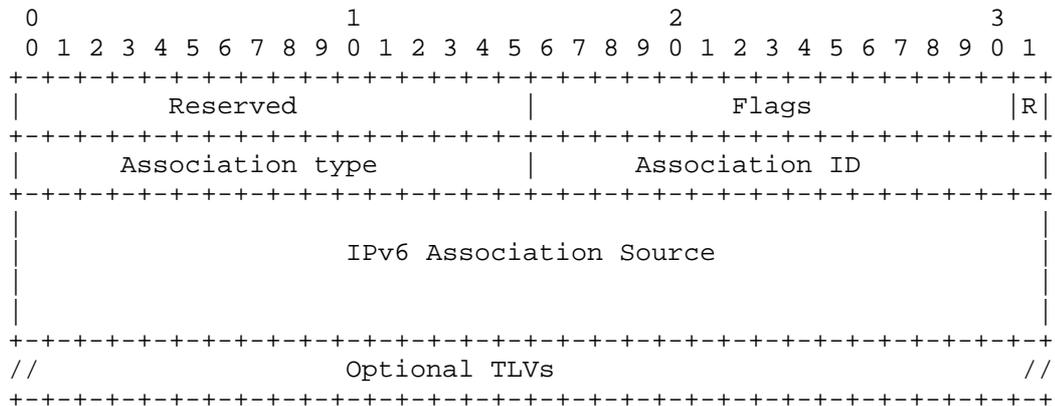
                Figure 2: The IPv6 ASSOCIATION Object format

   Reserved: 16 bits - MUST be set to 0 and ignored upon receipt.

   Flags: 16 bits - The following flags are currently defined:

   R (Removal - 1 bit): when set, the requesting PCE peer requires the
      removal of an LSP from the association group.

   Association type: 16 bits - the association type (for example
   protection). The association type will be defined in separate
   documents.

Association ID: 16 bits - the identifier of the association group.
When combined with Type and Association Source, this value uniquely
identifies an association group. The value 0xffff and 0x0 are
reserved. The value 0xffff is used to indicate all association
groups.

Association Source: 4 or 16 bytes - An IPv4 or IPv6 address, which is
associated to the node that originated the association.

Optional TLVs: The optional TLVs follow the PCEP TLV format of
[RFC5440]. This document defines two optional TLVs.

4.1.1. Global Association Source TLV

The Global Association Source TLV is an optional TLV for use in the
Association Object.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Type             |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Global Association Source                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
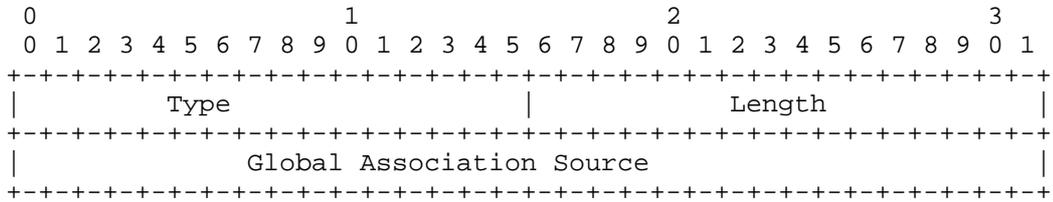
Figure 3: The Global Association Source TLV format

Type: To be allocated by IANA

Length: Fixed value of 4 bytes

Global Association Source: as defined in [RFC6780]

4.1.2. Extended Association ID TLV

The Extended Association ID TLV is an optional TLV for use in the
Association Object.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |              Type             |             Length            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   //                   Extended Association ID                   //
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
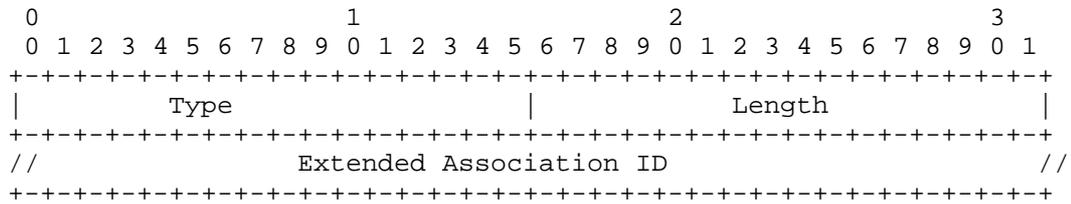
            Figure 4: The Extended Association ID TLV format

   Type: To be allocated by IANA

   Length: variable

   Extended Association ID: as defined in [RFC6780]

4.2. Object Encoding in PCEP messages

   The ASSOCIATION Object is OPTIONAL and MAY be carried in the Path
   Computation Update (PCUpd), Path Computation Report (PCRpt) and Path
   Computation Initiate (PCinit) messages.

   When carried in PCRpt message, it is used to report the association
   group membership information pertaining to a LSP to a stateful PCE.
   It can also be used to remove an LSP from one or more association
   groups by setting the R flag to 1. Unless, a PCE wants to delete an
   association from an LSP, it does not need to carry the ASSOCIATION
   object while updating other LSP attributes using the PCUpd message.

   The PCRpt message is defined in [I-D.ietf-pce-stateful-pce] and
   updated as below:

   <PCRpt Message> ::= <Common Header>
                       <state-report-list>
   Where:

      <state-report-list> ::= <state-report>[<state-report-list>]

       <state-report> ::= [<SRP>]
                          <LSP>
                          [<association-list>]
                          <path>
   Where:

   <association-list> ::= <ASSOCIATION> [<association-list>]

When an LSP is delegated to a stateful PCE, the stateful PCE can
initiate a new association group for this LSP, or associate it with
one or more existing association groups. This is done by including
the ASSOCIATION Object in a PCUpd message or in a PCInit message. A
stateful PCE can also remove a delegated LSP from one or more
association groups by setting the R flag to 1.

The PCUpd message is defined in [I-D.ietf-pce-stateful-pce] and
updated as below:

```
<PCUpd Message> ::= <Common Header>
                    <update-request-list>
Where:
   <update-request-list> ::= <update-request>[<update-request-list>]

   <update-request> ::= <SRP>
                        <LSP>
                        [<association-list>]
                        <path>

Where:  <association-list> ::= <ASSOCIATION> [<association-list>]
```

The PCInitiate message is defined in [I-D.ietf-pce-pce-initiated-lsp]
and updated as below:

```
<PCInitiate Message> ::= <Common Header>
                         <PCE-initiated-lsp-list>
Where:

   <PCE-initiated-lsp-list> ::=
   <PCE-initiated-lsp-request>[<PCE-initiated-lsp-list>]

   <PCE-initiated-lsp-request>::=
   (<PCE-initiated-lsp-instantiation>|<PCE-initiated-lsp-deletion>)

   <PCE-initiated-lsp-instantiation> ::= <SRP>
                                         <LSP>
                                         <END-POINTS>
                                         <ERO>
                                         [<association-list>]
                                         [<attribute-list>]

Where:
<association-list> ::= <ASSOCIATION> [<association-list>]
```

In case of passive stateful or stateless PCE, the ASSOCIATION Object
is OPTIONAL and MAY be carried in the Path Computation Request
(PCReq) message.

When carried in a PCReq message, the ASSOCIATION Object is used to
associate the path computation request to an association group, the
association might be further informed via PCRpt message in case of
passive stateful PCE later or it might be created out of band in case
of stateless PCE.

The PCReq message is defined in [RFC5440] and updated in [I-D.ietf-
pce-stateful-pce], it is further updated below for association:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>
```

Where:
```
      <svec-list>::= <SVEC>[<svec-list>]
      <request-list>::= <request>[<request-list>]

      <request>::= <RP>
                   <END-POINTS>
                   [<LSP>]
                   [<LSPA>]
                   [<BANDWIDTH>]
                   [<metric-list>]
                   [<association-list>]
                   [<RRO>[<BANDWIDTH>]]
                   [<IRO>]
                   [<LOAD-BALANCING>]
```

Where:
```
      <association-list> ::= <ASSOCIATION> [<association-list>]
```

Note that LSP object MAY be present for the passive stateful PCE.

4.3. Processing Rules

Both a PCC and a PCE can create one or more association groups for an
LSP. But a PCE peer cannot add new members for association group
created by another peer. If a PCE peer does not recognize the
ASSOCIATION object, it MUST return a PCErr message with Error-Type
"Unknown Object" as described in [RFC5440]. If a PCE peer is
unwilling or unable to process the ASSOCIATION object, it MUST return
a PCErr message with the Error-Type "Not supported object" and follow
the relevant procedures described in [RFC5440].

The association timeout interval is as a PCC-local value that can be
operator-configured or computed by the PCC based on local policy and
is used in the context of cleaning up associations on session
failure. The association timeout must be set to a value no larger

than the state timeout interval (defined in
[I-D.ietf-pce-stateful-pce]) and larger than the delegation timeout
interval (defined in [I-D.ietf-pce-stateful-pce].

When a PCC's PCEP session wih the PCE terminates unexpectedly, the
PCC MUST wait for the association timeout interval before cleaning up
the association. If this PCEP session can be re-established before
the association timeout interval time expires, no action is taken to
clean the association created by this PCE. During the time window of
the redelegation timeout interval and the association timeout
interval, the PCE, after re-establishing the session, can also ask
for redelegation following the procedure defined in
[I-D.ietf-pce-stateful-pce] and [I-D.ietf-pce-pce-initiated-lsp].
When the association timeout interval timers expires, the PCC clears
all the associations which are not delegated to any PCEs.

Upon LSP delegation revocation, the PCC MAY clear the association
created by the related PCE, but in order to avoid traffic loss, it
can perform this in a make-before-break fashion, which is the same as
what is defined in Stateful pce [I-D.ietf-pce-stateful-pce] for
handling LSP state cleanup.

Error handling for situations for multiple PCE scenarios will be
included in future versions of this draft.

5. IANA Considerations

The "PCEP Parameters" registry contains a subregistry "PCEP Objects".
This document request IANA to allocate the values from this registry.

Object-Class Value	Name	Reference
TBD	Association Object-Type 1: IPv4 2: IPv6	This document

This document defines the following new PCEP TLVs:

Value	Meaning	Reference
TBD	Global Association Source	This document
TBD	Extended Association Id	This document

This document requests IANA to create a subregistry of the "PCEP
Parameters" for the bits carried in the Flags field of the
ASSOCIATION object. The subregistry is called "ASSOCIATION Flags
Field".

The field contains 12 bits numbered from bit 0 as the most
significant bit.

```
        Bit; Name: Description                  Reference

        15   R: Removal                     This document
```

6. Security Considerations

   The security considerations described in [I-D.ietf-pce-stateful-pce]
   apply to the extensions described in this document. Additional
   considerations related to a malicious PCE are introduced, as the PCE
   may now create additional state on the PCC through the creation of
   association groups.

7. Acknowledgements

   We would like to thank Yuji Kamite and Joshua George for their
   contributions to this document. Also Thank Venugopal Reddy and Cyril
   Margaria for their useful comments.

8. Contributors

   Dhruv Dhody
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560037
   India
   Email: dhruv.ietf@gmail.com

9. References

9.1. Normative References

   [I-D.ietf-pce-pce-initiated-lsp]
            Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
            Extensions for PCE-initiated LSP Setup in a Stateful PCE
            Model", draft-ietf-pce-pce-initiated-lsp-05 (work in
            progress), October 2015.

   [I-D.ietf-pce-stateful-pce]
            Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
            Extensions for Stateful PCE", draft-ietf-pce-stateful-
            pce-12 (work in progress), October 2015.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <http://www.rfc-editor.org/info/rfc5440>.

   [RFC6780] Berger, L., Le Faucheur, F., and A. Narayanan, "RSVP
              ASSOCIATION Object Extensions", RFC 6780,
              DOI 10.17487/RFC6780, October 2012,
              <http://www.rfc-editor.org/info/rfc6780>.

9.2. Informative References

   [RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol Generic
              Requirements", RFC 4657, DOI 10.17487/RFC4657, September
              2006, <http://www.rfc-editor.org/info/rfc4657>.

Authors' Addresses

   Ina Minei
   Google, Inc.
   1600 Amphitheatre Parkway
   Mountain View, CA 94043
   US

   Email: inaminei@google.com

   Edward Crabbe

   Email: edward.crabbe@gmail.com

   Siva Sivabalan
   Cisco Systems, Inc.
   170 West Tasman Dr.
   San Jose, CA 95134
   US

   Email: msiva@cisco.com

Hariharan Ananthakrishnan
Packet Design

Email: hari@packetdesign.com

Xian Zhang
Huawei Technologies
F3-5-B R&D Center, Huawei Base Bantian, Longgang District
Shenzhen, Guangdong 518129
P.R.China

Email: zhang.xian@huawei.com

Yosuke Tanaka
NTT Communications Corporation
Granpark Tower 3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

Email: yosuke.tanaka@ntt.com

PCE Working Group   D. Dhody, Ed.
Internet-Draft   Huawei Technologies
Intended status: Standards Track   J. Hardwick
Expires: January 6, 2016   Metaswitch
                                                            V. Beeram
                                                     Juniper Networks
                                                              Z. Ali
                                                       Cisco Systems
                                                         J. Tantsura
                                                            Ericsson
                                                        July 5, 2015

        A YANG Data Model for Path Computation Element Communications Protocol
                                (PCEP)
                      draft-pkd-pce-pcep-yang-03

Abstract

   This document defines a YANG data model for the management of Path
   Computation Element communications Protocol (PCEP) for communications
   between a Path Computation Client (PCC) and a Path Computation
   Element (PCE), or between two PCEs. The data model includes
   configuration data and state data (status information and counters
   for the collection of statistics).

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on January 6, 2016.

Copyright Notice

This document is subject to BCP 78 and the IETF Trust's Legal
Provisions Relating to IETF Documents
(http://trustee.ietf.org/license-info) in effect on the date of
publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

The Path Computation Element (PCE) defined in [RFC4655] is an entity
that is capable of computing a network path or route based on a
network graph, and applying computational constraints. A Path
Computation Client (PCC) may make requests to a PCE for paths to be
computed.

PCEP is the communication protocol between a PCC and PCE and is
defined in [RFC5440]. PCEP interactions include path computation
requests and path computation replies as well as notifications of
specific states related to the use of a PCE in the context of
Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS)
Traffic Engineering (TE). [I-D.ietf-pce-stateful-pce] specifies
extensions to PCEP to enable stateful control of MPLS TE LSPs.

This document defines a YANG [RFC6020] data model for the management
of PCEP speakers. It is important to establish a common data model
for how PCEP speakers are identified, configured, and monitored. The
data model includes configuration data and state data (status
information and counters for the collection of statistics).

This document contains a specification of the PCEP YANG module,
"ietf-pcep" which provides the PCEP [RFC5440] data model.

2. 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 [RFC2119].

3. Terminology and Notation

This document uses the terminology defined in [RFC4655] and
[RFC5440]. In particular, it uses the following acronyms.

o Path Computation Request message (PCReq).

o Path Computation Reply message (PCRep).

o Notification message (PCNtf).

o Error message (PCErr).

o Request Parameters object (RP).

o Synchronization Vector object (SVEC).

o Explicit Route object (ERO).

This document also uses the following terms defined in [RFC7420]:

o PCEP entity: a local PCEP speaker.

o PCEP peer: to refer to a remote PCEP speaker.

   o PCEP speaker: where it is not necessary to distinguish between
      local and remote.

   Further, this document also uses the following terms defined in
   [I-D.ietf-pce-stateful-pce] :

   o Stateful PCE, Passive Stateful PCE, Active Stateful PCE

   o Delegation, Revocation, Redelegation

   o LSP State Report, Path Computation Report message (PCRpt).

   o LSP State Update, Path Computation Update message (PCUpd).

3.1. Tree Diagrams

   A graphical representation of the complete data tree is presented in
   Section 5. The meaning of the symbols in these diagrams is as
   follows and as per [I-D.ietf-netmod-rfc6087bis]:

   o Brackets "[" and "]" enclose list keys.

   o Curly braces "{" and "}" contain names of optional features that
      make the corresponding node conditional.

   o Abbreviations before data node names: "rw" means configuration
      (read-write), and "ro" state data (read-only).

   o Symbols after data node names: "?" means an optional node and "*"
      denotes a "list" or "leaf-list".

   o Parentheses enclose choice and case nodes, and case nodes are also
      marked with a colon (":").

   o Ellipsis ("...") stands for contents of subtrees that are not
      shown.

3.2. Prefixes in Data Node Names

   In this document, names of data nodes and other data model objects
   are often used without a prefix, as long as it is clear from the
   context in which YANG module each name is defined. Otherwise, names
   are prefixed using the standard prefix associated with the
   corresponding YANG module, as shown in Table 1.

```
                 +--------+----------------+-----------+
                 | Prefix | YANG module    | Reference |
                 +--------+----------------+-----------+
                 | yang   | ietf-yang-types | [RFC6991] |
                 | inet   | ietf-inet-types | [RFC6991] |
                 +--------+----------------+-----------+
```

            Table 1: Prefixes and corresponding YANG modules

4. Objectives

   This section describes some of the design objectives for the model:

   o In case of existing implementations, it needs to map the data
      model defined in this document to their proprietary native data
      model. To facilitate such mappings, the data model should be
      simple.

   o The data model should be suitable for new implementations to use
      as is.

   o Mapping to the PCEP MIB Module should be clear.

   o The data model should allow for static configurations of peers.

   o The data model should include read-only counters in order to
      gather statistics for sent and received PCEP messages, received
      messages with errors, and messages that could not be sent due to
      errors.

   o It should be fairly straightforward to augment the base data model
      for advanced PCE features.

5. The Design of PCEP Data Model

   The module, "ietf-pcep", defines the basic components of a PCE
   speaker.

```
module: ietf-pcep
+--rw pcep
|  +--rw entity
|     +--rw addr                          inet:ip-address
|     +--rw enabled?                      boolean
|     +--rw role                          pcep-role
|     +--rw description?                  string
|     +--rw domain
|     |  +--rw domain* [domain-type domain]
|     |     +--rw domain-type    domain-type
```

```
|    |     +--rw domain          domain
|    +--rw capability
|    |  +--rw gmpls?                boolean {gmpls}?
|    |  +--rw bi-dir?               boolean
|    |  +--rw diverse?              boolean
|    |  +--rw load-balance?         boolean
|    |  +--rw synchronize?          boolean {svec}?
|    |  +--rw objective-function?   boolean {obj-fn}?
|    |  +--rw add-path-constraint?  boolean
|    |  +--rw prioritization?       boolean
|    |  +--rw multi-request?        boolean
|    |  +--rw gco?                  boolean {gco}?
|    |  +--rw p2mp?                 boolean {p2mp}?
|    |  +--rw stateful {stateful}?
|    |     +--rw enabled?      boolean
|    |     +--rw active?       boolean
|    |     +--rw pce-initiated?  boolean
|    +--rw pce-info
|    |  +--rw scope
|    |  |  +--rw intra-area-scope?         boolean
|    |  |  +--rw intra-area-pref?          uint8
|    |  |  +--rw inter-area-scope?         boolean
|    |  |  +--rw inter-area-scope-default?  boolean
|    |  |  +--rw inter-area-pref?          uint8
|    |  |  +--rw inter-as-scope?           boolean
|    |  |  +--rw inter-as-scope-default?   boolean
|    |  |  +--rw inter-as-pref?            uint8
|    |  |  +--rw inter-layer-scope?        boolean
|    |  |  +--rw inter-layer-pref?         uint8
|    |  +--rw neigh-domains
|    |     +--rw domain* [domain-type domain]
|    |        +--rw domain-type    domain-type
|    |        +--rw domain         domain
|    +--rw connect-timer?           uint32
|    +--rw connect-max-retry?       uint32
|    +--rw init-backoff-timer?      uint32
|    +--rw max-backoff-timer?       uint32
|    +--rw open-wait-timer?         uint32
|    +--rw keep-wait-timer?         uint32
|    +--rw keep-alive-timer?        uint32
|    +--rw dead-timer?              uint32
|    +--rw allow-negotiation?       boolean
|    +--rw max-keep-alive-timer?    uint32
|    +--rw max-dead-timer?          uint32
|    +--rw min-keep-alive-timer?    uint32
|    +--rw min-dead-timer?          uint32
|    +--rw sync-timer?              uint32 {svec}?
|    +--rw request-timer?           uint32
```

```
   |      +--rw max-sessions?                uint32
   |      +--rw max-unknown-reqs?            uint32
   |      +--rw max-unknown-msgs?            uint32
   |      +--rw pcep-notification-max-rate   uint32
   |      +--rw stateful-timer {stateful}?
   |      |  +--rw state-timeout?         uint32
   |      |  +--rw redelegation-timeout?  uint32
   |      |  +--rw rpt-non-pcep-lsp?      boolean
   |      +--rw peers
   |         +--rw peer* [addr]
   |            +--rw addr                inet:ip-address
   |            +--rw description?        string
   |            +--rw domain
   |            |  +--rw domain* [domain-type domain]
   |            |     +--rw domain-type    domain-type
   |            |     +--rw domain         domain
   |            +--rw capability
   |            |  +--rw gmpls?               boolean {gmpls}?
   |            |  +--rw bi-dir?              boolean
   |            |  +--rw diverse?             boolean
   |            |  +--rw load-balance?        boolean
   |            |  +--rw synchronize?         boolean {svec}?
   |            |  +--rw objective-function?  boolean {obj-fn}?
   |            |  +--rw add-path-constraint? boolean
   |            |  +--rw prioritization?      boolean
   |            |  +--rw multi-request?       boolean
   |            |  +--rw gco?                 boolean {gco}?
   |            |  +--rw p2mp?                boolean {p2mp}?
   |            |  +--rw stateful {stateful}?
   |            |     +--rw enabled?        boolean
   |            |     +--rw active?         boolean
   |            |     +--rw pce-initiated?  boolean
   |            +--rw scope
   |            |  +--rw intra-area-scope?         boolean
   |            |  +--rw intra-area-pref?          uint8
   |            |  +--rw inter-area-scope?         boolean
   |            |  +--rw inter-area-scope-default? boolean
   |            |  +--rw inter-area-pref?          uint8
   |            |  +--rw inter-as-scope?           boolean
   |            |  +--rw inter-as-scope-default?   boolean
   |            |  +--rw inter-as-pref?            uint8
   |            |  +--rw inter-layer-scope?        boolean
   |            |  +--rw inter-layer-pref?         uint8
   |            +--rw neigh-domains
   |            |  +--rw domain* [domain-type domain]
   |            |     +--rw domain-type    domain-type
   |            |     +--rw domain         domain
   |            +--rw delegation-pref?   uint8 {stateful}?
```

```
+--ro pcep-state
   +--ro entity
      +--ro addr?                      inet:ip-address
      +--ro index?                     uint32
      +--ro admin-status?              pcep-admin-status
      +--ro oper-status?               pcep-admin-status
      +--ro role?                      pcep-role
      +--ro domain
      |  +--ro domain* [domain-type domain]
      |     +--ro domain-type    domain-type
      |     +--ro domain         domain
      +--ro capability
      |  +--ro gmpls?                 boolean {gmpls}?
      |  +--ro bi-dir?                boolean
      |  +--ro diverse?              boolean
      |  +--ro load-balance?         boolean
      |  +--ro synchronize?          boolean {svec}?
      |  +--ro objective-function?   boolean {obj-fn}?
      |  +--ro add-path-constraint?  boolean
      |  +--ro prioritization?       boolean
      |  +--ro multi-request?        boolean
      |  +--ro gco?                  boolean {gco}?
      |  +--ro p2mp?                 boolean {p2mp}?
      |  +--ro stateful {stateful}?
      |     +--ro enabled?        boolean
      |     +--ro active?         boolean
      |     +--ro pce-initiated?  boolean
      +--ro pce-info
      |  +--ro scope
      |  |  +--ro intra-area-scope?         boolean
      |  |  +--ro intra-area-pref?          uint8
      |  |  +--ro inter-area-scope?         boolean
      |  |  +--ro inter-area-scope-default? boolean
      |  |  +--ro inter-area-pref?          uint8
      |  |  +--ro inter-as-scope?           boolean
      |  |  +--ro inter-as-scope-default?   boolean
      |  |  +--ro inter-as-pref?            uint8
      |  |  +--ro inter-layer-scope?        boolean
      |  |  +--ro inter-layer-pref?         uint8
      |  +--ro neigh-domains
      |     +--ro domain* [domain-type domain]
      |        +--ro domain-type    domain-type
      |        +--ro domain         domain
      +--ro connect-timer?        uint32
      +--ro connect-max-retry?    uint32
      +--ro init-backoff-timer?   uint32
      +--ro max-backoff-timer?    uint32
      +--ro open-wait-timer?      uint32
```

```
      +--ro keep-wait-timer?       uint32
      +--ro keep-alive-timer?      uint32
      +--ro dead-timer?            uint32
      +--ro allow-negotiation?     boolean
      +--ro max-keep-alive-timer?  uint32
      +--ro max-dead-timer?        uint32
      +--ro min-keep-alive-timer?  uint32
      +--ro min-dead-timer?        uint32
      +--ro sync-timer?            uint32 {svec}?
      +--ro request-timer?         uint32
      +--ro max-sessions?          uint32
      +--ro max-unknown-reqs?      uint32
      +--ro max-unknown-msgs?      uint32
      +--ro stateful-timer {stateful}?
      |  +--ro state-timeout?        uint32
      |  +--ro redelegation-timeout?  uint32
      +--ro lsp-db {stateful}?
      |  +--ro lsp* [plsp-id pcc-id]
      |     +--ro plsp-id             uint32
      |     +--ro pcc-id              inet:ip-address
      |     +--ro admin-state?        boolean
      |     +--ro operational-state?  operational-state
      |     +--ro delegated
      |     |  +--ro enabled?   boolean
      |     |  +--ro pce?       leafref
      |     |  +--ro srp-id?    uint32
      |     +--ro symbolic-path-name?  string
      |     +--ro last-error?          lsp-error
      +--ro peers
         +--ro peer* [addr]
            +--ro addr                        inet:ip-address
            +--ro role?                       pcep-role
            +--ro domain
            |  +--ro domain* [domain-type domain]
            |     +--ro domain-type    domain-type
            |     +--ro domain         domain
            +--ro capability
            |  +--ro gmpls?                boolean {gmpls}?
            |  +--ro bi-dir?               boolean
            |  +--ro diverse?              boolean
            |  +--ro load-balance?         boolean
            |  +--ro synchronize?          boolean {svec}?
            |  +--ro objective-function?   boolean {obj-fn}?
            |  +--ro add-path-constraint?  boolean
            |  +--ro prioritization?       boolean
            |  +--ro multi-request?        boolean
            |  +--ro gco?                  boolean {gco}?
            |  +--ro p2mp?                 boolean {p2mp}?
```

```
            |  +--ro stateful {stateful}?
            |     +--ro enabled?        boolean
            |     +--ro active?         boolean
            |     +--ro pce-initiated?  boolean
            +--ro pce-info
            |  +--ro scope
            |  |  +--ro intra-area-scope?          boolean
            |  |  +--ro intra-area-pref?           uint8
            |  |  +--ro inter-area-scope?          boolean
            |  |  +--ro inter-area-scope-default?  boolean
            |  |  +--ro inter-area-pref?           uint8
            |  |  +--ro inter-as-scope?            boolean
            |  |  +--ro inter-as-scope-default?    boolean
            |  |  +--ro inter-as-pref?             uint8
            |  |  +--ro inter-layer-scope?         boolean
            |  |  +--ro inter-layer-pref?          uint8
            |  +--ro neigh-domains
            |     +--ro domain* [domain-type domain]
            |        +--ro domain-type    domain-type
            |        +--ro domain         domain
            +--ro delegation-pref?          uint8 {stateful}?
            +--ro discontinuity-time?       yang:timestamp
            +--ro initiate-session?         boolean
            +--ro session-exists?           boolean
            +--ro num-sess-setup-ok?        yang:counter32
            +--ro num-sess-setup-fail?      yang:counter32
            +--ro session-up-time?          yang:timestamp
            +--ro session-fail-time?        yang:timestamp
            +--ro session-fail-up-time?     yang:timestamp
            +--ro avg-rsp-time?             uint32
            +--ro lwm-rsp-time?             uint32
            +--ro hwm-rsp-time?             uint32
            +--ro num-pcreq-sent?           yang:counter32
            +--ro num-pcreq-rcvd?           yang:counter32
            +--ro num-pcrep-sent?           yang:counter32
            +--ro num-pcrep-rcvd?           yang:counter32
            +--ro num-pcerr-sent?           yang:counter32
            +--ro num-pcerr-rcvd?           yang:counter32
            +--ro num-pcntf-sent?           yang:counter32
            +--ro num-pcntf-rcvd?           yang:counter32
            +--ro num-keepalive-sent?       yang:counter32
            +--ro num-keepalive-rcvd?       yang:counter32
            +--ro num-unknown-rcvd?         yang:counter32
            +--ro num-corrupt-rcvd?         yang:counter32
            +--ro num-req-sent?             yang:counter32
            +--ro num-req-sent-pend-rep?    yang:counter32
            +--ro num-req-sent-ero-rcvd?    yang:counter32
            +--ro num-req-sent-nopath-rcvd? yang:counter32
```

```
           +--ro num-req-sent-cancel-rcvd?    yang:counter32
           +--ro num-req-sent-error-rcvd?     yang:counter32
           +--ro num-req-sent-timeout?        yang:counter32
           +--ro num-req-sent-cancel-sent?    yang:counter32
           +--ro num-req-rcvd?                yang:counter32
           +--ro num-req-rcvd-pend-rep?       yang:counter32
           +--ro num-req-rcvd-ero-sent?       yang:counter32
           +--ro num-req-rcvd-nopath-sent?    yang:counter32
           +--ro num-req-rcvd-cancel-sent?    yang:counter32
           +--ro num-req-rcvd-error-sent?     yang:counter32
           +--ro num-req-rcvd-cancel-rcvd?    yang:counter32
           +--ro num-rep-rcvd-unknown?        yang:counter32
           +--ro num-req-rcvd-unknown?        yang:counter32
           +--ro svec {svec}?
           |  +--ro num-svec-sent?        yang:counter32
           |  +--ro num-svec-req-sent?    yang:counter32
           |  +--ro num-svec-rcvd?        yang:counter32
           |  +--ro num-svec-req-rcvd?    yang:counter32
           +--ro stateful {stateful}?
           |  +--ro num-pcrpt-sent?               yang:counter32
           |  +--ro num-pcrpt-rcvd?               yang:counter32
           |  +--ro num-pcupd-sent?               yang:counter32
           |  +--ro num-pcupd-rcvd?               yang:counter32
           |  +--ro num-rpt-sent?                 yang:counter32
           |  +--ro num-rpt-rcvd?                 yang:counter32
           |  +--ro num-rpt-rcvd-error-sent?      yang:counter32
           |  +--ro num-upd-sent?                 yang:counter32
           |  +--ro num-upd-rcvd?                 yang:counter32
           |  +--ro num-upd-rcvd-unknown?         yang:counter32
           |  +--ro num-upd-rcvd-undelegated?     yang:counter32
           |  +--ro num-upd-rcvd-error-sent?      yang:counter32
           +--ro num-req-sent-closed?      yang:counter32
           +--ro num-req-rcvd-closed?      yang:counter32
           +--ro sessions
              +--ro session* [initiator]
                 +--ro initiator                pcep-initiator
                 +--ro state-last-change?       yang:timestamp
                 +--ro state?                   pcep-sess-state
                 +--ro connect-retry?           yang:counter32
                 +--ro local-id?                uint32
                 +--ro remote-id?               uint32
                 +--ro keepalive-timer?         uint32
                 +--ro peer-keepalive-timer?    uint32
                 +--ro dead-timer?              uint32
                 +--ro peer-dead-timer?         uint32
                 +--ro ka-hold-time-rem?        uint32
                 +--ro overloaded?              boolean
                 +--ro overload-time?           uint32
```

```
          +--ro peer-overloaded?             boolean
          +--ro peer-overload-time?          uint32
          +--ro lspdb-sync?                  sync-state {stateful}?
          +--ro discontinuity-time?          yang:timestamp
          +--ro avg-rsp-time?                uint32
          +--ro lwm-rsp-time?                uint32
          +--ro hwm-rsp-time?                uint32
          +--ro num-pcreq-sent?              yang:counter32
          +--ro num-pcreq-rcvd?              yang:counter32
          +--ro num-pcrep-sent?              yang:counter32
          +--ro num-pcrep-rcvd?              yang:counter32
          +--ro num-pcerr-sent?              yang:counter32
          +--ro num-pcerr-rcvd?              yang:counter32
          +--ro num-pcntf-sent?              yang:counter32
          +--ro num-pcntf-rcvd?              yang:counter32
          +--ro num-keepalive-sent?          yang:counter32
          +--ro num-keepalive-rcvd?          yang:counter32
          +--ro num-unknown-rcvd?            yang:counter32
          +--ro num-corrupt-rcvd?            yang:counter32
          +--ro num-req-sent?                yang:counter32
          +--ro num-req-sent-pend-rep?       yang:counter32
          +--ro num-req-sent-ero-rcvd?       yang:counter32
          +--ro num-req-sent-nopath-rcvd?    yang:counter32
          +--ro num-req-sent-cancel-rcvd?    yang:counter32
          +--ro num-req-sent-error-rcvd?     yang:counter32
          +--ro num-req-sent-timeout?        yang:counter32
          +--ro num-req-sent-cancel-sent?    yang:counter32
          +--ro num-req-rcvd?                yang:counter32
          +--ro num-req-rcvd-pend-rep?       yang:counter32
          +--ro num-req-rcvd-ero-sent?       yang:counter32
          +--ro num-req-rcvd-nopath-sent?    yang:counter32
          +--ro num-req-rcvd-cancel-sent?    yang:counter32
          +--ro num-req-rcvd-error-sent?     yang:counter32
          +--ro num-req-rcvd-cancel-rcvd?    yang:counter32
          +--ro num-rep-rcvd-unknown?        yang:counter32
          +--ro num-req-rcvd-unknown?        yang:counter32
          +--ro svec {svec}?
          |  +--ro num-svec-sent?        yang:counter32
          |  +--ro num-svec-req-sent?    yang:counter32
          |  +--ro num-svec-rcvd?        yang:counter32
          |  +--ro num-svec-req-rcvd?    yang:counter32
          +--ro stateful {stateful}?
             +--ro num-pcrpt-sent?            yang:counter32
             +--ro num-pcrpt-rcvd?            yang:counter32
             +--ro num-pcupd-sent?            yang:counter32
             +--ro num-pcupd-rcvd?            yang:counter32
             +--ro num-rpt-sent?             yang:counter32
             +--ro num-rpt-rcvd?             yang:counter32
```

```
                         +--ro num-rpt-rcvd-error-sent?      yang:counter32
                         +--ro num-upd-sent?                 yang:counter32
                         +--ro num-upd-rcvd?                 yang:counter32
                         +--ro num-upd-rcvd-unknown?         yang:counter32
                         +--ro num-upd-rcvd-undelegated?     yang:counter32
                         +--ro num-upd-rcvd-error-sent?      yang:counter32
  notifications:
  +---n pcep-session-up
  |  +--ro entity-addr?        leafref
  |  +--ro peer-addr?          leafref
  |  +--ro session-initiator?  leafref
  |  +--ro state-last-change?  yang:timestamp
  |  +--ro state?              pcep-sess-state
  +---n pcep-session-down
  |  +--ro entity-addr?        leafref
  |  +--ro peer-addr?          leafref
  |  +--ro session-initiator?  pcep-initiator
  |  +--ro state-last-change?  yang:timestamp
  |  +--ro state?              pcep-sess-state
  +---n pcep-session-local-overload
  |  +--ro entity-addr?        leafref
  |  +--ro peer-addr?          leafref
  |  +--ro session-initiator?  leafref
  |  +--ro overloaded?         boolean
  |  +--ro overload-time?      uint32
  +---n pcep-session-local-overload-clear
  |  +--ro entity-addr?  leafref
  |  +--ro peer-addr?    leafref
  |  +--ro overloaded?   boolean
  +---n pcep-session-peer-overload
  |  +--ro entity-addr?        leafref
  |  +--ro peer-addr?          leafref
  |  +--ro session-initiator?  leafref
  |  +--ro peer-overloaded?    boolean
  |  +--ro peer-overload-time? uint32
  +---n pcep-session-peer-overload-clear
     +--ro entity-addr?     leafref
     +--ro peer-addr?       leafref
     +--ro peer-overloaded? boolean
```

5.1. The Entity

   The PCEP yang module may contain status information for the local
   PCEP entity.

The data model for PCEP presented in this document uses a flat list
of entities. The entity has an IP address (using ietf-inet-types
[RFC6991]) and a "role" leaf (the local entity PCEP role) as
mandatory.

Note that, the PCEP MIB module [RFC7420] uses an entity list and a
system generated entity index as a primary index to the read only
entity table. If the device implements the PCEP MIB, the "index"
leaf MUST contain the value of the corresponding pcePcepEntityIndex
and only one entity is assumed.

5.2. The Peer Lists

The peer list contains peer(s) that the local PCEP entity knows
about. A PCEP speaker is identified by its IP address. If there is
a PCEP speaker in the network that uses multiple IP addresses then it
looks like multiple distinct peers to the other PCEP speakers in the
network.

Since PCEP sessions can be ephemeral, the peer list tracks a peer
even when no PCEP session currently exists to that peer. The
statistics contained are an aggregate of the statistics for all
successive sessions to that peer.

To limit the quantity of information that is stored, an
implementation MAY choose to discard this information if and only if
no PCEP session exists to the corresponding peer.

The data model for PCEP peer presented in this document uses a flat
list of peers. Each peer in the list is identified by its IP address
(addr-type, addr).

There is one list for static peer configuration ("/pcep/entity/
peers"), and a separate list for the operational state of all peers
(i.e. static as well as discovered)("/pcep-state/entity/peers").

5.3. The Session Lists

The session list contains PCEP session that the PCEP entity (PCE or
PCC) is currently participating in. The statistics in session are
semantically different from those in peer since the former applies to
the current session only, whereas the latter is the aggregate for all
sessions that have existed to that peer.

Although [RFC5440] forbids more than one active PCEP session between
a given pair of PCEP entities at any given time, there is a window
during session establishment where two sessions may exist for a given
pair, one representing a session initiated by the local PCEP entity

and the other representing a session initiated by the peer. If
either of these sessions reaches active state first, then the other
is discarded.

The data model for PCEP session presented in this document uses a
flat list of sessions. Each session in the list is identified by its
initiator. This index allows two sessions to exist transiently for a
given peer, as discussed above.

There is only one list for the operational state of all sessions
("/pcep-state/entity/peers/peer/sessions/session").

5.4. Notifications

This YANG model defines a list of notifications to inform client of
important events detected during the protocol operation. The
notifications defined cover the PCEP MIB notifications.

6. Advanced PCE Features

This document contains a specification of the base PCEP YANG module,
"ietf-pcep" which provides the basic PCEP [RFC5440] data model.

This document further handles advanced PCE features like -

o Capability and Scope

o Domain information (local/neighbour)

o Path-Key

o OF

o GCO

o P2MP

o GMPLS

o Inter-Layer

o Stateful PCE

o Segement Routing

[Editor's Note - Some of them would be added in a future revision.]

7. PCEP YANG Module

   RFC Ed.: In this section, replace all occurrences of 'XXXX' with the
   actual RFC number and all occurrences of the revision date below with
   the date of RFC publication (and remove this note).

<CODE BEGINS> file "ietf-pcep@2015-07-03.yang"


```
module ietf-pcep {
    namespace "urn:ietf:params:xml:ns:yang:ietf-pcep";
    prefix pcep;

    import ietf-inet-types {
        prefix inet;
    }

    import ietf-yang-types {
        prefix yang;
    }

    organization
        "IETF PCE (Path Computation Element) Working Group";

    contact
        "WG Web:   <http://tools.ietf.org/wg/pce/>
         WG List:  <mailto:pce@ietf.org>

         WG Chair: JP Vasseur
                   <mailto:jpv@cisco.com>

         WG Chair: Julien Meuric
                   <mailto:julien.meuric@orange.com>

         Editor:   Dhruv Dhody
                   <mailto:dhruv.ietf@gmail.com>";

    description
        "The YANG module defines a generic configuration and
         operational model for PCEP common across all of the
         vendor implementations.";
```

```
    revision 2015-07-03 {
        description "Initial revision.";
        reference
            "RFC XXXX:  A YANG Data Model for Path Computation
                        Element Communications Protocol
                        (PCEP)";
    }

    /*
     * Identities
     */

    identity pcep {
        description "Identity for the PCEP protocol.";
    }

    /*
     * Typedefs
     */
    typedef pcep-role {
        type enumeration {
            enum unknown {
                value "0";
                description
                "An unknown role";
            }
            enum pcc {
                value "1";
                description
                "The role of a Path Computation Client";
            }
            enum pce {
                value "2";
                description
                "The role of Path Computation Element";
            }
            enum pcc-and-pce {
                value "3";
                description
                "The role of both Path Computation Client and
                 Path Computation Element";
            }
        }

        description
            "The role of a PCEP speaker.
             Takes one of the following values
             - unknown(0): the role is not known.
```

```
              - pcc(1): the role is of a Path Computation
                Client (PCC).
              - pce(2): the role is of a Path Computation
                Server (PCE).
              - pccAndPce(3): the role is of both a PCC and
                a PCE.";

    }

    typedef pcep-admin-status {
        type enumeration {
                enum admin-status-up {
                value "1";
                description
                "Admin Status is Up";
            }
            enum admin-status-down {
                value "2";
                description
                "Admin Status is Down";
            }
        }

        description
        "The Admin Status of the PCEP entity.
         Takes one of the following values
             - admin-status-up(1): Admin Status is Up.
             - admin-status-down(2): Admin Status is Down";
    }

    typedef pcep-oper-status {
        type enumeration {
            enum oper-status-up {
                value "1";
                description
                "The PCEP entity is active";
            }
            enum oper-status-down {
                value "2";
                description
                "The PCEP entity is inactive";
            }
            enum oper-status-going-up {
                value "3";
                description
                "The PCEP entity is activating";
            }
            enum oper-status-going-down {
```

```
                value "4";
                description
                "The PCEP entity is deactivating";
            }
            enum oper-status-failed {
                value "5";
                description
                "The PCEP entity has failed and will recover
                 when possible.";
            }
            enum oper-status-failed-perm {
                value "6";
                description
                "The PCEP entity has failed and will not recover
                 without operator intervention";
            }
        }
        description
        "The operational status of the PCEP entity.
         Takes one of the following values
             - oper-status-up(1): Active
             - oper-status-down(2): Inactive
             - oper-status-going-up(3): Activating
             - oper-status-going-down(4): Deactivating
             - oper-status-failed(5): Failed
             - oper-status-failed-perm(6): Failed Permanantly";
    }

    typedef pcep-initiator {
        type enumeration {
            enum local {
                value "1";
                description
                "The local PCEP entity initiated the session";
            }

            enum remote {
                value "2";
                description
                "The remote PCEP peer initiated the session";
            }
        }
        description
        "The initiator of the session, that is, whether the TCP
         connection was initiated by the local PCEP entity or
         the remote peer.
         Takes one of the following values
             - local(1): Initiated locally
```

```
               - remote(2): Initiated remotely";
     }

     typedef pcep-sess-state {
         type enumeration {
             enum tcp-pending {
                 value "1";
                 description
                     "The tcp-pending state of PCEP session.";
             }

             enum open-wait {
                 value "2";
                 description
                     "The open-wait state of PCEP session.";
             }

             enum keep-wait {
                 value "3";
                 description
                     "The keep-wait state of PCEP session.";
             }

             enum session-up {
                 value "4";
                 description
                     "The session-up state of PCEP session.";
             }
         }
         description
             "The current state of the session.

              The set of possible states excludes the idle state
              since entries do not exist in the idle state.
              Takes one of the following values
                 - tcp-pending(1): PCEP TCP Pending state
                 - open-wait(2): PCEP Open Wait state
                 - keep-wait(3): PCEP Keep Wait state
                 - session-up(4): PCEP Session Up state";
     }

     typedef domain-type {
         type enumeration {
             enum ospf-area {
                 value "1";
                 description
                     "The OSPF area.";
             }
```

```
            enum isis-area {
                value "2";
                description
                    "The IS-IS area.";
            }
            enum as {
                value "3";
                description
                    "The Autonomous System (AS).";
            }
        }
        description
            "The PCE Domain Type";
    }

    typedef domain-ospf-area {
        type union {
            type uint32;
            type yang:dotted-quad;
        }
        description
            "OSPF Area ID.";
    }

    typedef domain-isis-area {
        type string {
            pattern '[0-9A-Fa-f]{2}\.([0-9A-Fa-f]{4}\.){0,3}';
        }
        description
            "IS-IS Area ID.";
    }

    typedef domain-as {
        type uint32;
        description
            "Autonomous System number.";

    }

    typedef domain {
        type union {
            type domain-ospf-area;
            type domain-isis-area;
            type domain-as;
        }
        description
            "The Domain Information";
    }
```

```
    typedef operational-state {
        type enumeration {
            enum down {
                value "0";
                description
                    "not active.";
            }
            enum up {
                value "1";
                description
                    "signalled.";
            }
            enum active {
                value "2";
                description
                    "up and carrying traffic.";
            }
            enum going-down {
                value "3";
                description
                    "LSP is being torn down, resources are
                     being released.";
            }
            enum going-up {
                value "4";
                description
                    "LSP is being signalled.";
            }
        }
        description
            "The operational status of the LSP";
    }

    typedef lsp-error {
        type enumeration {
            enum no-error {
                value "0";
                description
                    "No error, LSP is fine.";
            }
            enum unknown {
                value "1";
                description
                    "Unknown reason.";
            }
            enum limit {
                value "2";
                description
```

```
                    "Limit reached for PCE-controlled LSPs.";
            }
            enum pending {
                value "3";
                description
                    "Too many pending LSP update requests.";
            }
            enum unacceptable {
                value "4";
                description
                    "Unacceptable parameters.";
            }
            enum internal {
                value "5";
                description
                    "Internal error.";
            }
            enum admin {
                value "6";
                description
                    "LSP administratively brought down.";
            }
            enum preempted {
                value "7";
                description
                    "LSP preempted.";
            }
            enum rsvp {
                value "8";
                description
                    "RSVP signaling error.";
            }
        }
        description
            "The LSP Error Codes.";
    }

    typedef sync-state {
        type enumeration {
            enum pending {
                value "0";
                description
                    "The state synchronization
                     has not started.";
            }
            enum ongoing {
                value "1";
                description
```

```
                        "The state synchronization
                         is ongoing.";
                }
                enum finished {
                    value "2";
                    description
                        "The state synchronization
                         is finished.";
                }
            }
            description
                "The LSP-DB state synchronization operational status.";
        }
        /*
         * Features
         */

        feature svec {
            description
                "Support synchronized path computation.";
        }

        feature gmpls {
            description
                "Support GMPLS.";
        }

        feature obj-fn {
            description
                "Support OF as per RFC 5541.";
        }

        feature gco {
            description
                "Support GCO as per RFC 5557.";
        }

        feature pathkey {
            description
                "Support pathkey as per RFC 5520.";
        }

        feature p2mp {
            description
                "Support P2MP as per RFC 6006.";
        }

        feature stateful {
```

```
            description
                "Support stateful PCE.";
        }



    /*
     * Groupings
     */



    grouping pcep-entity-info{
        description
            "This grouping defines the attributes for PCEP entity.";
        leaf connect-timer {
            type uint32 {
                range "1..65535";
            }
            units "seconds";
            default 60;
            description
                "The time in seconds that the PCEP entity will wait
                 to establish a TCP connection with a peer.  If a
                 TCP connection is not established within this time
                 then PCEP aborts the session setup attempt.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                           Communication Protocol (PCEP)";
        }

        leaf connect-max-retry {
            type uint32;
            default 5;
            description
                "The maximum number of times the system tries to
                 establish a TCP connection to a peer before the
                 session with the peer transitions to the idle
                 state.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                           Communication Protocol (PCEP)";
        }

        leaf init-backoff-timer {
            type uint32 {
                range "1..65535";
            }
```

```
            units "seconds";
            description
               "The initial back-off time in seconds for retrying
                a failed session setup attempt to a peer.

                The back-off time increases for each failed
                session setup attempt, until a maximum back-off
                time is reached.  The maximum back-off time is
                max-backoff-timer.";
        }

        leaf max-backoff-timer {
            type uint32;
            units "seconds";
            description
               "The maximum back-off time in seconds for retrying
                a failed session setup attempt to a peer.

                The back-off time increases for each failed session
                setup attempt, until this maximum value is reached.
                Session setup attempts then repeat periodically
                without any further increase in back-off time.";
        }

        leaf open-wait-timer {
            type uint32 {
                range "1..65535";
            }
            units "seconds";
            default 60;
            description
               "The time in seconds that the PCEP entity will wait
                to receive an Open message from a peer after the
                TCP connection has come up.

                If no Open message is received within this time then
                PCEP terminates the TCP connection and deletes the
                associated sessions.";
            reference
               "RFC 5440: Path Computation Element (PCE)
                          Communication Protocol (PCEP)";
        }

        leaf keep-wait-timer {
            type uint32 {
                range "1..65535";
            }
            units "seconds";
```

```
        default 60;
        description
            "The time in seconds that the PCEP entity will wait
             to receive a Keepalive or PCErr message from a peer
             during session initialization after receiving an
             Open message.  If no Keepalive or PCErr message is
             received within this time then PCEP terminates the
             TCP connection and deletes the associated
             sessions.";
        reference
            "RFC 5440: Path Computation Element (PCE)
                       Communication Protocol (PCEP)";
    }

    leaf keep-alive-timer {
        type uint32 {
            range "0..255";
        }
        units "seconds";
        default 30;
        description
            "The keep alive transmission timer that this PCEP
             entity will propose in the initial OPEN message of
             each session it is involved in.  This is the
             maximum time between two consecutive messages sent
             to a peer. Zero means that the PCEP entity prefers
             not to send Keepalives at all.

             Note that the actual Keepalive transmission
             intervals, in either direction of an active PCEP
             session, are determined by negotiation between the
             peers as specified by RFC 5440, and so may differ
             from this configured value.";
        reference
            "RFC 5440: Path Computation Element (PCE)
                       Communication Protocol (PCEP)";
    }

    leaf dead-timer {
        type uint32 {
            range "0..255";
        }
        units "seconds";
        must ". >= ../keep-alive-timer" {
            error-message "The dead timer must be "
                    + "larger than the keep alive timer";
            description
                "This value MUST be greater than
```

```
                    keep-alive-timer.";

            }
            default 120;
            description
                "The dead timer that this PCEP entity will propose
                 in the initial OPEN message of each session it is
                 involved in. This is the time after which a peer
                 should declare a session down if it does not
                 receive any PCEP messages. Zero suggests that the
                 peer does not run a dead timer at all." ;
            reference
                "RFC 5440: Path Computation Element (PCE)
                          Communication Protocol (PCEP)";
        }

        leaf allow-negotiation{
            type boolean;
            description
                "Whether the PCEP entity will permit negotiation of
                 session parameters.";
        }

        leaf max-keep-alive-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in seconds,
                 the maximum value that this PCEP entity will
                 accept from a peer for the interval between
                 Keepalive transmissions. Zero means that the PCEP
                 entity will allow no Keepalive transmission at
                 all." ;
        }

        leaf max-dead-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in seconds,
                 the maximum value that this PCEP entity will accept
                 from a peer for the Dead timer.  Zero means that
                 the PCEP entity will allow not running a Dead
                 timer.";
```

```
        }

        leaf min-keep-alive-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in seconds,
                 the minimum value that this PCEP entity will
                 accept for the interval between Keepalive
                 transmissions. Zero means that the PCEP entity
                 insists on no Keepalive transmission at all.";
        }

        leaf min-dead-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiationin in
                 seconds, the minimum value that this PCEP entity
                 will accept for the Dead timer.  Zero means that
                 the PCEP entity insists on not running a Dead
                 timer.";
        }

        leaf sync-timer{
            if-feature svec;
            type uint32 {
                range "0..65535";
            }
            units "seconds";
            default 60;
            description
                "The value of SyncTimer in seconds is used in the
                 case of synchronized path computation request
                 using the SVEC object. Consider the case where a
                 PCReq message is received by a PCE that contains
                 the SVEC object referring to M synchronized path
                 computation requests.  If after the expiration of
                 the SyncTimer all the M path computation requests
                 have not been, received a protocol error is
                 triggered and the PCE MUST cancel the whole set
                 of path computation  requests.

                 The aim of the SyncTimer is to avoid the storage
```

```
                of unused synchronized requests should one of
                them get lost for some reasons (for example, a
                misbehaving PCC).

                Zero means that the PCEP entity does not use the
                SyncTimer.";
          reference
              "RFC 5440: Path Computation Element (PCE)
                         Communication Protocol (PCEP)";
      }


      leaf request-timer{
          type uint32 {
              range "1..65535";
          }
          units "seconds";
          description
              "The maximum time that the PCEP entity will wait
               for a response to a PCReq message.";
      }

      leaf max-sessions{
          type uint32;
          description
             "Maximum number of sessions involving this PCEP
              entity that can exist at any time.";
      }

      leaf max-unknown-reqs{
          type uint32;
          default 5;
          description
             "The maximum number of unrecognized requests and
              replies that any session on this PCEP entity is
              willing to accept per minute before terminating
              the session.

              A PCRep message contains an unrecognized reply
              if it contains an RP object whose request ID
              does not correspond to any in-progress request
              sent by this PCEP entity.

              A PCReq message contains an unrecognized request
              if it contains an RP object whose request ID is
              zero.";
          reference
              "RFC 5440: Path Computation Element (PCE)
```

```
                                Communication Protocol (PCEP)";
        }

        leaf max-unknown-msgs{
            type uint32;
            default 5;
            description
             "The maximum number of unknown messages that any
              session on this PCEP entity is willing to accept
              per minute before terminating the session.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                            Communication Protocol (PCEP)";
        }

    }//pcep-entity-info

    grouping  pce-scope{
        description
            "This grouping defines PCE path computation scope
             information which maybe relevant to PCE selection.
             This information corresponds to PCE auto-discovery
             information.";
        reference
            "RFC 5088: OSPF Protocol Extensions for Path
                        Computation Element (PCE)
                        Discovery
             RFC 5089: IS-IS Protocol Extensions for Path
                        Computation Element (PCE)
                        Discovery";
        leaf intra-area-scope{
            type boolean;
            default true;
            description
                "PCE can compute intra-area paths.";
        }
        leaf intra-area-pref{
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for intra-area TE LSP
              computation.";
        }
        leaf inter-area-scope{
            type boolean;
            default false;
            description
```

```
                "PCE can compute inter-area paths.";
        }
        leaf inter-area-scope-default{
            type boolean;
            default false;
            description
                "PCE can act as a default PCE for inter-area
                 path computation.";
        }
        leaf inter-area-pref{
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for inter-area TE LSP
              computation.";
        }
        leaf inter-as-scope{
            type boolean;
            default false;
            description
                "PCE can compute inter-AS paths.";
        }
        leaf inter-as-scope-default{
            type boolean;
            default false;
            description
                "PCE can act as a default PCE for inter-AS
                 path computation.";
        }
        leaf inter-as-pref{
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for inter-AS TE LSP
              computation.";
        }
        leaf inter-layer-scope{
            type boolean;
            default false;
            description
                "PCE can compute inter-layer paths.";
        }
        leaf inter-layer-pref{
            type uint8{
                range "0..7";
            }
```

```
            description
              "The PCE's preference for inter-layer TE LSP
              computation.";
          }
      }//pce-scope

      grouping  domain{
          description
              "This grouping specifies a Domain where the
               PCEP speaker has topology visibility.";
          leaf domain-type{
              type domain-type;
              description
                "The domain type.";
          }
          leaf domain{
              type domain;
              description
                "The domain Information.";
          }
      }//domain

      grouping capability{
          description
              "This grouping specifies a capability
               information of local PCEP entity. This maybe
               relevant to PCE selection as well. This
               information corresponds to PCE auto-discovery
               information.";
          reference
              "RFC 5088: OSPF Protocol Extensions for Path
                          Computation Element (PCE)
                          Discovery
               RFC 5089: IS-IS Protocol Extensions for Path
                          Computation Element (PCE)
                          Discovery";
          leaf gmpls{
              if-feature gmpls;
              type boolean;
              description
                "Path computation with GMPLS link
                 constraints.";
          }
          leaf bi-dir{
              type boolean;
              description
                "Bidirectional path computation.";
          }
```

```
        leaf diverse{
            type boolean;
            description
              "Diverse path computation.";
        }
        leaf load-balance{
            type boolean;
            description
              "Load-balanced path computation.";
        }
        leaf synchronize{
            if-feature svec;
            type boolean;
            description
              "Synchronized paths computation.";
        }
        leaf objective-function{
            if-feature obj-fn;
            type boolean;
            description
              "Support for multiple objective functions.";
        }
        leaf add-path-constraint{
            type boolean;
            description
              "Support for additive path constraints (max
               hop count, etc.).";
        }
        leaf prioritization{
            type boolean;
            description
              "Support for request prioritization.";
        }
        leaf multi-request{
            type boolean;
            description
              "Support for multiple requests per message.";
        }
        leaf gco{
            if-feature gco;
            type boolean;
            description
              "Support for Global Concurrent Optimization
               (GCO).";
        }
        leaf p2mp{
            if-feature p2mp;
            type boolean;
```

```
            description
              "Support for P2MP path computation.";
          }
        container stateful{
            if-feature stateful;
            description
                "If stateful PCE feature is present";
            leaf enabled{
                type boolean;
                description
                    "Enabled or Disabled";
            }
            leaf active{
                type boolean;
                description
                  "Support for active stateful PCE.";
            }
            leaf pce-initiated{
                type boolean;
                description
                  "Support for PCE-initiated LSP.";
            }
          }
     }//capability

     grouping  info{
         description
            "This grouping specifies all information which
             maybe relevant to both PCC and PCE.
             This information corresponds to PCE auto-discovery
             information.";
         container domain{
             description
                "The local domain for the PCEP entity";
             list domain{
                 key "domain-type domain";
                                 description
                     "The local domain.";
                                    uses domain{
                                            description
                                               "The local domain for the PCEP e
ntity.";
                                    }
             }
         }
         container capability{
             description
                "The PCEP entity capability";
             uses capability{
```

```
                   description
                       "The PCEP entity supported
                       capabilities.";
               }
           }
       }//info

       grouping  pce-info{
           description
               "This grouping specifies all PCE information
                which maybe relevant to the PCE selection.
                This information corresponds to PCE auto-discovery
                information.";
           container scope{
               description
                   "The path computation scope";
               uses pce-scope;
           }

           container neigh-domains{
               description
                   "The list of neighbour PCE-Domain
                    toward which a PCE can compute
                    paths";
               list domain{
                   key "domain-type domain";

                   description
                       "The neighbour domain.";
                   uses domain{
                       description
                           "The PCE neighbour domain.";
                   }
               }
           }
       }//pce-info

       grouping  pcep-stats{
           description
               "This grouping defines statistics for PCEP. It is used
                for both peer and current session.";
           leaf avg-rsp-time{
               type uint32;
               units "milliseconds";
               must "(/pcep-state/entity/peers/peer/role != 'pcc'" +
                   " or " +
                   "(/pcep-state/entity/peers/peer/role = 'pcc'" +
                   " and avg-rsp-time = 0))" {
```

```
            error-message
                "Invalid average response time";
            description
                "If role is pcc then this leaf is meaningless
                 and is set to zero.";
        }
        description
          "The average response time.

           If an average response time has not been
           calculated then this leaf has the value zero.";
    }

    leaf lwm-rsp-time{
        type uint32;
        units "milliseconds";
        must "(/pcep-state/entity/peers/peer/role != 'pcc'" +
             " or " +
             "(/pcep-state/entity/peers/peer/role = 'pcc'" +
             " and lwm-rsp-time = 0))" {
            error-message
                "Invalid smallest (low-water mark)
                 response time";
            description
                "If role is pcc then this leaf is meaningless
                 and is set to zero.";
        }
        description
         "The smallest (low-water mark) response time seen.

          If no responses have been received then this
          leaf has the value zero.";
    }

    leaf hwm-rsp-time{
        type uint32;
        units "milliseconds";
        must "(/pcep-state/entity/peers/peer/role != 'pcc'" +
             " or " +
             "(/pcep-state/entity/peers/peer/role = 'pcc'" +
             " and hwm-rsp-time = 0))" {
            error-message
                "Invalid greatest (high-water mark)
                 response time seen";
            description
                "If role is pcc then this field is
                 meaningless and is set to zero.";
        }
```

```
        description
         "The greatest (high-water mark) response time seen.

          If no responses have been received then this object
          has the value zero.";
    }

    leaf num-pcreq-sent{
        type yang:counter32;
        description
         "The number of PCReq messages sent.";
    }

    leaf num-pcreq-rcvd{
        type yang:counter32;
        description
          "The number of PCReq messages received.";
    }

    leaf num-pcrep-sent{
        type yang:counter32;
        description
          "The number of PCRep messages sent.";
    }

    leaf num-pcrep-rcvd{
        type yang:counter32;
        description
          "The number of PCRep messages received.";
    }

    leaf num-pcerr-sent{
        type yang:counter32;
        description
          "The number of PCErr messages sent.";
    }

    leaf num-pcerr-rcvd{
        type yang:counter32;
        description
          "The number of PCErr messages received.";
    }

    leaf num-pcntf-sent{
        type yang:counter32;
        description
          "The number of PCNtf messages sent.";
    }
```

```
        leaf num-pcntf-rcvd{
            type yang:counter32;
            description
              "The number of PCNtf messages received.";
        }

        leaf num-keepalive-sent{
            type yang:counter32;
            description
              "The number of Keepalive messages sent.";
        }

        leaf num-keepalive-rcvd{
            type yang:counter32;
            description
              "The number of Keepalive messages received.";
        }

        leaf num-unknown-rcvd{
            type yang:counter32;
            description
              "The number of unknown messages received.";
        }

        leaf num-corrupt-rcvd{
            type yang:counter32;
            description
              "The number of corrupted PCEP message received.";
        }

        leaf num-req-sent{
            type yang:counter32;
            description
              "The number of requests sent.  A request corresponds
               1:1 with an RP object in a PCReq message. This might
               be greater than num-pcreq-sent because multiple
               requests can be batched into a single PCReq
               message.";
        }

        leaf num-req-sent-pend-rep{
            type yang:counter32;
            description
              "The number of requests that have been sent for
               which a response is still pending.";
        }

        leaf num-req-sent-ero-rcvd{
```

```
        type yang:counter32;
        description
          "The number of requests that have been sent for
           which a response with an ERO object was received.
           Such responses indicate that a path was
           successfully computed by the peer.";
    }

    leaf num-req-sent-nopath-rcvd{
        type yang:counter32;
        description
          "The number of requests that have been sent for
           which a response with a NO-PATH object was
           received. Such responses indicate that the peer
           could not  find a path to satisfy the
           request.";
    }

    leaf num-req-sent-cancel-rcvd{
        type yang:counter32;
        description
          "The number of requests that were cancelled with
           a PCNtf message.

           This might be different than num-pcntf-rcvd because
           not all PCNtf messages are used to cancel requests,
           and a single PCNtf message can cancel multiple
           requests.";
    }

    leaf num-req-sent-error-rcvd{
        type yang:counter32;
        description
          "The number of requests that were rejected with a
           PCErr message.

           This might be different than num-pcerr-rcvd because
           not all PCErr messages are used to reject requests,
           and a single PCErr message can reject multiple
           requests.";
    }

    leaf num-req-sent-timeout{
        type yang:counter32;
        description
          "The number of requests that have been sent to a peer
           and have been abandoned because the peer has taken too
           long to respond to them.";
```

```
        }

        leaf num-req-sent-cancel-sent{
            type yang:counter32;
            description
             "The number of requests that were sent to the peer and
              explicitly cancelled by the local PCEP entity sending
              a PCNtf.";
        }

        leaf num-req-rcvd{
            type yang:counter32;
            description
             "The number of requests received.  A request
              corresponds 1:1 with an RP object in a PCReq
              message.

              This might be greater than num-pcreq-rcvd because
              multiple requests can be batched into a single
              PCReq message.";
        }

        leaf num-req-rcvd-pend-rep{
            type yang:counter32;
            description
             "The number of requests that have been received for
              which a response is still pending.";
        }

        leaf num-req-rcvd-ero-sent{
            type yang:counter32;
            description
             "The number of requests that have been received for
              which a response with an ERO object was sent.  Such
              responses indicate that a path was successfully
              computed by the local PCEP entity.";
        }

        leaf num-req-rcvd-nopath-sent{
            type yang:counter32;
            description
             "The number of requests that have been received for
              which a response with a NO-PATH object was sent. Such
              responses indicate that the local PCEP entity could
              not find a path to satisfy the request.";
        }

        leaf num-req-rcvd-cancel-sent{
```

```
        type yang:counter32;
        description
         "The number of requests received that were cancelled
          by the local PCEP entity sending a PCNtf message.

          This might be different than num-pcntf-sent because
          not all PCNtf messages are used to cancel requests,
          and a single PCNtf message can cancel multiple
          requests.";
    }

    leaf num-req-rcvd-error-sent{
        type yang:counter32;
        description
         "The number of requests received that were cancelled
          by the local PCEP entity sending a PCErr message.

          This might be different than num-pcerr-sent because
          not all PCErr messages are used to cancel requests,
          and a single PCErr message can cancel multiple
          requests.";
    }

    leaf  num-req-rcvd-cancel-rcvd{
        type yang:counter32;
        description
         "The number of requests that were received from the
          peer and explicitly cancelled by the peer sending
          a PCNtf.";
    }

    leaf  num-rep-rcvd-unknown{
        type yang:counter32;
        description
          "The number of responses to unknown requests
           received. A response to an unknown request is a
           response whose RP object does not contain the
           request ID of any request that is currently
           outstanding on the session.";
    }

    leaf  num-req-rcvd-unknown{
        type yang:counter32;
        description
          "The number of unknown requests that have been
           received. An unknown request is a request
           whose RP object contains a request ID of
           zero.";
```

```
            }

        container svec{
            if-feature svec;
            description
                "If synchronized path computation is supported";
            leaf num-svec-sent{
                type yang:counter32;
                description
                  "The number of SVEC objects sent in PCReq messages.
                   An SVEC object represents a set of synchronized
                   requests.";
            }

            leaf num-svec-req-sent{
                type yang:counter32;
                description
                  "The number of requests sent that appeared in one
                   or more SVEC objects.";
            }

            leaf num-svec-rcvd{
                type yang:counter32;
                description
                 "The number of SVEC objects received in PCReq
                  messages. An SVEC object represents a set of
                  synchronized requests.";
            }

            leaf num-svec-req-rcvd{
                type yang:counter32;
                description
                  "The number of requests received that appeared
                   in one or more SVEC objects.";
            }
        }
        container stateful{
            if-feature stateful;
            description
                "Stateful PCE related statistics";
            leaf num-pcrpt-sent{
                type yang:counter32;
                description
                 "The number of PCRpt messages sent.";
            }

            leaf num-pcrpt-rcvd{
                type yang:counter32;
```

```
            description
              "The number of PCRpt messages received.";
        }

        leaf num-pcupd-sent{
            type yang:counter32;
            description
             "The number of PCUpd messages sent.";
        }

        leaf num-pcupd-rcvd{
            type yang:counter32;
            description
              "The number of PCUpd messages received.";
        }

        leaf num-rpt-sent{
            type yang:counter32;
            description
              "The number of LSP Reports sent.  A LSP report
               corresponds 1:1 with an LSP object in a PCRpt
               message. This might be greater than
               num-pcrpt-sent because multiple reports can
               be batched into a single PCRpt message.";
        }

        leaf num-rpt-rcvd{
            type yang:counter32;
            description
             "The number of LSP Reports received.  A LSP report
              corresponds 1:1 with an LSP object in a PCRpt
              message.

              This might be greater than num-pcrpt-rcvd because
              multiple reports can be batched into a single
              PCRpt message.";
        }

        leaf  num-rpt-rcvd-error-sent{
            type yang:counter32;
            description
              "The number of reports of LSPs received that were
               responded by the local PCEP entity by sending a
               PCErr message.";
        }

        leaf num-upd-sent{
            type yang:counter32;
```

```
          description
            "The number of LSP updates sent.  A LSP update
             corresponds 1:1 with an LSP object in a PCUpd
             message. This might be greater than
             num-pcupd-sent because multiple updates can
             be batched into a single PCUpd message.";
          }

          leaf num-upd-rcvd{
              type yang:counter32;
              description
               "The number of LSP Updates received.  A LSP update
                corresponds 1:1 with an LSP object in a PCUpd
                message.

                This might be greater than num-pcupd-rcvd because
                multiple updates can be batched into a single
                PCUpd message.";
          }

          leaf  num-upd-rcvd-unknown{
              type yang:counter32;
              description
                "The number of updates to unknown LSPs
                 received. An update to an unknown LSP is a
                 update whose LSP object does not contain the
                 PLSP-ID of any LSP that is currently
                 present.";
          }

          leaf  num-upd-rcvd-undelegated{
              type yang:counter32;
              description
                "The number of updates to not delegated LSPs
                 received. An update to an undelegated LSP is a
                 update whose LSP object does not contain the
                 PLSP-ID of any LSP that is currently
                 delegated to current PCEP session.";
          }

          leaf  num-upd-rcvd-error-sent{
              type yang:counter32;
              description
                "The number of updates to LSPs received that were
                 responded by the local PCEP entity by sending a
                 PCErr message.";
          }
        }
```

```
    }//pcep-stats

    grouping lsp-state{
        description
            "This grouping defines the attributes for LSP in LSP-DB.
             These are the attributes specifically from the PCEP
             perspective";
        leaf plsp-id{
            type uint32{
                range "1..1048575";
            }
            description
                "A PCEP-specific identifier for the LSP.  A PCC
                 creates a unique PLSP-ID for each LSP that is
                 constant for the lifetime of a PCEP session.
                 PLSP-ID is 20 bits with 0 and 0xFFFFF are
                 reserved";
        }
        leaf pcc-id{
            type inet:ip-address;
            description
                "The local internet address of the PCC, that
                 generated the PLSP-ID.";
        }

        leaf admin-state{
            type boolean;
            description
                "The desired operational state";
        }
        leaf operational-state{
            type operational-state;
            description
                "The operational status of the LSP";
        }
        container delegated{
            description
                "The delegation related parameters";
            leaf enabled{
                type boolean;
                description
                    "LSP is delegated or not";
            }
            leaf pce{
                type leafref {
                    path "/pcep-state/entity/peers/peer/addr";
                }
                must "((../enabled == true)" +
```

```
                     " and " +
                     "((../../role == 'pcc')" +
                     " or " +
                     "(../../role == 'pcc-and-pce')))"
                {
                     error-message
                         "The PCEP entity must be PCC
                          and the LSP be delegated";
                     description
                         "When PCEP entity is PCC for
                         delegated LSP";
                }
                description
                     "The reference to the PCE peer to
                     which LSP is delegated";
             }
             leaf srp-id{
                 type uint32;
                 description
                     "The last SRP-ID-number associated with this
                     LSP.";
             }
        }
        leaf symbolic-path-name{
             type string;
             description
                 "The symbolic path name associated with the LSP.";
        }
        leaf last-error{
             type lsp-error;
             description
                 "The last error for the LSP.";
        }
    }//lsp-state

    grouping notification-instance-hdr {
        description
            "This group describes common instance specific data
             for notifications.";

        leaf entity-addr {
             type leafref {
                 path "/pcep-state/entity/addr";
             }
             description
                 "Reference to local entity address";
        }
```

```
        leaf peer-addr {
            type leafref {
                path "/pcep-state/entity/peers/peer/addr";
            }
            description
                "Reference to peer address";
        }

    }// notification-instance-hdr

    grouping notification-session-hdr {
        description
        "This group describes common session instance specific
         data for notifications.";

        leaf session-initiator {
            type leafref {
            path "/pcep-state/entity/peers/peer/sessions/" +
                "session/initiator";
            }
            description
                "Reference to pcep session initiator leaf";
        }
    }// notification-session-hdr

    /*
     * Configuration data nodes
     */
    container pcep{

        description
            "Parameters for list of configured PCEP entities
             on the device.";

        container entity {

            description
                "The configured PCEP entity on the device.";

            leaf addr {
                type inet:ip-address;
                mandatory true;
                description
                    "The local Internet address of this PCEP
                    entity.

                    If operating as a PCE server, the PCEP
                    entity listens on this address.
```

```
                    If operating as a PCC, the PCEP entity
                    binds outgoing TCP connections to this
                    address.

                    It is possible for the PCEP entity to
                    operate both as a PCC and a PCE Server, in
                    which case it uses this address both to
                    listen for incoming TCP connections and to
                    bind outgoing TCP connections.";
            }

            leaf enabled {
                type boolean;
                default true;
                description
                    "The administrative status of this PCEP
                     Entity.";
            }

            leaf role {
                type pcep-role;
                mandatory true;
                description
                    "The role that this entity can play.
                     Takes one of the following values.
                     - unknown(0): this PCEP Entity role is not
                       known.
                     - pcc(1): this PCEP Entity is a PCC.
                     - pce(2): this PCEP Entity is a PCE.
                     - pcc-and-pce(3): this PCEP Entity is both
                       a PCC and a PCE.";
            }

            leaf description {
                type string;
                description
                    "Description of the PCEP entity configured
                     by the user";
            }

            uses info {
                description
                    "Local PCEP entity information";
            }


            container pce-info {
                must "((../role == 'pce')" +
```

```
                " or " +
                "(../role == 'pcc-and-pce'))"
          {
               error-message "The PCEP entity must be PCE";
               description
                   "When PCEP entity is PCE";
          }
          uses pce-info {
               description
                   "Local PCE information";
          }
          description
               "The Local PCE Entity PCE information";
      }

      uses pcep-entity-info {
          description
              "The configuration related to the PCEP
               entity.";
      }



      leaf pcep-notification-max-rate {
          type uint32;
          mandatory true;
          description
              "This variable indicates the maximum number of
               notifications issued per second. If events occur
               more rapidly, the implementation may simply fail
               to emit these notifications during that period,
               or may queue them until an appropriate time. A
               value of 0 means no notifications are emitted
               and all should be discarded (that is, not
               queued).";
      }

      container stateful-timer{
          if-feature stateful;
          must "(../info/capability/stateful/active == true)"
          {
               error-message
                   "The Active Stateful PCE must be enabled";
               description
                   "When PCEP entity is active stateful
                    enabled";
          }
          leaf state-timeout{
```

```
                    type uint32;
                    units "seconds";
                    description
                        "When a PCEP session is terminated, a PCC
                         waits for this time period before flushing
                         LSP state associated with that PCEP session
                         and reverting to operator-defined default
                         parameters or behaviours.";
                }
                leaf redelegation-timeout{
                    type uint32;
                    units "seconds";
                    must "((../role == 'pcc')" +
                        " or " +
                        "(../role == 'pcc-and-pce'))"
                    {
                        error-message "The PCEP entity must be PCC";
                        description
                            "When PCEP entity is PCC";
                    }
                    description
                        "When a PCEP session is terminated, a PCC
                         waits for this time period before revoking
                         LSP delegation to a PCE and attempting to
                         redelegate LSPs associated with the
                         terminated PCEP session to an alternate
                         PCE.";
                }
                leaf rpt-non-pcep-lsp{
                    type boolean;
                    must "((../role == 'pcc')" +
                        " or " +
                        "(../role == 'pcc-and-pce'))"
                    {
                        error-message "The PCEP entity must be PCC";
                        description
                            "When PCEP entity is PCC";
                    }
                    description
                        "If set, a PCC reports LSPs that are not
                         controlled by any PCE (for example, LSPs
                         that are statically configured at the
                         PCC). ";
                }
                description
                    "The configured stateful parameters";
            }
```

```
            container peers{
                must "((../role == 'pcc')" +
                     " or " +
                     "(../role == 'pcc-and-pce'))"
                {
                    error-message
                        "The PCEP entity must be PCC";
                    description
                        "When PCEP entity is PCC, as remote
                         PCE peers are configured.";
                }
                description
                    "The list of configured peers for the
                     entity (remote PCE)";
                list peer{
                    key "addr";

                    description
                        "The peer configured for the entity.
                         (remote PCE)";

                    leaf addr {
                        type inet:ip-address;
                        description
                            "The local Internet address of this
                             PCEP peer.";
                    }

                    leaf description {
                        type string;
                        description
                            "Description of the PCEP peer
                             configured by the user";
                    }
                    uses info {
                        description
                            "PCE Peer information";
                    }
                    uses pce-info {
                        description
                            "PCE Peer information";
                    }

                    leaf delegation-pref{
                        if-feature stateful;
                        type uint8{
                            range "0..7";
                        }
```

```
                        must "(../../info/capability/stateful/active"
                             + "== true)"
                        {
                            error-message
                                "The Active Stateful PCE must be
                                 enabled";
                            description
                                "When PCEP entity is active stateful
                                 enabled";
                        }
                        description
                            "The PCE peer delegation preference.";
                    }
                }//peer
            }//peers
        }//entity
    }//pcep

    /*
     * Operational data nodes
     */

    container pcep-state{
        config false;
        description
            "The list of operational PCEP entities on the
             device.";

        container entity{
            description
                "The operational PCEP entity on the device.";

            leaf addr {
                type inet:ip-address;
                description
                    "The local Internet address of this PCEP
                    entity.

                    If operating as a PCE server, the PCEP
                    entity listens on this address.

                    If operating as a PCC, the PCEP entity
                    binds outgoing TCP connections to this
                    address.

                    It is possible for the PCEP entity to
                    operate both as a PCC and a PCE Server, in
                    which case it uses this address both to
```

```
                        listen for incoming TCP connections and to
                        bind outgoing TCP connections.";
            }

            leaf index{
                type uint32;
                description
                    "The index of the operational PECP
                     entity";
            }


            leaf admin-status {
                type pcep-admin-status;
                description
                    "The administrative status of this PCEP Entity.
                     This is the desired operational status as
                     currently set by an operator or by default in
                     the implementation.  The value of enabled
                     represents the current status of an attempt
                     to reach this desired status.";
            }

            leaf oper-status {
                type pcep-admin-status;
                description
                  "The operational status of the PCEP entity.
                   Takes one of the following values.
                   - oper-status-up(1): the PCEP entity is
                     active.
                   - oper-status-down(2): the PCEP entity is
                     inactive.
                   - oper-status-going-up(3): the PCEP entity is
                     activating.
                   - oper-status-going-down(4): the PCEP entity is
                     deactivating.
                   - oper-status-failed(5): the PCEP entity has
                     failed and will recover when possible.
                   - oper-status-failed-perm(6): the PCEP entity
                     has failed and will not recover without
                     operator intervention.";
            }

            leaf role {
                type pcep-role;
                description
                    "The role that this entity can play.
                     Takes one of the following values.
```

```
                    - unknown(0): this PCEP entity role is
                      not known.
                    - pcc(1): this PCEP entity is a PCC.
                    - pce(2): this PCEP entity is a PCE.
                    - pcc-and-pce(3): this PCEP entity is
                      both a PCC and a PCE.";
            }

            uses info {
                description
                    "Local PCEP entity information";
            }

            container pce-info {
                when "((../role == 'pce')" +
                     " or " +
                     "(../role == 'pcc-and-pce'))"
                {
                    description
                        "When PCEP entity is PCE";
                }
                uses pce-info {
                    description
                        "Local PCE information";
                }
                description
                    "The Local PCE Entity PCE information";
            }

            uses pcep-entity-info{
                description
                    "The operational information related to the
                     PCEP entity.";
            }

            container stateful-timer{
                if-feature stateful;
                must "(../info/capability/stateful/active == true)"
                {
                    error-message
                        "The Active Stateful PCE must be enabled";
                    description
                        "When PCEP entity is active stateful
                         enabled";
                }
                leaf state-timeout{
                    type uint32;
                    units "seconds";
```

```
                    description
                        "When a PCEP session is terminated, a PCC
                         waits for this time period before flushing
                         LSP state associated with that PCEP session
                         and reverting to operator-defined default
                         parameters or behaviours.";
                }
                leaf redelegation-timeout{
                    type uint32;
                    units "seconds";
                    must "((../role == 'pcc')" +
                         " or " +
                         "(../role == 'pcc-and-pce'))"
                    {
                        error-message "The PCEP entity must be PCC";
                        description
                            "When PCEP entity is PCC";
                    }
                    description
                        "When a PCEP session is terminated, a PCC
                         waits for this time period before revoking
                         LSP delegation to a PCE and attempting to
                         redelegate LSPs associated with the
                         terminated PCEP session to an alternate
                         PCE.";
                }
                description
                    "The operational stateful timer values";
            }


            container lsp-db{
                if-feature stateful;
                description
                    "The LSP-DB";
                list lsp{
                    key "plsp-id pcc-id";
                    description
                        "List of all LSPs in LSP-DB";
                    uses lsp-state{
                        description
                            "The PCEP specific attributes for
                             LSP-DB.";
                    }
                    // To Do - add groupings and useful information
                    // from TE yang model, once ready
                }
            }
```

```
            container peers{
                description
                    "The list of peers for the entity";

            list peer{
                key "addr";

                description
                    "The peer for the entity.";

                leaf addr {
                    type inet:ip-address;
                    description
                        "The local Internet address of this PCEP
                         peer.";
                }

                leaf role {
                    type pcep-role;
                    description
                        "The role of the PCEP Peer.
                         Takes one of the following values.
                         - unknown(0): this PCEP peer role
                           is not known.
                         - pcc(1): this PCEP peer is a PCC.
                         - pce(2): this PCEP peer is a PCE.
                         - pcc-and-pce(3): this PCEP peer
                           is both a PCC and a PCE.";
                }


                uses info {
                    description
                        "PCEP peer information";
                }


                container pce-info {
                    when "((../role == 'pce')" +
                    " or " +
                    "(../role == 'pcc-and-pce'))"
                    {
                        description
                            "When PCEP entity is PCE";
                    }
                    uses pce-info {
                        description
                            "PCE Peer information";
```

```
                        }
                    description
                        "The PCE Peer information";
                    }

                    leaf delegation-pref{
                        if-feature stateful;
                        type uint8{
                            range "0..7";
                        }
                        must "((../../role == 'pcc')" +
                            " or " +
                            "(../../role == 'pcc-and-pce'))"
                        {
                            error-message
                                "The PCEP entity must be PCC";
                            description
                                "When PCEP entity is PCC";
                        }
                        must "(../../info/capability/stateful/active"
                            + " == true)"
                        {
                            error-message
                                "The Active Stateful PCE must be
                                 enabled";
                            description
                                "When PCEP entity is active stateful
                                 enabled";
                        }
                        description
                            "The PCE peer delegation preference.";
                    }
                    leaf discontinuity-time {
                        type yang:timestamp;
                        description
                            "The timestamp of the time when the
                             information and statistics were
                             last reset.";
                    }

                    leaf initiate-session {
                        type boolean;
                        description
                            "Indicates whether the local PCEP
                             entity initiates sessions to this peer,
                             or waits for the peer to initiate a
                             session.";
                    }
```

```
                    leaf session-exists{
                        type boolean;
                        description
                            "Indicates whether a session with
                             this peer currently exists.";
                    }

                    leaf num-sess-setup-ok{
                        type yang:counter32;
                        description
                            "The number of PCEP sessions successfully
                             successfully established with the peer,
                             including any current session.  This
                             counter is incremented each time a
                             session with this peer is successfully
                             established.";
                    }

                    leaf num-sess-setup-fail{
                        type yang:counter32;
                        description
                           "The number of PCEP sessions with the peer
                            that have been attempted but failed
                            before being fully established. This
                            counter is incremented each time a
                            session retry to this peer fails.";
                    }

                    leaf session-up-time{
                        type yang:timestamp;
                        must "(../num-sess-setup-ok != 0  or " +
                            "(../num-sess-setup-ok = 0  and "  +
                            "session-up-time = 0))" {
                                error-message
                                    "Invalid Session Up timestamp";
                                description
                                    "If num-sess-setup-ok is zero,
                                     then this leaf contains zero.";
                            }
                        description
                           "The timestamp value of the last time a
                            session with this peer was successfully
                            established.";
                    }

                    leaf session-fail-time{
                        type yang:timestamp;
                        must "(../num-sess-setup-fail != 0  or " +
```

```
                    "(../num-sess-setup-fail = 0  and "  +
                    "session-fail-time = 0))" {
                        error-message
                            "Invalid Session Fail timestamp";
                        description
                            "If num-sess-setup-fail is zero,
                             then this leaf contains zero.";
                    }
                description
                  "The timestamp value of the last time a
                   session with this peer failed to be
                   established.";
            }

            leaf session-fail-up-time{
                type yang:timestamp;
                must "(../num-sess-setup-ok != 0  or "      +
                    "(../num-sess-setup-ok = 0  and "      +
                    "session-fail-up-time = 0))" {
                        error-message
                            "Invalid Session Fail from
                             Up timestamp";
                        description
                            "If num-sess-setup-ok is zero,
                             then this leaf contains zero.";
                    }
                description
                  "The timestamp value of the last time a
                   session with this peer failed from
                   active.";
            }

            uses pcep-stats{
                description
                    "Since PCEP sessions can be ephemeral,
                     the peer statistics tracks a peer even
                     when no PCEP session currently exists
                     to that peer. The statistics contained
                     are an aggregate of the statistics for
                     all successive sessions to that peer.";
            }

            leaf num-req-sent-closed{
                type yang:counter32;
                description
                    "The number of requests that were sent
                     to the peer and implicitly cancelled
                     when the session they were sent over
```

```
                              was closed.";
                    }

                    leaf  num-req-rcvd-closed{
                        type yang:counter32;
                        description
                            "The number of requests that were
                             received from the peer and implicitly
                             cancelled when the session they were
                             received over was closed.";
                    }



                    container sessions {
                        description
                            "This entry represents a single PCEP
                             session in which the local PCEP entity
                             participates.

                             This entry exists only if the
                             corresponding PCEP session has been
                             initialized by some event, such as
                             manual user configuration, auto-
                             discovery of a peer, or an incoming
                             TCP connection.";

                        list session {
                            key "initiator";

                            description
                                "The list of sessions, note that
                                 for a time being two sessions
                                 may exist for a peer";

                            leaf initiator {
                                type pcep-initiator;
                                description
                                    "The initiator of the session,
                                     that is, whether the TCP
                                     connection was initiated by
                                     the local PCEP entity or the
                                     peer.

                                     There is a window during
                                     session initialization where
                                     two sessions can exist between
```

```
                        a pair of PCEP speakers, each
                        initiated by one of the
                        speakers. One of these
                        sessions is always discarded
                        before it leaves OpenWait state.
                        However, before it is discarded,
                        two sessions to the given peer
                        appear transiently in this MIB
                        module. The sessions are
                        distinguished by who initiated
                        them, and so this field is the
                        key.";
            }

            leaf state-last-change {
                type yang:timestamp;
                description
                    "The timestamp value at the
                    time this session entered its
                    current state as denoted by
                    the state leaf.";
            }

            leaf state {
                type pcep-sess-state;
                description
                    "The current state of the
                    session.

                    The set of possible states
                    excludes the idle state since
                    entries do not exist in the
                    idle state.";
            }

            leaf connect-retry {
                type yang:counter32;
                description
                    "The number of times that the
                     local PCEP entity has
                     attempted to establish a TCP
                     connection for this session
                     without success. The PCEP
                     entity gives up when this
                     reaches connect-max-retry.";
            }

            leaf local-id {
```

```
                            type uint32 {
                              range "0..255";
                            }
                            description
                                "The value of the PCEP session
                                 ID used by the local PCEP
                                 entity in the Open message
                                 for this session.

                                 If state is tcp-pending then
                                 this is the session ID that
                                 will be used in the Open
                                 message. Otherwise, this is
                                 the session ID that was sent
                                 in the Open message.";
                        }

                        leaf remote-id {
                            type uint32 {
                                range "0..255";
                            }
                            must "((../state != 'tcp-pending'" +
                                "and " +
                                "../state != 'open-wait' )" +
                                "or " +
                                "((../state = 'tcp-pending'" +
                                " or " +
                                "../state = 'open-wait' )" +
                                "and remote-id = 0))" {
                                    error-message
                                        "Invalid remote-id";
                                    description
                                        "If state is tcp-pending
                                         or open-wait then this
                                         leaf is not used and
                                         MUST be set to zero.";
                                }
                            description
                                "The value of the PCEP session
                                 ID used by the peer in its
                                 Open message for this
                                 session.";
                        }

                        leaf keepalive-timer {
                            type uint32 {
                              range "0..255";
                            }
```

```
                        units "seconds";
                        must "(../state = 'session-up'" +
                            "or " +
                            "(../state != 'session-up'" +
                            "and keepalive-timer = 0))" {
                               error-message
                                   "Invalid keepalive
                                    timer";
                               description
                                   "This field is used if
                                    and only if state is
                                    session-up. Otherwise,
                                    it is not used and
                                    MUST be set to
                                    zero.";
                           }
                        description
                            "The agreed maximum interval at
                             which the local PCEP entity
                             transmits PCEP messages on this
                             PCEP session.  Zero means that
                             the local PCEP entity never
                             sends Keepalives on this
                             session.";
                    }

                    leaf peer-keepalive-timer {
                        type uint32 {
                          range "0..255";
                        }
                        units "seconds";
                        must "(../state = 'session-up'" +
                            "or "    +
                            "(../state != 'session-up'" +
                            "and " +
                            "peer-keepalive-timer = 0))" {
                               error-message
                                   "Invalid Peer keepalive
                                    timer";
                               description
                                   "This field is used if
                                    and only if state is
                                    session-up. Otherwise,
                                    it is not used and MUST
                                    be set to zero.";
                           }
                        description
                            "The agreed maximum interval at
```

```
                                    which the peer transmits PCEP
                                    messages on this PCEP session.
                                    Zero means that the peer never
                                    sends Keepalives on this
                                    session.";
                    }

                    leaf dead-timer {
                        type uint32 {
                          range "0..255";
                        }
                        units "seconds";
                        description
                            "The dead timer interval for
                             this PCEP session.";
                    }

                    leaf peer-dead-timer {
                        type uint32 {
                          range "0..255";
                        }
                        units "seconds";
                        must "((../state != 'tcp-pending'" +
                            "and " +
                            "../state != 'open-wait' )" +
                            "or " +
                            "((../state = 'tcp-pending'" +
                            " or " +
                            "../state = 'open-wait' )" +
                            "and " +
                            "peer-dead-timer = 0))" {
                               error-message
                                   "Invalid Peer Dead
                                    timer";
                               description
                                   "If state is tcp-
                                    pending or open-wait
                                    then this leaf is not
                                    used and MUST be set to
                                    zero.";
                        }
                        description
                            "The peer's dead-timer interval
                             for this PCEP session.";
                    }

                    leaf ka-hold-time-rem {
                        type uint32 {
```

```
                            range "0..255";
                         }
                         units "seconds";
                         must "((../state != 'tcp-pending'" +
                             "and " +
                             "../state != 'open-wait' ) " +
                             "or " +
                             "((../state = 'tcp-pending'" +
                             "or " +
                             "../state = 'open-wait' )" +
                             "and " +
                             "ka-hold-time-rem = 0))" {
                                error-message
                                    "Invalid Keepalive hold
                                     time remaining";
                                description
                                    "If state is tcp-pending
                                     or open-wait then this
                                     field is not used and
                                     MUST be set to zero.";
                            }
                         description
                             "The keep alive hold time
                              remaining for this session.";
                     }

                     leaf overloaded {
                         type boolean;
                         description
                             "If the local PCEP entity has
                              informed the peer that it is
                              currently overloaded, then this
                              is set to true.  Otherwise, it
                              is set to false.";
                     }

                     leaf overload-time {
                         type uint32;
                         units "seconds";
                         must "(../overloaded = true or" +
                             "(../overloaded != true and" +
                             " overload-time = 0))" {
                                error-message
                                    "Invalid overload-time";
                                description
                                    "This field is only used
                                     if overloaded is set to
                                     true. Otherwise, it is
```

```
                                not used and MUST be set
                                to zero.";
                    }
                description
                    "The interval of time that is
                     remaining until the local PCEP
                     entity will cease to be
                     overloaded on this session.";
            }

            leaf peer-overloaded {
                type boolean;
                description
                    "If the peer has informed the
                     local PCEP entity that it is
                     currently overloaded, then this
                     is set to true. Otherwise, it
                     is set to false.";
            }

            leaf peer-overload-time {
                type uint32;
                units "seconds";
                must "(../peer-overloaded = true" +
                    " or " +
                    "(../peer-overloaded != true" +
                    " and " +
                    "peer-overload-time = 0))" {
                        error-message
                            "Invalid peer overload
                             time";
                        description
                            "This field is only used
                             if peer-overloaded is
                             set to true. Otherwise,
                             it is not used and MUST
                             be set to zero.";
                    }
                description
                    "The interval of time that is
                     remaining until the peer will
                     cease to be overloaded.  If it
                     is not known how long the peer
                     will stay in overloaded state,
                     this leaf is set to zero.";
            }
            leaf lspdb-sync {
                if-feature stateful;
```

```
                                type sync-state;
                                description
                                    "The LSP-DB state synchronization
                                    status.";
                            }
                            leaf discontinuity-time {
                                type yang:timestamp;
                                description
                                        "The timestamp value of the time
                                         when the statistics were last
                                         reset.";
                            }

                            uses pcep-stats{
                                description
                                        "The statistics contained are
                                         for the current sessions to that
                                         peer. These are lost when the
                                         session goes down.";

                            }

                        } // session
                    } // sessions
                }//peer
            }//peers
        }//entity
    }//pcep-state

    /*
     * Notifications
     */
    notification pcep-session-up {
        description
            "This notification is sent when the value of
             '/pcep/pcep-state/peers/peer/sessions/session/state'
             enters the 'session-up' state.";

        uses notification-instance-hdr;

        uses notification-session-hdr;

        leaf state-last-change {
            type yang:timestamp;
            description
                "The timestamp value at the time this session entered
                its current state as denoted by the state leaf.";
        }
```

```
        leaf state {
            type pcep-sess-state;
            description
                "The current state of the session.

                 The set of possible states excludes the idle state
                 since entries do not exist in the idle state.";
        }
    } //notification

    notification pcep-session-down {
        description
            "This notification is sent when the value of
             '/pcep/pcep-state/peers/peer/sessions/session/state'
             leaves the 'session-up' state.";

        uses notification-instance-hdr;

        leaf session-initiator {
            type pcep-initiator;
            description
                "The initiator of the session.";
        }

        leaf state-last-change {
            type yang:timestamp;
            description
                "The timestamp value at the time this session entered
                 its current state as denoted by the state leaf.";
        }

        leaf state {
            type pcep-sess-state;
            description
                "The current state of the session.

                 The set of possible states excludes the idle state
                 since entries do not exist in the idle state.";
        }
    } //notification

    notification pcep-session-local-overload {
        description
            "This notification is sent when the local PCEP entity
             enters overload state for a peer.";

        uses notification-instance-hdr;
```

```
        uses notification-session-hdr;

        leaf overloaded {
            type boolean;
            description
                "If the local PCEP entity has informed the peer that
                it is currently overloaded, then this is set to
                true. Otherwise, it is set to false.";
        }

        leaf overload-time {
            type uint32;
            units "seconds";
            must "(../overloaded = true or "     +
                "(../overloaded != true   and "  +
                "overload-time = 0))" {
                    error-message
                        "Invalid overload-time";
                    description
                        "This field is only used if overloaded is
                         set to true. Otherwise, it is not used
                         and MUST be set to zero.";
                }
            description
                "The interval of time that is remaining until the
                local PCEP entity will cease to be overloaded on
                this session.";
        }
    } //notification

    notification pcep-session-local-overload-clear {
        description
            "This notification is sent when the local PCEP entity
            leaves overload state for a peer.";

        uses notification-instance-hdr;

        leaf overloaded {
            type boolean;
            description
                "If the local PCEP entity has informed the peer
                that it is currently overloaded, then this is set
                to true.  Otherwise, it is set to false.";
        }
    } //notification

    notification pcep-session-peer-overload {
        description
```

```
             "This notification is sent when a peer enters overload
             state.";

         uses notification-instance-hdr;

         uses notification-session-hdr;

         leaf peer-overloaded {
             type boolean;
             description
                 "If the peer has informed the local PCEP entity that
                 it is currently overloaded, then this is set to true.
                 Otherwise, it is set to false.";
         }

         leaf peer-overload-time {
             type uint32;
             units "seconds";
             must "(../peer-overloaded = true or "     +
                 "(../peer-overloaded != true  and "  +
                 "peer-overload-time = 0))" {
                     error-message
                         "Invalid peer-overload-time";
                     description
                         "This field is only used if
                          peer-overloaded is set to true.
                          Otherwise, it is not used and MUST
                          be set to zero.";
                 }
             description
                 "The interval of time that is remaining until the
                 peer will cease to be overloaded.  If it is not known
                 how long the peer will stay in overloaded state, this
                 leaf is set to zero.";
         }
     } //notification

    notification pcep-session-peer-overload-clear {
         description
             "This notification is sent when a peer leaves overload
             state.";

         uses notification-instance-hdr;

         leaf peer-overloaded {
             type boolean;
             description
                 "If the peer has informed the local PCEP entity that
```

```
                    it is currently overloaded, then this is set to true.
                    Otherwise, it is set to false.";
            }
      } //notification
}//module
```

<CODE ENDS>

8. Security Considerations

   The YANG module defined in this memo is designed to be accessed via
   the NETCONF protocol [RFC6241]. The lowest NETCONF layer is the
   secure transport layer and the mandatory-to-implement secure
   transport is SSH [RFC6242]. The NETCONF access control model
   [RFC6536] provides the means to restrict access for particular
   NETCONF users to a pre-configured subset of all available NETCONF
   protocol operations and content.

   There are a number of data nodes defined in the YANG module which are
   writable/creatable/deletable (i.e., config true, which is the
   default). These data nodes may be considered sensitive or vulnerable
   in some network environments. Write operations (e.g., <edit-config>)
   to these data nodes without proper protection can have a negative
   effect on network operations.

   TBD: List specific Subtrees and data nodes and their sensitivity/
   vulnerability.

9. Manageability Considerations

9.1. Control of Function and Policy

9.2. Information and Data Models

9.3. Liveness Detection and Monitoring

9.4. Verify Correct Operations

9.5. Requirements On Other Protocols

9.6. Impact On Network Operations

10. IANA Considerations

   This document registers a URI in the "IETF XML Registry" [RFC3688].
   Following the format in RFC 3688, the following registration has been
   made.

URI: urn:ietf:params:xml:ns:yang:ietf-pcep

Registrant Contact: The PCE WG of the IETF.

XML: N/A; the requested URI is an XML namespace.

This document registers a YANG module in the "YANG Module Names" registry [RFC6020].

```
Name:           ietf-pcep
Namespace:      urn:ietf:params:xml:ns:yang:ietf-pcep
Prefix:         pcep
Reference:      This I-D
```

11. Acknowledgements

The initial document is based on the PCEP MIB [RFC7420]. Further this document structure is based on Routing Yang Module [I-D.ietf-netmod-routing-cfg]. We would like to thank the authors of aforementioned documents.

12. References

12.1. Normative References

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

[RFC3688]   Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.

[RFC5440]   Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC6020]   Bjorklund, M., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, October 2010.

[RFC6991]   Schoenwaelder, J., "Common YANG Data Types", RFC 6991, July 2013.

[I-D.ietf-pce-stateful-pce]
            Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-11 (work in progress), April 2015.

12.2. Informative References

   [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
              Element (PCE)-Based Architecture", RFC 4655, August 2006.

   [RFC6241] Enns, R., Bjorklund, M., Schoenwaelder, J., and A.
              Bierman, "Network Configuration Protocol (NETCONF)", RFC
              6241, June 2011.

   [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure
              Shell (SSH)", RFC 6242, June 2011.

   [RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration
              Protocol (NETCONF) Access Control Model", RFC 6536, March
              2012.

   [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
              Hardwick, "Path Computation Element Communication Protocol
              (PCEP) Management Information Base (MIB) Module", RFC
              7420, December 2014.

   [I-D.ietf-netmod-routing-cfg]
              Lhotka, L. and A. Lindem, "A YANG Data Model for Routing
              Management", draft-ietf-netmod-routing-cfg-19 (work in
              progress), May 2015.

   [I-D.ietf-netmod-rfc6087bis]
              Bierman, A., "Guidelines for Authors and Reviewers of YANG
              Data Model Documents", draft-ietf-netmod-rfc6087bis-03
              (work in progress), June 2015.

Appendix A. Contributor Addresses

    Rohit Pobbathi
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560037
    India

    EMail: rohit.pobbathi@huawei.com

    Vinod KumarS
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560037
    India

    EMail: vinods.kumar@huawei.com

    Xufeng Liu
    Ericsson
    1595 Spring Hill Road, Suite 500
    Vienna, VA 22182
    USA

    EMail: xufeng.liu@ericsson.com

    Young Lee
    Huawei Technologies
    5340 Legacy Drive, Building 3
    Plano, TX 75023, USA

    Phone: (469) 277-5838
    EMail: leeyoung@huawei.com

    Udayasree Palle
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560037
    India

    EMail: udayasree.palle@huawei.com

    Xian Zhang
    Huawei Technologies
    Bantian, Longgang District
    Shenzhen 518129
    P.R.China

EMail: zhang.xian@huawei.com

Siva Sivabalan
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada

EMail: msiva@cisco.com

Avantika
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560037
India

EMail: avantika.sushilkumar@huawei.com

Authors' Addresses

Dhruv Dhody (editor)
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560037
India

EMail: dhruv.ietf@gmail.com

Jonathan Hardwick
Metaswitch
100 Church Street
Enfield EN2 6BQ
UK

EMail: jonathan.hardwick@metaswitch.com

Vishnu Pavan Beeram
Juniper Networks
USA

EMail: vbeeram@juniper.net

Zafar Ali
Cisco Systems

EMail: zali@cisco.com

Jeff Tantsura
Ericsson
300 Holger Way
San Jose, CA 95134
US

EMail: Jeff.Tantsura@ericsson.com

PCE Working Group   D. Dhody, Ed.
Internet-Draft   Huawei Technologies
Intended status: Standards Track   J. Hardwick
Expires: January 8, 2017   Metaswitch
                                                        V. Beeram
                                                 Juniper Networks
                                                      J. Tantsura
                                                     July 7, 2016

        A YANG Data Model for Path Computation Element Communications Protocol
                                (PCEP)
                       draft-pkd-pce-pcep-yang-06

Abstract

   This document defines a YANG data model for the management of Path
   Computation Element communications Protocol (PCEP) for communications
   between a Path Computation Client (PCC) and a Path Computation
   Element (PCE), or between two PCEs. The data model includes
   configuration data and state data (status information and counters
   for the collection of statistics).

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on January 8, 2017.

Copyright Notice

publication of this document. Please review these documents
carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   The Path Computation Element (PCE) defined in [RFC4655] is an entity
   that is capable of computing a network path or route based on a
   network graph, and applying computational constraints. A Path

Computation Client (PCC) may make requests to a PCE for paths to be computed.

PCEP is the communication protocol between a PCC and PCE and is defined in [RFC5440]. PCEP interactions include path computation requests and path computation replies as well as notifications of specific states related to the use of a PCE in the context of Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering (TE). [I-D.ietf-pce-stateful-pce] specifies extensions to PCEP to enable stateful control of MPLS TE LSPs.

This document defines a YANG [RFC6020] data model for the management of PCEP speakers. It is important to establish a common data model for how PCEP speakers are identified, configured, and monitored. The data model includes configuration data and state data (status information and counters for the collection of statistics).

This document contains a specification of the PCEP YANG module, "ietf-pcep" which provides the PCEP [RFC5440] data model.

2. 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 [RFC2119].

3. Terminology and Notation

This document uses the terminology defined in [RFC4655] and [RFC5440]. In particular, it uses the following acronyms.

o Path Computation Request message (PCReq).

o Path Computation Reply message (PCRep).

o Notification message (PCNtf).

o Error message (PCErr).

o Request Parameters object (RP).

o Synchronization Vector object (SVEC).

o Explicit Route object (ERO).

This document also uses the following terms defined in [RFC7420]:

o PCEP entity: a local PCEP speaker.

   o PCEP peer: to refer to a remote PCEP speaker.

   o PCEP speaker: where it is not necessary to distinguish between
      local and remote.

   Further, this document also uses the following terms defined in
   [I-D.ietf-pce-stateful-pce] :

   o Stateful PCE, Passive Stateful PCE, Active Stateful PCE

   o Delegation, Revocation, Redelegation

   o LSP State Report, Path Computation Report message (PCRpt).

   o LSP State Update, Path Computation Update message (PCUpd).

   [I-D.ietf-pce-pce-initiated-lsp] :

   o PCE-initiated LSP, Path Computation LSP Initiate Message
      (PCInitiate).

   [I-D.ietf-pce-lsp-setup-type] :

   o Path Setup Type (PST).

   [I-D.ietf-pce-segment-routing] :

   o Segment Routing (SR).

   o Segment Identifier (SID).

   o Maximum SID Depth (MSD).

3.1. Tree Diagrams

   A graphical representation of the complete data tree is presented in
   Section 5. The meaning of the symbols in these diagrams is as
   follows and as per [I-D.ietf-netmod-rfc6087bis]:

   o Brackets "[" and "]" enclose list keys.

   o Curly braces "{" and "}" contain names of optional features that
      make the corresponding node conditional.

   o Abbreviations before data node names: "rw" means configuration
      (read-write), and "ro" state data (read-only).

o Symbols after data node names: "?" means an optional node and "*"
   denotes a "list" or "leaf-list".

o Parentheses enclose choice and case nodes, and case nodes are also
   marked with a colon (":").

o Ellipsis ("...") stands for contents of subtrees that are not
   shown.

3.2. Prefixes in Data Node Names

   In this document, names of data nodes and other data model objects
   are often used without a prefix, as long as it is clear from the
   context in which YANG module each name is defined. Otherwise, names
   are prefixed using the standard prefix associated with the
   corresponding YANG module, as shown in Table 1.

```
+--------+-----------------+-----------+
| Prefix | YANG module     | Reference |
+--------+-----------------+-----------+
| yang   | ietf-yang-types | [RFC6991] |
| inet   | ietf-inet-types | [RFC6991] |
+--------+-----------------+-----------+
```

                Table 1: Prefixes and corresponding YANG modules

4. Objectives

   This section describes some of the design objectives for the model:

   o In case of existing implementations, it needs to map the data
      model defined in this document to their proprietary native data
      model. To facilitate such mappings, the data model should be
      simple.

   o The data model should be suitable for new implementations to use
      as is.

   o Mapping to the PCEP MIB Module should be clear.

   o The data model should allow for static configurations of peers.

   o The data model should include read-only counters in order to
      gather statistics for sent and received PCEP messages, received
      messages with errors, and messages that could not be sent due to
      errors.

   o It should be fairly straightforward to augment the base data model
      for advanced PCE features.

5. The Design of PCEP Data Model

   The module, "ietf-pcep", defines the basic components of a PCE
   speaker.

```
module: ietf-pcep
   +--rw pcep!
   |  +--rw entity
   |     +--rw addr                           inet:ip-address
   |     +--rw enabled?                       boolean
   |     +--rw role                           pcep-role
   |     +--rw description?                   string
   |     +--rw domain
   |     |  +--rw domain* [domain-type domain]
   |     |     +--rw domain-type   domain-type
   |     |     +--rw domain        domain
   |     +--rw capability
   |     |  +--rw gmpls?                boolean {gmpls}?
   |     |  +--rw bi-dir?               boolean
   |     |  +--rw diverse?              boolean
   |     |  +--rw load-balance?         boolean
   |     |  +--rw synchronize?          boolean {svec}?
   |     |  +--rw objective-function?   boolean {obj-fn}?
   |     |  +--rw add-path-constraint?  boolean
   |     |  +--rw prioritization?       boolean
   |     |  +--rw multi-request?        boolean
   |     |  +--rw gco?                  boolean {gco}?
   |     |  +--rw p2mp?                 boolean {p2mp}?
   |     |  +--rw stateful {stateful}?
   |     |  |  +--rw enabled?        boolean
   |     |  |  +--rw active?         boolean
   |     |  |  +--rw pce-initiated?  boolean {pce-initiated}?
   |     |  +--rw sr {sr}?
   |     |     +--rw enabled?   boolean
   |     |     +--rw msd?       uint8
   |     +--rw pce-info
   |     |  +--rw scope
   |     |  |  +--rw intra-area-scope?         boolean
   |     |  |  +--rw intra-area-pref?          uint8
   |     |  |  +--rw inter-area-scope?         boolean
   |     |  |  +--rw inter-area-scope-default? boolean
   |     |  |  +--rw inter-area-pref?          uint8
   |     |  |  +--rw inter-as-scope?           boolean
   |     |  |  +--rw inter-as-scope-default?   boolean
   |     |  |  +--rw inter-as-pref?            uint8
```

```
  |     |   | +--rw inter-layer-scope?        boolean
  |     |   | +--rw inter-layer-pref?         uint8
  |     | +--rw neigh-domains
  |     | | +--rw domain* [domain-type domain]
  |     | |    +--rw domain-type    domain-type
  |     | |    +--rw domain         domain
  |     | +--rw (auth-type-selection)?
  |     |    +--:(auth-key-chain)
  |     |    | +--rw key-chain?             key-chain:key-chain-ref
  |     |    +--:(auth-key)
  |     |    | +--rw key?                   string
  |     |    | +--rw crypto-algorithm
  |     |    |    +--rw (algorithm)?
  |     |    |       +--:(hmac-sha-1-12) {crypto-hmac-sha-1-12}?
  |     |    |       | +--rw hmac-sha1-12?          empty
  |     |    |       +--:(aes-cmac-prf-128) {aes-cmac-prf-128}?
  |     |    |       | +--rw aes-cmac-prf-128?       empty
  |     |    |       +--:(md5)
  |     |    |       | +--rw md5?                   empty
  |     |    |       +--:(sha-1)
  |     |    |       | +--rw sha-1?                 empty
  |     |    |       +--:(hmac-sha-1)
  |     |    |       | +--rw hmac-sha-1?            empty
  |     |    |       +--:(hmac-sha-256)
  |     |    |       | +--rw hmac-sha-256?          empty
  |     |    |       +--:(hmac-sha-384)
  |     |    |       | +--rw hmac-sha-384?          empty
  |     |    |       +--:(hmac-sha-512)
  |     |    |       | +--rw hmac-sha-512?          empty
  |     |    |       +--:(clear-text) {clear-text}?
  |     |    |       | +--rw clear-text?            empty
  |     |    |       +--:(replay-protection-only) {replay-protection-only}?
  |     |    |          +--rw replay-protection-only?  empty
  |     |    +--:(auth-tls) {tls}?
  |     |       +--rw tls
  |   +--rw connect-timer?            uint32
  |   +--rw connect-max-retry?        uint32
  |   +--rw init-backoff-timer?       uint32
  |   +--rw max-backoff-timer?        uint32
  |   +--rw open-wait-timer?          uint32
  |   +--rw keep-wait-timer?          uint32
  |   +--rw keep-alive-timer?         uint32
  |   +--rw dead-timer?               uint32
  |   +--rw allow-negotiation?        boolean
  |   +--rw max-keep-alive-timer?     uint32
  |   +--rw max-dead-timer?           uint32
  |   +--rw min-keep-alive-timer?     uint32
  |   +--rw min-dead-timer?           uint32
```

```
      |        +--rw sync-timer?                  uint32 {svec}?
      |        +--rw request-timer?               uint32
      |        +--rw max-sessions?                uint32
      |        +--rw max-unknown-reqs?            uint32
      |        +--rw max-unknown-msgs?            uint32
      |        +--rw pcep-notification-max-rate   uint32
      |        +--rw stateful-parameter {stateful}?
      |        |  +--rw state-timeout?          uint32
      |        |  +--rw redelegation-timeout?   uint32
      |        |  +--rw rpt-non-pcep-lsp?       boolean
      |        +--rw peers
      |           +--rw peer* [addr]
      |              +--rw addr                inet:ip-address
      |              +--rw description?        string
      |              +--rw domain
      |              |  +--rw domain* [domain-type domain]
      |              |     +--rw domain-type   domain-type
      |              |     +--rw domain        domain
      |              +--rw capability
      |              |  +--rw gmpls?                boolean {gmpls}?
      |              |  +--rw bi-dir?               boolean
      |              |  +--rw diverse?              boolean
      |              |  +--rw load-balance?         boolean
      |              |  +--rw synchronize?          boolean {svec}?
      |              |  +--rw objective-function?   boolean {obj-fn}?
      |              |  +--rw add-path-constraint?  boolean
      |              |  +--rw prioritization?       boolean
      |              |  +--rw multi-request?        boolean
      |              |  +--rw gco?                  boolean {gco}?
      |              |  +--rw p2mp?                 boolean {p2mp}?
      |              |  +--rw stateful {stateful}?
      |              |  |  +--rw enabled?        boolean
      |              |  |  +--rw active?         boolean
      |              |  |  +--rw pce-initiated?  boolean {pce-initiated}?
      |              |  +--rw sr {sr}?
      |              |     +--rw enabled?   boolean
      |              |     +--rw msd?       uint8
      |              +--rw scope
      |              |  +--rw intra-area-scope?         boolean
      |              |  +--rw intra-area-pref?          uint8
      |              |  +--rw inter-area-scope?         boolean
      |              |  +--rw inter-area-scope-default? boolean
      |              |  +--rw inter-area-pref?          uint8
      |              |  +--rw inter-as-scope?           boolean
      |              |  +--rw inter-as-scope-default?   boolean
      |              |  +--rw inter-as-pref?            uint8
      |              |  +--rw inter-layer-scope?        boolean
      |              |  +--rw inter-layer-pref?         uint8
```

```
    |                 +--rw neigh-domains
    |                 | +--rw domain* [domain-type domain]
    |                 |    +--rw domain-type    domain-type
    |                 |    +--rw domain         domain
    |                 +--rw delegation-pref?    uint8 {stateful}?
    |                 +--rw (auth-type-selection)?
    |                    +--:(auth-key-chain)
    |                    | +--rw key-chain?               key-chain:key-chain-ref
    |                    +--:(auth-key)
    |                    | +--rw key?                     string
    |                    | +--rw crypto-algorithm
    |                    |    +--rw (algorithm)?
    |                    |       +--:(hmac-sha-1-12) {crypto-hmac-sha-1-12}?
    |                    |       | +--rw hmac-sha1-12?           empty
    |                    |       +--:(aes-cmac-prf-128) {aes-cmac-prf-128}?
    |                    |       | +--rw aes-cmac-prf-128?       empty
    |                    |       +--:(md5)
    |                    |       | +--rw md5?                    empty
    |                    |       +--:(sha-1)
    |                    |       | +--rw sha-1?                  empty
    |                    |       +--:(hmac-sha-1)
    |                    |       | +--rw hmac-sha-1?             empty
    |                    |       +--:(hmac-sha-256)
    |                    |       | +--rw hmac-sha-256?           empty
    |                    |       +--:(hmac-sha-384)
    |                    |       | +--rw hmac-sha-384?           empty
    |                    |       +--:(hmac-sha-512)
    |                    |       | +--rw hmac-sha-512?           empty
    |                    |       +--:(clear-text) {clear-text}?
    |                    |       | +--rw clear-text?             empty
    |                    |       +--:(replay-protection-only) {replay-protection-only}
?
    |                    |                +--rw replay-protection-only?    empty
    |                    +--:(auth-tls) {tls}?
    |                       +--rw tls
    +--ro pcep-state
       +--ro entity
          +--ro addr?                  inet:ip-address
          +--ro index?                 uint32
          +--ro admin-status?          pcep-admin-status
          +--ro oper-status?           pcep-admin-status
          +--ro role?                  pcep-role
          +--ro domain
          | +--ro domain* [domain-type domain]
          |    +--ro domain-type    domain-type
          |    +--ro domain         domain
          +--ro capability
          | +--ro gmpls?                 boolean {gmpls}?
          | +--ro bi-dir?                boolean
```

```
         |  +--ro diverse?             boolean
         |  +--ro load-balance?        boolean
         |  +--ro synchronize?         boolean {svec}?
         |  +--ro objective-function?  boolean {obj-fn}?
         |  +--ro add-path-constraint? boolean
         |  +--ro prioritization?      boolean
         |  +--ro multi-request?       boolean
         |  +--ro gco?                 boolean {gco}?
         |  +--ro p2mp?                boolean {p2mp}?
         |  +--ro stateful {stateful}?
         |  |  +--ro enabled?       boolean
         |  |  +--ro active?        boolean
         |  |  +--ro pce-initiated? boolean {pce-initiated}?
         |  +--ro sr {sr}?
         |     +--ro enabled?   boolean
         |     +--ro msd?       uint8
         +--ro pce-info
         |  +--ro scope
         |  |  +--ro intra-area-scope?         boolean
         |  |  +--ro intra-area-pref?          uint8
         |  |  +--ro inter-area-scope?         boolean
         |  |  +--ro inter-area-scope-default? boolean
         |  |  +--ro inter-area-pref?          uint8
         |  |  +--ro inter-as-scope?           boolean
         |  |  +--ro inter-as-scope-default?   boolean
         |  |  +--ro inter-as-pref?            uint8
         |  |  +--ro inter-layer-scope?        boolean
         |  |  +--ro inter-layer-pref?         uint8
         |  +--ro neigh-domains
         |  |  +--ro domain* [domain-type domain]
         |  |     +--ro domain-type   domain-type
         |  |     +--ro domain        domain
         |  +--ro (auth-type-selection)?
         |     +--:(auth-key-chain)
         |     |  +--ro key-chain?          key-chain:key-chain-ref
         |     +--:(auth-key)
         |     |  +--ro key?                string
         |     |  +--ro crypto-algorithm
         |     |     +--ro (algorithm)?
         |     |        +--:(hmac-sha-1-12) {crypto-hmac-sha-1-12}?
         |     |        |  +--ro hmac-sha1-12?        empty
         |     |        +--:(aes-cmac-prf-128) {aes-cmac-prf-128}?
         |     |        |  +--ro aes-cmac-prf-128?    empty
         |     |        +--:(md5)
         |     |        |  +--ro md5?                 empty
         |     |        +--:(sha-1)
         |     |        |  +--ro sha-1?               empty
         |     |        +--:(hmac-sha-1)
```

```
   |      |        | +--ro hmac-sha-1?           empty
   |      |        +--:(hmac-sha-256)
   |      |        | +--ro hmac-sha-256?         empty
   |      |        +--:(hmac-sha-384)
   |      |        | +--ro hmac-sha-384?         empty
   |      |        +--:(hmac-sha-512)
   |      |        | +--ro hmac-sha-512?         empty
   |      |        +--:(clear-text) {clear-text}?
   |      |        | +--ro clear-text?           empty
   |      |        +--:(replay-protection-only) {replay-protection-only}?
   |      |           +--ro replay-protection-only?  empty
   |      +--:(auth-tls) {tls}?
   |         +--ro tls
   +--ro connect-timer?          uint32
   +--ro connect-max-retry?      uint32
   +--ro init-backoff-timer?     uint32
   +--ro max-backoff-timer?      uint32
   +--ro open-wait-timer?        uint32
   +--ro keep-wait-timer?        uint32
   +--ro keep-alive-timer?       uint32
   +--ro dead-timer?             uint32
   +--ro allow-negotiation?      boolean
   +--ro max-keep-alive-timer?   uint32
   +--ro max-dead-timer?         uint32
   +--ro min-keep-alive-timer?   uint32
   +--ro min-dead-timer?         uint32
   +--ro sync-timer?             uint32 {svec}?
   +--ro request-timer?          uint32
   +--ro max-sessions?           uint32
   +--ro max-unknown-reqs?       uint32
   +--ro max-unknown-msgs?       uint32
   +--ro stateful-parameter {stateful}?
   |  +--ro state-timeout?          uint32
   |  +--ro redelegation-timeout?   uint32
   |  +--ro rpt-non-pcep-lsp?       boolean
   +--ro lsp-db {stateful}?
   |  +--ro association-list* [id source global-source extended-id]
   |  |  +--ro type?          assoc-type
   |  |  +--ro id             uint16
   |  |  +--ro source         inet:ip-address
   |  |  +--ro global-source  uint32
   |  |  +--ro extended-id    string
   |  |  +--ro lsp* [plsp-id pcc-id]
   |  |     +--ro plsp-id   -> /pcep-state/entity/lsp-db/lsp/plsp-id
   |  |     +--ro pcc-id    -> /pcep-state/entity/lsp-db/lsp/pcc-id
   |  +--ro lsp* [plsp-id pcc-id]
   |     +--ro plsp-id               uint32
   |     +--ro pcc-id                inet:ip-address
```

```
          |            +--ro lsp-ref
          |            |  +--ro source?             -> /te:te/lsps-state/lsp/source
          |            |  +--ro destination?        -> /te:te/lsps-state/lsp/destinati
on
          |            |  +--ro tunnel-id?          -> /te:te/lsps-state/lsp/tunnel-id
          |            |  +--ro lsp-id?             -> /te:te/lsps-state/lsp/lsp-id
          |            |  +--ro extended-tunnel-id? -> /te:te/lsps-state/lsp/extended-
tunnel-id
          |            |  +--ro type?               -> /te:te/lsps-state/lsp/type
          |         +--ro admin-state?        boolean
          |         +--ro operational-state?  operational-state
          |         +--ro delegated
          |         |  +--ro enabled?   boolean
          |         |  +--ro pce?       -> /pcep-state/entity/peers/peer/addr
          |         |  +--ro srp-id?    uint32
          |         +--ro initiation {pce-initiated}?
          |         |  +--ro enabled?   boolean
          |         |  +--ro pce?       -> /pcep-state/entity/peers/peer/addr
          |         +--ro symbolic-path-name?  string
          |         +--ro last-error?          lsp-error
          |         +--ro pst?                 pst
          |         +--ro association-list* [id source global-source extended-id]
          |            +--ro id             -> /pcep-state/entity/lsp-db/associatio
n-list/id
          |            +--ro source         -> /pcep-state/entity/lsp-db/associatio
n-list/source
          |            +--ro global-source  -> /pcep-state/entity/lsp-db/associatio
n-list/global-source
          |            +--ro extended-id    -> /pcep-state/entity/lsp-db/associatio
n-list/extended-id
          +--ro peers
             +--ro peer* [addr]
                +--ro addr                 inet:ip-address
                +--ro role?                pcep-role
                +--ro domain
                |  +--ro domain* [domain-type domain]
                |     +--ro domain-type   domain-type
                |     +--ro domain        domain
                +--ro capability
                |  +--ro gmpls?               boolean {gmpls}?
                |  +--ro bi-dir?              boolean
                |  +--ro diverse?             boolean
                |  +--ro load-balance?        boolean
                |  +--ro synchronize?         boolean {svec}?
                |  +--ro objective-function?  boolean {obj-fn}?
                |  +--ro add-path-constraint? boolean
                |  +--ro prioritization?      boolean
                |  +--ro multi-request?       boolean
                |  +--ro gco?                 boolean {gco}?
                |  +--ro p2mp?                boolean {p2mp}?
                |  +--ro stateful {stateful}?
                |  |  +--ro enabled?        boolean
                |  |  +--ro active?         boolean
                |  |  +--ro pce-initiated?  boolean {pce-initiated}?
```

```
               | +--ro sr {sr}?
               |    +--ro enabled?   boolean
               |    +--ro msd?       uint8
               +--ro pce-info
               | +--ro scope
               | | +--ro intra-area-scope?         boolean
               | | +--ro intra-area-pref?          uint8
               | | +--ro inter-area-scope?         boolean
               | | +--ro inter-area-scope-default? boolean
               | | +--ro inter-area-pref?          uint8
               | | +--ro inter-as-scope?           boolean
               | | +--ro inter-as-scope-default?   boolean
               | | +--ro inter-as-pref?            uint8
               | | +--ro inter-layer-scope?        boolean
               | | +--ro inter-layer-pref?         uint8
               | +--ro neigh-domains
               |    +--ro domain* [domain-type domain]
               |       +--ro domain-type   domain-type
               |       +--ro domain        domain
               +--ro delegation-pref?       uint8 {stateful}?
               +--ro (auth-type-selection)?
               | +--:(auth-key-chain)
               | | +--ro key-chain?                key-chain:key-chain-ref
               | +--:(auth-key)
               | | +--ro key?                      string
               | | +--ro crypto-algorithm
               | |    +--ro (algorithm)?
               | |       +--:(hmac-sha-1-12) {crypto-hmac-sha-1-12}?
               | |       | +--ro hmac-sha1-12?          empty
               | |       +--:(aes-cmac-prf-128) {aes-cmac-prf-128}?
               | |       | +--ro aes-cmac-prf-128?      empty
               | |       +--:(md5)
               | |       | +--ro md5?                   empty
               | |       +--:(sha-1)
               | |       | +--ro sha-1?                 empty
               | |       +--:(hmac-sha-1)
               | |       | +--ro hmac-sha-1?            empty
               | |       +--:(hmac-sha-256)
               | |       | +--ro hmac-sha-256?          empty
               | |       +--:(hmac-sha-384)
               | |       | +--ro hmac-sha-384?          empty
               | |       +--:(hmac-sha-512)
               | |       | +--ro hmac-sha-512?          empty
               | |       +--:(clear-text) {clear-text}?
               | |       | +--ro clear-text?            empty
               | |       +--:(replay-protection-only) {replay-protection-only}
?
               | |             +--ro replay-protection-only?   empty
               | +--:(auth-tls) {tls}?
```

```
               |     +--ro tls
               +--ro discontinuity-time?      yang:timestamp
               +--ro initiate-session?        boolean
               +--ro session-exists?          boolean
               +--ro num-sess-setup-ok?       yang:counter32
               +--ro num-sess-setup-fail?     yang:counter32
               +--ro session-up-time?         yang:timestamp
               +--ro session-fail-time?       yang:timestamp
               +--ro session-fail-up-time?    yang:timestamp
               +--ro pcep-stats
               |  +--ro avg-rsp-time?                uint32
               |  +--ro lwm-rsp-time?                uint32
               |  +--ro hwm-rsp-time?                uint32
               |  +--ro num-pcreq-sent?              yang:counter32
               |  +--ro num-pcreq-rcvd?              yang:counter32
               |  +--ro num-pcrep-sent?              yang:counter32
               |  +--ro num-pcrep-rcvd?              yang:counter32
               |  +--ro num-pcerr-sent?              yang:counter32
               |  +--ro num-pcerr-rcvd?              yang:counter32
               |  +--ro num-pcntf-sent?              yang:counter32
               |  +--ro num-pcntf-rcvd?              yang:counter32
               |  +--ro num-keepalive-sent?          yang:counter32
               |  +--ro num-keepalive-rcvd?          yang:counter32
               |  +--ro num-unknown-rcvd?            yang:counter32
               |  +--ro num-corrupt-rcvd?            yang:counter32
               |  +--ro num-req-sent?                yang:counter32
               |  +--ro num-req-sent-pend-rep?       yang:counter32
               |  +--ro num-req-sent-ero-rcvd?       yang:counter32
               |  +--ro num-req-sent-nopath-rcvd?    yang:counter32
               |  +--ro num-req-sent-cancel-rcvd?    yang:counter32
               |  +--ro num-req-sent-error-rcvd?     yang:counter32
               |  +--ro num-req-sent-timeout?        yang:counter32
               |  +--ro num-req-sent-cancel-sent?    yang:counter32
               |  +--ro num-req-rcvd?                yang:counter32
               |  +--ro num-req-rcvd-pend-rep?       yang:counter32
               |  +--ro num-req-rcvd-ero-sent?       yang:counter32
               |  +--ro num-req-rcvd-nopath-sent?    yang:counter32
               |  +--ro num-req-rcvd-cancel-sent?    yang:counter32
               |  +--ro num-req-rcvd-error-sent?     yang:counter32
               |  +--ro num-req-rcvd-cancel-rcvd?    yang:counter32
               |  +--ro num-rep-rcvd-unknown?        yang:counter32
               |  +--ro num-req-rcvd-unknown?        yang:counter32
               |  +--ro svec {svec}?
               |  |  +--ro num-svec-sent?       yang:counter32
               |  |  +--ro num-svec-req-sent?   yang:counter32
               |  |  +--ro num-svec-rcvd?       yang:counter32
               |  |  +--ro num-svec-req-rcvd?   yang:counter32
               |  +--ro stateful {stateful}?
```

```
          |  |  +--ro num-pcrpt-sent?              yang:counter32
          |  |  +--ro num-pcrpt-rcvd?              yang:counter32
          |  |  +--ro num-pcupd-sent?              yang:counter32
          |  |  +--ro num-pcupd-rcvd?              yang:counter32
          |  |  +--ro num-rpt-sent?                yang:counter32
          |  |  +--ro num-rpt-rcvd?                yang:counter32
          |  |  +--ro num-rpt-rcvd-error-sent?     yang:counter32
          |  |  +--ro num-upd-sent?                yang:counter32
          |  |  +--ro num-upd-rcvd?                yang:counter32
          |  |  +--ro num-upd-rcvd-unknown?        yang:counter32
          |  |  +--ro num-upd-rcvd-undelegated?    yang:counter32
          |  |  +--ro num-upd-rcvd-error-sent?     yang:counter32
          |  |  +--ro initiation {pce-initiated}?
          |  |     +--ro num-pcinitiate-sent?             yang:counter32
          |  |     +--ro num-pcinitiate-rcvd?             yang:counter32
          |  |     +--ro num-initiate-sent?               yang:counter32
          |  |     +--ro num-initiate-rcvd?               yang:counter32
          |  |     +--ro num-initiate-rcvd-error-sent?    yang:counter32
          |  +--ro num-req-sent-closed?          yang:counter32
          |  +--ro num-req-rcvd-closed?          yang:counter32
          +--ro sessions
             +--ro session* [initiator]
                +--ro initiator               pcep-initiator
                +--ro state-last-change?      yang:timestamp
                +--ro state?                  pcep-sess-state
                +--ro session-creation?       yang:timestamp
                +--ro connect-retry?          yang:counter32
                +--ro local-id?               uint32
                +--ro remote-id?              uint32
                +--ro keepalive-timer?        uint32
                +--ro peer-keepalive-timer?   uint32
                +--ro dead-timer?             uint32
                +--ro peer-dead-timer?        uint32
                +--ro ka-hold-time-rem?       uint32
                +--ro overloaded?             boolean
                +--ro overload-time?          uint32
                +--ro peer-overloaded?        boolean
                +--ro peer-overload-time?     uint32
                +--ro lspdb-sync?             sync-state {stateful}?
                +--ro discontinuity-time?     yang:timestamp
                +--ro pcep-stats
                   +--ro avg-rsp-time?           uint32
                   +--ro lwm-rsp-time?           uint32
                   +--ro hwm-rsp-time?           uint32
                   +--ro num-pcreq-sent?         yang:counter32
                   +--ro num-pcreq-rcvd?         yang:counter32
                   +--ro num-pcrep-sent?         yang:counter32
                   +--ro num-pcrep-rcvd?         yang:counter32
```

```
                      +--ro num-pcerr-sent?            yang:counter32
                      +--ro num-pcerr-rcvd?            yang:counter32
                      +--ro num-pcntf-sent?            yang:counter32
                      +--ro num-pcntf-rcvd?            yang:counter32
                      +--ro num-keepalive-sent?        yang:counter32
                      +--ro num-keepalive-rcvd?        yang:counter32
                      +--ro num-unknown-rcvd?          yang:counter32
                      +--ro num-corrupt-rcvd?          yang:counter32
                      +--ro num-req-sent?              yang:counter32
                      +--ro num-req-sent-pend-rep?     yang:counter32
                      +--ro num-req-sent-ero-rcvd?     yang:counter32
                      +--ro num-req-sent-nopath-rcvd?  yang:counter32
                      +--ro num-req-sent-cancel-rcvd?  yang:counter32
                      +--ro num-req-sent-error-rcvd?   yang:counter32
                      +--ro num-req-sent-timeout?      yang:counter32
                      +--ro num-req-sent-cancel-sent?  yang:counter32
                      +--ro num-req-rcvd?              yang:counter32
                      +--ro num-req-rcvd-pend-rep?     yang:counter32
                      +--ro num-req-rcvd-ero-sent?     yang:counter32
                      +--ro num-req-rcvd-nopath-sent?  yang:counter32
                      +--ro num-req-rcvd-cancel-sent?  yang:counter32
                      +--ro num-req-rcvd-error-sent?   yang:counter32
                      +--ro num-req-rcvd-cancel-rcvd?  yang:counter32
                      +--ro num-rep-rcvd-unknown?      yang:counter32
                      +--ro num-req-rcvd-unknown?      yang:counter32
                      +--ro svec {svec}?
                      |  +--ro num-svec-sent?      yang:counter32
                      |  +--ro num-svec-req-sent?  yang:counter32
                      |  +--ro num-svec-rcvd?      yang:counter32
                      |  +--ro num-svec-req-rcvd?  yang:counter32
                      +--ro stateful {stateful}?
                         +--ro num-pcrpt-sent?            yang:counter32
                         +--ro num-pcrpt-rcvd?            yang:counter32
                         +--ro num-pcupd-sent?            yang:counter32
                         +--ro num-pcupd-rcvd?            yang:counter32
                         +--ro num-rpt-sent?              yang:counter32
                         +--ro num-rpt-rcvd?              yang:counter32
                         +--ro num-rpt-rcvd-error-sent?   yang:counter32
                         +--ro num-upd-sent?              yang:counter32
                         +--ro num-upd-rcvd?              yang:counter32
                         +--ro num-upd-rcvd-unknown?      yang:counter32
                         +--ro num-upd-rcvd-undelegated?  yang:counter32
                         +--ro num-upd-rcvd-error-sent?   yang:counter32
                         +--ro initiation {pce-initiated}?
                            +--ro num-pcinitiate-sent?            yang:counter
32
                            +--ro num-pcinitiate-rcvd?            yang:counter
32
                            +--ro num-initiate-sent?              yang:counter
32
                            +--ro num-initiate-rcvd?              yang:counter
32
```

```
                                  +--ro num-initiate-rcvd-error-sent?    yang:counter
32
notifications:
   +---n pcep-session-up
   |  +--ro peer-addr?           -> /pcep-state/entity/peers/peer/addr
   |  +--ro session-initiator?   -> /pcep-state/entity/peers/peer/sessions/sessi
on/initiator
   |  +--ro state-last-change?   yang:timestamp
   |  +--ro state?               pcep-sess-state
   +---n pcep-session-down
   |  +--ro peer-addr?           -> /pcep-state/entity/peers/peer/addr
   |  +--ro session-initiator?   pcep-initiator
   |  +--ro state-last-change?   yang:timestamp
   |  +--ro state?               pcep-sess-state
   +---n pcep-session-local-overload
   |  +--ro peer-addr?           -> /pcep-state/entity/peers/peer/addr
   |  +--ro session-initiator?   -> /pcep-state/entity/peers/peer/sessions/sessi
on/initiator
   |  +--ro overloaded?          boolean
   |  +--ro overload-time?       uint32
   +---n pcep-session-local-overload-clear
   |  +--ro peer-addr?    -> /pcep-state/entity/peers/peer/addr
   |  +--ro overloaded?   boolean
   +---n pcep-session-peer-overload
   |  +--ro peer-addr?           -> /pcep-state/entity/peers/peer/addr
   |  +--ro session-initiator?   -> /pcep-state/entity/peers/peer/sessions/sess
ion/initiator
   |  +--ro peer-overloaded?      boolean
   |  +--ro peer-overload-time?   uint32
   +---n pcep-session-peer-overload-clear
      +--ro peer-addr?           -> /pcep-state/entity/peers/peer/addr
      +--ro peer-overloaded?   boolean
```

5.1. The Entity

   The PCEP yang module may contain status information for the local
   PCEP entity.

   The entity has an IP address (using ietf-inet-types [RFC6991]) and a
   "role" leaf (the local entity PCEP role) as mandatory.

   Note that, the PCEP MIB module [RFC7420] uses an entity list and a
   system generated entity index as a primary index to the read only
   entity table. If the device implements the PCEP MIB, the "index"
   leaf MUST contain the value of the corresponding pcePcepEntityIndex
   and only one entity is assumed.

5.2. The Peer Lists

   The peer list contains peer(s) that the local PCEP entity knows
   about. A PCEP speaker is identified by its IP address. If there is
   a PCEP speaker in the network that uses multiple IP addresses then it
   looks like multiple distinct peers to the other PCEP speakers in the
   network.

   Since PCEP sessions can be ephemeral, the peer list tracks a peer
   even when no PCEP session currently exists to that peer. The
   statistics contained are an aggregate of the statistics for all
   successive sessions to that peer.

   To limit the quantity of information that is stored, an
   implementation MAY choose to discard this information if and only if
   no PCEP session exists to the corresponding peer.

   The data model for PCEP peer presented in this document uses a flat
   list of peers. Each peer in the list is identified by its IP address
   (addr-type, addr).

   There is one list for static peer configuration ("/pcep/entity/
   peers"), and a separate list for the operational state of all peers
   (i.e. static as well as discovered)("/pcep-state/entity/peers").
   The former is used to enable remote PCE configuration at PCC (or PCE)
   while the latter has the operational state of these peers as well as
   the remote PCE peer which were discovered and PCC peers that have
   initiated session.

5.3. The Session Lists

   The session list contains PCEP session that the PCEP entity (PCE or
   PCC) is currently participating in. The statistics in session are
   semantically different from those in peer since the former applies to
   the current session only, whereas the latter is the aggregate for all
   sessions that have existed to that peer.

   Although [RFC5440] forbids more than one active PCEP session between
   a given pair of PCEP entities at any given time, there is a window
   during session establishment where two sessions may exist for a given
   pair, one representing a session initiated by the local PCEP entity
   and the other representing a session initiated by the peer. If
   either of these sessions reaches active state first, then the other
   is discarded.

   The data model for PCEP session presented in this document uses a
   flat list of sessions. Each session in the list is identified by its

initiator. This index allows two sessions to exist transiently for a given peer, as discussed above.

There is only one list for the operational state of all sessions ("/pcep-state/entity/peers/peer/sessions/session").

5.4. Notifications

This YANG model defines a list of notifications to inform client of important events detected during the protocol operation. The notifications defined cover the PCEP MIB notifications.

6. Advanced PCE Features

This document contains a specification of the base PCEP YANG module, "ietf-pcep" which provides the basic PCEP [RFC5440] data model.

This document further handles advanced PCE features like -

o Capability and Scope

o Domain information (local/neighbour)

o Path-Key

o OF

o GCO

o P2MP

o GMPLS

o Inter-Layer

o Stateful PCE

o Segement Routing

o Authentication including PCEPS (TLS)

[Editor's Note - Some of them would be added in a future revision.]

6.1. Stateful PCE's LSP-DB

In the operational state of PCEP which supports stateful PCE mode, the list of LSP state are maintained in LSP-DB. The key is the PLSP-ID and the PCC IP address.

The PCEP data model contains the operational state of LSPs (/pcep-
state/entity/lsp-db/lsp/) with PCEP specific attributes. The generic
TE attributes of the LSP are defined in [I-D.ietf-teas-yang-te]. A
reference to LSP state in TE model is maintained.

7. Open Issues and Next Step

   This section is added so that open issues can be tracked. This
   section would be removed when the document is ready for publication.

7.1. The PCE-Initiated LSP

   The TE Model at [I-D.ietf-teas-yang-te] should support creationg of
   tunnels at the controller (PCE) and marking them as PCE-Initiated.
   The LSP-DB in the PCEP Yang (/pcep-state/entity/lsp-db/lsp/
   initiation) also marks the LSPs which are PCE-initiated.

7.2. PCEP over TLS (PCEPS)

   A future version of this document would add TLS related
   configurations.

8. PCEP YANG Module

   RFC Ed.: In this section, replace all occurrences of 'XXXX' with the
   actual RFC number and all occurrences of the revision date below with
   the date of RFC publication (and remove this note).


```
<CODE BEGINS> file "ietf-pcep@2016-07-07.yang"
module ietf-pcep {
    namespace "urn:ietf:params:xml:ns:yang:ietf-pcep";
    prefix pcep;

    import ietf-inet-types {
        prefix "inet";
    }

    import ietf-yang-types {
        prefix "yang";
    }

    import ietf-te {
        prefix "te";
    }

    import ietf-key-chain {
       prefix "key-chain";
```

```
    }


    organization
        "IETF PCE (Path Computation Element) Working Group";

    contact
        "WG Web:   <http://tools.ietf.org/wg/pce/>
         WG List:  <mailto:pce@ietf.org>
         WG Chair: JP Vasseur
                    <mailto:jpv@cisco.com>
         WG Chair: Julien Meuric
                    <mailto:julien.meuric@orange.com>
         WG Chair: Jonathan Hardwick
                    <mailto:Jonathan.Hardwick@metaswitch.com>
         Editor:   Dhruv Dhody
                    <mailto:dhruv.ietf@gmail.com>";

    description
        "The YANG module defines a generic configuration and
         operational model for PCEP common across all of the
         vendor implementations.";


    revision 2016-07-07 {
        description "Initial revision.";
        reference
            "RFC XXXX:  A YANG Data Model for Path Computation
                        Element Communications Protocol
                        (PCEP)";
    }

    /*
     * Identities
     */

    identity pcep {
        description "Identity for the PCEP protocol.";
    }

    /*
     * Typedefs
     */
    typedef pcep-role {
        type enumeration {
            enum unknown {
                value "0";
                description
```

```
                    "An unknown role";
            }
            enum pcc {
                value "1";
                description
                "The role of a Path Computation Client";
            }
            enum pce {
                value "2";
                description
                "The role of Path Computation Element";
            }
            enum pcc-and-pce {
                value "3";
                description
                "The role of both Path Computation Client and
                 Path Computation Element";
            }
        }

        description
            "The role of a PCEP speaker.
             Takes one of the following values
             - unknown(0): the role is not known.
             - pcc(1): the role is of a Path Computation
               Client (PCC).
             - pce(2): the role is of a Path Computation
               Server (PCE).
             - pccAndPce(3): the role is of both a PCC and
               a PCE.";

    }

    typedef pcep-admin-status {
        type enumeration {
                enum admin-status-up {
                value "1";
                description
                "Admin Status is Up";
            }
            enum admin-status-down {
                value "2";
                description
                "Admin Status is Down";
            }
        }

        description
```

```
        "The Admin Status of the PCEP entity.
         Takes one of the following values
             - admin-status-up(1): Admin Status is Up.
             - admin-status-down(2): Admin Status is Down";
    }

    typedef pcep-oper-status {
        type enumeration {
            enum oper-status-up {
                value "1";
                description
                "The PCEP entity is active";
            }
            enum oper-status-down {
                value "2";
                description
                "The PCEP entity is inactive";
            }
            enum oper-status-going-up {
                value "3";
                description
                "The PCEP entity is activating";
            }
            enum oper-status-going-down {
                value "4";
                description
                "The PCEP entity is deactivating";
            }
            enum oper-status-failed {
                value "5";
                description
                "The PCEP entity has failed and will recover
                 when possible.";
            }
            enum oper-status-failed-perm {
                value "6";
                description
                "The PCEP entity has failed and will not recover
                 without operator intervention";
            }
        }
        description
        "The operational status of the PCEP entity.
         Takes one of the following values
             - oper-status-up(1): Active
             - oper-status-down(2): Inactive
             - oper-status-going-up(3): Activating
             - oper-status-going-down(4): Deactivating
```

```
                - oper-status-failed(5): Failed
                - oper-status-failed-perm(6): Failed Permanantly";
    }

    typedef pcep-initiator {
        type enumeration {
            enum local {
                value "1";
                description
                "The local PCEP entity initiated the session";
            }

            enum remote {
                value "2";
                description
                "The remote PCEP peer initiated the session";
            }
        }
        description
        "The initiator of the session, that is, whether the TCP
         connection was initiated by the local PCEP entity or
         the remote peer.
         Takes one of the following values
            - local(1): Initiated locally
            - remote(2): Initiated remotely";
    }

    typedef pcep-sess-state {
        type enumeration {
            enum tcp-pending {
                value "1";
                description
                    "The tcp-pending state of PCEP session.";
            }

            enum open-wait {
                value "2";
                description
                    "The open-wait state of PCEP session.";
            }

            enum keep-wait {
                value "3";
                description
                    "The keep-wait state of PCEP session.";
            }

            enum session-up {
```

```
                     value "4";
                     description
                         "The session-up state of PCEP session.";
                 }
             }
             description
                 "The current state of the session.
                  The set of possible states excludes the idle state
                  since entries do not exist in the idle state.
                  Takes one of the following values
                      - tcp-pending(1): PCEP TCP Pending state
                      - open-wait(2): PCEP Open Wait state
                      - keep-wait(3): PCEP Keep Wait state
                      - session-up(4): PCEP Session Up state";
         }

         typedef domain-type {
             type enumeration {
                 enum ospf-area {
                     value "1";
                     description
                         "The OSPF area.";
                 }
                 enum isis-area {
                     value "2";
                     description
                         "The IS-IS area.";
                 }
                 enum as {
                     value "3";
                     description
                         "The Autonomous System (AS).";
                 }
             }
             description
                 "The PCE Domain Type";
         }

         typedef domain-ospf-area {
             type union {
                 type uint32;
                 type yang:dotted-quad;
             }
             description
                 "OSPF Area ID.";
         }

         typedef domain-isis-area {
```

```
        type string {
            pattern '[0-9A-Fa-f]{2}\.([0-9A-Fa-f]{4}\.){0,3}';
        }
        description
            "IS-IS Area ID.";
    }

    typedef domain-as {
        type uint32;
        description
            "Autonomous System number.";

    }

    typedef domain {
        type union {
            type domain-ospf-area;
            type domain-isis-area;
            type domain-as;
        }
        description
            "The Domain Information";
    }

    typedef operational-state {
        type enumeration {
            enum down {
                value "0";
                description
                    "not active.";
            }
            enum up {
                value "1";
                description
                    "signalled.";
            }
            enum active {
                value "2";
                description
                    "up and carrying traffic.";
            }
            enum going-down {
                value "3";
                description
                    "LSP is being torn down, resources are
                     being released.";
            }
            enum going-up {
```

```
                value "4";
                description
                    "LSP is being signalled.";
            }
        }
        description
            "The operational status of the LSP";
    }

    typedef lsp-error {
        type enumeration {
            enum no-error {
                value "0";
                description
                    "No error, LSP is fine.";
            }
            enum unknown {
                value "1";
                description
                    "Unknown reason.";
            }
            enum limit {
                value "2";
                description
                    "Limit reached for PCE-controlled LSPs.";
            }
            enum pending {
                value "3";
                description
                    "Too many pending LSP update requests.";
            }
            enum unacceptable {
                value "4";
                description
                    "Unacceptable parameters.";
            }
            enum internal {
                value "5";
                description
                    "Internal error.";
            }
            enum admin {
                value "6";
                description
                    "LSP administratively brought down.";
            }
            enum preempted {
                value "7";
```

```
                description
                    "LSP preempted.";
            }
            enum rsvp {
                value "8";
                description
                    "RSVP signaling error.";
            }
        }
        description
            "The LSP Error Codes.";
    }

    typedef sync-state {
        type enumeration {
            enum pending {
                value "0";
                description
                    "The state synchronization
                     has not started.";
            }
            enum ongoing {
                value "1";
                description
                    "The state synchronization
                     is ongoing.";
            }
            enum finished {
                value "2";
                description
                    "The state synchronization
                     is finished.";
            }
        }
        description
            "The LSP-DB state synchronization operational status.";
    }

        typedef pst{
                type enumeration{
                        enum rsvp-te{
                                value "0";
                                description
                                        "RSVP-TE signaling protocol";
                        }
                        enum sr{
                                value "1";
                                description
```

```
                                        "Segment Routing Traffic Engineering";
                    }
                }
                description
                    "The Path Setup Type";
        }

    typedef assoc-type{
        type enumeration{
            enum protection{
                value "1";
                description
                    "Path Protection Association Type";
            }
        }
        description
            "The PCEP Association Type";
    }

    /*
     * Features
     */

    feature svec {
        description
            "Support synchronized path computation.";
    }

    feature gmpls {
        description
            "Support GMPLS.";
    }

    feature obj-fn {
        description
            "Support OF as per RFC 5541.";
    }

    feature gco {
        description
            "Support GCO as per RFC 5557.";
    }

    feature pathkey {
        description
            "Support pathkey as per RFC 5520.";
    }
```

```
    feature p2mp {
        description
            "Support P2MP as per RFC 6006.";
    }

    feature stateful {
        description
            "Support stateful PCE.";
    }

    feature pce-initiated {
        description
            "Support PCE-Initiated LSP.";
    }

    feature tls {
        description
            "Support PCEP over TLS.";
    }

    feature sr {
        description
            "Support Segement Routing for PCE.";
    }

    /*
     * Groupings
     */



    grouping pcep-entity-info{
        description
            "This grouping defines the attributes for PCEP entity.";
        leaf connect-timer {
            type uint32 {
                range "1..65535";
            }
            units "seconds";
            default 60;
            description
                "The time in seconds that the PCEP entity will wait
                 to establish a TCP connection with a peer.  If a
                 TCP connection is not established within this time
                 then PCEP aborts the session setup attempt.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                           Communication Protocol (PCEP)";
```

```
        }

        leaf connect-max-retry {
            type uint32;
            default 5;
            description
                "The maximum number of times the system tries to
                 establish a TCP connection to a peer before the
                 session with the peer transitions to the idle
                 state.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                           Communication Protocol (PCEP)";
        }

        leaf init-backoff-timer {
            type uint32 {
                range "1..65535";
            }
            units "seconds";
            description
               "The initial back-off time in seconds for retrying
                a failed session setup attempt to a peer.
                The back-off time increases for each failed
                session setup attempt, until a maximum back-off
                time is reached.  The maximum back-off time is
                max-backoff-timer.";
        }

        leaf max-backoff-timer {
            type uint32;
            units "seconds";
            description
               "The maximum back-off time in seconds for retrying
                a failed session setup attempt to a peer.
                The back-off time increases for each failed session
                setup attempt, until this maximum value is reached.
                Session setup attempts then repeat periodically
                without any further increase in back-off time.";
        }

        leaf open-wait-timer {
            type uint32 {
                range "1..65535";
            }
            units "seconds";
            default 60;
            description
```

```
            "The time in seconds that the PCEP entity will wait
             to receive an Open message from a peer after the
             TCP connection has come up.
             If no Open message is received within this time then
             PCEP terminates the TCP connection and deletes the
             associated sessions.";
        reference
            "RFC 5440: Path Computation Element (PCE)
                       Communication Protocol (PCEP)";
    }

    leaf keep-wait-timer {
        type uint32 {
            range "1..65535";
        }
        units "seconds";
        default 60;
        description
            "The time in seconds that the PCEP entity will wait
             to receive a Keepalive or PCErr message from a peer
             during session initialization after receiving an
             Open message.  If no Keepalive or PCErr message is
             received within this time then PCEP terminates the
             TCP connection and deletes the associated
             sessions.";
        reference
            "RFC 5440: Path Computation Element (PCE)
                       Communication Protocol (PCEP)";
    }

    leaf keep-alive-timer {
        type uint32 {
            range "0..255";
        }
        units "seconds";
        default 30;
        description
            "The keep alive transmission timer that this PCEP
             entity will propose in the initial OPEN message of
             each session it is involved in.  This is the
             maximum time between two consecutive messages sent
             to a peer. Zero means that the PCEP entity prefers
             not to send Keepalives at all.
             Note that the actual Keepalive transmission
             intervals, in either direction of an active PCEP
             session, are determined by negotiation between the
             peers as specified by RFC 5440, and so may differ
             from this configured value.";
```

```
            reference
                "RFC 5440: Path Computation Element (PCE)
                        Communication Protocol (PCEP)";
        }

        leaf dead-timer {
            type uint32 {
                range "0..255";
            }
            units "seconds";
            must ". >= ../keep-alive-timer" {
                error-message "The dead timer must be "
                        + "larger than the keep alive timer";
                description
                    "This value MUST be greater than
                     keep-alive-timer.";

            }
            default 120;
            description
                "The dead timer that this PCEP entity will propose
                 in the initial OPEN message of each session it is
                 involved in. This is the time after which a peer
                 should declare a session down if it does not
                 receive any PCEP messages. Zero suggests that the
                 peer does not run a dead timer at all." ;
            reference
                "RFC 5440: Path Computation Element (PCE)
                        Communication Protocol (PCEP)";
        }

        leaf allow-negotiation{
            type boolean;
            description
                "Whether the PCEP entity will permit negotiation of
                 session parameters.";
        }

        leaf max-keep-alive-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in seconds,
                 the maximum value that this PCEP entity will
                 accept from a peer for the interval between
                 Keepalive transmissions. Zero means that the PCEP
```

```
                entity will allow no Keepalive transmission at
                all." ;
        }

        leaf max-dead-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in seconds,
                 the maximum value that this PCEP entity will accept
                 from a peer for the Dead timer.  Zero means that
                 the PCEP entity will allow not running a Dead
                 timer.";
        }

        leaf min-keep-alive-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in seconds,
                 the minimum value that this PCEP entity will
                 accept for the interval between Keepalive
                 transmissions. Zero means that the PCEP entity
                 insists on no Keepalive transmission at all.";
        }

        leaf min-dead-timer{
            type uint32 {
                range "0..255";
            }
            units "seconds";
            description
                "In PCEP session parameter negotiation in
                 seconds, the minimum value that this PCEP entity
                 will accept for the Dead timer.  Zero means that
                 the PCEP entity insists on not running a Dead
                 timer.";
        }

        leaf sync-timer{
            if-feature svec;
            type uint32 {
                range "0..65535";
            }
```

```
            units "seconds";
            default 60;
            description
                "The value of SyncTimer in seconds is used in the
                 case of synchronized path computation request
                 using the SVEC object. Consider the case where a
                 PCReq message is received by a PCE that contains
                 the SVEC object referring to M synchronized path
                 computation requests.  If after the expiration of
                 the SyncTimer all the M path computation requests
                 have not been, received a protocol error is
                 triggered and the PCE MUST cancel the whole set
                 of path computation  requests.
                 The aim of the SyncTimer is to avoid the storage
                 of unused synchronized requests should one of
                 them get lost for some reasons (for example, a
                 misbehaving PCC).
                 Zero means that the PCEP entity does not use the
                 SyncTimer.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                            Communication Protocol (PCEP)";
        }


        leaf request-timer{
            type uint32 {
                range "1..65535";
            }
            units "seconds";
            description
                "The maximum time that the PCEP entity will wait
                 for a response to a PCReq message.";
        }

        leaf max-sessions{
            type uint32;
            description
              "Maximum number of sessions involving this PCEP
               entity that can exist at any time.";
        }

        leaf max-unknown-reqs{
            type uint32;
            default 5;
            description
              "The maximum number of unrecognized requests and
               replies that any session on this PCEP entity is
```

```
                willing to accept per minute before terminating
                the session.
                A PCRep message contains an unrecognized reply
                if it contains an RP object whose request ID
                does not correspond to any in-progress request
                sent by this PCEP entity.
                A PCReq message contains an unrecognized request
                if it contains an RP object whose request ID is
                zero.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                           Communication Protocol (PCEP)";
        }

        leaf max-unknown-msgs{
            type uint32;
            default 5;
            description
             "The maximum number of unknown messages that any
              session on this PCEP entity is willing to accept
              per minute before terminating the session.";
            reference
                "RFC 5440: Path Computation Element (PCE)
                           Communication Protocol (PCEP)";
        }

    }//pcep-entity-info

    grouping  pce-scope{
        description
            "This grouping defines PCE path computation scope
             information which maybe relevant to PCE selection.
             This information corresponds to PCE auto-discovery
             information.";
        reference
            "RFC 5088: OSPF Protocol Extensions for Path
                       Computation Element (PCE)
                       Discovery
             RFC 5089: IS-IS Protocol Extensions for Path
                       Computation Element (PCE)
                       Discovery";
        leaf intra-area-scope{
            type boolean;
            default true;
            description
                "PCE can compute intra-area paths.";
        }
        leaf intra-area-pref{
```

```
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for intra-area TE LSP
              computation.";
        }
        leaf inter-area-scope{
            type boolean;
            default false;
            description
                "PCE can compute inter-area paths.";
        }
        leaf inter-area-scope-default{
            type boolean;
            default false;
            description
                "PCE can act as a default PCE for inter-area
                 path computation.";
        }
        leaf inter-area-pref{
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for inter-area TE LSP
              computation.";
        }
        leaf inter-as-scope{
            type boolean;
            default false;
            description
                "PCE can compute inter-AS paths.";
        }
        leaf inter-as-scope-default{
            type boolean;
            default false;
            description
                "PCE can act as a default PCE for inter-AS
                 path computation.";
        }
        leaf inter-as-pref{
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for inter-AS TE LSP
              computation.";
```

```
        }
        leaf inter-layer-scope{
            type boolean;
            default false;
            description
                "PCE can compute inter-layer paths.";
        }
        leaf inter-layer-pref{
            type uint8{
                range "0..7";
            }
            description
              "The PCE's preference for inter-layer TE LSP
              computation.";
        }
    }//pce-scope

    grouping  domain{
        description
            "This grouping specifies a Domain where the
             PCEP speaker has topology visibility.";
        leaf domain-type{
            type domain-type;
            description
              "The domain type.";
        }
        leaf domain{
            type domain;
            description
              "The domain Information.";
        }
    }//domain

    grouping capability{
        description
            "This grouping specifies a capability
             information of local PCEP entity. This maybe
             relevant to PCE selection as well. This
             information corresponds to PCE auto-discovery
             information.";
        reference
            "RFC 5088: OSPF Protocol Extensions for Path
                       Computation Element (PCE)
                       Discovery
             RFC 5089: IS-IS Protocol Extensions for Path
                       Computation Element (PCE)
                       Discovery";
        leaf gmpls{
```

```
            if-feature gmpls;
            type boolean;
            description
              "Path computation with GMPLS link
               constraints.";
        }
        leaf bi-dir{
            type boolean;
            description
              "Bidirectional path computation.";
        }
        leaf diverse{
            type boolean;
            description
              "Diverse path computation.";
        }
        leaf load-balance{
            type boolean;
            description
              "Load-balanced path computation.";
        }
        leaf synchronize{
            if-feature svec;
            type boolean;
            description
              "Synchronized paths computation.";
        }
        leaf objective-function{
            if-feature obj-fn;
            type boolean;
            description
              "Support for multiple objective functions.";
        }
        leaf add-path-constraint{
            type boolean;
            description
              "Support for additive path constraints (max
               hop count, etc.).";
        }
        leaf prioritization{
            type boolean;
            description
              "Support for request prioritization.";
        }
        leaf multi-request{
            type boolean;
            description
              "Support for multiple requests per message.";
```

```
        }
        leaf gco{
            if-feature gco;
            type boolean;
            description
              "Support for Global Concurrent Optimization
               (GCO).";
        }
        leaf p2mp{
            if-feature p2mp;
            type boolean;
            description
              "Support for P2MP path computation.";
        }

        container stateful{
            if-feature stateful;
            description
                "If stateful PCE feature is present";
            leaf enabled{
                type boolean;
                description
                    "Enabled or Disabled";
            }
            leaf active{
                type boolean;
                description
                  "Support for active stateful PCE.";
            }
            leaf pce-initiated{
                if-feature pce-initiated;
                type boolean;
                description
                  "Support for PCE-initiated LSP.";
            }
        }
                container sr{
                        if-feature sr;
                        description
                                "If segment routing is supported";
                        leaf enabled{
                type boolean;
                description
                    "Enabled or Disabled";
                        }
                        leaf msd{ /*should be in MPLS yang model (?)*/
                                type uint8;
                must "((../../role == 'pcc')" +
```

```
                " or " +
                "(../../role == 'pcc-and-pce')))"
            {
                    error-message
                        "The PCEP entity must be PCC";
                    description
                        "When PCEP entity is PCC for
                        MSD to be applicable";
            }
                            description
                                    "Maximum SID Depth";
                        }
                    }
    }//capability

    grouping  info{
        description
            "This grouping specifies all information which
             maybe relevant to both PCC and PCE.
             This information corresponds to PCE auto-discovery
             information.";
        container domain{
            description
                "The local domain for the PCEP entity";
            list domain{
                key "domain-type domain";
                description
                    "The local domain.";
                uses domain{
                    description
                        "The local domain for the PCEP entity.";
                }
            }
        }
        container capability{
            description
                "The PCEP entity capability";
            uses capability{
                description
                    "The PCEP entity supported
                    capabilities.";
            }
        }
    }//info

    grouping  pce-info{
        description
            "This grouping specifies all PCE information
```

```
         which maybe relevant to the PCE selection.
         This information corresponds to PCE auto-discovery
         information.";
    container scope{
        description
            "The path computation scope";
        uses pce-scope;
    }

    container neigh-domains{
        description
            "The list of neighbour PCE-Domain
             toward which a PCE can compute
             paths";
        list domain{
            key "domain-type domain";

            description
                "The neighbour domain.";
            uses domain{
                description
                    "The PCE neighbour domain.";
            }
        }
    }
}//pce-info

grouping  pcep-stats{
    description
        "This grouping defines statistics for PCEP. It is used
         for both peer and current session.";
    leaf avg-rsp-time{
        type uint32;
        units "milliseconds";
        must "(/pcep-state/entity/peers/peer/role != 'pcc'" +
             " or " +
             "(/pcep-state/entity/peers/peer/role = 'pcc'" +
             " and avg-rsp-time = 0))" {
            error-message
                "Invalid average response time";
            description
                "If role is pcc then this leaf is meaningless
                 and is set to zero.";
        }
        description
          "The average response time.
           If an average response time has not been
           calculated then this leaf has the value zero.";
```

```
        }

        leaf lwm-rsp-time{
            type uint32;
            units "milliseconds";
            must "(/pcep-state/entity/peers/peer/role != 'pcc'" +
                 " or " +
                 "(/pcep-state/entity/peers/peer/role = 'pcc'" +
                 " and lwm-rsp-time = 0))" {
                error-message
                    "Invalid smallest (low-water mark)
                     response time";
                description
                    "If role is pcc then this leaf is meaningless
                     and is set to zero.";
            }
            description
             "The smallest (low-water mark) response time seen.
              If no responses have been received then this
              leaf has the value zero.";
        }

        leaf hwm-rsp-time{
            type uint32;
            units "milliseconds";
            must "(/pcep-state/entity/peers/peer/role != 'pcc'" +
                 " or " +
                 "(/pcep-state/entity/peers/peer/role = 'pcc'" +
                 " and hwm-rsp-time = 0))" {
                error-message
                    "Invalid greatest (high-water mark)
                     response time seen";
                description
                    "If role is pcc then this field is
                     meaningless and is set to zero.";
            }
            description
             "The greatest (high-water mark) response time seen.
              If no responses have been received then this object
              has the value zero.";
        }

        leaf num-pcreq-sent{
            type yang:counter32;
            description
             "The number of PCReq messages sent.";
        }
```

```
        leaf num-pcreq-rcvd{
            type yang:counter32;
            description
              "The number of PCReq messages received.";
        }

        leaf num-pcrep-sent{
            type yang:counter32;
            description
              "The number of PCRep messages sent.";
        }

        leaf num-pcrep-rcvd{
            type yang:counter32;
            description
              "The number of PCRep messages received.";
        }

        leaf num-pcerr-sent{
            type yang:counter32;
            description
              "The number of PCErr messages sent.";
        }

        leaf num-pcerr-rcvd{
            type yang:counter32;
            description
              "The number of PCErr messages received.";
        }

        leaf num-pcntf-sent{
            type yang:counter32;
            description
              "The number of PCNtf messages sent.";
        }

        leaf num-pcntf-rcvd{
            type yang:counter32;
            description
              "The number of PCNtf messages received.";
        }

        leaf num-keepalive-sent{
            type yang:counter32;
            description
              "The number of Keepalive messages sent.";
        }
```

```
        leaf num-keepalive-rcvd{
            type yang:counter32;
            description
              "The number of Keepalive messages received.";
        }

        leaf num-unknown-rcvd{
            type yang:counter32;
            description
              "The number of unknown messages received.";
        }

        leaf num-corrupt-rcvd{
            type yang:counter32;
            description
              "The number of corrupted PCEP message received.";
        }

        leaf num-req-sent{
            type yang:counter32;
            description
              "The number of requests sent.  A request corresponds
               1:1 with an RP object in a PCReq message. This might
               be greater than num-pcreq-sent because multiple
               requests can be batched into a single PCReq
               message.";
        }

        leaf num-req-sent-pend-rep{
            type yang:counter32;
            description
              "The number of requests that have been sent for
               which a response is still pending.";
        }

        leaf num-req-sent-ero-rcvd{
            type yang:counter32;
            description
               "The number of requests that have been sent for
                which a response with an ERO object was received.
                Such responses indicate that a path was
                successfully computed by the peer.";
        }

        leaf num-req-sent-nopath-rcvd{
            type yang:counter32;
            description
              "The number of requests that have been sent for
```

```
                  which a response with a NO-PATH object was
                  received. Such responses indicate that the peer
                  could not  find a path to satisfy the
                  request.";
            }

         leaf num-req-sent-cancel-rcvd{
             type yang:counter32;
             description
               "The number of requests that were cancelled with
                a PCNtf message.
                This might be different than num-pcntf-rcvd because
                not all PCNtf messages are used to cancel requests,
                and a single PCNtf message can cancel multiple
                requests.";
            }

         leaf num-req-sent-error-rcvd{
             type yang:counter32;
             description
               "The number of requests that were rejected with a
                PCErr message.
                This might be different than num-pcerr-rcvd because
                not all PCErr messages are used to reject requests,
                and a single PCErr message can reject multiple
                requests.";
            }

         leaf num-req-sent-timeout{
             type yang:counter32;
             description
               "The number of requests that have been sent to a peer
                and have been abandoned because the peer has taken too
                long to respond to them.";
            }

         leaf num-req-sent-cancel-sent{
             type yang:counter32;
             description
               "The number of requests that were sent to the peer and
                explicitly cancelled by the local PCEP entity sending
                a PCNtf.";
            }

         leaf num-req-rcvd{
             type yang:counter32;
             description
               "The number of requests received.  A request
```

```
            corresponds 1:1 with an RP object in a PCReq
            message.
            This might be greater than num-pcreq-rcvd because
            multiple requests can be batched into a single
            PCReq message.";
        }

        leaf num-req-rcvd-pend-rep{
            type yang:counter32;
            description
             "The number of requests that have been received for
             which a response is still pending.";
        }

        leaf num-req-rcvd-ero-sent{
            type yang:counter32;
            description
             "The number of requests that have been received for
              which a response with an ERO object was sent.  Such
              responses indicate that a path was successfully
              computed by the local PCEP entity.";
        }

        leaf num-req-rcvd-nopath-sent{
            type yang:counter32;
            description
             "The number of requests that have been received for
              which a response with a NO-PATH object was sent. Such
              responses indicate that the local PCEP entity could
              not find a path to satisfy the request.";
        }

        leaf num-req-rcvd-cancel-sent{
            type yang:counter32;
            description
             "The number of requests received that were cancelled
              by the local PCEP entity sending a PCNtf message.
              This might be different than num-pcntf-sent because
              not all PCNtf messages are used to cancel requests,
              and a single PCNtf message can cancel multiple
              requests.";
        }

        leaf num-req-rcvd-error-sent{
            type yang:counter32;
            description
             "The number of requests received that were cancelled
              by the local PCEP entity sending a PCErr message.
```

```
            This might be different than num-pcerr-sent because
            not all PCErr messages are used to cancel requests,
            and a single PCErr message can cancel multiple
            requests.";
        }

        leaf  num-req-rcvd-cancel-rcvd{
            type yang:counter32;
            description
             "The number of requests that were received from the
             peer and explicitly cancelled by the peer sending
             a PCNtf.";
        }

        leaf  num-rep-rcvd-unknown{
            type yang:counter32;
            description
              "The number of responses to unknown requests
               received. A response to an unknown request is a
               response whose RP object does not contain the
               request ID of any request that is currently
               outstanding on the session.";
        }

        leaf  num-req-rcvd-unknown{
            type yang:counter32;
            description
              "The number of unknown requests that have been
               received. An unknown request is a request
               whose RP object contains a request ID of
               zero.";
        }

        container svec{
            if-feature svec;
            description
                "If synchronized path computation is supported";
            leaf num-svec-sent{
                type yang:counter32;
                description
                  "The number of SVEC objects sent in PCReq messages.
                   An SVEC object represents a set of synchronized
                   requests.";
            }

            leaf num-svec-req-sent{
                type yang:counter32;
                description
```

```
                    "The number of requests sent that appeared in one
                     or more SVEC objects.";
            }

            leaf num-svec-rcvd{
                type yang:counter32;
                description
                 "The number of SVEC objects received in PCReq
                  messages. An SVEC object represents a set of
                  synchronized requests.";
            }

            leaf num-svec-req-rcvd{
                type yang:counter32;
                description
                  "The number of requests received that appeared
                   in one or more SVEC objects.";
            }
        }
        container stateful{
            if-feature stateful;
            description
                "Stateful PCE related statistics";
            leaf num-pcrpt-sent{
                type yang:counter32;
                description
                 "The number of PCRpt messages sent.";
            }

            leaf num-pcrpt-rcvd{
                type yang:counter32;
                description
                  "The number of PCRpt messages received.";
            }

            leaf num-pcupd-sent{
                type yang:counter32;
                description
                 "The number of PCUpd messages sent.";
            }

            leaf num-pcupd-rcvd{
                type yang:counter32;
                description
                  "The number of PCUpd messages received.";
            }

            leaf num-rpt-sent{
```

```
            type yang:counter32;
            description
              "The number of LSP Reports sent.  A LSP report
               corresponds 1:1 with an LSP object in a PCRpt
               message. This might be greater than
               num-pcrpt-sent because multiple reports can
               be batched into a single PCRpt message.";
        }

        leaf num-rpt-rcvd{
            type yang:counter32;
            description
             "The number of LSP Reports received.  A LSP report
              corresponds 1:1 with an LSP object in a PCRpt
              message.
              This might be greater than num-pcrpt-rcvd because
              multiple reports can be batched into a single
              PCRpt message.";
        }

        leaf  num-rpt-rcvd-error-sent{
            type yang:counter32;
            description
              "The number of reports of LSPs received that were
              responded by the local PCEP entity by sending a
              PCErr message.";
        }

        leaf num-upd-sent{
            type yang:counter32;
            description
              "The number of LSP updates sent.  A LSP update
               corresponds 1:1 with an LSP object in a PCUpd
               message. This might be greater than
               num-pcupd-sent because multiple updates can
               be batched into a single PCUpd message.";
        }

        leaf num-upd-rcvd{
            type yang:counter32;
            description
             "The number of LSP Updates received.  A LSP update
              corresponds 1:1 with an LSP object in a PCUpd
              message.
              This might be greater than num-pcupd-rcvd because
              multiple updates can be batched into a single
              PCUpd message.";
        }
```

```
         leaf  num-upd-rcvd-unknown{
             type yang:counter32;
             description
               "The number of updates to unknown LSPs
                received. An update to an unknown LSP is a
                update whose LSP object does not contain the
                PLSP-ID of any LSP that is currently
                present.";
         }

         leaf  num-upd-rcvd-undelegated{
             type yang:counter32;
             description
               "The number of updates to not delegated LSPs
                received. An update to an undelegated LSP is a
                update whose LSP object does not contain the
                PLSP-ID of any LSP that is currently
                delegated to current PCEP session.";
         }

         leaf  num-upd-rcvd-error-sent{
             type yang:counter32;
             description
               "The number of updates to LSPs received that were
                responded by the local PCEP entity by sending a
                PCErr message.";
         }
         container initiation {
             if-feature pce-initiated;
             description
                 "PCE-Initiated related statistics";
             leaf num-pcinitiate-sent{
                 type yang:counter32;
                 description
                  "The number of PCInitiate messages sent.";
             }

             leaf num-pcinitiate-rcvd{
                 type yang:counter32;
                 description
                   "The number of PCInitiate messages received.";
             }

             leaf num-initiate-sent{
                 type yang:counter32;
                 description
                   "The number of LSP Initiation sent via PCE.
                    A LSP initiation corresponds 1:1 with an LSP
```

```
                          object in a PCInitiate message. This might be
                          greater than num-pcinitiate-sent because
                          multiple initiations can be batched into a
                          single PCInitiate message.";
                }

                leaf num-initiate-rcvd{
                    type yang:counter32;
                    description
                      "The number of LSP Initiation received from
                       PCE.  A LSP initiation corresponds 1:1 with
                       an LSP object in a PCInitiate message. This
                       might be greater than num-pcinitiate-rcvd
                       because multiple initiations can be batched
                       into a single PCInitiate message.";
                }

                leaf  num-initiate-rcvd-error-sent{
                    type yang:counter32;
                    description
                      "The number of initiations of LSPs received
                       that were responded by the local PCEP entity
                       by sending a PCErr message.";
                }
            }
        }
    }//pcep-stats

    grouping lsp-state{
        description
            "This grouping defines the attributes for LSP in LSP-DB.
             These are the attributes specifically from the PCEP
             perspective";
        leaf plsp-id{
            type uint32{
                range "1..1048575";
            }
            description
                "A PCEP-specific identifier for the LSP.  A PCC
                 creates a unique PLSP-ID for each LSP that is
                 constant for the lifetime of a PCEP session.
                 PLSP-ID is 20 bits with 0 and 0xFFFFF are
                 reserved";
        }
        leaf pcc-id{
            type inet:ip-address;
            description
                "The local internet address of the PCC, that
```

```
                   generated the PLSP-ID.";
        }

        container lsp-ref{
            description
                "reference to ietf-te lsp state";

            leaf source {
                type leafref {
                    path "/te:te/te:lsps-state/te:lsp/te:source";
                }
                description
                  "Tunnel sender address extracted from
                  SENDER_TEMPLATE  object";
                reference "RFC3209";
            }
            leaf destination {
                type leafref {
                    path "/te:te/te:lsps-state/te:lsp/te:"
                            + "destination";
                }
                description
                    "Tunnel endpoint address extracted from
                    SESSION object";
                reference "RFC3209";
            }
            leaf tunnel-id {
                type leafref {
                    path "/te:te/te:lsps-state/te:lsp/te:tunnel-id";
                }
                description
                    "Tunnel identifier used in the SESSION
                    that remains constant over the life
                    of the tunnel.";
                reference "RFC3209";
            }
            leaf lsp-id {
                type leafref {
                    path "/te:te/te:lsps-state/te:lsp/te:lsp-id";
                }
                description
                    "Identifier used in the SENDER_TEMPLATE
                    and the FILTER_SPEC that can be changed
                    to allow a sender to share resources with
                    itself.";
                reference "RFC3209";
            }
            leaf extended-tunnel-id {
```

```
                type leafref {
                    path "/te:te/te:lsps-state/te:lsp/te:"
                    + "extended-tunnel-id";
                }
                description
                    "Extended Tunnel ID of the LSP.";
                reference "RFC3209";
            }
            leaf type {
                type leafref {
                    path "/te:te/te:lsps-state/te:lsp/te:type";
                }
                description "LSP type P2P or P2MP";
            }
        }

        leaf admin-state{
            type boolean;
            description
                "The desired operational state";
        }
        leaf operational-state{
            type operational-state;
            description
                "The operational status of the LSP";
        }
        container delegated{
            description
                "The delegation related parameters";
            leaf enabled{
                type boolean;
                description
                    "LSP is delegated or not";
            }
            leaf pce{
                type leafref {
                    path "/pcep-state/entity/peers/peer/addr";
                }
                must "((../enabled == true)" +
                     " and " +
                     "((../../role == 'pcc')" +
                     " or " +
                     "(../../role == 'pcc-and-pce')))"
                {
                    error-message
                        "The PCEP entity must be PCC
                         and the LSP be delegated";
                    description
```

```
                        "When PCEP entity is PCC for
                        delegated LSP";
                }
                description
                    "The reference to the PCE peer to
                    which LSP is delegated";
            }
            leaf srp-id{
                type uint32;
                description
                    "The last SRP-ID-number associated with this
                    LSP.";
            }
        }
        container initiation {
            if-feature pce-initiated;
            description
                "The PCE initiation related parameters";
            leaf enabled{
                type boolean;
                description
                    "LSP is PCE-initiated or not";
            }
            leaf pce{
                type leafref {
                    path "/pcep-state/entity/peers/peer/addr";
                }
                must "(../enabled == true)"
                {
                    error-message
                        "The LSP must be PCE-Initiated";
                    description
                        "When the LSP must be PCE-Initiated";
                }
                description
                    "The reference to the PCE
                    that initiated this LSP";
            }
        }
        leaf symbolic-path-name{
            type string;
            description
                "The symbolic path name associated with the LSP.";
        }
        leaf last-error{
            type lsp-error;
            description
                "The last error for the LSP.";
```

```
                }
               leaf pst{
                        type pst;
                        default "rsvp-te";
                        description
                                "The Path Setup Type";

                }

     }//lsp-state

     grouping notification-instance-hdr {
         description
             "This group describes common instance specific data
              for notifications.";

         leaf peer-addr {
             type leafref {
                 path "/pcep-state/entity/peers/peer/addr";
             }
             description
                 "Reference to peer address";
         }

     }// notification-instance-hdr

     grouping notification-session-hdr {
         description
         "This group describes common session instance specific
          data for notifications.";

         leaf session-initiator {
             type leafref {
             path "/pcep-state/entity/peers/peer/sessions/" +
                 "session/initiator";
             }
             description
                 "Reference to pcep session initiator leaf";
         }
     }// notification-session-hdr

     grouping stateful-pce-parameter {
         description
         "This group describes stateful PCE specific
          parameters.";
         leaf state-timeout{
             type uint32;
             units "seconds";
```

```
            description
                "When a PCEP session is terminated, a PCC
                 waits for this time period before flushing
                 LSP state associated with that PCEP session
                 and reverting to operator-defined default
                 parameters or behaviours.";
        }
        leaf redelegation-timeout{
            type uint32;
            units "seconds";
            must "((../role == 'pcc')" +
                 " or " +
                 "(../role == 'pcc-and-pce'))"
            {
                error-message "The PCEP entity must be PCC";
                description
                    "When PCEP entity is PCC";
            }
            description
                "When a PCEP session is terminated, a PCC
                 waits for this time period before revoking
                 LSP delegation to a PCE and attempting to
                 redelegate LSPs associated with the
                 terminated PCEP session to an alternate
                 PCE.";
        }
        leaf rpt-non-pcep-lsp{
            type boolean;
            must "((../role == 'pcc')" +
                 " or " +
                 "(../role == 'pcc-and-pce'))"
            {
                error-message "The PCEP entity must be PCC";
                description
                    "When PCEP entity is PCC";
            }
            description
                "If set, a PCC reports LSPs that are not
                 controlled by any PCE (for example, LSPs
                 that are statically configured at the
                 PCC). ";
        }

    }

    grouping authentication {
        description "Authentication Information";
        choice auth-type-selection {
```

```
            description
                "Options for expressing authentication setting.";
            case auth-key-chain {
                leaf key-chain {
                    type key-chain:key-chain-ref;
                    description
                        "key-chain name.";
                }
            }
            case auth-key {
                leaf key {
                    type string;
                description
                    "Key string in ASCII format.";
                }
                container crypto-algorithm {
                    uses key-chain:crypto-algorithm-types;
                        description
                            "Cryptographic algorithm associated
                             with key.";
                }
            }
            case auth-tls {
                if-feature tls;
                container tls {
                    description
                        "TLS related information - TBD";
                }
            }
        }
    }

    grouping association {
        description
            "Generic Association parameters";
        leaf type {
            type "assoc-type";
            description
                "The PCEP association type";
        }
        leaf id {
            type uint16;
            description
                "PCEP Association ID";
        }
        leaf source {
          type inet:ip-address;
          description
```

```
                    "PCEP Association Source.";
            }
            leaf global-source {
              type uint32;
              description
                    "PCEP Association Global
                    Source.";
            }
            leaf extended-id{
                type string;
                description
                    "Additional information to
                    support unique identification.";
            }
     }
     grouping association-ref {
            description
                "Generic Association parameters";
            leaf id {
                type leafref {
                    path "/pcep-state/entity/lsp-db/"
                    + "association-list/id";
                }
                description
                    "PCEP Association ID";
            }
            leaf source {
                type leafref {
                    path "/pcep-state/entity/lsp-db/"
                    + "association-list/source";
                }
              description
                    "PCEP Association Source.";
            }
            leaf global-source {
                type leafref {
                    path "/pcep-state/entity/lsp-db/"
                    + "association-list/global-source";
                }
              description
                    "PCEP Association Global
                    Source.";
            }
            leaf extended-id{
                type leafref {
                    path "/pcep-state/entity/lsp-db/"
                    + "association-list/extended-id";
                }
```

```
                description
                    "Additional information to
                    support unique identification.";
            }
        }
        /*
         * Configuration data nodes
         */
        container pcep{

            presence
                "The PCEP is enabled";

                description
                "Parameters for list of configured PCEP entities
                 on the device.";

            container entity {

                description
                    "The configured PCEP entity on the device.";

                leaf addr {
                    type inet:ip-address;
                    mandatory true;
                    description
                        "The local Internet address of this PCEP
                        entity.
                        If operating as a PCE server, the PCEP
                        entity listens on this address.
                        If operating as a PCC, the PCEP entity
                        binds outgoing TCP connections to this
                        address.
                        It is possible for the PCEP entity to
                        operate both as a PCC and a PCE Server, in
                        which case it uses this address both to
                        listen for incoming TCP connections and to
                        bind outgoing TCP connections.";
                }

                leaf enabled {
                    type boolean;
                    default true;
                    description
                        "The administrative status of this PCEP
                         Entity.";
                }
```

```
            leaf role {
                type pcep-role;
                mandatory true;
                description
                    "The role that this entity can play.
                     Takes one of the following values.
                     - unknown(0): this PCEP Entity role is not
                       known.
                     - pcc(1): this PCEP Entity is a PCC.
                     - pce(2): this PCEP Entity is a PCE.
                     - pcc-and-pce(3): this PCEP Entity is both
                       a PCC and a PCE.";
            }

            leaf description {
                type string;
                description
                    "Description of the PCEP entity configured
                     by the user";
            }

            uses info {
                description
                    "Local PCEP entity information";
            }


            container pce-info {
                must "((../role == 'pce')" +
                     " or " +
                     "(../role == 'pcc-and-pce'))"
                {
                    error-message "The PCEP entity must be PCE";
                    description
                        "When PCEP entity is PCE";
                }
                uses pce-info {
                    description
                        "Local PCE information";
                }
                            uses authentication {
                                    description
                                        "Local PCE authentication inform
ation";
                    }


                description
                    "The Local PCE Entity PCE information";
```

```
            }

            uses pcep-entity-info {
                description
                    "The configuration related to the PCEP
                     entity.";
            }


            leaf pcep-notification-max-rate {
                type uint32;
                mandatory true;
                description
                    "This variable indicates the maximum number of
                     notifications issued per second. If events occur
                     more rapidly, the implementation may simply fail
                     to emit these notifications during that period,
                     or may queue them until an appropriate time. A
                     value of 0 means no notifications are emitted
                     and all should be discarded (that is, not
                     queued).";
            }

            container stateful-parameter{
                if-feature stateful;
                must "(../info/capability/stateful/active == true)"
                {
                    error-message
                        "The Active Stateful PCE must be enabled";
                    description
                        "When PCEP entity is active stateful
                         enabled";
                }
                uses stateful-pce-parameter;

                description
                    "The configured stateful parameters";
            }


            container peers{
                must "((../role == 'pcc')" +
                     " or " +
                     "(../role == 'pcc-and-pce'))"
                {
                    error-message
                        "The PCEP entity must be PCC";
```

```
                description
                    "When PCEP entity is PCC, as remote
                     PCE peers are configured.";
            }
            description
                "The list of configured peers for the
                 entity (remote PCE)";
            list peer{
                key "addr";

                description
                    "The peer configured for the entity.
                     (remote PCE)";

                leaf addr {
                    type inet:ip-address;
                    description
                        "The local Internet address of this
                         PCEP peer.";
                }

                leaf description {
                    type string;
                    description
                        "Description of the PCEP peer
                         configured by the user";
                }
                uses info {
                    description
                        "PCE Peer information";
                }
                uses pce-info {
                    description
                        "PCE Peer information";
                }

                leaf delegation-pref{
                    if-feature stateful;
                    type uint8{
                        range "0..7";
                    }
                    must "(../../info/capability/stateful/active"
                        + "== true)"
                    {
                        error-message
                            "The Active Stateful PCE must be
                             enabled";
                        description
```

```
                                "When PCEP entity is active stateful
                                 enabled";
                        }
                        description
                            "The PCE peer delegation preference.";
                    }
                    uses authentication {
                        description
                            "PCE Peer authentication";
                    }
                }//peer
            }//peers
        }//entity
    }//pcep

    /*
     * Operational data nodes
     */

    container pcep-state{
        config false;
        description
            "The list of operational PCEP entities on the
             device.";

        container entity{
            description
                "The operational PCEP entity on the device.";

            leaf addr {
                type inet:ip-address;
                description
                    "The local Internet address of this PCEP
                    entity.
                    If operating as a PCE server, the PCEP
                    entity listens on this address.
                    If operating as a PCC, the PCEP entity
                    binds outgoing TCP connections to this
                    address.
                    It is possible for the PCEP entity to
                    operate both as a PCC and a PCE Server, in
                    which case it uses this address both to
                    listen for incoming TCP connections and to
                    bind outgoing TCP connections.";
            }

            leaf index{
                type uint32;
```

```
            description
                "The index of the operational PECP
                 entity";
        }


        leaf admin-status {
            type pcep-admin-status;
            description
                "The administrative status of this PCEP Entity.
                 This is the desired operational status as
                 currently set by an operator or by default in
                 the implementation.  The value of enabled
                 represents the current status of an attempt
                 to reach this desired status.";
        }

        leaf oper-status {
            type pcep-admin-status;
            description
               "The operational status of the PCEP entity.
                Takes one of the following values.
                - oper-status-up(1): the PCEP entity is
                  active.
                - oper-status-down(2): the PCEP entity is
                  inactive.
                - oper-status-going-up(3): the PCEP entity is
                  activating.
                - oper-status-going-down(4): the PCEP entity is
                  deactivating.
                - oper-status-failed(5): the PCEP entity has
                  failed and will recover when possible.
                - oper-status-failed-perm(6): the PCEP entity
                  has failed and will not recover without
                  operator intervention.";
        }

        leaf role {
            type pcep-role;
            description
                "The role that this entity can play.
                 Takes one of the following values.
                 - unknown(0): this PCEP entity role is
                   not known.
                 - pcc(1): this PCEP entity is a PCC.
                 - pce(2): this PCEP entity is a PCE.
                 - pcc-and-pce(3): this PCEP entity is
                   both a PCC and a PCE.";
```

```
            }

            uses info {
                description
                    "Local PCEP entity information";
            }

            container pce-info {
                when "((../role == 'pce')" +
                     " or " +
                     "(../role == 'pcc-and-pce'))"
                {
                    description
                        "When PCEP entity is PCE";
                }
                uses pce-info {
                    description
                        "Local PCE information";
                }
                                uses authentication {
                                    description
                                        "Local PCE authentication inform
ation";
                    }
                description
                    "The Local PCE Entity PCE information";
            }

            uses pcep-entity-info{
                description
                    "The operational information related to the
                     PCEP entity.";
            }

            container stateful-parameter{
                if-feature stateful;
                must "(../info/capability/stateful/active == true)"
                {
                    error-message
                        "The Active Stateful PCE must be enabled";
                    description
                        "When PCEP entity is active stateful
                         enabled";
                }
                uses stateful-pce-parameter;

                description
                    "The operational stateful parameters";
            }
```

```
            container lsp-db{
                if-feature stateful;
                description
                    "The LSP-DB";
                list association-list {
                    key "id source global-source extended-id";
                    description
                        "List of all PCEP associations";
                    uses association {
                        description
                            "The Association attributes";
                    }
                    list lsp {
                        key "plsp-id pcc-id";
                        description
                            "List of all LSP in this association";
                        leaf plsp-id {
                            type leafref {
                                path "/pcep-state/entity/lsp-db/"
                                + "lsp/plsp-id";
                            }
                            description
                                "Reference to PLSP-ID in LSP-DB";
                        }
                        leaf pcc-id {
                            type leafref {
                                path "/pcep-state/entity/lsp-db/"
                                + "lsp/pcc-id";
                            }
                            description
                                "Reference to PCC-ID in LSP-DB";
                        }
                    }
                }
                list lsp{
                    key "plsp-id pcc-id";
                    description
                        "List of all LSPs in LSP-DB";
                    uses lsp-state{
                        description
                            "The PCEP specific attributes for
                             LSP-DB.";
                    }
                    list association-list {
                        key "id source global-source extended-id";
                        description
                            "List of all PCEP associations";
                        uses association-ref {
```

```
                            description
                                "Reference to the Association
                                 attributes";
                        }
                    }

                }
            }
            container peers{
                description
                        "The list of peers for the entity";

                list peer{
                    key "addr";

                    description
                        "The peer for the entity.";

                    leaf addr {
                        type inet:ip-address;
                        description
                            "The local Internet address of this PCEP
                             peer.";
                    }

                    leaf role {
                        type pcep-role;
                        description
                            "The role of the PCEP Peer.
                             Takes one of the following values.
                             - unknown(0): this PCEP peer role
                               is not known.
                             - pcc(1): this PCEP peer is a PCC.
                             - pce(2): this PCEP peer is a PCE.
                             - pcc-and-pce(3): this PCEP peer
                               is both a PCC and a PCE.";
                    }


                    uses info {
                        description
                            "PCEP peer information";
                    }


                    container pce-info {
                        when "((../role == 'pce')" +
                        " or " +
```

```
                          "(../role == 'pcc-and-pce'))"
                          {
                              description
                                  "When PCEP entity is PCE";
                          }
                          uses pce-info {
                              description
                                  "PCE Peer information";
                          }
                      description
                          "The PCE Peer information";
                      }

                      leaf delegation-pref{
                          if-feature stateful;
                          type uint8{
                              range "0..7";
                          }
                          must "((../../role == 'pcc')" +
                               " or " +
                               "(../../role == 'pcc-and-pce'))"
                          {
                              error-message
                                  "The PCEP entity must be PCC";
                              description
                                  "When PCEP entity is PCC";
                          }
                          must "(../../info/capability/stateful/active"
                               + " == true)"
                          {
                              error-message
                                  "The Active Stateful PCE must be
                                   enabled";
                              description
                                  "When PCEP entity is active stateful
                                   enabled";
                          }
                          description
                              "The PCE peer delegation preference.";
                      }

                      uses authentication {
                          description
                              "PCE Peer authentication";
                      }

                      leaf discontinuity-time {
                          type yang:timestamp;
```

```
                    description
                        "The timestamp of the time when the
                         information and statistics were
                         last reset.";
                }

                leaf initiate-session {
                    type boolean;
                    description
                        "Indicates whether the local PCEP
                         entity initiates sessions to this peer,
                         or waits for the peer to initiate a
                         session.";
                }

                leaf session-exists{
                    type boolean;
                    description
                        "Indicates whether a session with
                         this peer currently exists.";
                }

                leaf num-sess-setup-ok{
                    type yang:counter32;
                    description
                        "The number of PCEP sessions successfully
                         successfully established with the peer,
                         including any current session.  This
                         counter is incremented each time a
                         session with this peer is successfully
                         established.";
                }

                leaf num-sess-setup-fail{
                    type yang:counter32;
                    description
                       "The number of PCEP sessions with the peer
                        that have been attempted but failed
                        before being fully established. This
                        counter is incremented each time a
                        session retry to this peer fails.";
                }

                leaf session-up-time{
                    type yang:timestamp;
                    must "(../num-sess-setup-ok != 0  or " +
                        "(../num-sess-setup-ok = 0  and " +
                        "session-up-time = 0))" {
```

```
                             error-message
                                 "Invalid Session Up timestamp";
                             description
                                 "If num-sess-setup-ok is zero,
                                  then this leaf contains zero.";
                         }
                     description
                       "The timestamp value of the last time a
                        session with this peer was successfully
                        established.";
                 }

                 leaf session-fail-time{
                     type yang:timestamp;
                     must "(../num-sess-setup-fail != 0  or " +
                         "(../num-sess-setup-fail = 0  and "  +
                         "session-fail-time = 0))" {
                             error-message
                                 "Invalid Session Fail timestamp";
                             description
                                 "If num-sess-setup-fail is zero,
                                  then this leaf contains zero.";
                         }
                     description
                       "The timestamp value of the last time a
                        session with this peer failed to be
                        established.";
                 }

                 leaf session-fail-up-time{
                     type yang:timestamp;
                     must "(../num-sess-setup-ok != 0  or "     +
                         "(../num-sess-setup-ok = 0  and "      +
                         "session-fail-up-time = 0))" {
                             error-message
                                 "Invalid Session Fail from
                                  Up timestamp";
                             description
                                 "If num-sess-setup-ok is zero,
                                  then this leaf contains zero.";
                         }
                     description
                       "The timestamp value of the last time a
                        session with this peer failed from
                        active.";
                 }

                 container pcep-stats {
```

```
                        description
                            "The container for all statistics at peer
                             level.";
                        uses pcep-stats{
                            description
                                "Since PCEP sessions can be
                                ephemeral, the peer statistics tracks
                                a peer even when no PCEP session
                                currently exists to that peer. The
                                statistics contained are an aggregate
                                of the statistics for all successive
                                sessions to that peer.";
                        }

                        leaf num-req-sent-closed{
                            type yang:counter32;
                            description
                                "The number of requests that were
                                 sent to the peer and implicitly
                                 cancelled when the session they were
                                 sent over was closed.";
                        }

                        leaf  num-req-rcvd-closed{
                            type yang:counter32;
                            description
                                "The number of requests that were
                                received from the peer and
                                implicitly cancelled when the
                                session they were received over
                                was closed.";
                        }
                    }//pcep-stats


                    container sessions {
                        description
                            "This entry represents a single PCEP
                             session in which the local PCEP entity
                             participates.
                             This entry exists only if the
                             corresponding PCEP session has been
                             initialized by some event, such as
                             manual user configuration, auto-
                             discovery of a peer, or an incoming
                             TCP connection.";
```

```
list session {
    key "initiator";

    description
        "The list of sessions, note that
         for a time being two sessions
         may exist for a peer";

    leaf initiator {
        type pcep-initiator;
        description
            "The initiator of the session,
             that is, whether the TCP
             connection was initiated by
             the local PCEP entity or the
             peer.
             There is a window during
             session initialization where
             two sessions can exist between
             a pair of PCEP speakers, each
             initiated by one of the
             speakers. One of these
             sessions is always discarded
             before it leaves OpenWait state.
             However, before it is discarded,
             two sessions to the given peer
             appear transiently in this MIB
             module. The sessions are
             distinguished by who initiated
             them, and so this field is the
             key.";
    }

    leaf state-last-change {
        type yang:timestamp;
        description
            "The timestamp value at the
             time this session entered its
             current state as denoted by
             the state leaf.";
    }

    leaf state {
        type pcep-sess-state;
        description
            "The current state of the
             session.
             The set of possible states
```

```
                            excludes the idle state since
                            entries do not exist in the
                            idle state.";
                    }

                    leaf session-creation {
                        type yang:timestamp;
                        description
                            "The timestamp value at the
                            time this session was
                            created.";
                    }

                    leaf connect-retry {
                        type yang:counter32;
                        description
                            "The number of times that the
                             local PCEP entity has
                             attempted to establish a TCP
                             connection for this session
                             without success. The PCEP
                             entity gives up when this
                             reaches connect-max-retry.";
                    }

                    leaf local-id {
                        type uint32 {
                          range "0..255";
                        }
                        description
                            "The value of the PCEP session
                             ID used by the local PCEP
                             entity in the Open message
                             for this session.
                             If state is tcp-pending then
                             this is the session ID that
                             will be used in the Open
                             message. Otherwise, this is
                             the session ID that was sent
                             in the Open message.";
                    }

                    leaf remote-id {
                        type uint32 {
                            range "0..255";
                        }
                        must "((../state != 'tcp-pending'" +
                            "and " +
```

```
                        "../state != 'open-wait' )" +
                        "or " +
                        "((../state = 'tcp-pending'" +
                        " or " +
                        "../state = 'open-wait' )" +
                        "and remote-id = 0))" {
                           error-message
                               "Invalid remote-id";
                           description
                                 "If state is tcp-pending
                                  or open-wait then this
                                  leaf is not used and
                                  MUST be set to zero.";
                        }
                   description
                        "The value of the PCEP session
                         ID used by the peer in its
                         Open message for this
                         session.";
                }

                leaf keepalive-timer {
                    type uint32 {
                      range "0..255";
                    }
                    units "seconds";
                    must "(../state = 'session-up'" +
                         "or " +
                         "(../state != 'session-up'" +
                         "and keepalive-timer = 0))" {
                            error-message
                                "Invalid keepalive
                                 timer";
                            description
                                "This field is used if
                                 and only if state is
                                 session-up. Otherwise,
                                 it is not used and
                                 MUST be set to
                                 zero.";
                        }
                   description
                        "The agreed maximum interval at
                         which the local PCEP entity
                         transmits PCEP messages on this
                         PCEP session.  Zero means that
                         the local PCEP entity never
                         sends Keepalives on this
```

```
                                    session.";
                            }

                            leaf peer-keepalive-timer {
                                type uint32 {
                                  range "0..255";
                                }
                                units "seconds";
                                must "(../state = 'session-up'" +
                                        "or "    +
                                        "(../state != 'session-up'" +
                                        "and " +
                                        "peer-keepalive-timer = 0))" {
                                            error-message
                                                "Invalid Peer keepalive
                                                 timer";
                                            description
                                                "This field is used if
                                                 and only if state is
                                                 session-up. Otherwise,
                                                 it is not used and MUST
                                                 be set to zero.";
                                }
                                description
                                        "The agreed maximum interval at
                                         which the peer transmits PCEP
                                         messages on this PCEP session.
                                         Zero means that the peer never
                                         sends Keepalives on this
                                         session.";
                            }

                            leaf dead-timer {
                                type uint32 {
                                  range "0..255";
                                }
                                units "seconds";
                                description
                                        "The dead timer interval for
                                         this PCEP session.";
                            }

                            leaf peer-dead-timer {
                                type uint32 {
                                  range "0..255";
                                }
                                units "seconds";
                                must "((../state != 'tcp-pending'" +
```

```
                            "and " +
                            "../state != 'open-wait' )" +
                            "or " +
                            "((../state = 'tcp-pending'" +
                            " or " +
                            "../state = 'open-wait' )" +
                            "and " +
                            "peer-dead-timer = 0))" {
                                error-message
                                    "Invalid Peer Dead
                                     timer";
                                description
                                    "If state is tcp-
                                     pending or open-wait
                                     then this leaf is not
                                     used and MUST be set to
                                     zero.";
                            }
                        description
                            "The peer's dead-timer interval
                             for this PCEP session.";
                    }

                    leaf ka-hold-time-rem {
                        type uint32 {
                          range "0..255";
                        }
                        units "seconds";
                        must "((../state != 'tcp-pending'" +
                                "and " +
                                "../state != 'open-wait' ) " +
                                "or " +
                                "((../state = 'tcp-pending'" +
                                "or " +
                                "../state = 'open-wait' )" +
                                "and " +
                                "ka-hold-time-rem = 0))" {
                                    error-message
                                        "Invalid Keepalive hold
                                         time remaining";
                                    description
                                        "If state is tcp-pending
                                         or open-wait then this
                                         field is not used and
                                         MUST be set to zero.";
                                }
                        description
                            "The keep alive hold time
```

```
                                remaining for this session.";
                    }

                    leaf overloaded {
                        type boolean;
                        description
                            "If the local PCEP entity has
                             informed the peer that it is
                             currently overloaded, then this
                             is set to true.  Otherwise, it
                             is set to false.";
                    }

                    leaf overload-time {
                        type uint32;
                        units "seconds";
                        must "(../overloaded = true or" +
                            "(../overloaded != true and" +
                            " overload-time = 0))" {
                                error-message
                                    "Invalid overload-time";
                                description
                                    "This field is only used
                                     if overloaded is set to
                                     true. Otherwise, it is
                                     not used and MUST be set
                                     to zero.";
                            }
                        description
                            "The interval of time that is
                             remaining until the local PCEP
                             entity will cease to be
                             overloaded on this session.";
                    }

                    leaf peer-overloaded {
                        type boolean;
                        description
                            "If the peer has informed the
                             local PCEP entity that it is
                             currently overloaded, then this
                             is set to true. Otherwise, it
                             is set to false.";
                    }

                    leaf peer-overload-time {
                        type uint32;
                        units "seconds";
```

```
                        must "(../peer-overloaded = true" +
                            " or " +
                            "(../peer-overloaded != true" +
                            " and " +
                            "peer-overload-time = 0))" {
                              error-message
                                  "Invalid peer overload
                                   time";
                              description
                                  "This field is only used
                                   if peer-overloaded is
                                   set to true. Otherwise,
                                   it is not used and MUST
                                   be set to zero.";
                        }
                    description
                        "The interval of time that is
                         remaining until the peer will
                         cease to be overloaded.  If it
                         is not known how long the peer
                         will stay in overloaded state,
                         this leaf is set to zero.";
                }
                leaf lspdb-sync {
                    if-feature stateful;
                    type sync-state;
                    description
                        "The LSP-DB state synchronization
                         status.";
                }
                leaf discontinuity-time {
                    type yang:timestamp;
                    description
                        "The timestamp value of the time
                         when the statistics were last
                         reset.";
                }

                container pcep-stats {
                    description
                        "The container for all statistics
                         at session level.";
                    uses pcep-stats{
                        description
                            "The statistics contained are
                             for the current sessions to
                             that peer. These are lost
                             when the session goes down.
```

```
                                               ";
                        }
                }//pcep-stats

                        } // session
                    } // sessions
                }//peer
            }//peers
        }//entity
    }//pcep-state

    /*
     * Notifications
     */
    notification pcep-session-up {
        description
            "This notification is sent when the value of
             '/pcep/pcep-state/peers/peer/sessions/session/state'
             enters the 'session-up' state.";

        uses notification-instance-hdr;

        uses notification-session-hdr;

        leaf state-last-change {
            type yang:timestamp;
            description
                "The timestamp value at the time this session entered
                its current state as denoted by the state leaf.";
        }

        leaf state {
            type pcep-sess-state;
            description
                "The current state of the session.
                 The set of possible states excludes the idle state
                 since entries do not exist in the idle state.";
        }
    } //notification

    notification pcep-session-down {
        description
            "This notification is sent when the value of
             '/pcep/pcep-state/peers/peer/sessions/session/state'
             leaves the 'session-up' state.";

        uses notification-instance-hdr;
```

```
        leaf session-initiator {
            type pcep-initiator;
            description
                "The initiator of the session.";
        }

        leaf state-last-change {
            type yang:timestamp;
            description
                "The timestamp value at the time this session entered
                its current state as denoted by the state leaf.";
        }

        leaf state {
            type pcep-sess-state;
            description
                "The current state of the session.
                 The set of possible states excludes the idle state
                 since entries do not exist in the idle state.";
        }
    } //notification

    notification pcep-session-local-overload {
        description
            "This notification is sent when the local PCEP entity
            enters overload state for a peer.";

        uses notification-instance-hdr;

        uses notification-session-hdr;

        leaf overloaded {
            type boolean;
            description
                "If the local PCEP entity has informed the peer that
                it is currently overloaded, then this is set to
                true. Otherwise, it is set to false.";
        }

        leaf overload-time {
            type uint32;
            units "seconds";
            must "(../overloaded = true or "    +
                "(../overloaded != true  and "  +
                "overload-time = 0))" {
                    error-message
                        "Invalid overload-time";
                    description
```

```
                            "This field is only used if overloaded is
                             set to true. Otherwise, it is not used
                             and MUST be set to zero.";
                }
            description
                "The interval of time that is remaining until the
                local PCEP entity will cease to be overloaded on
                this session.";
        }
    } //notification

    notification pcep-session-local-overload-clear {
        description
            "This notification is sent when the local PCEP entity
            leaves overload state for a peer.";

        uses notification-instance-hdr;

        leaf overloaded {
            type boolean;
            description
                "If the local PCEP entity has informed the peer
                that it is currently overloaded, then this is set
                to true.  Otherwise, it is set to false.";
        }
    } //notification

    notification pcep-session-peer-overload {
        description
            "This notification is sent when a peer enters overload
            state.";

        uses notification-instance-hdr;

        uses notification-session-hdr;

        leaf peer-overloaded {
            type boolean;
            description
                "If the peer has informed the local PCEP entity that
                it is currently overloaded, then this is set to true.
                Otherwise, it is set to false.";
        }

        leaf peer-overload-time {
            type uint32;
            units "seconds";
            must "(../peer-overloaded = true or "     +
```

```
                    "(../peer-overloaded != true  and "  +
                    "peer-overload-time = 0))" {
                        error-message
                            "Invalid peer-overload-time";
                        description
                            "This field is only used if
                             peer-overloaded is set to true.
                             Otherwise, it is not used and MUST
                             be set to zero.";
                    }
                description
                    "The interval of time that is remaining until the
                    peer will cease to be overloaded.  If it is not known
                    how long the peer will stay in overloaded state, this
                    leaf is set to zero.";
            }
        } //notification

    notification pcep-session-peer-overload-clear {
        description
            "This notification is sent when a peer leaves overload
            state.";

        uses notification-instance-hdr;

        leaf peer-overloaded {
            type boolean;
            description
                "If the peer has informed the local PCEP entity that
                it is currently overloaded, then this is set to true.
                Otherwise, it is set to false.";
        }
    } //notification
}//module

<CODE ENDS>
```

9. Security Considerations

    The YANG module defined in this memo is designed to be accessed via
    the NETCONF protocol [RFC6241]. The lowest NETCONF layer is the
    secure transport layer and the mandatory-to-implement secure
    transport is SSH [RFC6242]. The NETCONF access control model
    [RFC6536] provides the means to restrict access for particular
    NETCONF users to a pre-configured subset of all available NETCONF
    protocol operations and content.

There are a number of data nodes defined in the YANG module which are writable/creatable/deletable (i.e., config true, which is the default). These data nodes may be considered sensitive or vulnerable in some network environments. Write operations (e.g., <edit-config>) to these data nodes without proper protection can have a negative effect on network operations.

TBD: List specific Subtrees and data nodes and their sensitivity/ vulnerability.

10. Manageability Considerations

10.1. Control of Function and Policy

10.2. Information and Data Models

10.3. Liveness Detection and Monitoring

10.4. Verify Correct Operations

10.5. Requirements On Other Protocols

10.6. Impact On Network Operations

11. IANA Considerations

This document registers a URI in the "IETF XML Registry" [RFC3688]. Following the format in RFC 3688, the following registration has been made.

URI: urn:ietf:params:xml:ns:yang:ietf-pcep

Registrant Contact: The PCE WG of the IETF.

XML: N/A; the requested URI is an XML namespace.

This document registers a YANG module in the "YANG Module Names" registry [RFC6020].

```
   Name:           ietf-pcep
   Namespace:      urn:ietf:params:xml:ns:yang:ietf-pcep
   Prefix:         pcep
   Reference:      This I-D
```

12. Acknowledgements

   The initial document is based on the PCEP MIB [RFC7420]. Further
   this document structure is based on Routing Yang Module
   [I-D.ietf-netmod-routing-cfg]. We would like to thank the authors of
   aforementioned documents.

13. References

13.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688,
              DOI 10.17487/RFC3688, January 2004,
              <http://www.rfc-editor.org/info/rfc3688>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <http://www.rfc-editor.org/info/rfc5440>.

   [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for
              the Network Configuration Protocol (NETCONF)", RFC 6020,
              DOI 10.17487/RFC6020, October 2010,
              <http://www.rfc-editor.org/info/rfc6020>.

   [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types",
              RFC 6991, DOI 10.17487/RFC6991, July 2013,
              <http://www.rfc-editor.org/info/rfc6991>.

   [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE", draft-ietf-pce-stateful-
              pce-14 (work in progress), March 2016.

   [I-D.ietf-pce-pce-initiated-lsp]
              Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
              Extensions for PCE-initiated LSP Setup in a Stateful PCE
              Model", draft-ietf-pce-pce-initiated-lsp-05 (work in
              progress), October 2015.

[I-D.ietf-pce-lsp-setup-type]
          Sivabalan, S., Medved, J., Minei, I., Crabbe, E., Varga,
          R., Tantsura, J., and J. Hardwick, "Conveying path setup
          type in PCEP messages", draft-ietf-pce-lsp-setup-type-03
          (work in progress), June 2015.

[I-D.ietf-pce-segment-routing]
          Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E.,
          Lopez, V., Tantsura, J., Henderickx, W., and J. Hardwick,
          "PCEP Extensions for Segment Routing", draft-ietf-pce-
          segment-routing-07 (work in progress), March 2016.

13.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
          Element (PCE)-Based Architecture", RFC 4655,
          DOI 10.17487/RFC4655, August 2006,
          <http://www.rfc-editor.org/info/rfc4655>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed.,
          and A. Bierman, Ed., "Network Configuration Protocol
          (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011,
          <http://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure
          Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011,
          <http://www.rfc-editor.org/info/rfc6242>.

[RFC6536] Bierman, A. and M. Bjorklund, "Network Configuration
          Protocol (NETCONF) Access Control Model", RFC 6536,
          DOI 10.17487/RFC6536, March 2012,
          <http://www.rfc-editor.org/info/rfc6536>.

[RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
          Hardwick, "Path Computation Element Communication Protocol
          (PCEP) Management Information Base (MIB) Module",
          RFC 7420, DOI 10.17487/RFC7420, December 2014,
          <http://www.rfc-editor.org/info/rfc7420>.

[I-D.ietf-netmod-routing-cfg]
          Lhotka, L. and A. Lindem, "A YANG Data Model for Routing
          Management", draft-ietf-netmod-routing-cfg-22 (work in
          progress), July 2016.

[I-D.ietf-netmod-rfc6087bis]
          Bierman, A., "Guidelines for Authors and Reviewers of YANG
          Data Model Documents", draft-ietf-netmod-rfc6087bis-06
          (work in progress), March 2016.

   [I-D.ietf-teas-yang-te]
              Saad, T., Gandhi, R., Liu, X., Beeram, V., Shah, H., Chen,
              X., Jones, R., and B. Wen, "A YANG Data Model for Traffic
              Engineering Tunnels and Interfaces", draft-ietf-teas-yang-
              te-03 (work in progress), March 2016.

Appendix A. Contributor Addresses

    Rohit Pobbathi
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: rohit.pobbathi@huawei.com

    Vinod KumarS
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: vinods.kumar@huawei.com

    Zafar Ali
    Cisco Systems
    Canada

    EMail: zali@cisco.com

    Xufeng Liu
    Ericsson
    1595 Spring Hill Road, Suite 500
    Vienna, VA 22182
    USA

    EMail: xufeng.liu@ericsson.com

    Young Lee
    Huawei Technologies
    5340 Legacy Drive, Building 3
    Plano, TX 75023, USA

    Phone: (469) 277-5838
    EMail: leeyoung@huawei.com

    Udayasree Palle
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: udayasree.palle@huawei.com

    Xian Zhang
    Huawei Technologies
    Bantian, Longgang District
    Shenzhen 518129
    P.R.China

    EMail: zhang.xian@huawei.com

    Avantika
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: avantika.sushilkumar@huawei.com

Authors' Addresses

    Dhruv Dhody (editor)
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: dhruv.ietf@gmail.com

    Jonathan Hardwick
    Metaswitch
    100 Church Street
    Enfield EN2 6BQ
    UK

    EMail: jonathan.hardwick@metaswitch.com

    Vishnu Pavan Beeram
    Juniper Networks
    USA

    EMail: vbeeram@juniper.net

    Jeff Tantsura
    USA

    EMail: jefftant@gmail.com

PCE Working Group   S. Sivabalan
Internet-Draft   C. Filsfils
Intended status: Standards Track   S. Previdi
Expires: October 25, 2015   Cisco Systems, Inc.
                                                              J. Tantsura
                                                                 Ericsson
                                                               J. Hardwick
                                                       Metaswitch Networks
                                                                M. Nanduri
                                                     Microsoft Corporation
                                                           April 23, 2015

              Carrying Binding Label/Segment-ID in PCE-based Networks.
                   draft-sivabalan-pce-binding-label-sid-00.txt

Abstract

   It is possible to associate a binding label to RSVP-TE signaled
   Traffic Engineering Label Switching Path or binding Segment-ID (SID)
   to Segment Routed Traffic Engineering path. Such a binding label/SID
   can be used by an upstream node for steering traffic into the
   appropriate TE path to enforce TE policies. This document proposes
   an approach for reporting binding label/SID to Path Computation
   Element (PCE) for supporting PCE-based Traffic Engineering policies.

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 [RFC2119].

Copyright Notice

Table of Contents

1. Introduction

   A PCE can compute Traffic Engineering paths (TE paths) through a
   network that are subject to various constraints. Currently, TE paths
   are either set up using the RSVP-TE signaling protocol or Segment
   Routed (SR). We refer to such paths as RSVP-TE paths and SR-TE paths
   respectively in this document.

   Similar to assigning label to a Forwarding Equivalence Class (FEC)
   via Label Distribution Protocol (LDP), a binding label can be
   assigned to a RSVP-TE LSP. If the topmost label of an incoming
   packet is the binding label, the packet is steered onto the RSVP-TE
   LSP. As such, any upstream node can use binding labels to steer the
   packets that it originates to appropriate TE LSPs to enforce TE
   policy. Similarly, a binding SID (see
   [I-D.ietf-isis-segment-routing-extensions] and
   [I-D.ietf-ospf-segment-routing-extensions]) can be used to enforce TE
   policy with SR-TE path. Note that if an SR-TE path is represented as
   a forwarding-adjacency, then the corresponding adjacency SID can be
   used as the binding SID. In such case, the path is advertised using
   the routing protocols as described in [RFC5440]. The binding SID
   provides an alternate mechanism without additional overhead on
   routing protocols.

   [RFC5440] describes the Path Computation Element Protocol (PCEP) for
   communication between a Path Computation Client (PCC) and a PCE or
   between a pair of PCEs.[I-D.ietf-pce-stateful-pce] specifies
   extension to PCEP that allows a PCC to delegate its LSPs to a PCE.
   The PCE can then update the state of LSPs delegated to it.
   [I-D.ietf-pce-pce-initiated-lsp] specifies a mechanism allowing a PCE
   to dynamically instantiate an LSP on a PCC by sending the path and
   characteristics of the LSP. The PCEP extension to setup and maintain
   SR-TE paths is specified in [I-D.ietf-pce-segment-routing].

   Binding label/SID has local significance to the ingress node of the
   corresponding TE path. When a stateful PCE is deployed for setting
   up TE paths, it may be desirable to report the binding label or SID
   to the PCE for the purpose of enforcing end-to-end TE policy. A
   sample Data Center (DC) use-case is illustrated in the following
   diagram. In the MPLS DC network, an SR LSP (without traffic
   engineering) is established using a prefix SID advertised by BGP (see
   [I-D.keyupate-idr-bgp-prefix-sid]). In IP/MPLS WAN, an SR-TE LSP is
   setup using the PCE. The list of SIDs of the SR-TE LSP is {A, B, C,
   D}. The gateway node 1 (which is the PCC) allocates a binding SID X
   and reports it to the PCE. In order for the access node to steer the
   traffic over the SR-TE LSP, the PCE passes the SID stack {Y, X} where
   Y is the prefix SID of the gateway node-1 to the access node. In the
   absence of the binding SID X, the PCE should pass the SID stack {Y,

A, B, C, D} to the access node. This example also illustrates the
additional benefit of using the binding SID to reduce the number of
SIDs imposed on the access nodes with a limited forwarding capacity.

```
               SID stack
               {Y, X}                   +-----+
          _ _ _ _ _ _ _ _ _ _ _ _ _ _ _| PCE |
          |                             +-----+
          |                                ^
          |                                |
          |                                | Binding
          |          .-----.               | SID (X)     .-----.
          |        (         )             |           (         )
          V       .--(         )--.        |          .--(         )--.
      +------+   (                 ) +-------+   (                 ) +-------+
      |Access|_(   MPLS DC Network  )_|Gateway|_(    IP/MPLS WAN    )_|Gateway|
      | Node | (  ==============>   ) |Node-1 | (  ===============> ) |Node-2 |
      +------+  (    SR path        ) +-------+ (     SR-TE path    ) +-------+
               '--(         )--'     Prefix    '--(         )--'
                  (         )        SID of        (         )
                   '-----'          Node-1          '-----'
                                    is Y            SIDs for SR-TE LSP:
                                                    {A, B, C, D}
```

Figure 1: A sample Use-case of Binding SID

In this document, we introduce a new OPTIONAL TLV that a PCC can use
in order to report the binding label associated with a TE LSP. This
TLV is intended for TE LSPs established using RSVP-TE, SR, or any
other future method. Also, in the case of SR-TE LSPs, the TLV can
carry an MPLS label binding (for SR-TE path with MPLS data-plane) or
a binding SID (e.g., IPv6 address for SR-TE paths with IPv6 data-
plane). However, use of this TLV for non-MPLS label binding will be
described in separate document(s).

2. Terminology

   The following terminologies are used in this document:

   LER: Label Edge Router.

   LSP: Label Switched Path.

LSR: Label Switching Router.

PCC: Path Computation Client.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

SID: Segment ID.

SR: Segment Routing.

TLV: Type, Length, and Value.

3. Path Binding TLV

The new optional TLV is called "TE-PATH-BINDING TLV" whose format is
shown in the diagram below is defined to carry binding label or SID
for a TE path. This TLV is associated with the LSP object specified
in ([I-D.ietf-pce-stateful-pce]). The type of this TLV is to be
allocated by IANA.
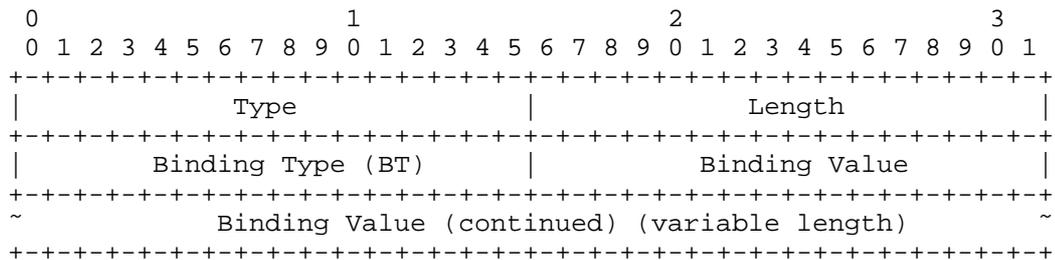
```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |             Type              |            Length             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |      Binding Type (BT)        |          Binding Value        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~          Binding Value (continued) (variable length)         ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: TE-PATH-BINDING TLV

TE-PATH-BINDING TLV is a generic TLV such that it is able to carry
MPLS label binding as well as other types of future bindings (e.g.,
IPv6 SR path). The one octet Binding Type (BT) field identifies the
type of binding included in the TLV. This document specifies the
following BT value:

o BT = 0: MPLS label (default).

4. Operation

The binding value is allocated by PCC and reported to PCE via PCRpt
message. If a PCE does not recognize the TE-PATH-BINDING TLV, it

MUST ignore the TLV in accordance with ([RFC5440]). If a PCE
recognizes the TLV but does not support the TLV, it MUST send PCErr
with Error-Type = 2 (Capability not supported).

If a TE-PATH-BINDING TLV is absent in PCRpt message, PCE MUST assume
that the corresponding LSP does not have any binding. If there are
more than one PATH-BINDING TLVs, only the first TLV MUST be processed
and the rest MUST be silently ignored. If PCE recognizes an invalid
binding value (e.g., label value from the reserved label space when
MPLS label binding is used), it MUST send the PCE error message with
Error-Type = 10 ("Reception of an invalid object") and Error Value =
TBD ("Bad label value") as specified in
[I-D.ietf-pce-segment-routing].

If a PCC receives TE-PATH-BINDING TLV in any message, it MUST close
the corresponding PCEP session with the reason "Reception of a
malformed PCEP message" according ([RFC5440]). Similarly, if a PCE
receives a TE-PATH-BINDING TLV in any message other than a PCRpt or
if the TE-PATH-BINDING TLV is associated with any object other than
LSP object, the PCE MUST close the corresponding PCEP session with
the reason "Reception of a malformed PCEP message" according
([RFC5440]).

If a PCC wants to withdraw or modify a previously reported binding
value, it MUST send a PCRpt message without any TE-PATH-BINDING TLV
and with the TE-PATH-BINDING TLV containing the new binding value
respectively.

5. Security Considerations

   No additional security measure is required.

6. IANA Considerations

   IANA is requested to allocate a new TLV type (recommended value is
   31)for TE-PATH-BINDING TLV specified in this document.

   This document requests that a registry is created to manage the value
   of the Binding Type field in the TE-PATH-BINDING TLV.

```
Value    Description              Reference

  0      MPLS Label               This document
```

7. Acknowledgements

8. Normative References

   [I-D.ietf-isis-segment-routing-extensions]
           Previdi, S., Filsfils, C., Bashandy, A., Gredler, H.,
           Litkowski, S., Decraene, B., and J. Tantsura, "IS-IS
           Extensions for Segment Routing",
           draft-ietf-isis-segment-routing-extensions-03 (work in
           progress), October 2014.

   [I-D.ietf-ospf-segment-routing-extensions]
           Psenak, P., Previdi, S., Filsfils, C., Gredler, H.,
           Shakir, R., Henderickx, W., and J. Tantsura, "OSPF
           Extensions for Segment Routing",
           draft-ietf-ospf-segment-routing-extensions-04 (work in
           progress), February 2015.

   [I-D.ietf-pce-pce-initiated-lsp]
           Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
           Extensions for PCE-initiated LSP Setup in a Stateful PCE
           Model", draft-ietf-pce-pce-initiated-lsp-04 (work in
           progress), April 2015.

   [I-D.ietf-pce-segment-routing]
           Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E.,
           Lopez, V., Tantsura, J., Henderickx, W., and J. Hardwick,
           "PCEP Extensions for Segment Routing",
           draft-ietf-pce-segment-routing-03 (work in progress),
           April 2015.

   [I-D.ietf-pce-stateful-pce]
           Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
           Extensions for Stateful PCE",
           draft-ietf-pce-stateful-pce-11 (work in progress),
           April 2015.

   [I-D.keyupate-idr-bgp-prefix-sid]
           Patel, K., Ray, S., Previdi, S., and C. Filsfils, "Segment
           Routing Prefix SID extensions for BGP",
           draft-keyupate-idr-bgp-prefix-sid-01 (work in progress),
           April 2015.

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

   [RFC4206] Kompella, K. and Y. Rekhter, "Label Switched Paths (LSP)
              Hierarchy with Generalized Multi-Protocol Label Switching
              (GMPLS) Traffic Engineering (TE)", RFC 4206, October 2005.

   [RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element
              (PCE) Communication Protocol (PCEP)", RFC 5440,
              March 2009.

Authors' Addresses

   Siva Sivabalan
   Cisco Systems, Inc.
   2000 Innovation Drive
   Kanata, Ontario K2K 3E8
   Canada

   Email: msiva@cisco.com

   Clarence Filsfils
   Cisco Systems, Inc.
   Pegasus Parc
   De kleetlaan 6a, DIEGEM BRABANT 1831
   BELGIUM

   Email: cfilsfil@cisco.com

   Stefano Previdi
   Cisco Systems, Inc.
   Via Del Serafico, 200
   Rome, Rome 00142
   Italy

   Email: sprevidi@cisco.com

Jeff Tantsura
Ericsson
300 Holger Way
San Jose, CA 95134
USA

Email: jeff.tantsura@ericsson.com

Jonathan Hardwick
Metaswitch Networks
100 Church Street
Enfield, Middlesex
UK

Email: Jonathan.Hardwick@metaswitch.com

Mohan Nanduri
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
USA

Email: mnanduri@microsoft.com

PCE Working Group   S. Sivabalan
Internet-Draft   C. Filsfils
Intended status: Standards Track   Cisco Systems, Inc.
Expires: January 9, 2020   J. Tantsura
                                                           Apstra, Inc.
                                                            J. Hardwick
                                                   Metaswitch Networks
                                                            S. Previdi
                                                                  C. Li
                                                   Huawei Technologies
                                                          July 8, 2019

            Carrying Binding Label/Segment-ID in PCE-based Networks.
                  draft-sivabalan-pce-binding-label-sid-07

Abstract

   In order to provide greater scalability, network opacity, and service
   independence, SR utilizes a Binding Segment Identifier (BSID). It is
   possible to associate a BSID to RSVP-TE signaled Traffic Engineering
   Label Switching Path or binding Segment-ID (SID) to Segment Routed
   (SR) Traffic Engineering path. Such a binding label/SID can be used
   by an upstream node for steering traffic into the appropriate TE path
   to enforce SR policies. This document proposes an approach for
   reporting binding label/SID to Path Computation Element (PCE) for
   supporting PCE-based Traffic Engineering policies.

Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in BCP
   14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on January 9, 2020.

Table of Contents

1. Introduction

   A PCE can compute Traffic Engineering paths (TE paths) through a
   network that are subject to various constraints. Currently, TE paths
   are either set up using the RSVP-TE signaling protocol or Segment
   Routing (SR). We refer to such paths as RSVP-TE paths and SR-TE
   paths respectively in this document.

   As per [RFC8402] SR allows a headend node to steer a packet flow
   along any path. The headend node is said to steer a flow into an
   Segment Routing Policy (SR Policy). Further, as per
   [I-D.ietf-spring-segment-routing-policy], an SR Policy is a framework
   that enables instantiation of an ordered list of segments on a node
   for implementing a source routing policy with a specific intent for
   traffic steering from that node.

   As described in [RFC8402], Binding Segment Identifier (BSID) is bound
   to an Segment Routed (SR) Policy, instantiation of which may involve
   a list of SIDs. Any packets received with an active segment equal to
   BSID are steered onto the bound SR Policy. A BSID may be either a
   local (SRLB) or a global (SRGB) SID. As per Section 6.4 of
   [I-D.ietf-spring-segment-routing-policy] a BSID can also be
   associated with any type of interfaces or tunnel to enable the use of
   a non-SR interface or tunnels as segments in a SID-list.

   [RFC5440] describes the Path Computation Element Protocol (PCEP) for
   communication between a Path Computation Client (PCC) and a PCE or
   between a pair of PCEs as per [RFC4655]. [RFC8231] specifies
   extension to PCEP that allows a PCC to delegate its LSPs to a
   stateful PCE. A stateful PCE can then update the state of LSPs
   delegated to it. [RFC8281] specifies a mechanism allowing a PCE to
   dynamically instantiate an LSP on a PCC by sending the path and
   characteristics. The PCEP extension to setup and maintain SR-TE
   paths is specified in [I-D.ietf-pce-segment-routing].

   [I-D.ietf-pce-segment-routing] provides a mechanism for a network
   controller (acting as a PCE) to instantiate candidate paths for an SR
   Policy onto a head-end node (acting as a PCC) using PCEP. For more
   information on the SR Policy Architecture, see
   [I-D.ietf-spring-segment-routing-policy].

   Binding label/SID has local significance to the ingress node of the
   corresponding TE path. When a stateful PCE is deployed for setting
   up TE paths, it may be desirable to report the binding label or SID
   to the stateful PCE for the purpose of enforcing end-to-end TE/SR
   policy. A sample Data Center (DC) use-case is illustrated in the
   following diagram. In the MPLS DC network, an SR LSP (without
   traffic engineering) is established using a prefix SID advertised by

BGP (see [I-D.ietf-idr-bgp-prefix-sid]). In IP/MPLS WAN, an SR-TE
LSP is setup using the PCE. The list of SIDs of the SR-TE LSP is {A,
B, C, D}. The gateway node 1 (which is the PCC) allocates a binding
SID X and reports it to the PCE. In order for the access node to
steer the traffic over the SR-TE LSP, the PCE passes the SID stack
{Y, X} where Y is the prefix SID of the gateway node-1 to the access
node. In the absence of the binding SID X, the PCE should pass the
SID stack {Y, A, B, C, D} to the access node. This example also
illustrates the additional benefit of using the binding SID to reduce
the number of SIDs imposed on the access nodes with a limited
forwarding capacity.

```
                  SID stack
                  {Y, X}                    +-----+
  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _| PCE |
  |                                         +-----+
  |                                            ^
  |                                            |   Binding
  |                 .-----.                    |   SID (X)      .-----.
  |                (       )                    |             (        )
  V             .--(         )--.               |           .--(          )--.
+------+       (                 ) +-------+   ( IP/MPLS WAN    ) +-------+
|Access|_(  MPLS DC Network  )_|Gateway|_(                 )_|Gateway|
| Node | ( ==============> ) |Node-1 | ( ================> ) |Node-2 |
+------+ (     SR path      ) +-------+ (     SR-TE path   ) +-------+
          '--(          )--'  Prefix    '--(         )--'
             (        )       SID of       (        )
              '-----'         Node-1        '-----'
                              is Y          SIDs for SR-TE LSP:
                                            {A, B, C, D}
```
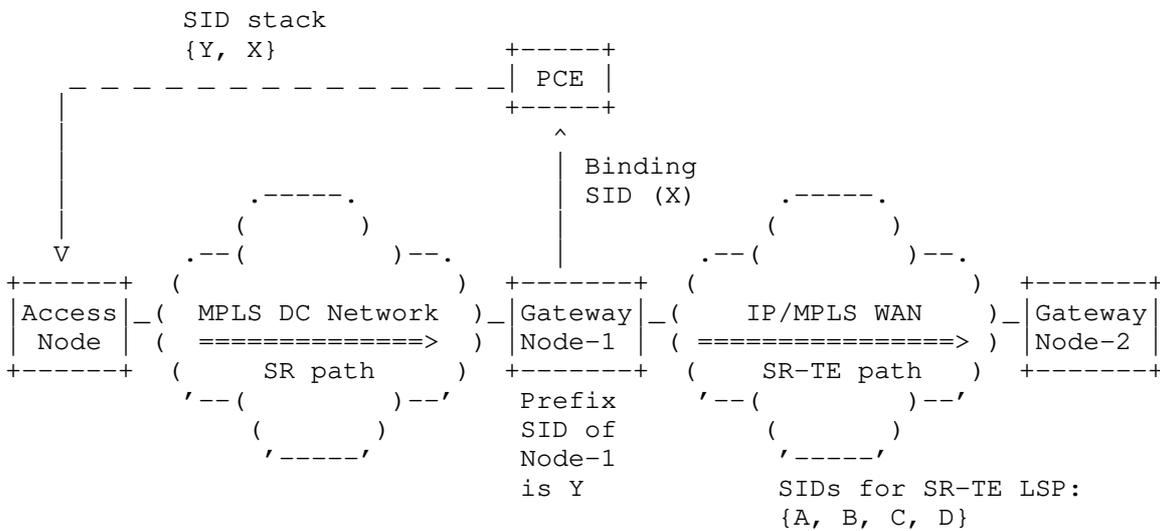
                Figure 1: A sample Use-case of Binding SID

   A PCC could report the binding label/SID allocated by it to the
   stateful PCE via Path Computation State Report (PCRpt) message. It
   is also possible for a stateful PCE to request a PCC to allocate a
   specific binding label/SID by sending an Path Computation Update
   Request (PCUpd) message. If the PCC can successfully allocate the
   specified binding value, it reports the binding value to the PCE.
   Otherwise, the PCC sends an error message to the PCE indicating the
   cause of the failure. A local policy or configuration at the PCC
   SHOULD dictate if the binding label/SID needs to be assigned.

   In this document, we introduce a new OPTIONAL TLV that a PCC can use
   in order to report the binding label/SID associated with a TE LSP, or

a PCE to request a PCC to allocate a specific binding label/SID
value. This TLV is intended for TE LSPs established using RSVP-TE,
SR, or any other future method. Also, in the case of SR-TE LSPs, the
TLV can carry a binding MPLS label (for SR-TE path with MPLS data-
plane) or a binding IPv6 SID (e.g., IPv6 address for SR-TE paths with
IPv6 data-plane). Binding value means either MPLS label or SID
throughout this document.

Additionally, to support the PCE based central controller [RFC8283]
operation where the PCE would take responsibility for managing some
part of the MPLS label space for each of the routers that it
controls, the PCE could directly make the binding label/SID
allocation and inform the PCC. See
[I-D.ietf-pce-pcep-extension-for-pce-controller] for details.

2. Terminology

The following terminologies are used in this document:

BSID: Binding Segment Identifier.

LER: Label Edge Router.

LSP: Label Switched Path.

LSR: Label Switching Router.

PCC: Path Computation Client.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

RSVP-TE: Resource ReserVation Protocol-Traffic Engineering.

SID: Segment Identifier.

SR: Segment Routing.

SRGB: Segment Routing Global Block.

SRLB: Segment Routing Local Block.

TLV: Type, Length, and Value.

3. Path Binding TLV

   The new optional TLV is called "TE-PATH-BINDING TLV" whose format is
   shown in the diagram below is defined to carry binding label or SID
   for a TE path. This TLV is associated with the LSP object specified
   in ([RFC8231]). The type of this TLV is to be allocated by IANA.
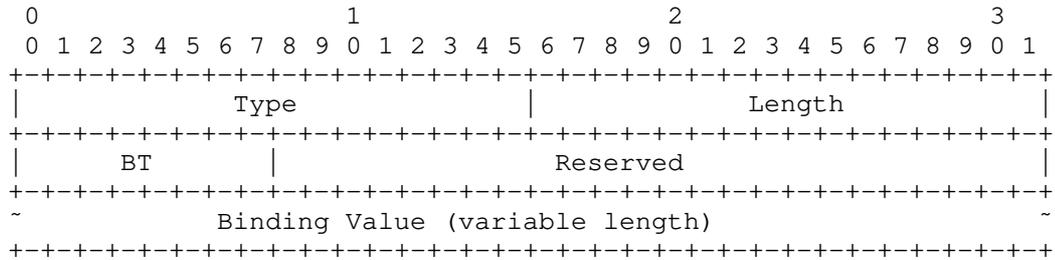
```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type               |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     BT        |              Reserved                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~             Binding Value (variable length)                   ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

                    Figure 2: TE-PATH-BINDING TLV

   TE-PATH-BINDING TLV is a generic TLV such that it is able to carry
   MPLS label binding as well as SRV6 Binding SID. It is formatted
   according to the rules specified in [RFC5440].

   Binding Type (BT): A one byte field identifies the type of binding
   included in the TLV. This document specifies the following BT
   values:

   o BT = 0: The binding value is an MPLS label carried in the format
      specified in [RFC5462] where only the label value is valid, and
      other fields (TC, S, and TTL) fields MUST be considered invalid.
      The Length MUST be set to 6.

   o BT = 1: Similar to the case where BT is 0 except that all the
      fields on the MPLS label entry are set on transmission. However,
      the receiver MAY choose to override TC, S, and TTL values
      according its local policy.

   o BT = 2: The binding value is a SRv6 SID with a format of an 16
      byte IPv6 address, representing the binding SID for SRv6.

   Reserved: MUST be set to 0 while sending and ignored on receipt.

   Binding Value: A variable length field, padded with trailing zeros to
   a 4-byte boundary. For the BT as 0, the 20 bits represents the MPLS
   label. For the BT as 1, the 32-bits represents the label stack entry
   as per [RFC5462]. For the BT as 2, the 128-bits represent the SRv6
   SID.

4. Operation

   The binding value is allocated by the PCC and reported to a PCE via
   PCRpt message. If a PCE does not recognize the TE-PATH-BINDING TLV,
   it MUST ignore the TLV in accordance with ([RFC5440]). If a PCE
   recognizes the TLV but does not support the TLV, it MUST send PCErr
   with Error-Type = 2 (Capability not supported).

   If a TE-PATH-BINDING TLV is absent in PCRpt message, PCE MUST assume
   that the corresponding LSP does not have any binding. If there are
   more than one TE-PATH-BINDING TLVs, only the first TLV MUST be
   processed and the rest MUST be silently ignored. If a PCE recognizes
   an invalid binding value (e.g., label value from the reserved label
   space when MPLS label binding is used), it MUST send the PCErr
   message with Error-Type = 10 ("Reception of an invalid object") and
   Error Value = TBD ("Bad label value") as specified in
   [I-D.ietf-pce-segment-routing].

   If a PCE requires a PCC to allocate a specific binding value, it may
   do so by sending a PCUpd or PCInitiate message containing a TE-PATH-
   BINDING TLV. If the value can be successfully allocated, the PCC
   reports the binding value to the PCE. If the PCC considers the
   binding value specified by the PCE invalid, it MUST send a PCErr
   message with Error-Type = TBD ("Binding label/SID failure") and Error
   Value = TBD ("Invalid SID"). If the binding value is valid, but the
   PCC is unable to allocate the binding value, it MUST send a PCErr
   message with Error-Type = TBD ("Binding label/SID failure") and Error
   Value = TBD ("Unable to allocate the specified label/SID").

   If a PCC receives TE-PATH-BINDING TLV in any message other than PCUpd
   or PCInitiate, it MUST close the corresponding PCEP session with the
   reason "Reception of a malformed PCEP message" (according to
   [RFC5440]). Similarly, if a PCE receives a TE-PATH-BINDING TLV in
   any message other than a PCRpt or if the TE-PATH-BINDING TLV is
   associated with any object other than LSP object, the PCE MUST close
   the corresponding PCEP session with the reason "Reception of a
   malformed PCEP message" (according to [RFC5440]).

   If a PCC wishes to withdraw or modify a previously reported binding
   value, it MUST send a PCRpt message without any TE-PATH-BINDING TLV
   or with the TE-PATH-BINDING TLV containing the new binding value
   respectively.

   If a PCE wishes to modify a previously requested binding value, it
   MUST send a PCUpd message with TE-PATH-BINDING TLV containing the new
   binding value. Absence of TE-PATH-BINDING TLV in PCUpd message means
   that the PCE does not specify a binding value in which case the
   binding value allocation is governed by the PCC's local policy.

If a PCC receives a valid binding value from a PCE which is different
than the current binding value, it MUST try to allocate the new
value. If the new binding value is successfully allocated, the PCC
MUST report the new value to the PCE. Otherwise, it MUST send a
PCErr message with Error-Type = TBD ("Binding label/SID failure") and
Error Value = TBD ("Unable to allocate the specified label/SID").

In some cases, a stateful PCE can request the PCC to allocate a
binding value. It may do so by sending a PCUpd message containing an
empty TE-PATH-BINDING TLV, i.e., no binding value is specified
(making the length field of the TLV as 2). A PCE can also make the
request PCC to allocate a binding at the time of initiation by
sending a PCInitiate message with an empty TE-PATH-BINDING TLV.

5. Binding SID in SR-ERO

   In PCEP messages, LSP route information is carried in the Explicit
   Route Object (ERO), which consists of a sequence of subobjects.
   [I-D.ietf-pce-segment-routing] defines a new ERO subobject "SR-ERO
   subobject" capable of carrying a SID as well as the identity of the
   node/adjacency (NAI) represented by the SID. The NAI Type (NT) field
   indicates the type and format of the NAI contained in the SR-ERO. In
   case of binding SID, the NAI MUST NOT be included and NT MUST be set
   to zero. So as per Section 5.2.1 of [I-D.ietf-pce-segment-routing],
   for NT=0, the F bit MUST be 1, the S bit needs to be zero and the
   Length MUST be 8. Further the M bit MUST be set. If these
   conditions are not met, the entire ERO MUST be considered invalid and
   a PCErr message is sent with Error-Type = 10 ("Reception of an
   invalid object") and Error-Value = 11 ("Malformed object").

6. Binding SID in SRv6-ERO/

   [I-D.ietf-pce-segment-routing] defines a new ERO subobject "SRv6-ERO
   subobject" for SRv6 SID. The NAI MUST NOT be included and NT MUST be
   set to zero. So as per Section 5.2.1 of
   [I-D.ietf-pce-segment-routing], for NT=0, the F bit MUST be 1, the S
   bit needs to be zero and the Length MUST be 24. If these conditions
   are not met, the entire ERO is considered invalid and a PCErr message
   is sent with Error-Type = 10 ("Reception of an invalid object") and
   Error-Value = 11 ("Malformed object") (as per
   [I-D.ietf-pce-segment-routing]).

7. Implementation Status

   [Note to the RFC Editor - remove this section before publication, as
   well as remove the reference to RFC 7942.]

This section records the status of known implementations of the
protocol defined by this specification at the time of posting of this
Internet-Draft, and is based on a proposal described in [RFC7942].
The description of implementations in this section is intended to
assist the IETF in its decision processes in progressing drafts to
RFCs. Please note that the listing of any individual implementation
here does not imply endorsement by the IETF. Furthermore, no effort
has been spent to verify the information presented here that was
supplied by IETF contributors. This is not intended as, and must not
be construed to be, a catalog of available implementations or their
features. Readers are advised to note that other implementations may
exist.

According to [RFC7942], "this will allow reviewers and working groups
to assign due consideration to documents that have the benefit of
running code, which may serve as evidence of valuable experimentation
and feedback that have made the implemented protocols more mature.
It is up to the individual working groups to use this information as
they see fit".

7.1. Huawei

   o Organization: Huawei

   o Implementation: Huawei's Router and Controller

   o Description: An experimental code-point is used and plan to
      request early code-point allocation from IANA after WG adoption.

   o Maturity Level: Production

   o Coverage: Full

   o Contact: mahendrasingh@huawei.com

8. Security Considerations

   The security considerations described in [RFC5440], [RFC8231],
   [RFC8281] and [I-D.ietf-pce-segment-routing] are applicable to this
   specification. No additional security measure is required.

   As described [I-D.ietf-pce-segment-routing], SR allows a network
   controller to instantiate and control paths in the network. A rouge
   PCE can manipulate binding SID allocations to move traffic around for
   some other LSPs that uses BSID in its SR-ERO.

   Thus, as per [RFC8231], it is RECOMMENDED that these PCEP extensions
   only be activated on authenticated and encrypted sessions across PCEs

   and PCCs belonging to the same administrative authority, using
   Transport Layer Security (TLS) [RFC8253], as per the recommendations
   and best current practices in [RFC7525] (unless explicitly set aside
   in [RFC8253]).

9. Manageability Considerations

   All manageability requirements and considerations listed in
   [RFC5440], [RFC8231], and [I-D.ietf-pce-segment-routing] apply to
   PCEP protocol extensions defined in this document. In addition,
   requirements and considerations listed in this section apply.

9.1. Control of Function and Policy

   A PCC implementation SHOULD allow the operator to configure the
   policy based on which PCC needs to allocates the binding label/SID.

9.2. Information and Data Models

   The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to
   include policy configuration for binding label/SID allocation.

9.3. Liveness Detection and Monitoring

   Mechanisms defined in this document do not imply any new liveness
   detection and monitoring requirements in addition to those already
   listed in [RFC5440].

9.4. Verify Correct Operations

   Mechanisms defined in this document do not imply any new operation
   verification requirements in addition to those already listed in
   [RFC5440], [RFC8231], and [I-D.ietf-pce-segment-routing].

9.5. Requirements On Other Protocols

   Mechanisms defined in this document do not imply any new requirements
   on other protocols.

9.6. Impact On Network Operations

   Mechanisms defined in [RFC5440], [RFC8231], and
   [I-D.ietf-pce-segment-routing] also apply to PCEP extensions defined
   in this document. Further, the mechanism described in this document
   can help the operator to request control of the LSPs at a particular
   PCE.

10. IANA Considerations

10.1. PCEP TLV Type Indicators

   This document defines a new PCEP TLV; IANA is requested to make the
   following allocations from the "PCEP TLV Type Indicators" sub-
   registry of the PCEP Numbers registry, as follows:

               Value   Name   Reference

                TBD   TE-PATH-BINDING   This document

10.1.1. TE-PATH-BINDING TLV

   IANA is requested to create a sub-registry to manage the value of the
   Binding Type field in the TE-PATH-BINDING TLV.

               Value   Description   Reference

                0   MPLS Label   This document
                1   MPLS Label Stack   This document
                        Entry
                2   SRv6 SID   This document

10.2. PCEP Error Type and Value

   This document defines a new Error-type and Error-Values for the PCErr
   message. IANA is requested to allocate new error-type and error-
   values within the "PCEP-ERROR Object Error Types and Values"
   subregistry of the PCEP Numbers registry, as follows:

   Error-Type   Meaning
   ----------   -------
   TBD   Binding label/SID failure:

                 Error-value = TBD: Invalid SID
                 Error-value = TBD: Unable to allocate
                                             the specified
                                             label/SID

11. Acknowledgements

   We like to thank Milos Fabian for his valuable comments.

12. References

12.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <https://www.rfc-editor.org/info/rfc5440>.

   [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching
              (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic
              Class" Field", RFC 5462, DOI 10.17487/RFC5462, February
              2009, <https://www.rfc-editor.org/info/rfc5462>.

   [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre,
              "Recommendations for Secure Use of Transport Layer
              Security (TLS) and Datagram Transport Layer Security
              (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May
              2015, <https://www.rfc-editor.org/info/rfc7525>.

   [RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running
              Code: The Implementation Status Section", BCP 205,
              RFC 7942, DOI 10.17487/RFC7942, July 2016,
              <https://www.rfc-editor.org/info/rfc7942>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for Stateful PCE", RFC 8231,
              DOI 10.17487/RFC8231, September 2017,
              <https://www.rfc-editor.org/info/rfc8231>.

   [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody,
              "PCEPS: Usage of TLS to Provide a Secure Transport for the
              Path Computation Element Communication Protocol (PCEP)",
              RFC 8253, DOI 10.17487/RFC8253, October 2017,
              <https://www.rfc-editor.org/info/rfc8253>.

   [RFC8281]   Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path
               Computation Element Communication Protocol (PCEP)
               Extensions for PCE-Initiated LSP Setup in a Stateful PCE
               Model", RFC 8281, DOI 10.17487/RFC8281, December 2017,
               <https://www.rfc-editor.org/info/rfc8281>.

   [RFC8402]   Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L.,
               Decraene, B., Litkowski, S., and R. Shakir, "Segment
               Routing Architecture", RFC 8402, DOI 10.17487/RFC8402,
               July 2018, <https://www.rfc-editor.org/info/rfc8402>.

   [I-D.ietf-pce-segment-routing]
               Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W.,
               and J. Hardwick, "PCEP Extensions for Segment Routing",
               draft-ietf-pce-segment-routing-16 (work in progress),
               March 2019.

12.2. Informative References

   [RFC4655]   Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
               Element (PCE)-Based Architecture", RFC 4655,
               DOI 10.17487/RFC4655, August 2006,
               <https://www.rfc-editor.org/info/rfc4655>.

   [RFC8283]   Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An
               Architecture for Use of PCE and the PCE Communication
               Protocol (PCEP) in a Network with Central Control",
               RFC 8283, DOI 10.17487/RFC8283, December 2017,
               <https://www.rfc-editor.org/info/rfc8283>.

   [I-D.ietf-spring-segment-routing-policy]
               Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d.,
               bogdanov@google.com, b., and P. Mattes, "Segment Routing
               Policy Architecture", draft-ietf-spring-segment-routing-
               policy-03 (work in progress), May 2019.

   [I-D.ietf-idr-bgp-prefix-sid]
               Previdi, S., Filsfils, C., Lindem, A., Sreekantiah, A.,
               and H. Gredler, "Segment Routing Prefix SID extensions for
               BGP", draft-ietf-idr-bgp-prefix-sid-27 (work in progress),
               June 2018.

   [I-D.ietf-pce-pcep-extension-for-pce-controller]
               Zhao, Q., Li, Z., Negi, M., and C. Zhou, "PCEP Procedures
               and Protocol Extensions for Using PCE as a Central
               Controller (PCECC) of LSPs", draft-ietf-pce-pcep-
               extension-for-pce-controller-01 (work in progress),
               February 2019.

   [I-D.ietf-pce-pcep-yang]
              Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A
              YANG Data Model for Path Computation Element
              Communications Protocol (PCEP)", draft-ietf-pce-pcep-
              yang-12 (work in progress), July 2019.

Appendix A. Contributor Addresses

   Dhruv Dhody
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   EMail: dhruv.ietf@gmail.com

   Mahendra Singh Negi
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   EMail: mahendrasingh@huawei.com

Authors' Addresses

   Siva Sivabalan
   Cisco Systems, Inc.
   2000 Innovation Drive
   Kanata, Ontario K2K 3E8
   Canada

   EMail: msiva@cisco.com

   Clarence Filsfils
   Cisco Systems, Inc.
   Pegasus Parc
   De kleetlaan 6a, DIEGEM BRABANT 1831
   BELGIUM

   EMail: cfilsfil@cisco.com

   Jeff Tantsura
   Apstra, Inc.

   EMail: jefftant.ietf@gmail.com

   Jonathan Hardwick
   Metaswitch Networks
   100 Church Street
   Enfield, Middlesex
   UK

   EMail: Jonathan.Hardwick@metaswitch.com

   Stefano Previdi
   Huawei Technologies

   EMail: stefano@previdi.net

   Cheng Li
   Huawei Technologies
   Huawei Campus, No. 156 Beiqing Rd.
   Beijing 100095
   China

   EMail: chengli13@huawei.com

      IGP extension for PCEP security capability support in the PCE discovery
                draft-wu-pce-discovery-pceps-support-03

Abstract

   When a Path Computation Element (PCE) is a Label Switching Router
   (LSR) participating in the Interior Gateway Protocol (IGP), or even a
   server participating in IGP, its presence and path computation
   capabilities can be advertised using IGP flooding. The IGP
   extensions for PCE discovery (RFC 5088 and RFC 5089) define a method
   to advertise path computation capabilities using IGP flooding for
   OSPF and IS-IS respectively. However these specifications lack a
   method to advertise PCEP security (e.g., Transport Layer
   Security(TLS)) support capability.

   This document proposes new capability flag bit for PCE-CAP-FLAGS sub-
   TLV that can be announced as attribute in the IGP advertisement to
   distribute PCEP security support information.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   As described in [RFC5440], PCEP communication privacy is one
   importance issue, as an attacker that intercepts a Path Computation
   Element (PCE) message could obtain sensitive information related to
   computed paths and resources.

   Among the possible solutions mentioned in these documents, Transport
   Layer Security (TLS) [RFC5246] provides support for peer
   authentication, and message encryption and integrity. In order for a
   Path Computation Client(PCC) to begin a connection with a PCE server
   using TLS, PCC SHOULD know whether PCE server supports TLS as a
   secure transport.

   [RFC5088] and [RFC5089] define a method to advertise path computation
   capabilities using IGP flooding for OSPF and IS-IS respectively.
   However [RFC5088] and [RFC5089] lacks a method to advertise PCEP
   security (e.g., TLS) support capability.

This document proposes new capability flag bits for PCE-CAP-FLAGS
sub- TLV that can be announced as attributes in the IGP advertisement
(defined in [RFC5088] and [RFC5089]) to distribute PCEP security
support information.

2. 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 [RFC2119].

3. IGP extension for PCEP security capability support

The PCE-CAP-FLAGS sub-TLV is defined in section 4.5 of [RFC5088] and
[RFC5089] as an optional sub-TLV used to advertise PCE capabilities.
In this section, we extend the PCE-CAP-FLAGS sub-TLV to include the
capability and indications that are described for PCEP security
(e.g., TLS) support in the current document.

In the PCE-CAP-FLAGS sub-TLV defined in [RFC5088] and [RFC5089], nine
capability flags defined in [RFC5088] (as per [RFC4657]) and two
capability flags defined [RFC5557], [RFC6006] are included and
follows the following format:

   o TYPE: 5
   o LENGTH: Multiple of 4
   o VALUE: This contains an array of units of 32 bit flags with
      the most significant bit as 0. Each bit represents one PCE
      capability.

and the processing rule of these flag bits are defined in [RFC5088]
and [RFC5089]. In this document, we define three new capability flag
bits that indicate TCP MD5 support, TCP Authentication Option (TCP-
AO) support, PCEP over TLS support respectively as follows:

       Bit   Capability Description
       xx   TCP MD5 support
       xx   TCP AO Support
       xx   PCEP over TLS support

Editor Note: TCP-MD5 is a MUST in RFC5440, do we need a capability
for it

3.1. Use of PCEP security capability support for PCE discovery

TCP MD5, TCP-AO, PCEP over TLS support flag bits are advertised using
IGP flooding.

   o PCE supports TCP MD5: IGP advertisement SHOULD include TCP MD5
      support flag bit.

   o PCE supports TCP-AO: IGP advertisement SHOULD include TCP-AO
      support flag bit.

   o PCE supports TLS: IGP advertisement SHOULD include PCEP over TLS
      support flag bit.

   If PCE supports multiple security mechanisms, it SHOULD include all
   corresponding flag bits in IGP advertisement.

   If the client is looking for connecting with PCE server with TCP-MD5
   support, the client MUST check if TCP-MD5 support flag bit in the
   PCE- CAP-FLAGS sub-TLV is set. If not, the client SHOULD not
   consider this PCE. If the client is looking for connecting with PCE
   server with TCP-AO support, the client MUST check if TCP-AO support
   flag bit in the PCE- CAP-FLAGS sub-TLV is set. If not, the client
   SHOULD not consider this PCE. If the client is looking for
   connecting with PCE server using TLS, the client MUST check if PCEP
   over TLS support flag bit in the PCE-CAP-FLAGS sub-TLV is set. If
   not, the client SHOULD not consider this PCE.

4. Backward Compatibility Consideration

   An LSR that does not support the new IGP PCE capability bits
   specified in this document silently ignores those bits.

   IGP extensions defined in this document do not introduce any new
   interoperability issues.

5. Management Considerations

   A configuration option may be provided for advertising and
   withdrawing PCE security capability via IGP.

6. Security Considerations

   This document raises no new security issues beyond those described in
   [RFC5088] and [RFC5089].

7. IANA Considerations

   IANA is requested to allocate a new bit in "PCE Security Capability
   Flags" registry for PCEP Security support capability.

```
       Bit             Meaning                  Reference
       xx              TCP MD5 support          [This.I.D]
       xx              TCP-AO Support           [This.I.D]
       xx              PCEP over TLS support    [This.I.D]
```

8. References

8.1. Normative References

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

   [RFC5088] Le Roux, JL., "OSPF Protocol Extensions for Path
              Computation Element (PCE) Discovery", RFC 5088, January
              2008.

   [RFC5089] Le Roux, JL., "IS-IS Protocol Extensions for Path
              Computation Element (PCE) Discovery", RFC 5089, January
              2008.

8.2. Informative References

   [RFC4657] Ash, J. and J. Le Roux, "Path Computation Element (PCE)
              Communication Protocol Generic Requirements", RFC 4657,
              September 2006.

   [RFC5246] Dierks, T., "The Transport Layer Security (TLS) Protocol
              Version 1.2", RFC 5246, August 2008.

   [RFC5440] Le Roux, JL., "Path Computation Element (PCE)
              Communication Protocol (PCEP)", RFC 5440, March 2009.

   [RFC5557] Lee, Y., Le Roux, JL., King, D., and E. Oki, "Path
              Computation Element Communication Protocol (PCEP)
              Requirements and Protocol Extensions in Support of Global
              Concurrent Optimization", RFC 5557, July 2009.

   [RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z.,
              and J. Meuric, "Extensions to the Path Computation Element
              Communication Protocol (PCEP) for Point-to-Multipoint
              Traffic Engineering Label Switched Paths", RFC 6006,
              September 2010.

Authors' Addresses

Diego R. Lopez
Telefonica I+D
Spain

Email: diego.r.lopez@telefonica.com

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

Dhruv Dhody
Huawei
Leela Palace
Bangalore, Karnataka 560008
India

Email: dhruv.ietf@gmail.com

Daniel King
Old Dog Consulting
UK

Email: daniel@olddog.co.uk

Michael Wang
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: wangzitao@huawei.com

PCE working group   D. Lopez
Internet-Draft   Telefonica I+D
Intended status: Standards Track   Q. Wu
Expires: September 29, 2017   D. Dhody
                                                               Z. Wang
                                                                Huawei
                                                               D. King
                                                     Old Dog Consulting

                                                        March 28, 2017

       IGP extension for PCEP security capability support in the PCE discovery
                  draft-wu-pce-discovery-pceps-support-07

Abstract

   When a Path Computation Element (PCE) is a Label Switching Router
   (LSR) participating in the Interior Gateway Protocol (IGP), or even a
   server participating in IGP, its presence and path computation
   capabilities can be advertised using IGP flooding. The IGP
   extensions for PCE discovery (RFC 5088 and RFC 5089) define a method
   to advertise path computation capabilities using IGP flooding for
   OSPF and IS-IS respectively. However these specifications lack a
   method to advertise PCEP security (e.g., Transport Layer
   Security(TLS),TCP Authentication Option (TCP-AO)) support capability.

   This document proposes new capability flag bits for PCE-CAP-FLAGS
   sub- TLV that can be announced as attribute in the IGP advertisement
   to distribute PCEP security support information.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on September 29, 2017

Copyright Notice

Table of Contents

1. Introduction

   As described in [RFC5440], PCEP communication privacy is one
   importance issue, as an attacker that intercepts a Path Computation
   Element (PCE) message could obtain sensitive information related to
   computed paths and resources.

   Among the possible solutions mentioned in these documents, Transport
   Layer Security (TLS) [RFC5246] provides support for peer
   authentication, and message encryption and integrity while TCP
   Authentication Option (TCP-AO) offer significantly improved security
   for applications using TCP. In order for a Path Computation
   Client(PCC) to begin a connection with a PCE server using TLS or TCP-
   AO, PCC SHOULD know whether PCE server supports TLS or TCP-AO as a
   secure transport.

[RFC5088] and [RFC5089] define a method to advertise path computation capabilities using IGP flooding for OSPF and IS-IS respectively. However [RFC5088] and [RFC5089] lacks a method to advertise PCEP security (e.g., TLS) support capability.

This document proposes new capability flag bits for PCE-CAP-FLAGS sub- TLV that can be announced as attributes in the IGP advertisement (defined in [RFC5088] and [RFC5089]) to distribute PCEP security support information.

2. 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 [RFC2119].

3. IGP extension for PCEP security capability support

The PCE-CAP-FLAGS sub-TLV is defined in section 4.5 of [RFC5088] and [RFC5089] as an optional sub-TLV used to advertise PCE capabilities. In this section, we extend the PCE-CAP-FLAGS sub-TLV to include the capability and indications that are described for PCEP security (e.g., TLS) support in the current document.

In the PCE-CAP-FLAGS sub-TLV defined in [RFC5088] and [RFC5089], nine capability flags defined in [RFC5088] (as per [RFC4657]) and two capability flags defined [RFC5557], [RFC6006] are included and follows the following format:

    o TYPE: 5
    o LENGTH: Multiple of 4
    o VALUE: This contains an array of units of 32 bit flags with
       the most significant bit as 0. Each bit represents one PCE
       capability.

and the processing rule of these flag bits are defined in [RFC5088] and [RFC5089]. In this document, we define two new capability flag bits that indicate TCP Authentication Option (TCP-AO) support, PCEP over TLS support respectively as follows:

        Bit   Capability Description
        xx   TCP AO Support
        xx   PCEP over TLS support

3.1. Use of PCEP security capability support for PCE discovery

   TCP-AO, PCEP over TLS support flag bits are advertised using IGP
   flooding.

   o PCE supports TCP-AO: IGP advertisement SHOULD include TCP-AO
      support flag bit.

   o PCE supports TLS: IGP advertisement SHOULD include PCEP over TLS
      support flag bit.

   If PCE supports multiple security mechanisms, it SHOULD include all
   corresponding flag bits in IGP advertisement.

   If the client is looking for connecting with PCE server with TCP-AO
   support, the client MUST check if TCP-AO support flag bit in the PCE-
   CAP-FLAGS sub-TLV is set. If not, the client SHOULD NOT consider
   this PCE. If the client is looking for connecting with PCE server
   using TLS, the client MUST check if PCEP over TLS support flag bit in
   the PCE-CAP-FLAGS sub-TLV is set. If not, the client SHOULD NOT
   consider this PCE.

4. Backward Compatibility Consideration

   An LSR that does not support the new IGP PCE capability bits
   specified in this document silently ignores those bits.

   IGP extensions defined in this document do not introduce any new
   interoperability issues.

5. Management Considerations

   A configuration option may be provided for advertising and
   withdrawing PCE security capability via IGP.

6. Security Considerations

   This document raises no new security issues beyond those described in
   [RFC5088] and [RFC5089].

7. IANA Considerations

   IANA is requested to allocate a new bit in "PCE Security Capability
   Flags" registry for PCEP Security support capability.

```
        Bit            Meaning               Reference
        xx             TCP-AO Support        [This.I.D]
        xx             PCEP over TLS support [This.I.D]
```

8. References

8.1. Normative References

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

   [RFC5088] Le Roux, JL., "OSPF Protocol Extensions for Path
              Computation Element (PCE) Discovery", RFC 5088, January
              2008.

   [RFC5089] Le Roux, JL., "IS-IS Protocol Extensions for Path
              Computation Element (PCE) Discovery", RFC 5089, January
              2008.

8.2. Informative References

   [RFC4657] Ash, J. and J. Le Roux, "Path Computation Element (PCE)
              Communication Protocol Generic Requirements", RFC 4657,
              September 2006.

   [RFC5246] Dierks, T., "The Transport Layer Security (TLS) Protocol
              Version 1.2", RFC 5246, August 2008.

   [RFC5440] Le Roux, JL., "Path Computation Element (PCE)
              Communication Protocol (PCEP)", RFC 5440, March 2009.

   [RFC5557] Lee, Y., Le Roux, JL., King, D., and E. Oki, "Path
              Computation Element Communication Protocol (PCEP)
              Requirements and Protocol Extensions in Support of Global
              Concurrent Optimization", RFC 5557, July 2009.

   [RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z.,
              and J. Meuric, "Extensions to the Path Computation Element
              Communication Protocol (PCEP) for Point-to-Multipoint
              Traffic Engineering Label Switched Paths", RFC 6006,
              September 2010.

Appendix A. Appendix A: No MD5 Capability Support

   To be compliant with Section 10.2 of RFC5440, this document doesn't
   consider to add capability for TCP-MD5. Therefore by default, PCEP
   Speaker in communication supports capability for TCP-MD5 (See section
   10.2, [RFC5440]). A method to advertise TCP-MD5 Capability support
   using IGP flooding is not required. If the client is looking for
   connecting with PCE server with other Security capability support
   (e.g., TLS support) than TCP-MD5, the client MUST check if flag bit

   in the PCE- CAP-FLAGS sub-TLV for specific capability is set (See
   section 3.1).

Authors' Addresses

   Diego R. Lopez
   Telefonica I+D
   Spain

   Email: diego.r.lopez@telefonica.com

   Qin Wu
   Huawei Technologies
   12 Mozhou East Road, Jiangning District
   Nanjing, Jiangsu 210012
   China

   Email: bill.wu@huawei.com

   Dhruv Dhody
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560037
   India

   Email: dhruv.ietf@gmail.com

   Michael Wang
   Huawei
   12 Mozhou East Road, Jiangning District
   Nanjing, Jiangsu 210012
   China

   Email: wangzitao@huawei.com

   Daniel King
   Old Dog Consulting
   UK

   Email: daniel@olddog.co.uk

PCE Working Group   Q. Wu
Internet-Draft   D. Dhody
Intended status: Standards Track   Huawei
Expires: January 1, 2016   M. Boucadair
                                                          C. Jacquenet
                                                       France Telecom
                                                          J. Tantsura
                                                             Ericsson
                                                        June 30, 2015

        PCEP Extensions for traffic steering support in Service Function
                                 Chaining
                    draft-wu-pce-traffic-steering-sfc-07

Abstract

   This document provides an overview of the usage of Path Computation
   Element (PCE) with Service Function Chaining (SFC); which is
   described as the definition and instantiation of an ordered set of
   such service functions (such as firewalls, load balancers), and the
   subsequent "steering" of traffic flows through those service
   functions.

   This document specifies extensions to the Path Computation Element
   Protocol (PCEP) that allow a stateful PCE to compute and instantiate
   Service Function Paths (SFP).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   Service chaining enables the creation of composite services that
   consist of an ordered set of Service Functions (SF) that must be
   applied to packets and/or frames selected as a result of
   classification as described in [I-D.ietf-sfc-architecture] and
   referred to as Service Function Chain (SFC). A Service Function Path
   (SFP) is the instantiation of a SFC in the network. Packets follow a

Service Function Path from a classifier through the requisite Service
Functions (SF) and Service Function Forwarders (SFF).

[RFC5440] describes the Path Computation Element Protocol (PCEP) as
the communication between a Path Computation Client (PCC) and a Path
Control Element (PCE), or between PCE and PCE, enabling computation
of Multiprotocol Label Switching (MPLS) for Traffic Engineering Label
Switched Path (TE LSP).

[I-D.ietf-pce-stateful-pce] specifies extensions to PCEP to enable
stateful control of MPLS TE LSPs. [I-D.ietf-pce-pce-initiated-lsp]
provides the fundamental extensions needed for stateful PCE-initiated
LSP instantiation.

This document specifies extensions to the PCEP that allow a stateful
PCE to compute and instantiate Service Function Paths (SFP).

2. 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 [RFC2119].

The following terminologies are used in this document:

PCC: Path Computation Client.

PCE: Path Computation Element.

PCEP: Path Computation Element Protocol.

PDP: Policy Decision Point.

SF: Service Function.

SFC: Service Function Chain.

SFP: Service Function Path.

SFF: Service Forwarder Function.

UNI: User-Network Interface.

3. Service Function Paths and PCE

Services are constructed as a sequence of SFs that represent an SFC,
where a SF can be a virtual instance or be embedded in a physical
network element, and one or more SFs may be supported by the same

physical network element. A SFC creates an abstracted view of a
service and specifies the set of required SFs as well as the order in
which they must be executed.

When an SFC is instantiated into the network it is necessary to
select the specific instances of SFs that will be used, and to create
the service function path for that SFC using SF network locators.
Thus, the instantiation of a SFC results in the establishment of a
Service Function Path, either a la hop-by-hop through the ordered
sequence of SF functions, or in a pre-computed, traffic-engineered
fashion. In other words, an SFP is the instantiation of the defined
SFC as described in [I-D.ietf-sfc-architecture].

The selection of SFP can be based on a set of policy attributes
(forwarding and routing, QoS, security, etc., or a combination
thereof), ranging from simple to more elaborate selection criteria
and the use of stateful PCE with extensions to PCEP are one such way
to achieve this.

Stateful pce [I-D.ietf-pce-stateful-pce] specifies a set of
extensions to PCEP to enable stateful control of TE LSPs.
[I-D.ietf-pce-pce-initiated-lsp] provides the fundamental motivations
and extensions needed for stateful PCE-initiated LSP instantiation.
This document specifies extensions that allow a stateful PCE to
compute and instantiate Service Function Paths (SFP) via PCEP.

```
                +-----------------------+
                |        stateful PCE   |
                | +-------+  +-------+   |
                | |Policy |  | TE-DB |   |          +-------+
                | +-------+  +-------+   |          |  SFC  |
    +----------|        +------------+   |<---|control|
    |SFP       |        |LSP-DB/SFP-DB|  |    | plane |
    |Instan-   |        +------------+   |    +-------+
    |tiation   +-----------------------+
    |             +-----+ +-----+          +-----+
    |             |SF-1 | |SF-2 |          |SF-3 |
    |             |     | |     |          |     |
    |             +---+-+ +-+---+          +--+--+
    |                 |     |                 |
    |               ++-----++             +----+-+
    V               |      |              |    |
 +-----+  Signaling |      | Signaling   |    | Signaling
 | SF  |----------->|  SFF-1 | --------->|  SFF-2 |----------->
 Classifier         |      |              |    |
 |Node |            |      |              |    |
 +-----+            +-------+             +-------+
```
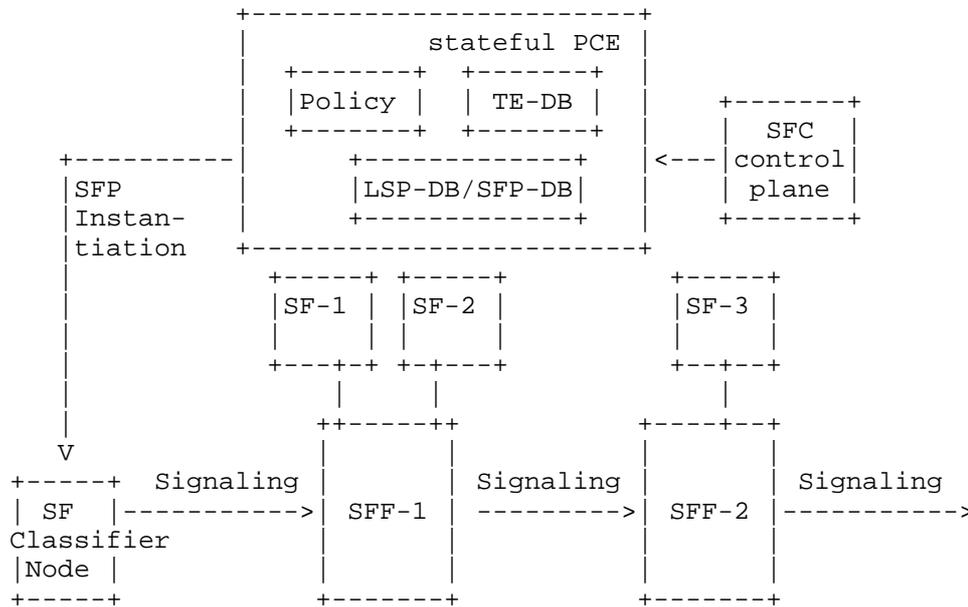
Figure 1: PCE based SFP instantiation

SFC Control plane components are responsible for maintaining SFC
Policy Tables and enforcing appropriate policies in SF Classifier and
SFF Nodes as described in
[I-D.ietf-sfc-architecture][I-D.ww-sfc-control-plane]. The SFC
Control plane component can be seen as a policy Decision point
(PDP,[RFC5394]). Such PDP can then operates a stateful PCE and its
instantiation mechanism to compute and instantiate Service Function
Paths (SFP). The PCE maybe co-located with the SFC Control plane
component or an external entity.

4. Overview of PCEP Operation in SFC-enabled Networks

   A PCEP speaker indicates its ability to support PCE provisioned
   dynamic SFP paths during the PCEP Initialization phase via a
   mechanism described in Section 5.1. A PCE can initiate SFPs only for
   PCCs that advertised this capability and a PCC will follow the
   procedures described in this document only on sessions where the PCE
   advertised this capability.

   As per section 5.1 of [I-D.ietf-pce-pce-initiated-lsp], the PCE sends
   a Path Computation LSP Initiate Request (PCInitiate) message to the
   PCC to instantiate or delete a LSP. The Explicit Route Object (ERO)
   can be used to encode either a sequence of SF functions or a
   combination of SFs and SFFs to establish a SFP. If the said SFFs and

SFs can be identified with an IP address, the IP sub-object can be
used as a SF/SFF identification means. This document makes no change
to the PCInitiate message format but extends LSP objects described in
Section 5.2.

Editor-Note: In case a PCE-Initiated Signaling mechanism is used to
setup the service function path, then does the classifier / PCE-
Initiated signaling protocol needs to understand if the IP address is
for SFF or SF or the signaling protocol is only used to signal IP
address for SFs?

4.1. SFP Instantiation

The Instantiation operation of a SFP is the same as defined in
section 5.3[I-D.ietf-pce-pce-initiated-lsp]. Rules of processing and
error codes remain unchanged.

4.2. SFP Withdrawal

The withdrawal operation of a SFP is the same as defined in section
5.4 of [I-D.ietf-pce-pce-initiated-lsp] : the PCE sends an LSP
Initiate Message with an LSP object carrying the PLSP-ID of the SFP
to be removed and an SRP object with the R flag set (LSP-REMOVE as
per section 5.2 of [I-D.ietf-pce-pce-initiated-lsp]). Rules of
processing and error codes remain unchanged.

4.3. SFP Delegation and Cleanup

SFP delegation and cleanup operations are similar to those defined in
section 6 of [I-D.ietf-pce-pce-initiated-lsp]. Rules of processing
and error codes remains unchanged.

4.4. SFP State Synchronization

State Synchronization operations described in Section 5.4 of
[I-D.ietf-pce-stateful-pce]can be applied for SFP state maintenance
as well.

4.5. SFP Update and Report

A PCE can send an SFP Update request to a PCC to update one or more
attributes of an SFP and to re-signal the SFP with the updated
attributes. A PCC can send an SFP state report to a PCE, and which
contains the SFP State information. The mechanism is described in
[I-D.ietf-pce-stateful-pce] and can be applied for SFPs as well.

5. Object Formats

5.1. The OPEN Object

   This document defines a new optional TLV for use in the OPEN Object
   to indicate the PCEP speaker's Service function Chaining capability.

   The SFC-PCE-CAPABILITY TLV is an optional TLV for use in the OPEN
   Object to advertise the SFC capability during the PCEP session. The
   format of the SFC-PCE-CAPABILITY TLV is shown in the
   followingFigure 2 :

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=TBD            |           length=4            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Reserved            |           Flags               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
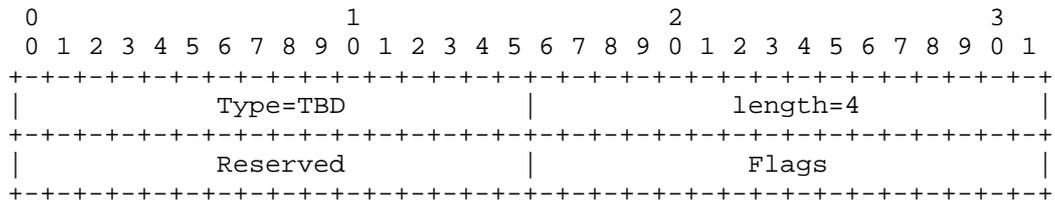
                     SFC-PCE-CAPABILITY TLV Format

   The code point for the TLV type is to be defined by IANA. The TLV
   length is 4 octets.

   The value is TBD.

   As per [I-D.ietf-pce-stateful-pce], a PCEP speaker advertises the
   capability of instantiating PCE-initiated LSPs via the Stateful PCE
   Capability TLV (LSP-INSTANTIATION-CAPABILITY bit) conveyed in an Open
   message. The inclusion of the SFC-PCE-CAPABILITY TLV in an OPEN
   object indicates that the sender is SFC-capable. Both mechanisms
   indicate the SFP instantiation capability of the PCEP speaker.

5.2. The LSP Object

   The LSP object is defined in [I-D.ietf-pce-pce-initiated-lsp] and
   included here for reference (Figure 3).

```
  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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                  PLSP-ID                | Flags |F|C| O|A|R|S|D|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 //                       TLVs                                  //
 |                                                              |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
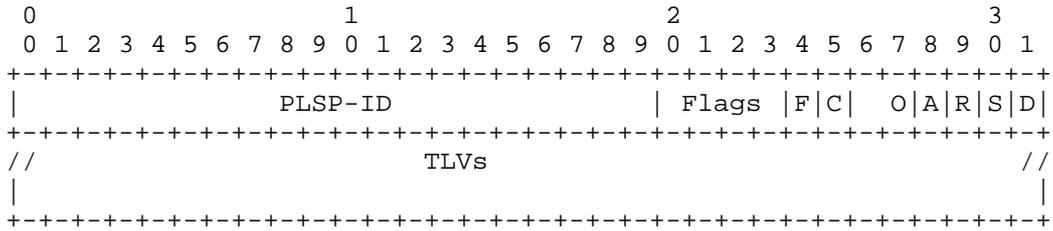
                        LSP Object Format

   A new flag, called the SFC (F) flag, is introduced. The F Flag set
   to 1 indicates that this LSP is actually an SFP. The C flag will
   also be set to indicate it was created via a PCInitiate message.

5.2.1. SFP Identifiers TLV

   The SFP Identifiers TLV MUST be included in the LSP object for
   Service Function Paths (SFP). The SFP Identifier TLV is used by the
   classifier to enable SFP selection for the traffic,i.e.,direct
   traffic to specific SFP[I-D.ietf-sfc-architecture]. The SFP
   Identifier carried in the SFC encapsulation can be further used by
   SFF to select service functions and next SFF,e.g., enable a packet
   that repeatedly arrives at the same SFF to get the correct services
   provided each time it arrives, and to go to the correct next SFF each
   time it arrives.

   The format of SFP Identifier TLV is shown in the following figure.

```
   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
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |              Service Path ID               | Service Index |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
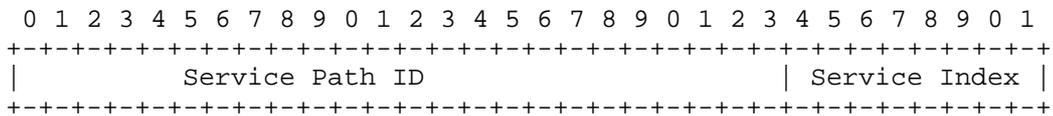
     Service path ID (SPI): 24 bits
     Service index (SI): 8 bits

   SPI: identifies a service path. The same ID is used by the
   participating nodes for path setup/selection. An administrator can
   use the SPI for reporting and troubleshooting packets along a
   specific path. SPI along with PLSP-ID is used in PCEP to identify
   the Service Path.

   SI: provides location within the service path.

6. Backward Compatibility

   The SFP instantiation capability PCEP protocol extensions described
   in this document MUST NOT be used if PCCs or the PCE did not
   advertise its SFP instantiation stateful capability, as per
   Section 5.1. If this is not the case and stateful operations on SFPs
   are attempted, then a PCErr with error-type 19 (Invalid Operation)
   and error-value TBD needs to be generated.

   [Editor Note: more information on exact error value is needed]

7. SFP signaling and forwarding consideration

   The SFP instantiation mechanism described in this document is not
   tightly coupled to any SFP signaling mechanism. For example,SR-based
   approach [I-D.ietf-pce-segment-routing] can utilize the mechanism
   described here and does not need any other specific protocol
   extensions. Generic SFC Encapsulation [I-D.quinn-sfc-nsh] can also
   be used together with the mechanism described here to enable SFP
   forwarding.

8. Security Considerations

   The security considerations described in [RFC5440] and
   [I-D.ietf-pce-pce-initiated-lsp] are applicable to this
   specification. No additional security measure is required.

9. IANA Considerations

   TBD

10. Acknowledgements

   Many thanks to Ron Parker, Hao Wang,Dave Dolson,Jing Huang,Joel M.
   Halpern for the discussion in formulating the content for the draft.

11. References

11.1. Normative References

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

   [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE", draft-ietf-pce-stateful-
              pce-11 (work in progress), April 2015.

[I-D.ietf-pce-pce-initiated-lsp]
          Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
          Extensions for PCE-initiated LSP Setup in a Stateful PCE
          Model", draft-ietf-pce-pce-initiated-lsp-04 (work in
          progress), April 2015.

11.2. Informative References

   [RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework
              for Policy-based Admission Control", RFC 2753, January
              2000.

   [RFC5394] Bryskin, I., Papadimitriou, D., Berger, L., and J. Ash,
              "Policy-Enabled Path Computation Framework", RFC 5394,
              December 2008.

   [RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element
              (PCE) Communication Protocol (PCEP)", RFC 5440, March
              2009.

   [I-D.ietf-sfc-architecture]
              Halpern, J. and C. Pignataro, "Service Function Chaining
              (SFC) Architecture", draft-ietf-sfc-architecture-09 (work
              in progress), June 2015.

   [I-D.ww-sfc-control-plane]
              Li, H., Wu, Q., Boucadair, M., Jacquenet, C., Haeffner,
              W., Lee, S., Parker, R., Dunbar, L., Malis, A., Halpern,
              J., Reddy, T., and P. Patil, "Service Function Chaining
              (SFC) Control Plane Components & Requirements", draft-ww-
              sfc-control-plane-06 (work in progress), June 2015.

   [I-D.ietf-pce-segment-routing]
              Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E.,
              Lopez, V., Tantsura, J., Henderickx, W., and J. Hardwick,
              "PCEP Extensions for Segment Routing", draft-ietf-pce-
              segment-routing-05 (work in progress), May 2015.

   [I-D.quinn-sfc-nsh]
              Quinn, P., Guichard, J., Surendra, S., Smith, M.,
              Henderickx, W., Nadeau, T., Agarwal, P., Manur, R.,
              Chauhan, A., Halpern, J., Majee, S., Elzur, U., Melman,
              D., Garg, P., McConnell, B., Wright, C., and K. Kevin,
              "Network Service Header", draft-quinn-sfc-nsh-07 (work in
              progress), February 2015.

Authors' Addresses

   Qin Wu
   Huawei
   101 Software Avenue, Yuhua District
   Nanjing, Jiangsu 210012
   China

   EMail: sunseawq@huawei.com

   Dhruv Dhody
   Huawei
   Leela Palace
   Bangalore, Karnataka 560008
   INDIA

   EMail: dhruv.ietf@gmail.com

   Mohamed Boucadair
   France Telecom
   Rennes 35000
   France

   EMail: mohamed.boucadair@orange.com

   Christian Jacquenet
   France Telecom
   Rennes 35000
   France

   EMail: christian.jacquenet@orange.com

   Jeff Tantsura
   Ericsson
   300 Holger Way
   San Jose, CA 95134
   US

   EMail: Jeff.Tantsura@ericsson.com

PCE Working Group   Q. Wu
Internet-Draft   D. Dhody
Intended status: Standards Track   Huawei
Expires: December 29, 2017   M. Boucadair
                                                             C. Jacquenet
                                                                  Orange
                                                             J. Tantsura
                                                           June 27, 2017

           PCEP Extensions for Service Function Chaining (SFC)
                   draft-wu-pce-traffic-steering-sfc-12

Abstract

   This document provides an overview of the usage of Path Computation
   Element (PCE) to dynamically structure service function chains.
   Service Function Chaining (SFC) is a technique that is meant to
   facilitate the dynamic enforcement of differentiated traffic
   forwarding policies within a domain. Service function chains are
   composed of an ordered set of elementary Service Functions (such as
   firewalls, load balancers) that need to be invoked according to the
   design of a given service. Corresponding traffic is thus forwarded
   along a Service Function Path (SFP) that can be computed by means of
   PCE.

   This document specifies extensions to the Path Computation Element
   Protocol (PCEP) that allow a stateful PCE to compute and instantiate
   Service Function Paths.

Copyright Notice

Table of Contents

1. Introduction

   Service Function Chaining (SFC) enables the creation of composite
   services that consist of an ordered set of Service Functions (SF)
   that must be applied to packets and/or frames and/or flows selected
   as a result of service-inferred traffic classification as described
   in [RFC7665]. A Service Function Path (SFP) is a path along which
   traffic that is bound to a specific service function chain will be

forwarded. Packets typically follow a Service Function Path from a
classifier through the Service Functions (SF) that need to be invoked
according to the SFC instructions. Forwarding decisions are made by
Service Function Forwarders (SFF) according to such instructions.

[RFC5440] describes the Path Computation Element Protocol (PCEP) as
the protocol used by a Path Computation Client (PCC) and a Path
Control Element (PCE) to exchange information, thereby enabling the
computation of Multiprotocol Label Switching (MPLS) for Traffic
Engineering Label Switched Path (TE LSP), in particular.

[I-D.ietf-pce-stateful-pce] specifies extensions to PCEP to enable a
stateful control of MPLS TE LSPs. [I-D.ietf-pce-pce-initiated-lsp]
provides the extensions needed for stateful PCE-initiated LSP
instantiation.

This document specifies PCEP extensions that allow a stateful PCE to
compute and instantiate traffic-engineered Service Function Paths
(SFP).

2. 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 [RFC2119].

   This document makes use of these acronyms:

   PCC: Path Computation Client.

   PCE: Path Computation Element.

   PCEP: Path Computation Element Protocol.

   PDP: Policy Decision Point.

   SF: Service Function.

   SFC: Service Function Chain.

   SFP: Service Function Path.

   RSP: Rendered Service Path.

   SFF: Service Function Forwarder.

   UNI: User-Network Interface.

3. Service Function Paths and PCE

   Service function chains are constructed as a sequence of SFs, where a
   SF can be virtualized or embedded in a physical network element. One
   or several SFs may be supported by the same physical network element.
   A SFC creates an abstracted view of a service and specifies the set
   of required SFs as well as the order in which they must be executed.

   When an SFC is created, it is necessary to select the specific
   instances of SFs that will be used. A service function path for that
   SFC will then be established (notion of rendered service path) or can
   be precomputed, based upon the sequence of SFs that need to be
   invoked by the corresponding traffic, i.e., the traffic that is bound
   to the corresponding SFC. Note that a SF instance can be serviced by
   one or multiple SFFs. One or multiple SF instances can be serviced
   by one SFF. Thus, the instantiation of an SFC results in the
   establishment of a Service Function Path, either in a hop-by-hop
   fashion, or by means of traffic-engineering capabilities. In the
   latter case, the SFP is precomputed, i.e., an SFP is an instantiation
   of the defined SFC as described in [RFC7665].

   The computation, the selection, and the establishment of a traffic-
   engineered SFP can rely upon a set of (service-specific) policies
   (forwarding and routing, QoS, security, etc., or a combination
   thereof). Stateful PCE with appropriate SFC-aware PCEP extensions
   can be used to compute traffic-engineered SFPs.

```
                      SFC Control Plane
                +-----------------------+
                |PCE based Controller   |
         I2     | +-------++-------+     |
   +------------ |Policy || TE-DB |     |
   |     I1     | +-------++-------+     <----+
   |            |        +-------------+  |   |
   | +---------||        |LSP-DB/SFP-DB|  |   |I2
   | |SFP      |        +-------------+  |   |
   | |Instan-  |                          |   |
   | |tiation  +-----------^-----------+  |policy-provisioning
   | |PCEP                 |               |information/
   | |Signaling         I2 |               |Carried by NETCONF,
   | |                     |               |BGP, for example
   | |                     |               |
   | |                     |               |
   | |                     |               |
   V V               +------V+        +----+--+
 +-----+  Forwarding |      |        |       |  Forwarding
 | SFC |------------>| SFF-1 |  Forwarding|       |
 Classifier         |      | --------->| SFF-2 |----------->
 |     |             |      |        |       |
 +-----+             ++-----++       +-----+-+
                      |    |              |
                   +--+--+ ++----+     +--+--+
                   |SF-1 | |SF-2 |     |SF-3 |
                   |     | |     |     |     |
                   +--+--+ +---+-+     +--+--+
                      |I2      |I2        |I2
                      V        V          V
```
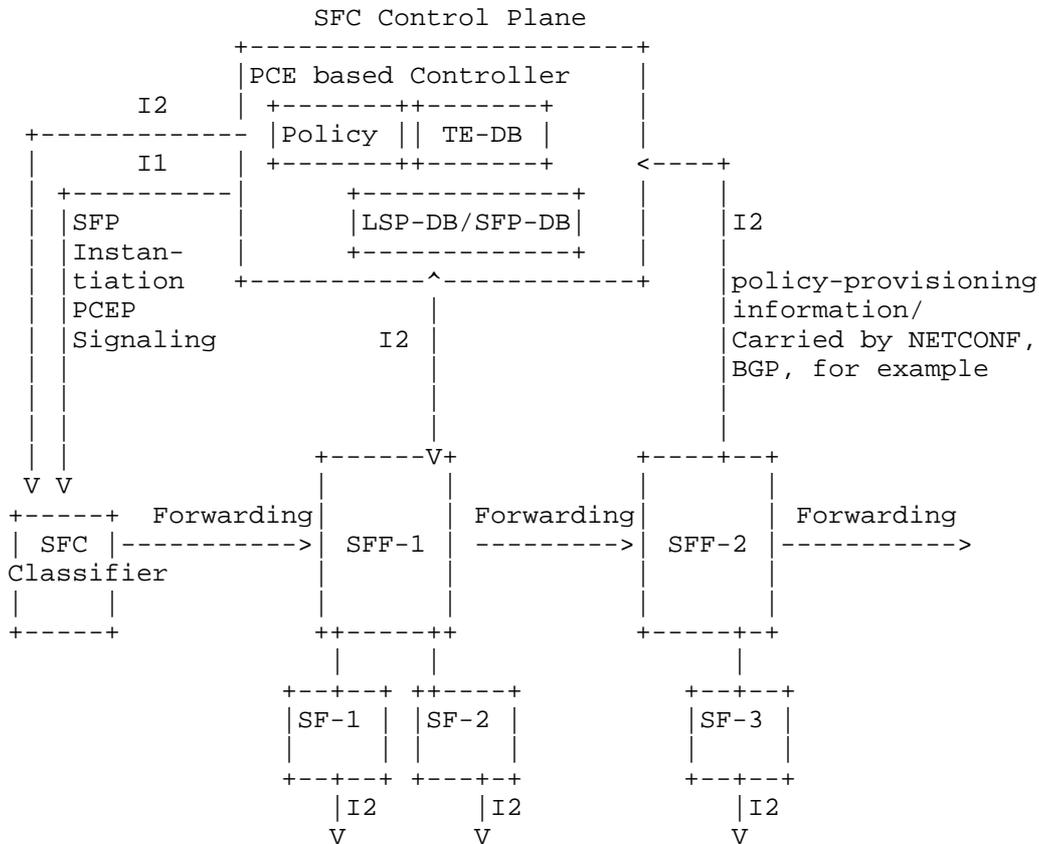
Figure 1: PCE-based SFP instantiation

In Figure 1, the PCE-based Controller [I-D.ietf-teas-pce-central-
control] in the SFC Control plane is responsible for computing the
path for a given service function chain. This PCE-based controller
can operate as a stateful PCE ([I-D.draft_ietf-stateful-pce]) that
will provide a classifier (a headend from a PCE standpoint) with the
PCEP-formatted information to instantiate a given SFP. As a
consequence, the PCE-based controller derives the set of policy-
provisioning information (namely SFP configuration information and
traffic classification rules) that will be provided to the various
elements (Classifier, SFF) involved in the establishment of the SFP.

By doing so, SFC Classifier can bind a flow to a service function
chain and forward such flow along the corresponding SFP. The SFC
Control Plane [I-D.ietf-sfc-control-plane] is also responsible for
defining the appropriate policies (traffic classification, forwarding
and routing, etc.) that will be enforced by SFC Classifiers,SFF Nodes

and SF Nodes, as described in [RFC7665]. From that standpoint, the
SFC Control Plane embeds a Policy Decision Point that is responsible
for defining the SFC policies. SFC policies will be provided by the
PDP and enforced by SFC components like classifiers and SFFs by means
of policy-provision information. A protocol like NETCONF, BGP can be
used to carry such policy-provisioning information.

4. Overview of PCEP Operation in SFC-Enabled Networks

   A PCEP speaker indicates its ability to support PCE-computed SFP
   paths during the PCEP Initialization phase via a mechanism described
   in Section 5.1. A PCE may initiate SFPs only for PCCs that
   advertised this capability; a PCC follows the procedures described in
   this document only for sessions where the PCE advertised this
   capability.

   As per Section 5.1 of [I-D.ietf-pce-pce-initiated-lsp], the PCE sends
   a Path Computation LSP Initiate Request (PCInitiate) message to the
   PCC to instantiate or delete a LSP. The Explicit Route Object (ERO)
   is used to encode either a full sequence of SF instances or a
   specific sequence of SFFs and SFs to establish an SFP. If the said
   SFFs and SFs are identified with an IP address, the IP sub-object can
   be used as a SF/SFF identification means. This document makes no
   change to the PCInitiate message format but extends LSP objects
   described in Section 5.2.

   Editor's note: In case a PCE-Initiated signaling mechanism is used to
   set up the service function path, does the classifier / PCE-Initiated
   signaling protocol need to understand whether an IP address is
   assigned to a SFF or a SF, or the signaling protocol is only used to
   signal IP addresses for SFs?

   To prevent multiple classifiers assign the same SFP ID to one Service
   Function Path(SFP ID assignment conflict),in this document, we assume
   SFP ID can be predetermined and assigned by stateful PCE when
   stateful PCE can be used to compute traffic-engineered SFPs.

4.1. SFP Instantiation

   The instantiation of a SFP is the same as defined in Section 5.3 of
   [I-D.ietf-pce-pce-initiated-lsp]. Rules for processing and error
   codes remain unchanged.

4.2. SFP Withdrawal

   The withdrawal of an SFP is the same as defined in Section 5.4 of
   [I-D.ietf-pce-pce-initiated-lsp]: the PCE sends an LSP Initiate
   Message with an LSP object carrying the PLSP-ID of the SFP and the

SFP Identifier to be removed, as well as an SRP object with the R
flag set (LSP-REMOVE as per Section 5.2 of
[I-D.ietf-pce-pce-initiated-lsp]). Rules for processing and error
codes remain unchanged.

4.3. SFP Delegation and Cleanup

SFP delegation and cleanup operations are similar to those defined in
Section 6 of [I-D.ietf-pce-pce-initiated-lsp]. Rules for processing
and error codes remain unchanged.

4.4. SFP State Synchronization

State Synchronization operations described in Section 5.4 of
[I-D.ietf-pce-stateful-pce] can be applied to SFP state maintenance
as well.

4.5. SFP Update and Report

A PCE can send an SFP Update request to a PCC to update one or more
attributes of an SFP and to re-signal the SFP with the updated
attributes. A PCC can send an SFP state report to a PCE, and which
contains the SFP State information. The mechanism is described in
[I-D.ietf-pce-stateful-pce] and can be applied to SFPs as well.

5. Object Formats

5.1. The OPEN Object

The optional TLV shown in Figure 2 is defined for use in the OPEN
Object to indicate the PCEP speaker's Service Function Chaining
capability.

The SFC-PCE-CAPABILITY TLV is an optional TLV to be carried in the
OPEN Object to advertise the SFC capability during the PCEP session.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=TBD          |           length=4            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Reserved          |             Flags             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
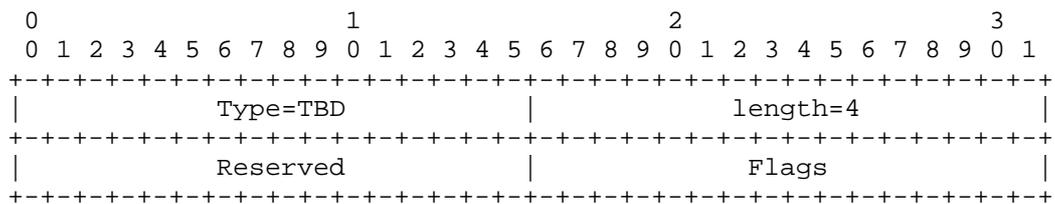
SFC-PCE-CAPABILITY TLV Format

The code point for the TLV type is to be defined by IANA (see
Section 9). The TLV length is 4 octets.

As per [I-D.ietf-pce-stateful-pce], a PCEP speaker advertises the
capability of instantiating PCE-initiated LSPs via the Stateful PCE
Capability TLV (LSP-INSTANTIATION-CAPABILITY bit) carried in an Open
message. The inclusion of the SFC-PCE-CAPABILITY TLV in an OPEN
object indicates that the sender is SFC-capable. Both mechanisms
indicate the SFP instantiation capability of the PCEP speaker.

5.2. The LSP Object

The LSP object is defined in [I-D.ietf-pce-pce-initiated-lsp] and
included here for reference (Figure 3).

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                PLSP-ID                | Flags |F|C| O|A|R|S|D|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        TLVs                                //
|                                                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
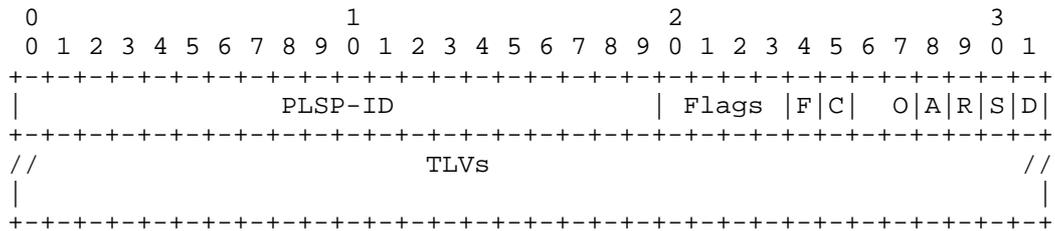
LSP Object Format

A new flag, called the SFC flag (F-bit), is introduced. The F-bit
set to "1" indicates that this LSP is actually an SFP. The C flag
will also be set to indicate it was created via a PCInitiate message.

5.2.1. SFP Identifiers TLV

As described in section 4, SFP ID is predetermined and assigned by
stateful PCE. The SFP Identifiers TLV MUST be included in the LSP
object for SFPs. The SFP Identifier TLV is used by the classifier to
select the SFP along which some traffic will be forwarded, according
to the traffic classification rules applied by the classifier
[RFC7665]. The SFP Identifier is part of the SFC metadata carried in
packets and is used by the SFF to invoke service functions and
identify the next SFF.

The format of the SFP Identifier TLV is shown in Figure 4.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Service Path ID               | Service Index |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

  Service Path ID (SPI): 24 bits
  Service Index (SI): 8 bits
```
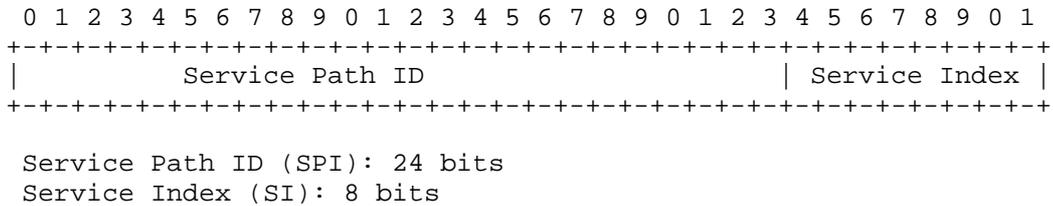
Figure 4

SPI: identifies a service path. The same ID is used by the
participating nodes for path setup/selection. An administrator can
use the SPI for reporting and troubleshooting packets along a
specific path. SPI along with PLSP-ID is used by PCEP to identify
the Service Path.

SI: provides location within the service path.

6. Backward Compatibility

The SFP instantiation capability defined as a PCEP extension and
documented in this draft MUST NOT be used if PCCs or the PCE did not
advertise their stateful SFP instantiation capability,Section 5.1.
If this is not the case and stateful operations on SFPs are
attempted, then a PCErr message with error-type 19 (Invalid
Operation) and error-value TBD needs to be generated.

[Editor's note: more information on exact error value is needed]

7. SFP Instantiation Signaling and Forwarding Considerations

The PCE-initiated SFP instantiation signaling described in this
document is exchanged between PCE server and SFC Classifier and does
not assume any specific mechanism to exchange SFP
information(e.g.,path identification information,metadata
[I-D.ietf-sfc-nsh]) between SFFs or between SFF and SF, or between
the controller and SFF and establish SFP in the data plane throughout
a SFC domain. For example, such mechanism can rely upon the use of
the SFC Encapsulation defined in [I-D.ietf-sfc-nsh] to exchange SFP
information between SFFs or rely upon the use of BGP Control plane
defined in [I-D.ietf-bess-nsh-bgp-control-plane] to exchange SFP
information between the Controller and SFF.

Likewise, [I-D.ietf-teas-pce-central-control] can use the signaling
mechanism described in this draft to enforce SFC-inferred traffic
engineering policies and provide load balancing between service
function nodes. The approach that relies upon the Segment Routing
technique [I-D.ietf-pce-segment-routing] can also take advantage of

the signaling mechanism described in this document to support Service
Path instantiation, which does not require any additional specific
extension to the Segment Routing machinery.

8. Security Considerations

   The security considerations described in [RFC5440] and
   [I-D.ietf-pce-pce-initiated-lsp] are applicable to this
   specification. This document does not raise any additional security
   issue.

9. IANA Considerations

   IANA is requested to allocate a new code point in the PCEP TLV Type
   Indicators registry, as follows:

   Value   Meaning   Reference
   ------- ---------------------------- --------------
   TBD   SFC-PCE-CAPABILITY   This document

10. Acknowledgements

   Many thanks to Ron Parker, Hao Wang, Dave Dolson, Jing Huang, and
   Joel M. Halpern for the discussion about the content for the
   document.

11. References

11.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE", draft-ietf-pce-stateful-
              pce-21 (work in progress), June 2017.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <http://www.rfc-editor.org/info/rfc5440>.

   [I-D.ietf-pce-pce-initiated-lsp]
             Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
             Extensions for PCE-initiated LSP Setup in a Stateful PCE
             Model", draft-ietf-pce-pce-initiated-lsp-10 (work in
             progress), June 2017.

   [I-D.ietf-teas-pce-central-control]
             Farrel, A., Zhao, Q., Li, Z., and C. Zhou, "An
             Architecture for Use of PCE and PCEP in a Network with
             Central Control", draft-ietf-teas-pce-central-control-03
             (work in progress), June 2017.

11.2. Informative References

   [RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework
             for Policy-based Admission Control", RFC 2753,
             DOI 10.17487/RFC2753, January 2000,
             <http://www.rfc-editor.org/info/rfc2753>.

   [RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function
             Chaining (SFC) Architecture", RFC 7665,
             DOI 10.17487/RFC7665, October 2015,
             <http://www.rfc-editor.org/info/rfc7665>.

   [RFC5394] Bryskin, I., Papadimitriou, D., Berger, L., and J. Ash,
             "Policy-Enabled Path Computation Framework", RFC 5394,
             DOI 10.17487/RFC5394, December 2008,
             <http://www.rfc-editor.org/info/rfc5394>.

   [I-D.ietf-sfc-control-plane]
             Boucadair, M., "Service Function Chaining (SFC) Control
             Plane Components & Requirements", draft-ietf-sfc-control-
             plane-08 (work in progress), October 2016.

   [I-D.ietf-pce-segment-routing]
             Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W.,
             and J. Hardwick, "PCEP Extensions for Segment Routing",
             draft-ietf-pce-segment-routing-09 (work in progress),
             April 2017.

   [I-D.ietf-sfc-nsh]
             Quinn, P. and U. Elzur, "Network Service Header", draft-
             ietf-sfc-nsh-12 (work in progress), February 2017.

   [I-D.ietf-bess-nsh-bgp-control-plane]
             Farrel, A., Drake, J., Rosen, E., Uttaro, J., and L.
             Jalil, "BGP Control Plane for NSH SFC", draft-ietf-bess-
             nsh-bgp-control-plane-00 (work in progress), March 2017.

Authors' Addresses

    Qin Wu
    Huawei
    101 Software Avenue, Yuhua District
    Nanjing, Jiangsu 210012
    China

    EMail: bill.wu@huawei.com

    Dhruv Dhody
    Huawei
    Leela Palace
    Bangalore, Karnataka 560008
    INDIA

    EMail: dhruv.ietf@gmail.com

    Mohamed Boucadair
    Orange
    Rennes 35000
    France

    EMail: mohamed.boucadair@orange.com

    Christian Jacquenet
    Orange
    Rennes
    France

    EMail: christian.jacquenet@orange.com

    Jeff Tantsura
    2330 Central Expressway
    Santa Clara, CA 95050
    US

    EMail: jefftant.ietf@gmail.com

PCE Working Group   Quintin Zhao
Internet-Draft   Robin Li
Intended status: Standards Track   Huawei Technologies
Expires: January 7, 2016   King Ke
                                              Tencent Holdings Ltd.
                                                       Luyuan Fang
                                                          Microsoft
                                                          Chao Zhou
                                                      Cisco Systems
                                                        Boris Zhang
                                               Telus Communications
                                                       July 6, 2015

     The Use Cases for Using PCE as the Central Controller(PCECC) of LSPs
              draft-zhao-pce-central-controller-user-cases-02

Abstract

   In certain networks deployment scenarios, service providers would
   like to keep all the existing MPLS functionalities in both MPLS and
   GMPLS network while removing the complexity of existing signaling
   protocols such as LDP and RSVP-TE. In this document, we propose to
   use the PCE as a central controller so that LSP can be calculated/
   signaled/initiated/downloaded/managed through a centralized PCE
   server to each network devices along the LSP path while leveraging
   the existing PCE technologies as much as possible.

   This draft describes the use cases for using the PCE as the central
   controller where LSPs are calculated/setup/initiated/downloaded/
   maintained through extending the current PCE architectures and
   extending the PCEP.

Status of This Memo

   This Internet-Draft will expire on January 7, 2016.

Copyright Notice

   Copyright (c) 2015 IETF Trust and the persons identified as the
   document authors. All rights reserved.

   This document is subject to BCP 78 and the IETF Trust's Legal
   Provisions Relating to IETF Documents
   (http://trustee.ietf.org/license-info) in effect on the date of
   publication of this document. Please review these documents
   carefully, as they describe your rights and restrictions with respect
   to this document. Code Components extracted from this document must
   include Simplified BSD License text as described in Section 4.e of
   the Trust Legal Provisions and are provided without warranty as
   described in the Simplified BSD License.

Table of Contents

1. Introduction

1.1. Background

   In certain network deployment scenarios, service providers would like
   to have the ability to dynamically adapt to a wide range of
   customer's requests for the sake of flexible network service
   delivery, SDN has provides additional flexibility in how the network
   is operated comparing the traditional network.

   The existing networking ecosystem has become awfully complex and
   highly demanding in terms of robustness, performance, scalability,
   flexibility, agility, etc. By migrating to the SDN enabled network
   from the existing network, service providers and network operators
   must have a solution which they can evolve easily from the existing
   network into the SDN enabled network while keeping the network
   services remain scalable, guarantee robustness and availability etc.

   Taking the smooth transition between traditional network and the new
   SDN enabled network into account, especially from a cost impact
   assessment perspective, using the existing PCE components from the
   current network to function as the central controller of the SDN
   network is one choice, which not only achieves the goal of having a
   centralized controller to provide the functionalities needed for the
   central controller, but also leverages the existing PCE network
   components.

   The Path Computation Element communication Protocol (PCEP) provides
   mechanisms for Path Computation Elements (PCEs) to perform route
   computations in response to Path Computation Clients (PCCs) requests.
   PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model
   draft [I-D. draft-ietf-pce- stateful-pce] describes a set of
   extensions to PCEP to enable active control of MPLS-TE and GMPLS
   tunnels.

   [I-D.crabbe-pce-pce-initiated-lsp] describes the setup and teardown
   of PCE-initiated LSPs under the active stateful PCE model, without
   the need for local configuration on the PCC, thus allowing for a
   dynamic MPLS network that is centrally controlled and deployed.

   [I-D.ali-pce-remote-initiated-gmpls-lsp] complements [I-D.
   draft-crabbe-pce-pce-initiated-lsp] by addressing the requirements
   for remote-initiated GMPLS LSPs.

   SR technology leverages the source routing and tunneling paradigms.
   A source node can choose a path without relying on hop-by-hop
   signaling protocols such as LDP or RSVP-TE. Each path is specified
   as a set of "segments" advertised by link-state routing protocols

(IS-IS or OSPF). [I-D.filsfils-spring-segment-routing] provides an
introduction to SR technology. The corresponding IS-IS and OSPF
extensions are specified in [I-D.ietf-isis-segment-routing-
extensions] and [I-D.psenak-ospf-segment-routing-extensions],
respectively.

A Segment Routed path (SR path) can be derived from an IGP Shortest
Path Tree (SPT). Segment Routed Traffic Engineering paths (SR-TE
paths) may not follow IGP SPT. Such paths may be chosen by a
suitable network planning tool and provisioned on the source node of
the SR-TE path.

It is possible to use a stateful PCE for computing one or more SR-TE
paths taking into account various constraints and objective
functions. Once a path is chosen, the stateful PCE can instantiate
an SR-TE path on a PCC using PCEP extensions specified in
[I-D.crabbe-pce-pce-initiated-lsp] using the SR specific PCEP
extensions described in [I-D.sivabalan-pce-segment-routing].

By using the solutions provided from above drafts, LSP in both MPLS
and GMPLS network can be setup/delete/maintained/synchronized through
a centrally controlled dynamic MPLS network. Since in these
solutions, the LSP is need to be signaled through the head end LER to
the tail end LER, there are either RSVP-TE signaling protocol need to
be deployed in the MPLS/GMPLS network, or extend TGP protocol with
node/adjacency segment identifiers signaling capability to be
deployed.

The PCECC solution proposed in this document allow for a dynamic MPLS
network that is eventually controlled and deployed without the
deployment of RSVP-TE protocol or extended IGP protocol with node/
adjacency segment identifiers signaling capability while providing
all the key MPLS functionalities needed by the service providers.
These key MPLS features include MPLS P2P LSP, P2MP/MP2MP LSP, MPLS
protection mechanism etc. In the case that one LSP path consists
legacy network nodes and the new network nodes which are centrally
controlled, the PCECC solution provides a smooth transition step for
users.

1.2. Using the PCE as the Central Controller (PCECC) Approach

   With PCECC, it not only removes the existing MPLS signaling totally
   from the control plane without losing any existing MPLS
   functionalities, but also PCECC achieves this goal through utilizing
   the existing PCEP without introducing a new protocol into the
   network.

   The following diagram illustrates the PCECC architecture.

```
+------------------------------------------------------------------+
|                            PCECC                                 |
|    +---------------------------------------------------------+   |
|    |  LSP-Database    RSVP-TE Signal Control Module          |   |
|    |  TE-Database     LDP signaling Control Module           |   |
|    |  Label-Database  LSP/label/TE MGRs                      |   |
|    +---------------------------------------------------------+   |
|      ^                ^                ^              ^       ^    |
| IGP|LDP/RSVP-TE   |PCEP           |PCEP       PCEP|      IGP|LDP/  |
|    |PCEP          |               |               |         |RSVP-TE/ |
|    V              V               V               V         V PCE |
| +--------+    +--------+     +--------+     +--------+    +--------+ |
| |NODE 1  |    | NODE 2 |     | NODE 3 |     | NODE 4 |    | NODE 5 | |
| |       |...|        |...|          |...|          |...|        | |
| | Legacy |IGP|        |IGP|          |IGP|  PCC4    |IGP| Legacy | |
| | Node   |   |        |   |          |   |          |   | Node   | |
| +--------+    +--------+     +--------+     +--------+    +--------+ |
|                                                                  |
+------------------------------------------------------------------+
```
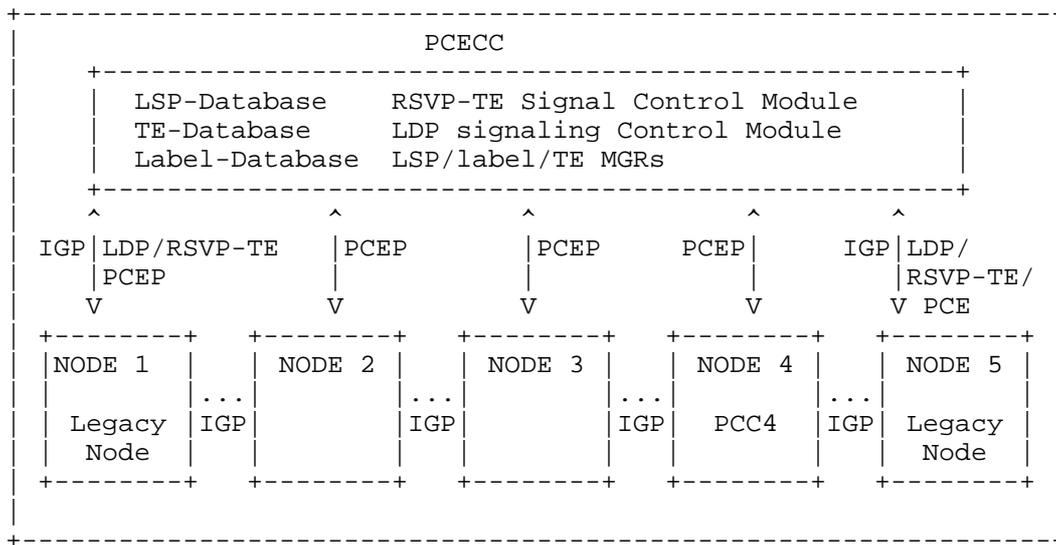
Figure 1: PCECC Architecture

Through the draft, we call the combination of the functionality for
global label range signaling and the functionality of LSP setup/
download/cleanup using the combination of global labels and local
labels as PCECC functionality.

Current MPLS label has local meaning. That is, MPLS label allocated
locally and signaled through the LDP/RSVP-TE/BGP etc dynamic
signaling protocol.

As the SDN(Service-Driven Network) technology develops, MPLS global
label has been proposed again for new solutions. [I-D.li-mpls-
global-label-usecases] proposes possible usecases of MPLS global
label. MPLS global label can be used for identification of the
location, the service and the network in different application
scenarios. From these usecases we can see that no matter SDN or
traditional application scenarios, the new solutions based on MPLS
global label can gain advantage over the existing solutions to
facilitate service provisions. The solution choices are described in
[I-D.li-mpls-global-label-framework].

To ease the label allocation and signaling mechanism, also with the
new applications such as concentrated LSP controller is introduced,
PCE can be conveniently used as a central controller and MPLS global
label range negotiator.

The later section of this draft describes the user cases for PCE
server and PCE clients to have the global label range negotiation and
local label range negotiation functionality.

To empower networking with centralized controllable modules, there
are many choices for downloading the forwarding entries to the data
plane, one way is the use of the OpenFlow protocol, which helps
devices populate their forwarding tables according to a set of
instructions to the data plane. There are other candidate protocols
to convey specific configuration information towards devices also.
Since the PCEP protocol is already deployed in some of the service
network, to leverage the PCEP to populated the MPLS forwarding table
is a possible good choice.

For the centralized network, the performance achieved through
distributed system can not be easy matched if all of the forwarding
path is computed, downloaded and maintained by the centralized
controller. The performance can be improved by supporting part of
the forwarding path in the PCECC network through the segment routing
mechanism except that the adjacency IDs for all the network nodes and
links are propagated through the centralized controller instead of
using the IGP extension.

The node and link adjacency IDs can be negotiated through the PCECC
with each PCECC clients and these IDs can be just taken from the
global label range which has been negotiated already.

With the capability of supporting SR within the PCECC architecture,
all the p2p forwarding path protection use cases described in the
draft [I-D.ietf-spring-resiliency-use-cases] will be supported too
within the PCECC network. These protection alternatives include end-
to-end path protection, local protection without operator management
and local protection with operator management.

With the capability of global label and local label existing at the
same time in the PCECC network, PCECC will use compute, setup and
maintain the P2MP and MP2MP lsp using the local label range for each
network nodes.

With the capability of setting up/maintaining the P2MP/MP2MP LSP
within the PCECC network, it is easy to provide the end-end managed
path protection service and the local protection with the operation
management in the PCECC network for the P2MP/MP2MP LSP, which
includes both the RSVP-TE P2MP based LSP and also the mLDP based LSP.

2. Terminology

   The following terminology is used in this document.

   IGP: Interior Gateway Protocol. Either of the two routing
      protocols, Open Shortest Path First (OSPF) or Intermediate System
      to Intermediate System (IS-IS).

   PCC: Path Computation Client: any client application requesting a
      path computation to be performed by a Path Computation Element.

   PCE: Path Computation Element. An entity (component, application,
      or network node) that is capable of computing a network path or
      route based on a network graph and applying computational
      constraints.

   TE: Traffic Engineering.

3. PCEP Requirements

   Following key requirements associated PCECC should be considered when
   designing the PCECC based solution:

   1. Path Computation Element (PCE) clients supporting this draft MUST
       have the capability to advertise its PCECC capability to the
       PCECC.

   2. Path Computation Element (PCE) supporting this draft MUST have
       the capability to negotiate a global label range for a group of
       clients.

   3. Path Computation Client (PCC) MUST be able ask for global label
       range assigned in path request message .

   4. PCE are not required to support label reserve service.
       Therefore, it MUST be possible for a PCE to reject a Path
       Computation Request message with a reason code that indicates no
       support for label reserve service.

   5. PCEP SHOULD provide a means to return global label range and LSP
       label assignments of the computed path in the reply message.

   6. PCEP SHOULD provide a means to download the MPLS forwarding entry
       to the PCECC's clients.

4. Use Cases of PCECC for Label Resource Reservations

   Example 1 to 2 are based on network configurations illustrated using
   the following figure:

```
+-----------------------------+     +------------------------------+
|      PCE DOMAIN 1           |     |      PCE DOMAIN 2            |
|      +--------+             |     |      +--------+             |
|      |        |             |     |      |        |             |
|      | PCECC1 | ----------------------- | PCECC2 |             |
|      |        |             |     |      |        |             |
|      |        |             |     |      |        |             |
|      +--------+             |     |      +--------+             |
|       ^         ^           |     |       ^         ^           |
|      /           \          |     |      /           \          |
|     V             V         |     |     V             V         |
| +--------+   +--------+     |     | +--------+   +--------+     |
| |NODE 11 |   | NODE 1n|     |     | |NODE 21 |   | NODE 2n|     |
| |        |...|        |     |     | |        |...|        |     |
| | PCECC  |   | PCECC  |     |     | | PCECC  |   |PCECC   |     |
| |Enabled |   | Enabled|     |     | |Enabled |   |Enabled |     |
| +--------+   +--------+     |     | +--------+   +--------+     |
|                             |     |                              |
+-----------------------------+     +------------------------------+
```
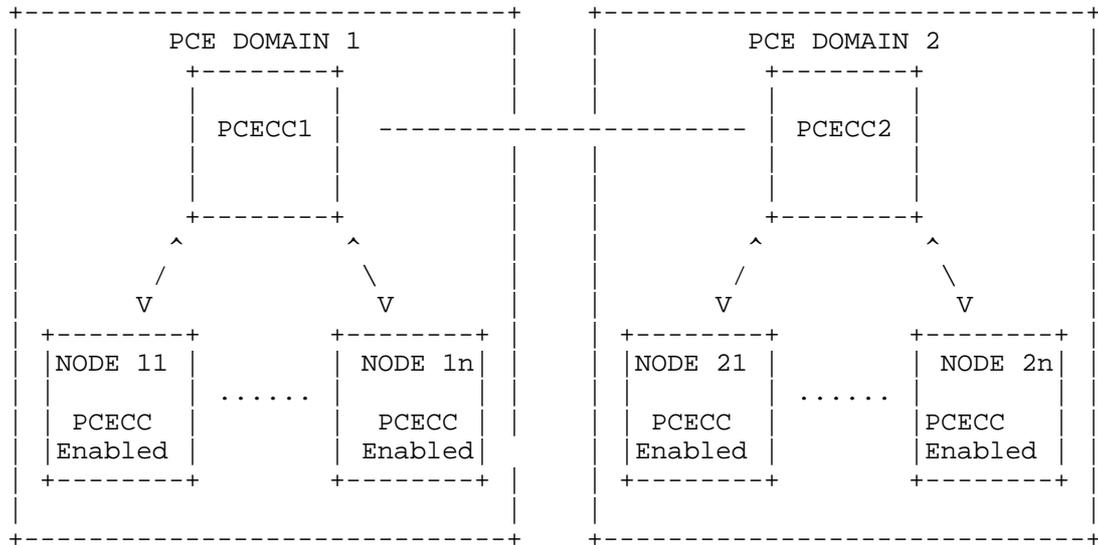

           Figure 2: Using PCECC for Global Label Allocation

   Example 1: Shared Global Label Range Reservation

   o PCECC Clients nodes report MPLS label capability to the central
      controller PCECC.

   o The central controller PCECC collects MPLS label capability of all
      nodes. Then PCECC can calculate the shared MPLS global label
      range for all the PCECC client nodes.

   o In the case that the shared global label range need to be
      negotiated across multiple domains, the central controllers of
      these domains need to be communicate to negotiate a common global
      label range.

   o The central controller PCECC notifies the shared global label
      range to all PCECC client nodes.

   Example 2: Global Label Allocation

o PCECC Client node1 send global label allocation request to the
   central controller PCECC1.

o The central controller PCECC1 allocates the global label for FEC1
   from the shared global label range and sends the reply to the
   client node1.

o The central controller PCECC1 notifies the allocated label for
   FEC1 to all PCECC client nodes within domain 1.

5. Using PCECC for SR without the IGP Extension

   For the centralized network, the performance achieved through
   distributed system can not be easy matched if all of the forwarding
   path is computed, downloaded and maintained by the centralized
   controller. The performance can be improved by supporting part of
   the forwarding path in the PCECC network through the segment routing
   mechanism except that node segment ids and adjacency segment IDs for
   all the network are allocated dynamically and propagated through the
   centralized controller instead of using the IGP extension.

   When the PCECC is used for the distribution of the node segment ID
   and adjacency segment ID, the node segment ID is allocated from the
   global label pool. For the allocation of adjacency segment ID, there
   are two choices, the first choice is that it is allocated from the
   local label pool, the second choice is that it is allocated from the
   global label pool. The advantage for the second choice is that the
   depth of the label stack for the forwarding path encoding will be
   reduced since adjacency segment ID can signal the forwarding path
   without adding the node segment ID in front of it. In this version
   of the draft, we use the fist choice for now. We may update the
   draft to reflect the use of the second choice.

   Same as the SR solutions, when PCECC is used as the central
   controller, the support of FRR on any topology can be pre-computated
   and setup without any additional signaling (other than the regular
   IGP/BGP protocols) including the support of shared risk constraints,
   support of node and link protection and support of microloop
   avoidance.

   The following example illustrate the use case where the node segment
   ID and adjacency segment ID are allocated from the global label
   allocated for SR path.

```
                        192.0.2.1/32
                        +----------+
                        | R1(1001) |
                        +----------+
                             |
                        +----------+
                        | R2(1002) |  192.0.2.2/32
                        +----------+
                       *   |   *     *
                      *    |   *      *
                    *link1|    *       *
         192.0.2.4/32  *  |   *link2  *  192.0.2.5/32
         +----------+ 9001|    *    +----------+
         | R4(1004) |     |    *    | R5(1005) |
         +----------+     |    *    +----------+
                   *      |  *9003  *   +
                  *       |  *    *    +
                 *        |  *   *    +
         +----------+   +----------+
192.0.2.3/32 | R3(1003) |   |R6(1006)  |192.0.2.6/32
         +----------+   +----------+
              |
         +----------+
         | R8(1008) |  192.0.2.8/32
         +----------+
```

                  Figure 3: Using PCECC for SR Path

5.1. Use Cases of PCECC for SR Best Effort(BE) Path

   In this mode of the solution, the PCECC just need to allocate the
   node segment ID and adjacency ID without calculating the explicit
   path for the SR path. The ingress of the forwarding path just need
   to encapsulate the destination node segment ID on top of the packet.
   All the intermediate nodes will forward the packet based on the final
   destination node segment id. It is similar to the LDP LSP forwarding
   except that label swapping is using the same global label both for
   the in segment and out segment in each hop.

   The p2p SR BE path examples are explained as bellow:

   Note that the node segment id for each node from the shared global
   labels ranges negotiated already.

   Example 1:

   R1 may send a packet to R8 simply by pushing an SR header with
   segment list {1008}. The path can be: R1-R2-R3-R8 or R1-R2-R5-R8

depending on the route calculation on node R2.

Example 2: local link/node protection:

For the packet which has destination of R3 and after that, R2 may preinstalled the backup forwarding entry to protect the R4 node, the pre-installed the backup path can go through either node5 or link1 or link2 between R2 and R3. The backup path calculation is locally decided by R2 and any existing IP FRR algorithms can be used here.

5.2. Use Cases of PCECC for SR Traffic Engineering (TE) Path

In the case of traffic engineering path is needed, the PCECC need to allocate the node segment ID and adjacency ID, and at the same time PCECC calculates the explicit path for the SR path and pass this explicit path represented with a sequence of node segment id and adjacency id. The ingress of the forwarding path need to encapsulate the stack of node segment id and adjacency id on top of the packet. For the case where strict traffic engineering path is needed, all the intermediate nodes and links will be specified through the stack of labels so that the packet is forwarded exactly as it is wanted.

Even though it is similar to TE LSP forwarding where forwarding path is engineered, but the Qos is only guaranteed through the enforce of the bandwidth admission control. As for the RSVP-TE LSP case, Qos is guaranteed through the link bandwidth reservation in each hop of the forwarding path.

The p2p SR traffic engineering path examples are explained as bellow:

Note that the node segment id for each node is allocated from the shared global labels ranges negotiated already and adjacency segment ids for each link are allocated from the local label pool for each node.

Example 1:

R1 may send a packet P1 to R8 simply by pushing an SR header with segment list {1008}. The path should be: R1-R2-R3-R8.

Example 2:

R1 may send a packet P2 to R8 by pushing an SR header with segment list {1002, 9001, 1008}. The path should be: R1-R2-(1)link-R3-R8.

Example 3:

R1 may send a packet P3 to R8 while avoiding the links between R2 and

R3 by pushing an SR header with segment list {1004, 1008}. The path
should be : R1-R2-R4-R3-R8

The p2p local protection examples for SR TE path are explained as
below:

Example 4: local link protection:

o R1 may send a packet P4 to R8 by pushing an SR header with segment
   list {1002, 9001, 1008}. The path should be: R1-R2-(1)link-R3-R8.

o When node R2 receives the packet from R1 which has the header of
   R2- (1)link-R3-R8, and also find out there is a link failure of
   link1, then it will send out the packet with header of R3-R8
   through link2.

Example 5: local node protection:

o R1 may send a packet P5 to R8 by pushing an SR header with segment
   list {1004, 1008}. The path should be : R1-R2-R4-R3-R8.

o When node R2 receives the packet from R1 which has the header of
   {1004, 1008}, and also find out there is a node failure for node4,
   then it will send out the packet with header of {1005, 1008} to
   node5 instead of node4.

6. Use Cases of PCECC for TE LSP

   In the previous sections, we have discussed the cases where the SR
   path is setup through the PCECC. Although those cases give the
   simplicity and scalability, but there are existing functionalities
   for the traffic engineering path such as the bandwidth guarantee
   through the full forwarding path and the multicast forwarding path
   which SR based solution cannot solve. Also there are cases where the
   depth of the label stack may have been an issue for existing
   deployment and certain vendors.

   So to address these issues, PCECC architecture should also support
   the TE LSP and multicast LSP functionalities. To achieve this, the
   existing PCEP can be used to communicate between the PCE server and
   PCE's client PCC for exchanging the path request and reply
   information regarding to the TE LSP info. In this case, the TE LSP
   info is not only the path info itself, but it includes the full
   forwarding info. Instead of letting the ingress of LSP to initiate
   the LSP setup through the RSVP-TE signaling protocol, with minor
   extensions, we can use the PCEP to download the complete TE LSP
   forwarding entries for each node in the network.

```
                       192.0.2.1/32
                      +----------+
                      | R1(1001) |
                      +----------+
                        |      |
                   6001|link1  |
                        |   6002|link2
                      +----------+
                      | R2(1002) |  192.0.2.2/32
                      +----------+
              link3 *   |   *     * link4
              7002 *    |   *       *7001
                  *link1|   *        *
        192.0.2.4/32 *  |   *link2  *  192.0.2.5/32
          +----------+ 5001|   *   +----------+
          | R4(1004) |    |   *   | R5(1005) |
          +----------+    |   *   +----------+
                 *        |  *5003  *           +
            9001*         |   *    *link1   +
                 *        |   *  *9002      +
              +----------+    +----------+
    192.0.2.3/32 | R3(1003) |  |R6(1006)  |192.0.2.6/32
              +----------+    +----------+
                 |       |
             3001|link1  |
                 |    3002|link2
              +----------+
              | R8(1008) |  192.0.2.8/32
              +----------+
```

                 Figure 4: Using PCECC for TE LSP

   TE LSP Setup Example

   o Node1 sends a path request message for the setup of TE LSP from R1
      to R8.

   o PCECC program each node along the path from R1 to R8 with the
      primary path: {R1, link1, 6001}, {R2, link3, 7002], {R4, link0,
      9001}, {R3, link1, 3001}, {R8}.

   o For the end to end protection, PCECC program each node along the
      path from R1 to R8 with the secondary path: {R1, link2, 6002},
      {R2, link4, 7001], {R5, link1, 9002}, {R3, link2, 3002}, {R8}.

   o It is also possible to have a secondary backup path for the local
      node protection setup by PCECC. For exampleGBP[not] the primary
      path is still same as what we have setup so far, then to protect

the node R4 locally, PCECC can program the secondary path like
this: {R1, link1, 6001}, {R2, link1, 5001}, {R3, link1, 3001},
{R8}. By doing this, the node R4 is locally protected.

7. Use Cases of PCECC for Multicast LSPs

The current multicast LSPs are setup either using the RSVP-TE P2MP or
mLDP protocols. The setup of these LSPs not only need a lot of
manual configurations, but also it is also complex when the
protection is considered. By using the PCECC solution, the multicast
LSP can be computed and setup through centralized controller which
has the full picture of the topology and bandwidth usage for each
link. It not only reduces the complex configurations comparing the
distributed RSVP-TE P2MP or mLDP signal lings, but also it can
compute the disjoint primary path and secondary path efficiently.

7.1. Using PCECC for P2MP/MP2MP LSPs' Setup

With the capability of global label and local label existing at the
same time in the PCECC network, PCECC will use compute, setup and
maintain the P2MP and MP2MP lsp using the local label range for each
network nodes.

```
                        +----------+
                        |    R1    | Root node of the multicast LSP
                        +----------+
                           |6000
                        +----------+
        Transit Node    |    R2    |
                        +----------+
                         *  |  *   *
                    9001*   |  *    *9002
                       *    |  *     *
        +-----------+       |  *      +-----------+
        |    R4     |       |  *      |    R5     | Transit Nodes
        +-----------+       |  *      +-----------+
              *             |  *     *     +
           9003*            |  *     *     +9004
              *             |  *    *      +
            +----------+  +-----------+
            |    R3    |  |    R5     | Leaf Node
            +----------+  +----------+
             9005|
            +-----------+
            |    R8     | Leaf Node
            +-----------+
```
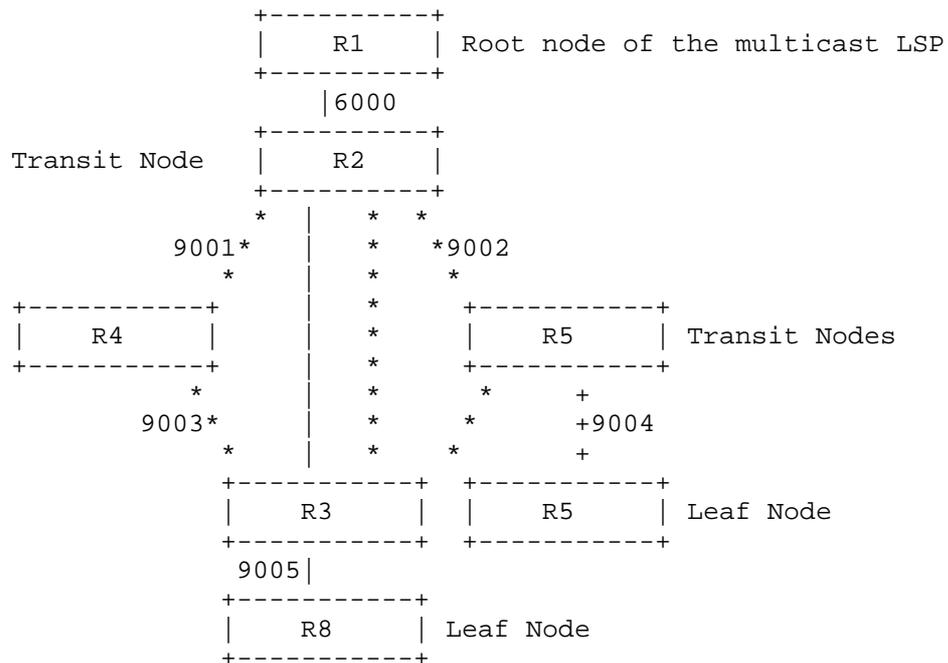
Figure 5: Using PCECC for P2MP TE LSP

The P2MP examples are explained here:

Step1: R1 may send a packet P1 to R2 simply by pushing an label of 6000 to the packet.

Step2: After R2 receives the packet with label 6000, it will forwarding to R4 by pushing header of 9001 and R5 by pusing header of 9002.

Step3: After R4 receives the packet with label 9001, it will forwarding to R3 by pushing header of 9003. After R5 receives the packet with label 9002, it will forwarding to R5 by pushing header of 9004.

Step3: After R3 receives the packet with label 9003, it will forwarding to R8 by pushing header of 9005

7.2. Use Cases of PCECC for the Resiliency of P2MP/MP2MP LSPs

7.2.1. PCECC for the End-to-End Protection of the P2MP/MP2MP LSPs

In this section we describe the end-end managed path protection service and the local protection with the operation management in the PCECC network for the P2MP/MP2MP LSP, which includes both the RSVP-TE P2MP based LSP and also the mLDP based LSP.

An end-to-end protection (for nodes and links) principle can be applied for computing backup P2MP or MP2MP LSPs. During computation of the primarily multicast trees, PCECC server may also be taken into consideration to compute a secondary tree. A PCE may compute the primary and backup P2MP or MP2Mp LSP together or sequentially.

```
                          +----+  +----+
         Root node of LSP | R1 |--| R11|
                          +---+   +----+
                           /         +
                         10/         +20
                         /            +
             +----------+        +-----------+
Transit Node |   R2     |        |    R3     |
             +----------+        +-----------+
              |     \      +          +
              |      \      +          +
            10|     10\    +20       20+
              |        \  +            +
              |         \ +            +
              |        + \             +
             +----------+     +-----------+ Leaf Nodes
             |   R4     |     |    R5     | (Downstream LSR)
             +----------+     +-----------+
```
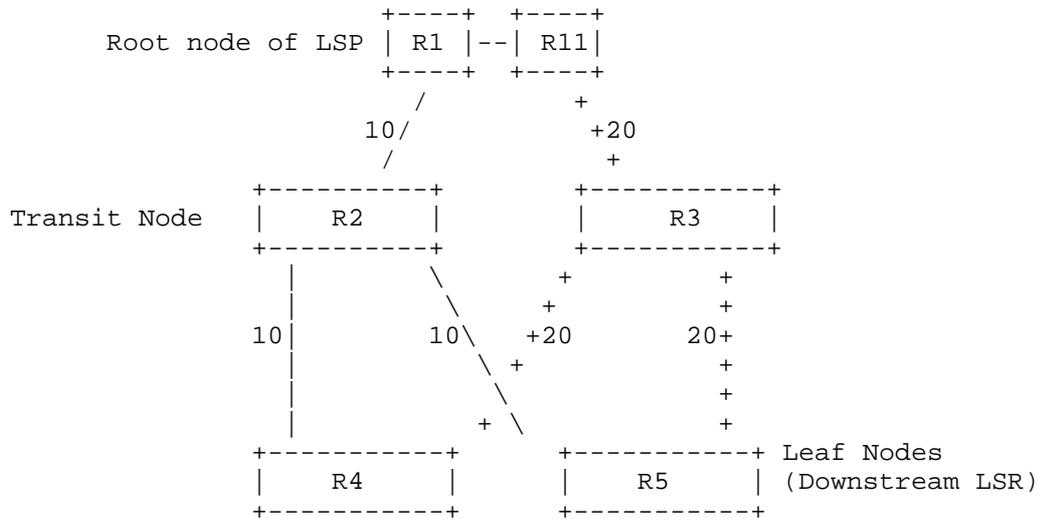
Figure 6: Using PCECC for P2MP TE End-to-End Protection

In the example above, when the PCECC setup the primary multicast tree
from the root node R1 to the leafs, which is R1->R2->{R4, R5}, at
same time, it can setup the backup tree, which is R11->R3->{R4, R5}.
Both the these two primary forwarding tree and secondary forwarding
tree will be downloaded to each routers along the primary path and
the secondary path. The traffic will be forwarded through the
R1->R2->{R4, R5} path normally, and when there is a node in the
primary tree, then the root node R1 will switch the flow to the
backup tree, which is R11->R3->{R4, R5}. By using the PCECC, the
path computation and forwarding path downloading can all be done
without the complex signaling used in the P2MP RSVP-TE or mLDP.

7.2.2. PCECC for the Local Protection of the P2MP/MP2MP LSPs

In this section we describe the local protection service in the PCECC
network for the P2MP/MP2MP LSP.

While the PCECC sets up the primary multicast tree, it can also build
the back LSP among PLR, the protected node, and MPs (the downstream
nodes of the protected node). In the cases where the amount of
downstream nodes are huge, this mechanism can avoid unnecessary
packet duplication on PLR, so that protect the network from traffic
congestion risk.

```
                          +------------+
                          |    R1      | Root Node
                          +------------+
                               .
                               .
                               .
                          +------------+ Point of Local Repair/
                          |    R10     | Switchover Point
                          +-----------+ (Upstream LSR)
                          /        +
                      10/          +20
                        /            +
            +----------+         +-----------+
Protected Node |   R20    |         |    R30    |
            +----------+         +-----------+
            |      \     +          +
            |       \     +          +
          10|     10\  +20        20+
            |        \  +            +
            |         \ +            +
            |        + \             +
      +-----------+       +-----------+ Merge Point
      |   R40     |       |    R50    | (Downstream LSR)
      +-----------+       +-----------+
           .                   .
           .                   .
```
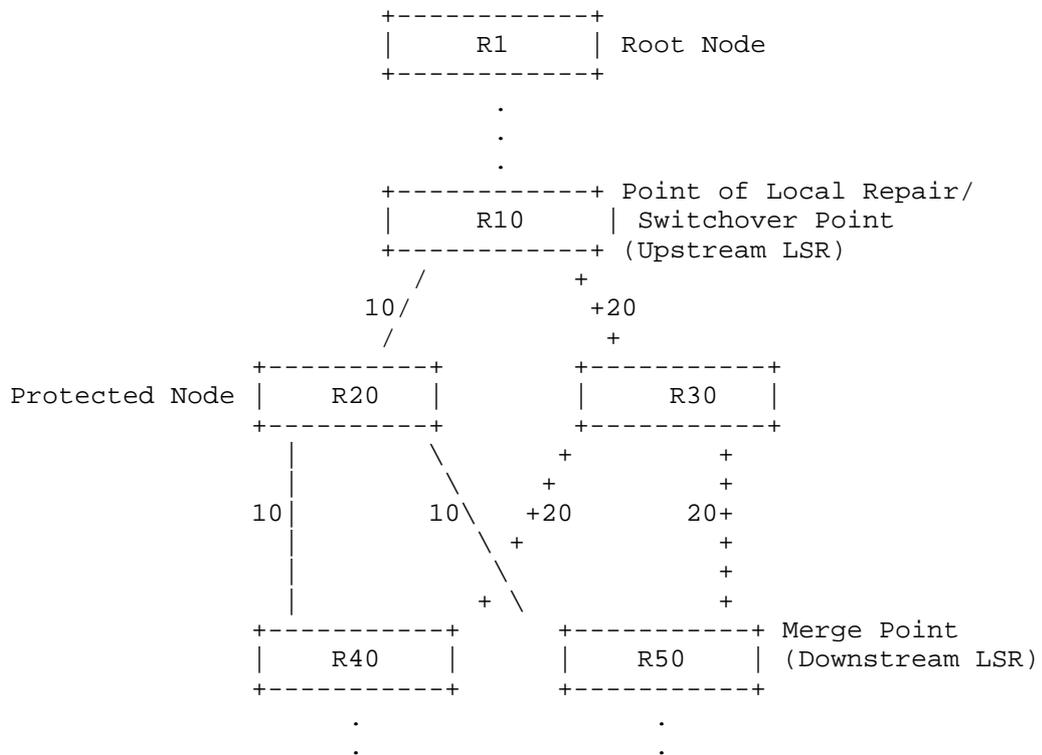
                Figure 7: Using PCECC for P2MP TE LocalProtection

   In the example above, when the PCECC setup the primary multicast path
   around the PLR node R10 to protect node R20, which is R10->R20->{R40,
   R50}, at same time, it can setup the backup path R10->R30->{R40,
   R50}. Both the these two primary forwarding path and secondary
   forwarding path will be downloaded to each routers along the primary
   path and the secondary path. The traffic will be forwarded through
   the R10->R20->{R40, R50} path normally, and when there is a node
   failure for node R20, then the PLR node R10 will switch the flow to
   the backup path, which is R10->R30->{R40, R50}. By using the PCECC,
   the path computation and forwarding path downloading can all be done
   without the complex signaling used in the P2MP RSVP-TE or mLDP.

8. Use Cases of PCECC for LSP in the Network Migration

   One of the main advantages for PCECC solution is that it has backward
   compatibility naturally since the PCE server itself can function as a
   proxy node of MPLS network for all the new nodes which don't support
   the existing MPLS signaling protocol anymore.

As it is illustrated in the following example, the current network
will migrate to a total PCECC controlled network gradually by
replacing the legacy nodes. During the migration, the legacy nodes
still need to signal using the existing MPLS protocol such as LDP and
RSVP-TE, and the new nodes setup their portion of the forwarding path
through PCECC directly. With the PCECC function as the proxy of
these new nodes, MPLS signaling can populate through network as
normal.

Example described in this section is based on network configurations
illustrated using the following figure:

```
+-----------------------------------------------------------------+
|                          PCE DOMAIN                             |
|   +-----------------------------------------------------+       |
|   |                        PCECC                        |       |
|   +-----------------------------------------------------+       |
|       ^                 ^                 ^         ^            |
|       |     RSVP-TE |  if22           if44|RSVP-TE  |           |
|       V                 V                 V         V           |
|   +--------+   +--------+   +--------+   +--------+   +--------+ |
|   | NODE 1 |   | NODE 2 |   | NODE x |   | NODE 4 |   | NODE 5 | |
|   |        |...|        |...|        |...|        |...|        | |
|   | Legacy |if1| Legacy |if2|PCCEC   |if3| PCECC  |if4| Legacy | |
|   | Node   |   | Node   |   |Enabled |   |Enabled |   | Node   | |
|   +--------+   +--------+   +--------+   +--------+   +--------+ |
|                                                                 |
+-----------------------------------------------------------------+
```

Figure 8: Using PCECC During Migration

Example: PCECC Initiated LSP Setup In the Network Migration

In this example, there are five nodes for the TE LSP from head end
(node1) to the tail end (node5). Where the NodeX is central
controlled and other nodes are legacy nodes.

o Node1 sends a path request message for the setup of LSP
   destinating to Node5.

o PCECC sends a reply message for LSP setup with path (node1, if1),
   (node2, if22), (node-PCECC, if44), (node4, if4), Nnode5.

o Node1, Node2, Node-PCECC, Node 5 will setup the LSP to Node5
   normally using the local label as normal.

o Then the PCECC will program the outsegment of Node2, the insegment
   of Node4, and the insegment/outsegment for NodeX.

9. Use Cases of PCECC for L3VPN and PWE3

   The existing services using MPLS LSP tunnels based on MPLS signalling
   mechanism such L3VPN, PWE3 and IPv6 can be simplified by using the
   PCECC to negoitate the label assignments for the L3VPN, PWE3 and
   Ipv6.

   In the case of L3VPN, VPN labels can be negotiated and distributed
   through the PCECC PCEP among the PE router instead of using the BGP
   protocols.

```
         +-------------------------------------------+
         |                PCE DOMAIN                 |
         |    +-------------------------------+      |
         |    |             PCECC             |      |
         |    +-------------------------------+      |
         |           ^            ^            ^     |
         |PWE3/L3VPN  | PCEP  PCEP|LSP PWE3/L3VPN|PCEP|
         |           V            V            V     |
  +--------+ |    +--------+  +--------+  +--------+  | +--------+
  |  CE    | |    | PE1    |  | NODE x |  | PE2    |  | |  CE    |
  |        | |......|        |...|        |...|        |.....| |        |
  | Legacy | | |if1 | PCECC  |if2|PCCEC   |if3| PCECC  |if4| | Legacy |
  | Node   | |    | Enabled|  |Enabled |  |Enabled |  | | Node   |
  +--------+ |    +--------+  +--------+  +--------+  | +--------+
         |                                           |
         +-------------------------------------------+
```
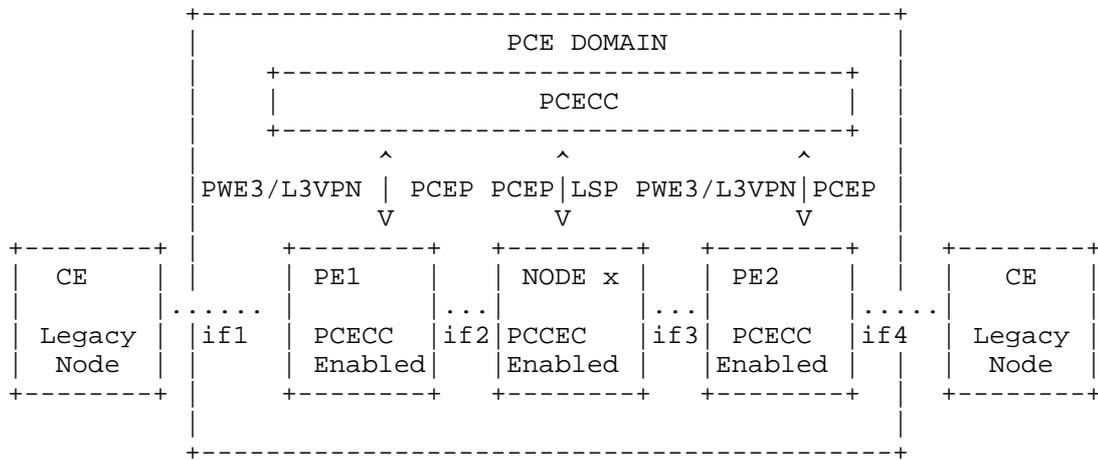
Figure 9: Using PCECC for L3VPN and PWE3

   In the cast PWE3, instead of using the LDP signalling protocols, the
   lable and port pairs assigned to each pseudowire can be negotiated
   through PCECC among the PE rotuers and the corresponding forwarding
   entries will be distributed into each PE routers through the extended
   PCEP protocols.

10. The Considerations for PCECC Procedure and PCEP extensions

   The PCECC's procedures and PCEP extensions is defined in [I-D.zhao-
   pce-pcep-extension-for-pce-controller].

11. IANA Considerations

   This document does not require any action from IANA.

12. Security Considerations

   TBD.

13. Acknowledgments

   We would like to thank Robert Tao, Changjiang Yan, Tieying Huang for
   their useful comments and suggestions.

14. References

14.1. Normative References

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

   [RFC5440]   Vasseur, JP. and
                                            JL. Le Roux, "Path
                                            Computation Element
                                            (PCE) Communication
                                            Protocol (PCEP)",
                                            RFC 5440,
                                            March 2009.

14.2. Informative References

   [RFC5441]   Vasseur, JP.,
                                            Zhang, R., Bitar,
                                            N., and JL. Le
                                            Roux, "A Backward-
                                            Recursive PCE-Based
                                            Computation (BRPC)
                                            Procedure to
                                            Compute Shortest
                                            Constrained Inter-
                                            Domain Traffic
                                            Engineering Label
                                            Switched Paths",
                                            RFC 5441,

                                             April 2009.

        [RFC5541]   Le Roux, JL.,
                                             Vasseur, JP., and
                                             Y. Lee, "Encoding
                                             of Objective
                                             Functions in the
                                             Path Computation
                                             Element
                                             Communication
                                             Protocol (PCEP)",
                                             RFC 5541,
                                             June 2009.

        [I-D.filsfils-spring-segment-routing]   Filsfils, C.,
                                             Previdi, S.,
                                             Bashandy, A.,
                                             Decraene, B.,
                                             Litkowski, S.,
                                             Horneffer, M.,
                                             Milojevic, I.,
                                             Shakir, R., Ytti,
                                             S., Henderickx, W.,
                                             Tantsura, J., and
                                             E. Crabbe, "Segment
                                             Routing
                                             Architecture", draf
                                             t-filsfils-spring-
                                             segment-routing-04
                                             (work in progress),
                                             July 2014.

        [I-D.ietf-pce-stateful-pce]   Crabbe, E., Minei,
                                             I., Medved, J., and
                                             R. Varga, "PCEP
                                             Extensions for
                                             Stateful PCE", draf
                                             t-ietf-pce-
                                             stateful-pce-11
                                             (work in progress),
                                             April 2015.

        [I-D.crabbe-pce-pce-initiated-lsp]   Crabbe, E., Minei,
                                             I., Sivabalan, S.,
                                             and R. Varga, "PCEP
                                             Extensions for PCE-
                                             initiated LSP Setup
                                             in a Stateful PCE

                                               Model", draft-
                                               crabbe-pce-pce-
                                               initiated-lsp-03
                                               (work in progress),
                                               October 2013.

     [I-D.ali-pce-remote-initiated-gmpls-lsp]   Ali, Z., Sivabalan,
                                               S., Filsfils, C.,
                                               Varga, R., Lopez,
                                               V., Dios, O., and
                                               X. Zhang, "Path
                                               Computation Element
                                               Communication
                                               Protocol (PCEP)
                                               Extensions for
                                               remote-initiated
                                               GMPLS LSP Setup", d
                                               raft-ali-pce-
                                               remote-initiated-
                                               gmpls-lsp-03 (work
                                               in progress),
                                               February 2014.

     [I-D.ietf-isis-segment-routing-extensions]   Previdi, S.,
                                               Filsfils, C.,
                                               Bashandy, A.,
                                               Gredler, H.,
                                               Litkowski, S.,
                                               Decraene, B., and
                                               J. Tantsura, "IS-IS
                                               Extensions for
                                               Segment Routing", d
                                               raft-ietf-isis-
                                               segment-routing-
                                               extensions-05 (work
                                               in progress),
                                               June 2015.

     [I-D.psenak-ospf-segment-routing-extensions]   Psenak, P.,
                                               Previdi, S.,
                                               Filsfils, C.,
                                               Gredler, H.,
                                               Shakir, R.,
                                               Henderickx, W., and
                                               J. Tantsura, "OSPF
                                               Extensions for
                                               Segment Routing", d
                                               raft-psenak-ospf-

                                              segment-routing-
                                              extensions-05 (work
                                              in progress),
                                              June 2014.

     [I-D.sivabalan-pce-segment-routing]   Sivabalan, S.,
                                              Medved, J.,
                                              Filsfils, C.,
                                              Crabbe, E., Raszuk,
                                              R., Lopez, V., and
                                              J. Tantsura, "PCEP
                                              Extensions for
                                              Segment Routing", d
                                              raft-sivabalan-pce-
                                              segment-routing-03
                                              (work in progress),
                                              July 2014.

     [I-D.li-mpls-global-label-usecases]   Li, Z., Zhao, Q.,
                                              Yang, T., and R.
                                              Raszuk, "Use Cases
                                              of MPLS Global
                                              Label", draft-li-
                                              mpls-global-label-
                                              usecases-02 (work
                                              in progress),
                                              July 2014.

     [I-D.li-mpls-global-label-framework]   Li, Z., Zhao, Q.,
                                              Chen, X., Yang, T.,
                                              and R. Raszuk, "A
                                              Framework of MPLS
                                              Global Label", draf
                                              t-li-mpls-global-
                                              label-framework-02
                                              (work in progress),
                                              July 2014.

     [I-D.zhao-pce-pcep-extension-for-pce-controller] Zhao, Q., Zhao, K.,
                                              Li, Z., Dhody, D.,
                                              Palle, U., and T.
                                              Communications,
                                              "PCEP Procedures
                                              and Protocol
                                              Extensions for
                                              Using PCE as a
                                              Central Controller
                                              (PCECC) of LSPs", d

                                                       raft-zhao-pce-pcep-
                                                       extension-for-pce-
                                                       controller-01 (work
                                                       in progress),
                                                       March 2015.

   [I-D.ietf-spring-resiliency-use-cases]   Francois, P.,
                                                       Filsfils, C.,
                                                       Decraene, B., and
                                                       R. Shakir, "Use-
                                                       cases for
                                                       Resiliency in
                                                       SPRING", draft-
                                                       ietf-spring-
                                                       resiliency-use-
                                                       cases-01 (work in
                                                       progress),
                                                       March 2015.

Authors' Addresses

   Quintin Zhao
   Huawei Technologies
   125 Nagog Technology Park
   Acton, MA 01719
   US

   EMail: quintin.zhao@huawei.com

   Robin Li
   Huawei Technologies
   Huawei Bld., No.156 Beiqing Rd.
   Beijing 100095
   China

   EMail: lizhenbin@huawei.com

   King Ke
   Tencent Holdings Ltd.
   Shenzhen
   China

   EMail: kinghe@tencent.com

Luyuan Fang
Microsoft

EMail: lufang@microsoft.com

Chao Zhou
Cisco Systems

EMail: chao.zhou@cisco.com

Boris Zhang
Telus Communications
200 Consilium Pl Floor 15
Toronto, ON   M1H 3J3
Canada

EMail: Boris.Zhang@telus.com

Katherine Zhao
Huawei Technologies
2330 Central Expressway
Santa Clara, CA   95050
USA

EMail: Katherine.zhao@huawei.com

Richard Li
Huawei Technologies
2330 Central Expressway
Santa Clara, CA   95050
USA

EMail: renwei.li@huawei.com

PCE Working Group   Q. Zhao
Internet-Draft   K. Zhao
Intended status: Experimental   Z. Li
Expires: September 3, 2015   D. Dhody
                                                              U. Palle
                                                   Huawei Technologies
                                                              B. Zhang
                                                            Telus Ltd.
                                                         March 2, 2015

        PCEP Procedures and Protocol Extensions for Using PCE as a Central
                         Controller (PCECC) of LSPs
              draft-zhao-pce-pcep-extension-for-pce-controller-01

Abstract

   In certain networks deployment scenarios, service providers would
   like to keep all the existing MPLS functionalities in both MPLS and
   GMPLS while removing the complexity of existing signaling protocols
   such as LDP and RSVP-TE. In
   [I-D.zhao-pce-central-controller-user-cases], we propose to use the
   PCE [RFC5440] as a central controller (PCECC) so that LSP can be
   calculated/ signaled/initiated and label forwarding entries are
   downloaded through a centralized PCE server to each network devices
   along the LSP path while leveraging the existing PCE technologies as
   much as possible.

   This draft specify the procedures and PCEP protocol extensions for
   using the PCE as the central controller and user cases where LSPs are
   calculated/setup/initiated and label forwarding entries are
   downloaded through extending the existing PCE architectures and PCEP.

   This document also discuss the role of PCECC in Segment Routing (SR).

Status of This Memo

time. It is inappropriate to use Internet-Drafts as reference
material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 3, 2015.

Copyright Notice

Table of Contents

1. Introduction

   In certain network deployment scenarios, service providers would like
   to have the ability to dynamically adapt to a wide range of
   customer's requests for the sake of flexible network service
   delivery, Software Defined Networks(SDN) has provides additional
   flexibility in how the network is operated compared to the
   traditional network.

   The existing networking ecosystem has become awfully complex and
   highly demanding in terms of robustness, performance, scalability,
   flexibility, agility, etc. By migrating to the SDN enabled network
   from the existing network, service providers and network operators
   must have a solution which they can evolve easily from the existing
   network into the SDN enabled network while keeping the network
   services remain scalable, guarantee robustness and availability etc.

   Taking the smooth transition between traditional network and the new
   SDN enabled network into account, especially from a cost impact
   assessment perspective, using the existing PCE components from the
   current network to function as the central controller of the SDN
   network is one choice, which not only achieves the goal of having a
   centralized controller, but also leverages the existing PCE network
   components.

The Path Computation Element communication Protocol (PCEP) provides
mechanisms for Path Computation Elements (PCEs) to perform route
computations in response to Path Computation Clients (PCCs) requests.
PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model
[I-D.ietf-pce-stateful-pce] describes a set of extensions to PCEP to
enable active control of MPLS-TE and GMPLS tunnels.

[I-D.ietf-pce-pce-initiated-lsp] describes the setup and teardown of
PCE-initiated LSPs under the active stateful PCE model, without the
need for local configuration on the PCC, thus allowing for a dynamic
MPLS network that is centrally controlled and deployed.

[I-D.ietf-pce-remote-initiated-gmpls-lsp] complements
[I-D.ietf-pce-pce-initiated-lsp] by addressing the requirements for
remote-initiated GMPLS LSPs.

Segment Routing (SR) technology leverages the source routing and
tunneling paradigms. A source node can choose a path without relying
on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path
is specified as a set of "segments" advertised by link-state routing
protocols (IS-IS or OSPF). [I-D.ietf-spring-segment-routing]
provides an introduction to SR technology. The corresponding IS-IS
and OSPF extensions are specified in
[I-D.ietf-isis-segment-routing-extensions] and
[I-D.ietf-ospf-segment-routing-extensions], respectively.

A Segment Routed path (SR path) can be derived from an IGP Shortest
Path Tree (SPT). Segment Routed Traffic Engineering paths (SR-TE
paths) may not follow IGP SPT. Such paths may be chosen by a
suitable network planning tool and provisioned on the source node of
the SR-TE path.

It is possible to use a stateful PCE for computing one or more SR-TE
paths taking into account various constraints and objective
functions. Once a path is chosen, the stateful PCE can instantiate
an SR-TE path on a PCC using PCEP extensions specified in
[I-D.ietf-pce-pce-initiated-lsp] using the SR specific PCEP
extensions described in [I-D.ietf-pce-segment-routing].

PCECC may further use PCEP protocol for SR label distribution instead
of IGP extensions with some benefits.

Current MPLS label has local meaning. That is, MPLS label is always
allocated by the downstream node to the upstream node. Then the MPLS
label is only identified by the neighboring upstream node and
downstream node. The label allocation is done locally and signaled
through the LDP/RSVP-TE/BGP protocol. To ease the label allocation
and signaling mechanism, PCE can be conveniently used as a central

controller with Label download capability. Further PCE can also be
used to manage the label range and SRGB etc.

The PCECC solution introduced in
[I-D.zhao-pce-central-controller-user-cases] allow for a dynamic MPLS
network that is eventually controlled and deployed without the
deployment of RSVP-TE protocol or extended IGP protocol with node/
adjacency segment identifiers signaling capability while providing
all the key MPLS functionalities needed by the service providers.

This draft specify the procedures and PCEP protocol extensions for
using the PCE as the central controller and user cases where LSPs are
calculated/setup/initiated/downloaded through extending the existing
PCE architectures and PCEP.

1.1. 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 [RFC2119].

2. Terminology

The following terminology is used in this document.

IGP: Interior Gateway Protocol. Either of the two routing
   protocols, Open Shortest Path First (OSPF) or Intermediate System
   to Intermediate System (IS-IS).

PCC: Path Computation Client: any client application requesting a
   path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application,
   or network node) that is capable of computing a network path or
   route based on a network graph and applying computational
   constraints.

TE: Traffic Engineering.

3. PCECC Modes

The following PCECC modes are supported -

o Basic PCECC.

o PCECC SR.

   * PCECC SR-BE (Best Effort).

* PCECC SR-TE (Traffic Engineered).

In basic PCECC mode, the forwarding is similar to RSVP-TE signalled LSP without the RSVP-TE signaling. The PCECC allocates and download the label entries along the LSP. The rest of processing is similar to the existing stateful PCE mechanism.

In case of SR, there are two modes for SR-BE and SR-TE. For SR-BE, the forwarding is similar to LDP LSP without LDP signaling or IGP-SR extension. The SR Node label are allocated and distributed in the domain centrally by the PCE via PCEP. Each node (PCC) rely on local IGP for the nexthop calculation. For SR-TE, the forwarding uses label stack similar to IGP based SR-TE without IGP-SR extension. The SR node and adj labels are allocated and distributed in the domain centrally by the PCE via PCEP by PCECC. Rest of the processing is similar to existing stateful PCE with SR mechanism.

4. PCEP Requirements

Following key requirements associated PCECC should be considered when designing the PCECC based solution:

1. PCEP speaker supporting this draft MUST have the capability to advertise its PCECC capability to its peers.

2. Path Computation Client (PCC) supporting this draft MUST have a capability to communicate local label range or global label range or both to PCE.

3. Path Computation Element (PCE) supporting this draft SHOULD have the capability to negotiate a global label range for a group of clients and communicate the final global label range to PCC.

4. PCEP speaker not supporting this draft MUST be able to reject PCECC related message with a reason code that indicates no support for PCECC.

5. PCEP SHOULD provide a means to identify PCECC based LSP in the PCEP messages.

6. PCEP SHOULD provide a means to update (or cleanup) the label-download or label-map entry to the PCC.

5. Procedures for Using the PCE as the Central Controller (PCECC)

5.1. Stateful PCE Model

   Active stateful PCE is described in [I-D.ietf-pce-stateful-pce]. PCE
   as a central controller (PCECC) reuses existing Active stateful PCE
   mechanism as much as possible to control the LSP.

5.2. New LSP Functions

   This document defines the following new PCEP messages and extends the
   existing messages to support PCECC:

   (PCLRResv): a PCEP message sent by a PCC to a PCE to ask for the
      label range reservation or a PCE to a PCC to send the reserved
      label range. The PCLRResv message described in Section 6.1.

   (PCLabelUpd): a PCEP message sent by a PCE to a PCC to download or
      cleanup the Label entry. The PCLabelUpd message described in
      Section 6.2.

   (PCRpt): a PCEP message described in [I-D.ietf-pce-stateful-pce].
      PCRpt message MAYBE used to send PCECC LSP Reports.

   (PCInitiate): a PCEP message described in
      [I-D.ietf-pce-pce-initiated-lsp]. PCInitiate message is used to
      setup PCE-Initiated LSP based on PCECC mechanism.

   (PCUpd): a PCEP message described in [I-D.ietf-pce-stateful-pce].
      PCUpd message is used to send PCECC LSP Update.

   The new LSP functions defined in this document are mapped onto the
   messages as shown in the following table.

   +---+-------------------------+
   | Function | Message |
   +---+-------------------------+
PCECC Capability advertisement	Open
Label Range Reservation	PCLRResv
Label entry Update	PCLabelUpd
Label entry Cleanup	PCLabelUpd
PCECC Initiated LSP	PCInitiate
PCECC LSP Update	PCUpd
PCECC LSP State Report	PCRpt
PCECC LSP Delegation	PCRpt
   +---+-------------------------+

5.3. PCECC Capability Advertisement

   During PCEP Initialization Phase, PCEP Speakers (PCE or PCC)
   advertise their support of PCECC extensions. A PCEP Speaker includes
   the "PCECC Capability" TLV, described in Section 7.1.1 of this
   document, in the OPEN Object to advertise its support for PCECC
   extensions.

   The presence of the PCECC Capability TLV in PCC's OPEN Object
   indicates that the PCC is willing to function as a PCECC client.

   The presence of the PCECC Capability TLV in PCE's OPEN message
   indicates that the PCE is interested in function as a PCECC server.

   The PCEP protocol extensions for PCECC MUST NOT be used if one or
   both PCEP Speakers have not included the PCECC Capability TLV in
   their respective OPEN message. If the PCEP Speakers support the
   extensions of this draft but did not advertise this capability then a
   PCErr message with Error-Type=19(Invalid Operation) and Error-
   Value=[TBD] (Attempted LSP setup/download/label-range reservation if
   PCECC capability was not advertised) will be generated and the PCEP
   session will be terminated.

   L flag and G flag defined in PCECC Capability TLV specifies the local
   and global label range reservation capability.

   A PCC or a PCE MUST include both PCECC-CAPABILITY TLV and STATEFUL-
   PCE-CAPABILITY TLV in OPEN Object to support the extensions defined
   in this document. If PCECC-CAPABILITY TLV is advertised and
   STATEFUL-PCE-CAPABILITY TLV is not advertised in OPEN Object, it
   SHOULD send a PCErr message with Error-Type=19 (Invalid Operation)
   and Error-value=[TBD](stateful pce capability was not advertised) and
   terminate the session.

5.4. Label Range Reservation

   After PCEP initial state synchronization, the label range is
   reserved.

   If L flag is advertised in OPEN Object by PCEP speakers, a PCC
   reserves a local label range and is communicated using PCLRResv
   message to a PCE. The PCE maintains the local label range of each
   node and further during LSP setup, a label is assigned to each node
   from the corresponding local label range reserved.

   If G flag is advertised in OPEN Object by PCEP speakers,a PCC
   reserves a global label range and is advertised in PCLRResv message
   to a PCE. The PCE MAY negotiate and reserves the global label range

and also sends the negotiated global label range in PCLRResv message
to the PCC. Please refer [I-D.li-mpls-global-label-framework] for
MPLS global label allocation.

A PCC MUST send PCLRResv message immediately after the initial LSP
synchronization completion. A PCE SHOULD not send PCLabelUpd message
to a PCC before PCLRResv message received. If the PCC received
PCLabelUpd message and not initiated label range reservation, it
SHOULD send a PCErr message with Error-type=[TBD] (label range not
reserved) and Error-value=[TBD].

The label range reservation sequence is shown below.

```
    +-+-+                          +-+-+
    |PCC|                          |PCE|
    +-+-+                          +-+-+
      |                              |
      |--- PCLRResv, label type=1 --->|local label range reserved
      |              (100-500)        |global label range negotiated
      |              label type=2     |
      |              (600-1000)       |
      |                              |
      |<--- PCLRResv, label type=2 ---|global label range reserved
      |              (700-1000)       |
      |                              |
```

[Editor's Note: This section of the document would be updated with
more details about Label Block Negotiation, Reservation, Adjustment
etc in a future revision of the document.]

5.5. LSP Operations

The PCEP messages pertaining to PCECC MUST include PATH-SETUP-TYPE
TLV [I-D.sivabalan-pce-lsp-setup-type] in the SRP object to clearly
identify the PCECC LSP is intended.

5.5.1. Basic PCECC Mode

5.5.1.1. PCECC LSP Setup

Inorder to setup a LSP based on PCECC mechanism, a PCC MUST delegate
the LSP by sending a PCRpt message with Path Setup Type set for basic
PCECC (see Section 7.3) and D (Delegate) flag (see
[I-D.ietf-pce-stateful-pce]) set in the LSP object.

The LSP-ID in LSP-IDENTIFIER TLV (which usually corresponds to the
RSVP-TE LSP-ID) for PCECC LSP MUST always be generated by the PCE.
In the first PCRpt message of PCECC LSP, LSP ID of LSP-IDENTIFIER TLV
is set to zero.

When a PCE received PCRpt message with P and D flags set, it
generates LSP ID; calculates the path and assign labels along the
path; and setup the path by sending PCLabelUpd message to each node
along the path of the LSP.

The PCE SHOULD send the PCUpd message with the same PLSP-ID to the
Ingress PCC in response to the delegate PCRpt message.

The PCECC LSPs MUST be delegated to a PCE at all times.

LSP deletion operation for PCECC LSP is same as defined in
[I-D.ietf-pce-stateful-pce]. If the PCE received PCRpt message for
LSP deletion then it does Label cleanup operation as described in
Section 5.5.1.3 for the corresponding LSP.

The Basic PCECC LSP setup sequence is as shown below.

```
                        +-------+                    +-------+
                        |PCC    |                    | PCE   |
                        |Ingress|                    +-------+
                +------|       |                        |
                | PCC  +-------+|                        |
                | Transit| |                            |
        +------|        | ||--- PCRpt,PLSP-ID=1, P=1, D=1  --->| PCECC LSP
        |PCC    +--------+ |       (LSP ID=0)             |(LSPID=1)
        |Egress |  |   |                                  |
        +-------+  |   |                                  |
           |   |   |   |                                  |
           |<------ PCLabelUpd,PLSP-ID=1  ------------------ | Label
           |   |   |        (LSP ID=1)                    | download
           |   |   |                                      |
           |   |   |<----- PCLabelUpd,PLSP-ID=1  ----------- | Label
           |   |   |        (LSP ID=1)                    | download
           |   |   |                                      |
           |   |   |<--- PCLabelUpd,PLSP-ID=1  ------- | Label
           |   |   |        (LSP ID=1)                    | download
           |   |   |                                      |
           |   |   |<-- PCUpd,PLSP-ID=1, P=1, D=1 ---- | PCECC LSP
           |   |   |        (LSP ID=1)                    | Update
           |   |   |                                      |
```
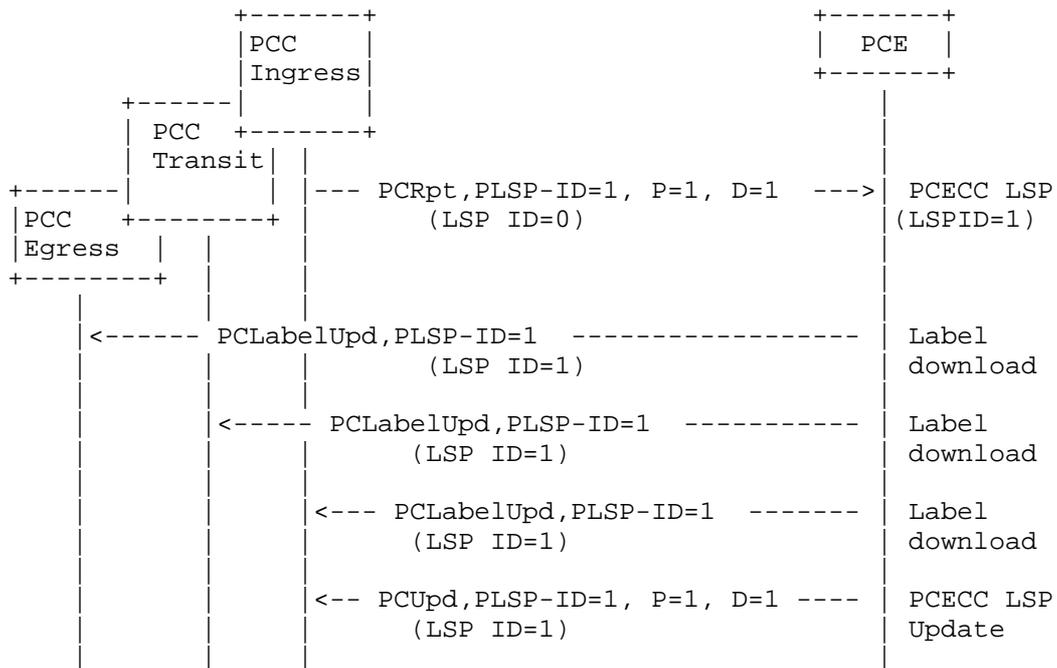
The PCECC LSP are considered to be 'up' by default. The Ingress MAY
further choose to deploy a data plane check mechanism and report the
status back to the PCE via PCRpt message.

5.5.1.2. Label Download

Inorder to setup an LSP based on PCECC, the PCE sends a PCLabelUpd
message to each node of the LSP to download the Label entry as
described in Section 5.5.1.1.

The LSP object in PCLabelUpd MUST include the LSP-IDENTIFIER TLV.

If a node (PCC) received a PCLabelUpd message but failed to download
the Label entry, it MUST send a PCErr message with Error-type=[TBD]
(label download failed) and Error-value=[TBD].

5.5.1.3. Label Cleanup

Inorder to delete an LSP based on PCECC, the PCE sends a PCLabelUpd
message to each node along the path of the LSP to cleanup the Label
entry.

If the PCC received a PCLabelUpd message but does not recognize the
label, the PCC MUST generate a PCErr message with Error-Type
19(Invalid operation) and Error-Value=3, "Unknown Label".

The R flag in SRP object defined in [I-D.ietf-pce-pce-initiated-lsp]
specifies the deletion of Label Entry in the PCLabelUpd message.

```
                   +-------+                      +-------+
                   |PCC    |                      | PCE   |
                   |Ingress|                      +-------+
            +------|       |                          |
            | PCC  +-------+                          |
            | Transit| |                              |
   +------|         | |--- PCRpt,PLSP-ID=1,P=1,D=1,R=1--->| PCECC LSP
   |PCC    +--------+ |           (LSP ID=1)             | remove
   |Egress |  |   |                                      |
   +--------+  |   |                                      |
       |       |   |                                      |
       |<------ PCLabelUpd, PLSP-ID=1 ------------------ | Label
       |       |   |          (LSP ID=1, R=1)            | cleanup
       |       |   |                                      |
       |       |<----- PCLabelUpd, PLSP-ID=1 ----------- | Label
       |       |   |          (LSP ID=1, R=1)            | cleanup
       |       |   |                                      |
       |       |   |<--- PCLabelUpd, PLSP-ID=1 ------- | Label
       |       |   |          (LSP ID=1, R=1)            | cleanup
       |       |   |                                      |
```
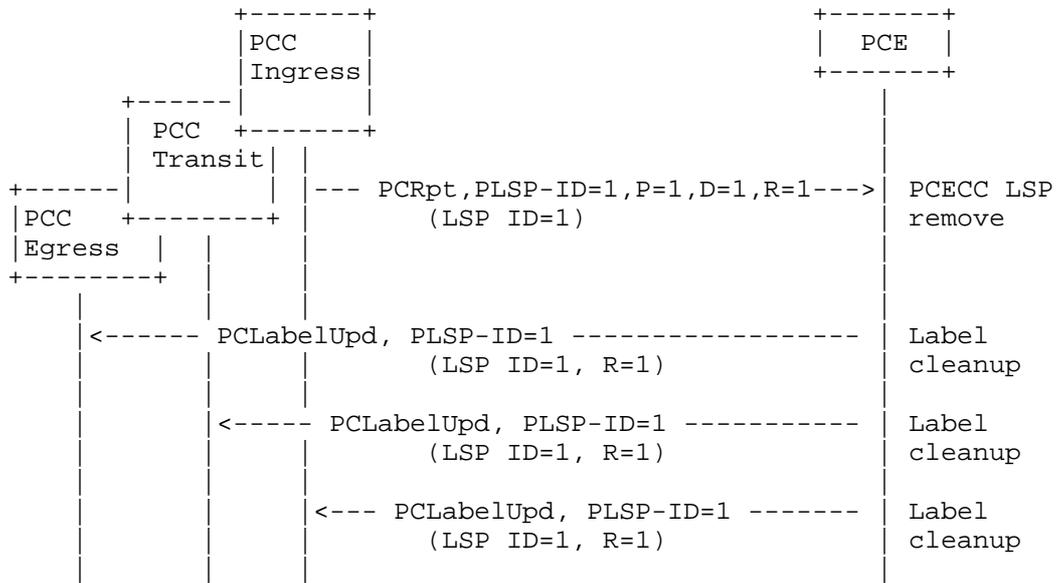
5.5.1.4. PCE Initiated PCECC LSP

   The LSP Instantiation operation is same as defined in
   [I-D.ietf-pce-pce-initiated-lsp].

   Inorder to setup a PCE Initiated LSP based on PCECC mechanism, a PCE
   sends PCInitiate message with Path Setup Type set for basic PCECC
   (see Section 7.3) to the Ingress PCC.

   The Ingress PCC MUST also set D (Delegate) flag (see
   [I-D.ietf-pce-stateful-pce]) and C (Create) flag (see
   [I-D.ietf-pce-pce-initiated-lsp]) in LSP object of PCRpt message.
   The PCC responds with first PCRpt message with the status as "GOING-
   UP" and assigned PLSP-ID.

   The rest of the PCECC LSP setup operations are same as those
   described in Section 5.5.1.1.

   The LSP deletion operation for PCE Initiated PCECC LSP is same as
   defined in [I-D.ietf-pce-pce-initiated-lsp]. The PCE should further
   perform Label entry cleanup operation as described in Section 5.5.1.3
   for the corresponding LSP.

   The PCE Initiated PCECC LSP setup sequence is shown below.

```
                      +-------+                  +-------+
                      |PCC    |                  |  PCE  |
                      |Ingress|                  +-------+
              +------|       |                      |
              | PCC  +-------+                       |
              | Transit| |                           |
   +------|          | |<---PCInitiate,PLSP-ID=0,P=1,D=1---| PCECC LSP
   |PCC   +--------+ |                              | Initiate
   |Egress  | |       |--- PCRpt,PLSP-ID=2,P=1,D=1,C=1--->| PCECC LSP
   +--------+ |       |         (LSP ID=0,GOING-UP)        |(LSPID=2
      |       | |     |                              | assigned)
      |<------ PCLabelUpd, PLSP-ID=2 -----------------| Label
      |       | |          (LSP ID=2)                | download
      |       | |     |                              |
      |       | |<----- PCLabelUpd, PLSP-ID=2 -----------| Label
      |       | |          (LSP ID=2)                | download
      |       | |     |                              |
      |       | |     |<--- PCLabelUpd, PLSP-ID=2 -------| Label
      |       | |     |     (LSP ID=2)                | download
      |       | |     |                              |
      |       | |     |<-- PCUpd, PLSP-ID=2, P=1, D=1----| PCECC LSP
      |       | |     |     (LSP ID=2)                | Update
      |       | |     |                              |
```
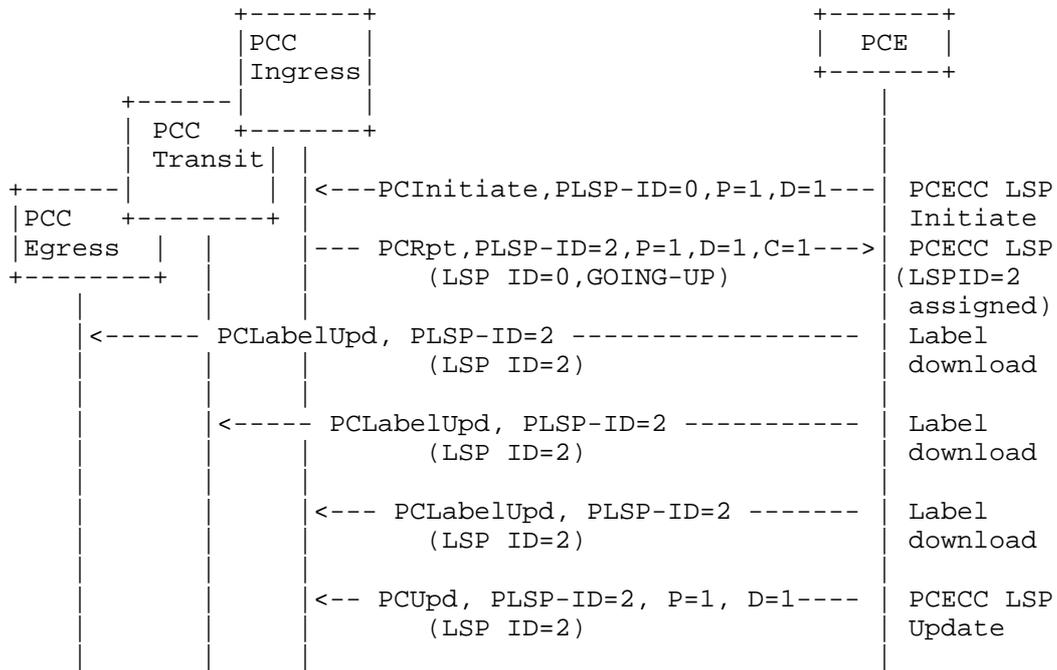
5.5.1.5. PCECC LSP Update

   Incase of a modification of PCECC LSP with a new path, a PCE sends a
   PCUpd message to the Ingress PCC.

   When a PCC received a PCUpd message for an existing LSP, a PCC MUST
   follow the make-before-break procedure. On successful traffic switch
   over to the new LSP, PCC sends a PCRpt message to the PCE for the
   deletion of old LSP. Further the PCE does cleanup operation for the
   old LSP described in Section 5.5.1.3.

   The PCECC LSP Update and make-before-break sequence is shown below.
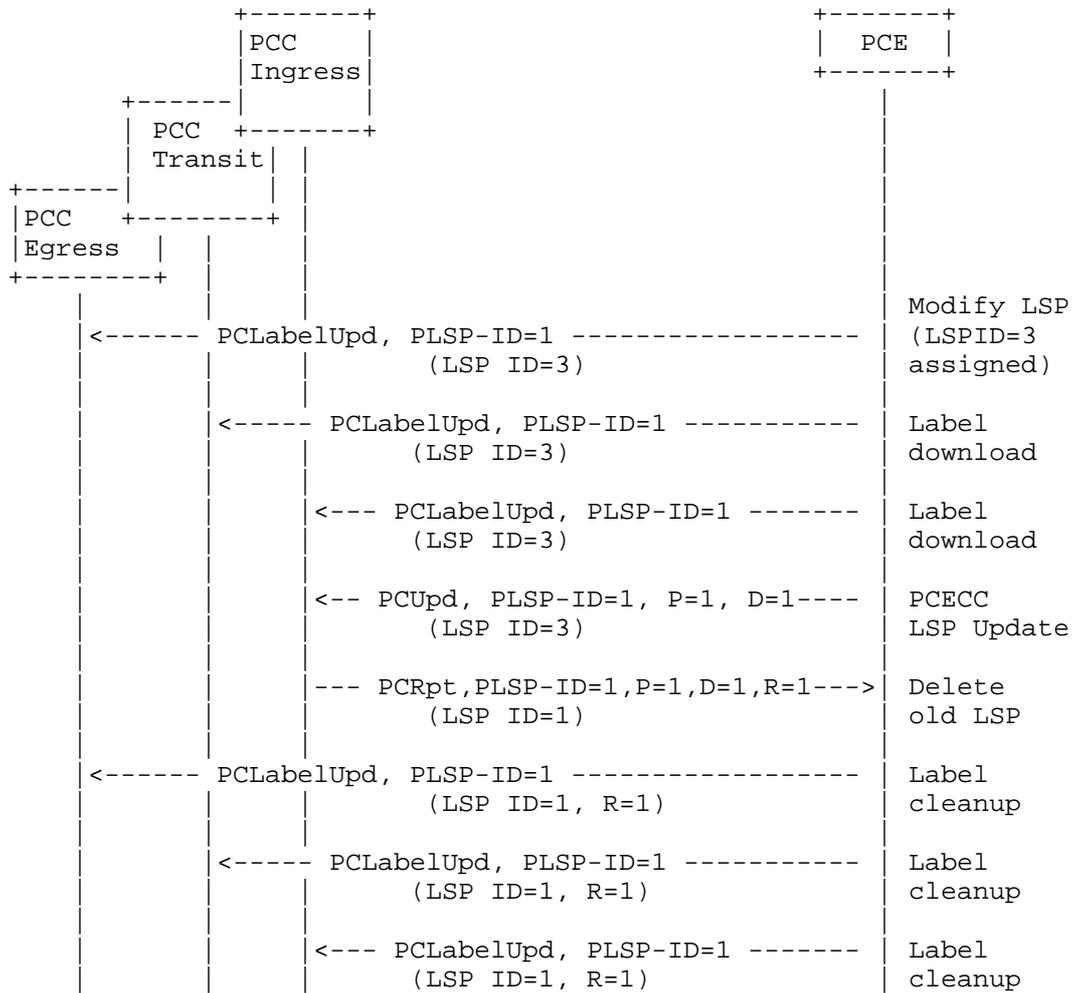
```
                    +-------+                      +-------+
                    |PCC    |                      | PCE   |
                    |Ingress|                      +-------+
            +------|       |                       |
            | PCC  +-------+                        |
            | Transit| |                            |
      +------|       | |                            |
      |PCC   +--------+ |                            |
      |Egress  | |     |                            |
      +-------+ | |    |                            |
         |      | |    |                            | Modify LSP
         |<------ PCLabelUpd, PLSP-ID=1 -----------------| (LSPID=3
         |      | |    | (LSP ID=3)                 | assigned)
         |      | |    |                            |
         |      |<----- PCLabelUpd, PLSP-ID=1 -----------| Label
         |      | |    | (LSP ID=3)                 | download
         |      | |    |                            |
         |      | |    |<--- PCLabelUpd, PLSP-ID=1 -------| Label
         |      | |    | (LSP ID=3)                 | download
         |      | |    |                            |
         |      | |    |<-- PCUpd, PLSP-ID=1, P=1, D=1----| PCECC
         |      | |    | (LSP ID=3)                 | LSP Update
         |      | |    |                            |
         |      | |    |--- PCRpt,PLSP-ID=1,P=1,D=1,R=1-->| Delete
         |      | |    | (LSP ID=1)                 | old LSP
         |      | |    |                            |
         |<------ PCLabelUpd, PLSP-ID=1 -----------------| Label
         |      | |    | (LSP ID=1, R=1)            | cleanup
         |      | |    |                            |
         |      |<----- PCLabelUpd, PLSP-ID=1 -----------| Label
         |      | |    | (LSP ID=1, R=1)            | cleanup
         |      | |    |                            |
         |      | |    |<--- PCLabelUpd, PLSP-ID=1 -------| Label
         |      | |    | (LSP ID=1, R=1)            | cleanup
```

The modified PCECC LSP are considered to be 'up' by default. The
Ingress MAY further choose to deploy a data plane check mechanism and
report the status back to the PCE via PCRpt message.

5.5.1.6. PCECC LSP State Report

As mentioned before, an Ingress PCC MAY choose to apply any OAM
mechanism to check the status of LSP in the Data plane and MAY
further send its status in PCRpt message to the PCE.

5.5.2. PCECC Segment Routing (SR)

   Segment Routing (SR) as described in
   [I-D.ietf-spring-segment-routing] depends on "segments" that are
   advertised by Interior Gateway Protocols (IGPs). The SR-node
   allocate and advertise the SID (node, adj etc) and flood via the IGP.
   This document proposes a new mechanism where PCE allocate the SID
   (label) centrally and uses PCEP to advertise the SID. In some
   deployments PCE (and PCEP) are better suited than IGP because of
   centralized nature of PCE and direct TCP based PCEP session to the
   node.

5.5.2.1. PCECC SR-BE

   Each node (PCC) is allocated a node-SID (label) by the PCECC. The
   PCECC sends PCLabelUpd to update the label map of each node to all
   the nodes in the domain. Each node (PCC) uses the local information
   to determines the next-hop and download the label forwarding
   instructions accordingly. The PCLabelUpd message in this case MUST
   not have LSP object but uses new FEC object.

```
                        +-------+                    +-------+
                        |PCC    |                    | PCE   |
                        |3.3.3.3|                    +-------+
                 +------|       |                        |
                 | PCC  +-------+                         |
                 | 2.2.2.2| |                            |
          +------|      | |                              |
          |PCC   +--------+ |                            |
          |1.1.1.1 |   |    |                            |
          +--------+   |    |                            |
              |        |    |                            |
              |<------ PCLabelUpd, FEC=1.1.1.1----------------- | Label Map
              |        |    |         Label=X                  | update
              |Find    |    |                                  |
              |Nexthop|<----- PCLabelUpd, FEC=1.1.1.1---------- | Label Map
              |locally|    |            Label=X                | update
              |        |    |                                  |
              |        |    |<--- PCLabelUpd, FEC=1.1.1.1------ | Label Map
              |        |    |              Label=X             | update
              |        |    |                                  |
```
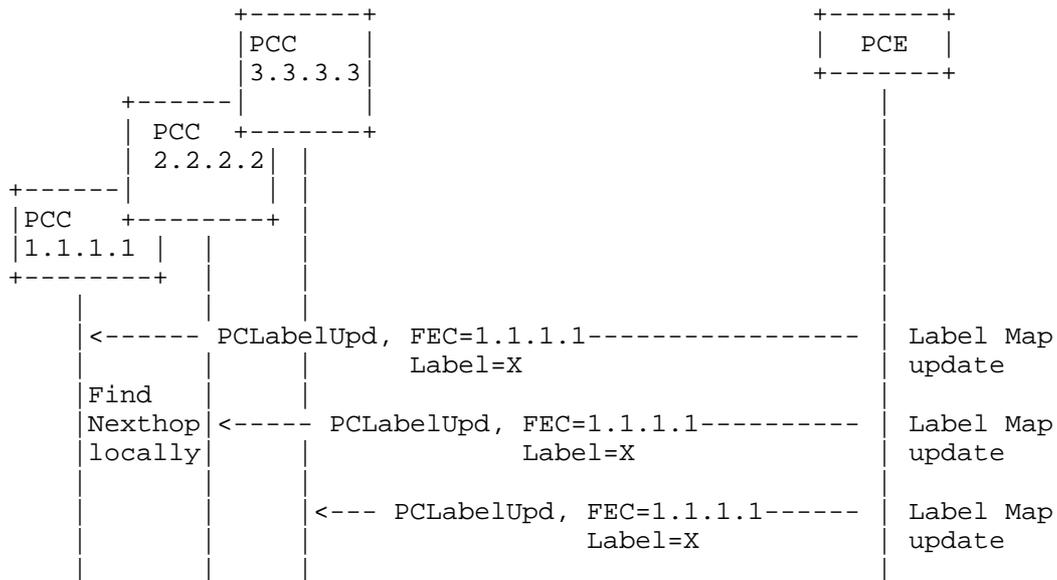
   The forwarding behavior and the end result is similar to IGP based
   "Node-SID" in SR. Thus, from anywhere in the domain, it enforces the
   ECMP-aware shortest- path forwarding of the packet towards the
   related node.

PCE rely on the Node label cleanup using the same PCLabelUpd message.

5.5.2.2. PCECC SR-TE

   A Segment Routed Best Effort path (SR-BE path) can be derived from an
   IGP Shortest Path Tree (SPT) as explained above. On the other hand,
   SR-TE paths may not follow IGP SPT. Such paths may be chosen by a
   PCE and provisioned on the source node of the SR-TE path.

   [I-D.ietf-pce-segment-routing] extends PCEP to allow a stateful PCE
   to compute and initiate SR-TE paths, as well as a PCC to request a
   path subject to certain constraint(s) and optimization criteria in SR
   networks.

   For SR-TE, apart from node-SID, Adj-SID is used where each adjacency
   is allocated an Adj-SID (label) by the PCECC. The PCECC sends
   PCLabelUpd to update the label map of each Adj to the corresponding
   nodes in the domain. Each node (PCC) download the label forwarding
   instructions accordingly. Similar to SR-BE, the PCLabelUpd message
   in this case MUST not have LSP object but uses new FEC object.

```
                    +-------+                      +-------+
                    |PCC    |                      |  PCE  |
                    |3.3.3.3|                      +-------+
            +------|       |                      |
            | PCC  +-------+                      |
            | 2.2.2.2| |                          |
    +------|       | |                            |
    |PCC   +--------+ |                           |
    |1.1.1.1 |  |    |                            |
    +--------+  |    |                            |
       |        |    |                            |
       |<------ PCLabelUpd, FEC=10.1.1.1 / ------------- |  Label Map
       |        |    |         10.1.1.2           |  update
       |        |    |        Label=A             |
       |        |    |                            |
       |        |<----- PCLabelUpd, FEC=10.1.1.2--------- |  Label Map
       |        |    |         10.1.1.1           |  update
       |        |    |         Label=B            |
       |        |    |                            |
```
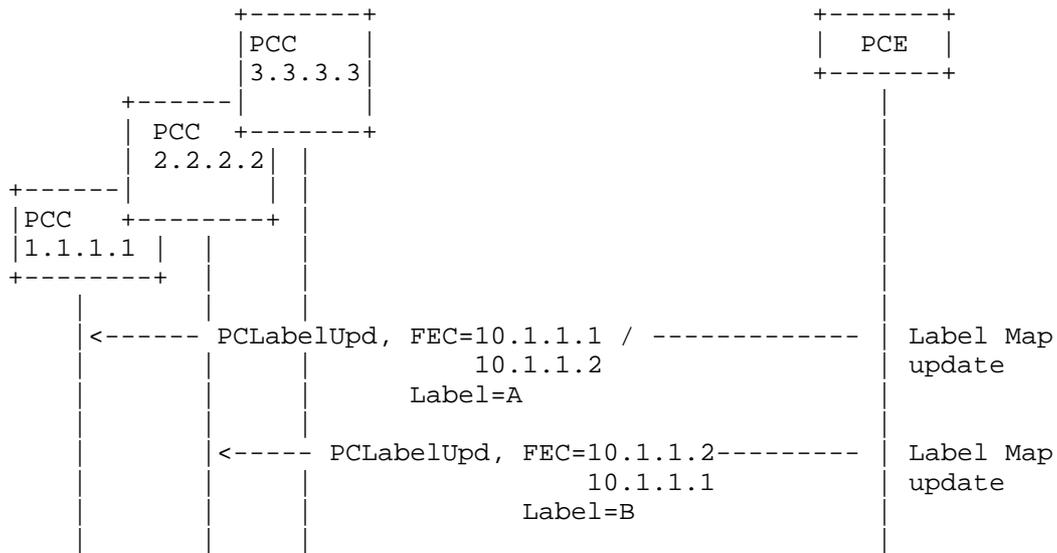
   The forwarding behavior and the end result is similar to IGP based
   "Adj-SID" in SR.

The Path Setup Type MUST be set for PCECC SR-TE (see Section 7.3).
The rest of the PCEP procedures and mechanism are similar to
[I-D.ietf-pce-segment-routing].

PCE rely on the Adj label cleanup using the same PCLabelUpd message.

6. PCEP messages

As defined in [RFC5440], a PCEP message consists of a common header
followed by a variable-length body made of a set of objects that can
be either mandatory or optional. An object is said to be mandatory
in a PCEP message when the object must be included for the message to
be considered valid. For each PCEP message type, a set of rules is
defined that specify the set of objects that the message can carry.
An implementation MUST form the PCEP messages using the object
ordering specified in this document.

6.1. The PCLRResv message

A Label Range Reservation message (also referred to as PCLRResv
message) is a PCEP message sent by a PCC to a PCE for the reservation
of label range or by PCE to PCC to send reserved label range for the
network. The Message-Type field of the PCEP common header for the
PCLRResv message is set to [TBD].

The format of a PCLRResv message is as follows:

PCLRResv Message>::= <Common Header>
                                <label-range>
    Where:

      <label-range> ::= <SRP>
                        <labelrange-list>

    Where
<labelrange-list>::=<LABEL-RANGE>[<labelrange-list>]

There are two mandatory objects that MUST be included within each
<label-range> in the PCLRResv message: the SRP Object and LABEL-RANGE
object.

SRP object is defined in [I-D.ietf-pce-stateful-pce] and this
document extends the use of SRP object in PCLRResv message. If the
SRP object is missing, the receiving PCE MUST send a PCErr message
with Error-type=6 (Mandatory Object missing) and Error-value=10 (SRP
object missing).

PCC generates the value of SRP-ID-number in SRP object of PCLRResv
message send to a PCE. The PCE MUST include the same SRP-ID-number
in SRP object of PCLRResv message sent to the PCC in response to
PCLRResv message.

LABEL-RANGE object is defined in Section 7.2. If the LABEL-RANGE
object is missing, the receiving PCE MUST send a PCErr message with
Error-type=6 (Mandatory Object missing) and Error-value=[TBD] (Label
object missing).

[Editor's Note: This section of the document would be updated with
more details about Label Block Negotiation, Reservation, Adjustment
etc in a future revision of the document.]

6.2. The PCLabelUpd message

The Label Update Message (also referred to as PCLabelUpd) is a PCEP
message sent by a PCE to a PCC to download label or update the label
map. The same message is also used to cleanup the Label entry. The
Message-Type field of the PCEP common header for the PCLabelUpd
message is set to [TBD].

The format of the PCLabelUpd message is as follows:

<PCLabelUpd Message> ::= <Common Header>
                               <pce-label-update-list>
Where:
<pce-label-update-list> ::= <pce-label-update>
                         [<pce-label-update-list>]

<pce-label-update> ::= (<pce-label-download>|<pce-label-map>)

Where:
<pce-label-download> ::= <SRP>
                                <LSP>
                         <label-list>

<pce-label-map> ::= <SRP>
                       <LABEL>
                       <FEC>

<label-list > ::= <LABEL>
                       [<label-list>]

The PCLabelUpd message is used to download label along the path of
the LSP for the basic PCECC mode, as well as to update the label map
for the Node and Adjacency Label in case of SR.

The SRP object is defined in [I-D.ietf-pce-stateful-pce] and this
document extends the use of SRP object in PCLabelUpd message. The
SRP object is mandatory and MUST be included in PCLabelUpd message.
If the SRP object is missing, the receiving PCC MUST send a PCErr
message with Error-type=6 (Mandatory Object missing) and Error-
value=10 (SRP object missing).

The LSP object is defined in [I-D.ietf-pce-stateful-pce] and this
document extends the use of LSP object in PCLabelUpd message. The
LSP is an optional object and used in the basic PCECC mode in
PCLabelUpd message. LSP Identifiers TLV is defined in
[I-D.ietf-pce-stateful-pce], it MUST be included in the LSP object in
PCLabelUpd message. If the TLV is missing, the PCC will generate a
PCErr message with Error-Type=6 (mandatory object missing) and Error-
Value=11 (LSP-IDENTIFIERS TLV missing) and close the session.

The LABEL object is defined in Section 7.4. The LABEL is the
mandatory object and MUST be included in PCLabelUpd message. If the
LABEL object is missing, the receiving PCC MUST send a PCErr message
with Error-type=6 (Mandatory Object missing) and Error-value=[TBD]
(LABEL object missing). More than one LABEL object MAY be included
in the PCLabelUpd message for the transit LSR.

The FEC object is defined in Section 7.5. The FEC is an optional
object and used in PCECC SR mode in PCLabelUpd message. The FEC
object encodes the Node and Adjacency information of the Label Map.

To cleanup the SRP object must set the R (remove) bit.

7. PCEP Objects

   The PCEP objects defined in this document are compliant with the PCEP
   object format defined in [RFC5440]. The P flag and the I flag of the
   PCEP objects defined in this document MUST always be set to 0 on
   transmission and MUST be ignored on receipt since these flags are
   exclusively related to path computation requests.

7.1. OPEN Object

   This document defines a new optional TLV for use in the OPEN Object.

7.1.1. PCECC Capability TLV

   The PCECC-CAPABILITY TLV is an optional TLV for use in the OPEN
   Object for PCECC capability advertisement. Advertisement of the
   PCECC capability implies support of LSPs that are setup through PCECC
   as per PCEP extensions defined in this document.

   Its format is shown in the following figure:

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |            Type=[TBD]          |            Length=4           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Flags                          |G|L|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
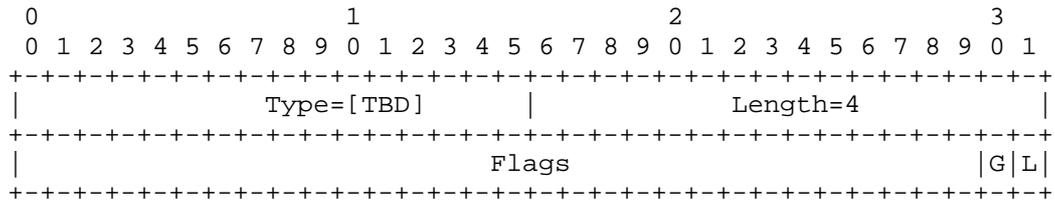
   The type of the TLV is [TBD] and it has a fixed length of 4 octets.

   The value comprises a single field - Flags (32 bits):

   L (LOCAL-LABEL-RANGE-CAPABILITY - 1 bit): If set to 1 by a PCEP
      speaker, it indicates that the PCEP speaker is capable for local
      label range reservation.

   G (GLOBAL-LABEL-RANGE-CAPABILITY - 1 bit): If set to 1 by a PCEP
      speaker, it indicates that the PCEP speaker capable for global
      label range reservation.

   Unassigned bits are considered reserved. They MUST be set to 0 on
   transmission and MUST be ignored on receipt.

7.2. LABEL-RANGE Object

   The LABEL-RANGE object MUST be carried within PCLRResv message. The
   LABEL-RANGE object is used to carry the label range information based
   on the label type.

   LABEL-RANGE Object-Class is TBD.

   LABEL-RANGE Object-Type is 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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | label type   |                 range size                    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                          label  base                         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    //                      Optional TLVs                         //
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
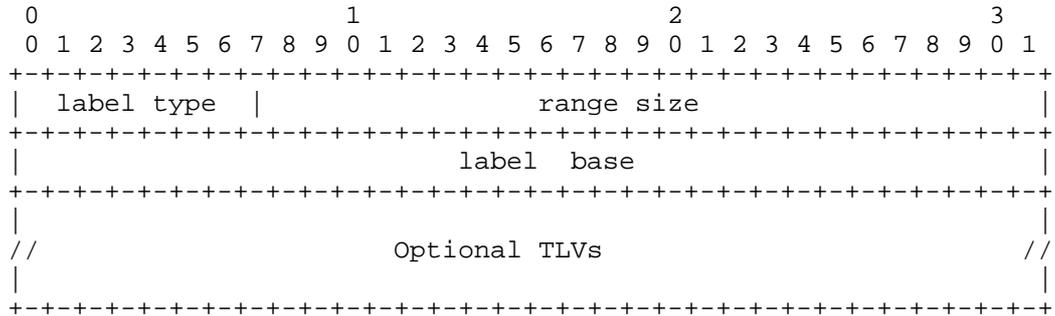
   label type (8 bit): The values defined for label type are label type
      1 specifies the local label. It means the label range is non
      negotiable. label type 2 specifies the global label. It means
      the label range is negotiable. Refer
      [I-D.li-mpls-global-label-framework] for global label.

   Range size (24 bit): It specifies the size of label range.

   Label base (32 bit): It specifies the minimum label of label range.

7.3. PATH-SETUP-TYPE TLV

   The PATH-SETUP-TYPE TLV is defined in
   [I-D.sivabalan-pce-lsp-setup-type]; this document defines following
   new PST value:

   o PST = 2: Path is setup via Basic PCECC mode.

   o PST = 3: Path is setup via PCECC SR-TE mode.

   On a PCRpt or PCInitiate message, the PST=2 in PATH-SETUP-TYPE TLV in
   SRP object indicates that this LSP was setup via a basic PCECC based
   mechanism; the PST=3 indicates that this LSP was setup via a PCECC
   SR-TE based mechanism.

7.4. Label Object

   The LABEL Object is used to specify the Label information and MUST be
   carried within PCLabelUpd message.

   LABEL Object-Class is TBD.

   LABEL Object-Type is 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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |             Reserved          |              Flags          |O|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                             Label                             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 //                       Optional TLV                          //
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
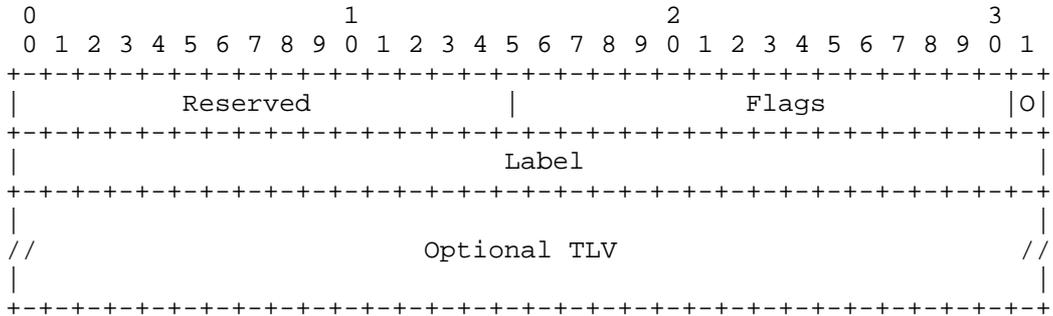
The fields in the LABEL object are as follows:

Flags: is used to carry any additional information pertaining to the
   label. Currently, the following flag bit is defined:

   * O bit(Out-label) : if the bit is set it specifies the label is
      the OUT label and it is mandatory to encode the nexthop
      information (via NEXTHOP-IPV4-ADDRESS TLV or NEXTHOP-
      IPV6-ADDRESS TLV or NEXTHOP-UNNUMBERED-IPV4-ID TLV in LABEL
      object).

Label (32-bit): The Label information encoded such that the 20
   rightmost bits represent a label.

7.4.1. NextHop TLV

   This document defines the following TLV for the LABEL object to
   associate the nexthop information incase of an outgoing label.

NEXTHOP-IPV4-ADDRESS TLV:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type=TBD           |          Length = 8           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     nexthop IPv4 address                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

NEXTHOP-IPV6-ADDRESS TLV:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type=TBD           |          Length = 20          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//              nexthop IPv6 address (16 bytes)               //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

NEXTHOP-UNNUMBERED-IPV4-ID TLV:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type=TBD           |          Length = 12          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Node-ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Interface ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
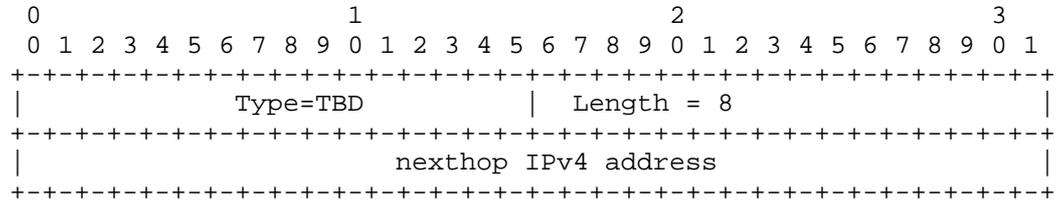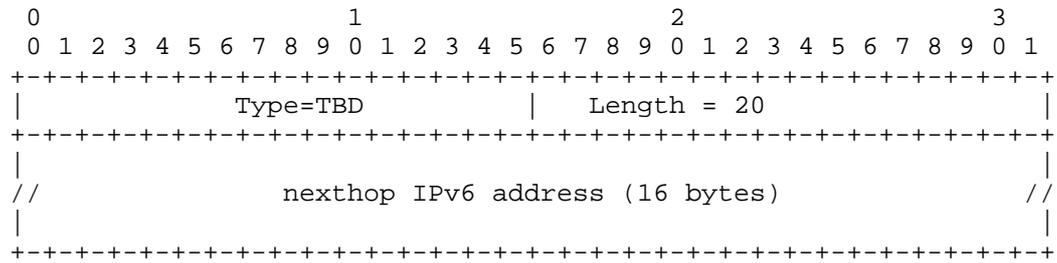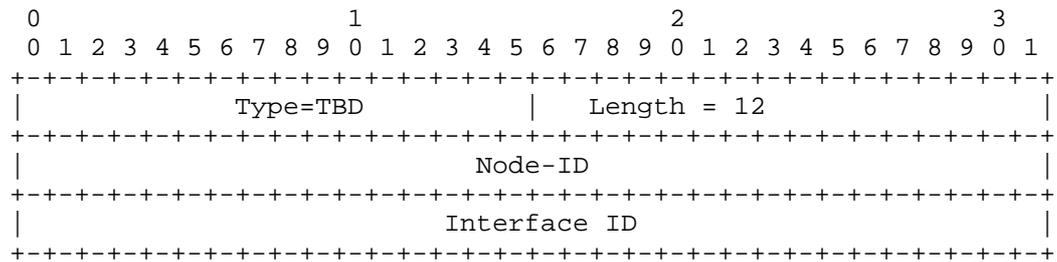
The NextHop TLVs are as follows:

NEXTHOP-IPV4-ADDRESS TLV: where Nexthop IPv4 address is specified as
   an IPv4 address of the Nexthop.

NEXTHOP-IPV6-ADDRESS TLV: where Nexthop IPv6 address is specified as
   an IPv6 address of the Nexthop.

NEXTHOP-UNNUMBERED-IPV4-ID TLV: where a pair of Node ID / Interface
   ID tuples is used for the Nexthop.

7.5. FEC Object

   The FEC Object is used to specify the FEC information and MAY be
   carried within PCLabelUpd message.

   FEC Object-Class is TBD.

   FEC Object-Type is 1 'IPv4 Node ID'.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          IPv4 Node ID                         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
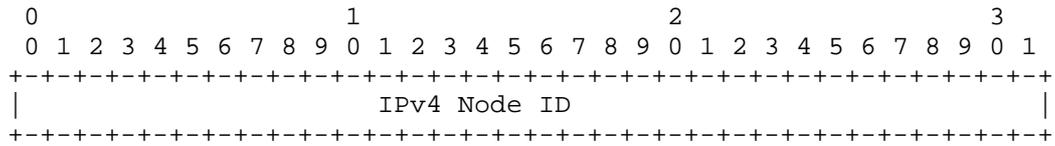
   FEC Object-Type is 2 'IPv6 Node ID'.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   //                    IPv6 Node ID (16 bytes)                  //
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
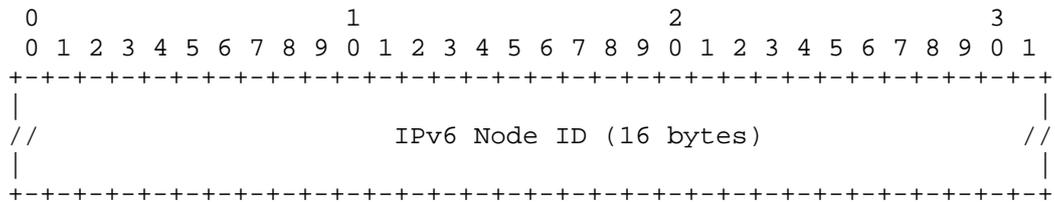
   FEC Object-Type is 3 'IPv4 Adjacency'.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Local IPv4 address                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Remote IPv4 address                     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
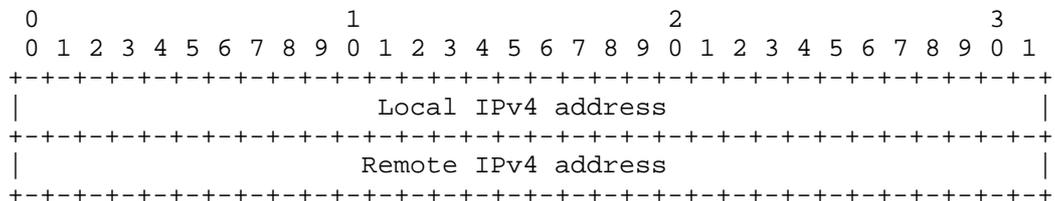
   FEC Object-Type is 4 'IPv6 Adjacency'.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   //                 Local IPv6 address (16 bytes)               //
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   //                 Remote IPv6 address (16 bytes)              //
```

```
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   FEC Object-Type is 5 'Unnumbered Adjacency with IPv4 NodeIDs'.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                         Local Node-ID                         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                       Local Interface ID                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                        Remote Node-ID                         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      Remote Interface ID                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   The FEC objects are as follows:

   IPv4 Node ID: where IPv4 Node ID is specified as an IPv4 address of
      the Node. FEC Object-type is 1, and the Object-Length is 4 in
      this case.

   IPv6 Node ID: where IPv6 Node ID is specified as an IPv6 address of
      the Node. FEC Object-type is 2, and the Object-Length is 16 in
      this case.

   IPv4 Adjacency: where Local and Remote IPv4 address is specified as
      pair of IPv4 address of the adjacency. FEC Object-type is 3, and
      the Object-Length is 8 in this case.

   IPv6 Adjacency: where Local and Remote IPv6 address is specified as
      pair of IPv6 address of the adjacency. FEC Object-type is 4, and
      the Object-Length is 32 in this case.

   Unnumbered Adjacency with IPv4 NodeID: where a pair of Node ID /
      Interface ID tuples is used. FEC Object-type is 5, and the
      Object-Length is 16 in this case.

8. Security Considerations

   TBD

9. Manageability Considerations

9.1. Control of Function and Policy

   TBD.

9.2. Information and Data Models

   TBD.

9.3. Liveness Detection and Monitoring

   TBD.

9.4. Verify Correct Operations

   TBD.

9.5. Requirements On Other Protocols

   TBD.

9.6. Impact On Network Operations

   TBD.

10. IANA Considerations

   TBD

11. Acknowledgments

   We would like to thank Robert Tao, Changjing Yan, Tieying Huang for
   their useful comments and suggestions.

12. References

12.1. Normative References

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

   [RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element
              (PCE) Communication Protocol (PCEP)", RFC 5440, March
              2009.

   [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE", draft-ietf-pce-stateful-
              pce-10 (work in progress), October 2014.

   [I-D.ietf-pce-pce-initiated-lsp]
              Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
              Extensions for PCE-initiated LSP Setup in a Stateful PCE
              Model", draft-ietf-pce-pce-initiated-lsp-02 (work in
              progress), October 2014.

12.2. Informative References

   [I-D.li-mpls-global-label-framework]
              Li, Z., Zhao, Q., Chen, X., Yang, T., and R. Raszuk, "A
              Framework of MPLS Global Label", draft-li-mpls-global-
              label-framework-02 (work in progress), July 2014.

   [I-D.zhao-pce-central-controller-user-cases]
              Zhao, Q., Zhao, K., and Z. Ke, "The Use Cases for Using
              PCE as the Central Controller(PCECC) of LSPs", draft-zhao-
              pce-central-controller-user-cases-01 (work in progress),
              July 2014.

   [I-D.ietf-pce-remote-initiated-gmpls-lsp]
              Ali, Z., Sivabalan, S., Filsfils, C., Varga, R., Lopez,
              V., Dios, O., and X. Zhang, "Path Computation Element
              Communication Protocol (PCEP) Extensions for remote-
              initiated GMPLS LSP Setup", draft-ietf-pce-remote-
              initiated-gmpls-lsp-00 (work in progress), March 2014.

   [I-D.ietf-spring-segment-routing]
              Filsfils, C., Previdi, S., Bashandy, A., Decraene, B.,
              Litkowski, S., Horneffer, M., Shakir, R., Tantsura, J.,
              and E. Crabbe, "Segment Routing Architecture", draft-ietf-
              spring-segment-routing-01 (work in progress), February
              2015.

   [I-D.ietf-isis-segment-routing-extensions]
              Previdi, S., Filsfils, C., Bashandy, A., Gredler, H.,
              Litkowski, S., Decraene, B., and J. Tantsura, "IS-IS
              Extensions for Segment Routing", draft-ietf-isis-segment-
              routing-extensions-03 (work in progress), October 2014.

   [I-D.ietf-ospf-segment-routing-extensions]
              Psenak, P., Previdi, S., Filsfils, C., Gredler, H.,
              Shakir, R., Henderickx, W., and J. Tantsura, "OSPF
              Extensions for Segment Routing", draft-ietf-ospf-segment-
              routing-extensions-04 (work in progress), February 2015.

   [I-D.ietf-pce-segment-routing]
              Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E.,
              Raszuk, R., Lopez, V., and J. Tantsura, "PCEP Extensions
              for Segment Routing", draft-ietf-pce-segment-routing-00
              (work in progress), October 2014.

   [I-D.sivabalan-pce-lsp-setup-type]
              Sivabalan, S., Medved, J., Minei, I., Crabbe, E., and R.
              Varga, "Conveying path setup type in PCEP messages",
              draft-sivabalan-pce-lsp-setup-type-02 (work in progress),
              June 2014.

Authors' Addresses

   Quintin Zhao
   Huawei Technologies
   125 Nagog Technology Park
   Acton, MA 01719
   USA

   EMail: quintin.zhao@huawei.com

   Katherine Zhao
   Huawei Technologies
   2330 Central Expressway
   Santa Clara, CA 95050
   USA

   EMail: katherine.zhao@huawei.com

   Zhenbin Li
   Huawei Technologies
   Huawei Bld., No.156 Beiqing Rd.
   Beijing   100095
   China

   EMail: lizhenbin@huawei.com

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560037
India

EMail: dhruv.ietf@gmail.com

Udayasree Palle
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560037
India

EMail: udayasree.palle@huawei.com

Boris Zhang
Telus Ltd.
Toronto
Canada

EMail: boris.zhang@telus.com

          PCEP Procedures and Protocol Extensions for Using PCE as a Central
                        Controller (PCECC) of LSPs
             draft-zhao-pce-pcep-extension-for-pce-controller-08

Abstract

   The Path Computation Element (PCE) is a core component of Software-
   Defined Networking (SDN) systems. It can compute optimal paths for
   traffic across a network and can also update the paths to reflect
   changes in the network or traffic demands.

   PCE was developed to derive paths for MPLS Label Switched Paths
   (LSPs), which are supplied to the head end of the LSP using the Path
   Computation Element Communication Protocol (PCEP). But SDN has a
   broader applicability than signaled (G)MPLS traffic-engineered (TE)
   networks, and the PCE may be used to determine paths in a range of
   use cases. PCEP has been proposed as a control protocol for use in
   these environments to allow the PCE to be fully enabled as a central
   controller.

   A PCE-based central controller (PCECC) can simplify the processing of
   a distributed control plane by blending it with elements of SDN and
   without necessarily completely replacing it. Thus, the LSP can be
   calculated/setup/initiated and the label forwarding entries can also
   be downloaded through a centralized PCE server to each network
   devices along the path while leveraging the existing PCE technologies
   as much as possible.

   This document specifies the procedures and PCEP protocol extensions
   for using the PCE as the central controller.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   The Path Computation Element (PCE) [RFC4655] was developed to offload
   path computation function from routers in an MPLS traffic-engineered
   network. Since then, the role and function of the PCE has grown to
   cover a number of other uses (such as GMPLS [RFC7025]) and to allow
   delegated control [RFC8231] and PCE-initiated use of network
   resources [RFC8281].

   According to [RFC7399], Software-Defined Networking (SDN) refers to a
   separation between the control elements and the forwarding components
   so that software running in a centralized system, called a
   controller, can act to program the devices in the network to behave
   in specific ways. A required element in an SDN architecture is a
   component that plans how the network resources will be used and how
   the devices will be programmed. It is possible to view this

component as performing specific computations to place traffic flows
within the network given knowledge of the availability of network
resources, how other forwarding devices are programmed, and the way
that other flows are routed. This is the function and purpose of a
PCE, and the way that a PCE integrates into a wider network control
system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths
for MPLS Label Switched Paths (LSPs), paths were requested by network
elements (known as Path Computation Clients (PCCs)), and the results
of the path computations were supplied to network elements using the
Path Computation Element Communication Protocol (PCEP) [RFC5440].
This protocol was later extended to allow a PCE to send unsolicited
requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller
as an extension of the architecture described in [RFC4655] and
assumes the continued use of PCEP as the protocol used between PCE
and PCC. [RFC8283] further examines the motivations and
applicability for PCEP as a Southbound Interface (SBI), and
introduces the implications for the protocol.
[I-D.ietf-teas-pcecc-use-cases] describes the use cases for the PCECC
architecture.

A PCE-based central controller (PCECC) can simplify the processing of
a distributed control plane by blending it with elements of SDN and
without necessarily completely replacing it. Thus, the LSP can be
calculated/setup/initiated and the label forwarding entries can also
be downloaded through a centralized PCE server to each network
devices along the path while leveraging the existing PCE technologies
as much as possible.

This draft specify the procedures and PCEP protocol extensions for
using the PCE as the central controller for static LSPs, where LSPs
can be provisioned as explicit label instructions at each hop on the
end-to-end path. Each router along the path must be told what label-
forwarding instructions to program and what resources to reserve.
The PCE-based controller keeps a view of the network and determines
the paths of the end-to-end LSPs, and the controller uses PCEP to
communicate with each router along the path of the end-to-end LSP.

The extension for PCECC in Segment Routing (SR) is specified in a
separate draft [I-D.zhao-pce-pcep-extension-pce-controller-sr].

1.1. Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
   "OPTIONAL" in this document are to be interpreted as described in BCP
   14 [RFC2119] [RFC8174] when, and only when, they appear in all
   capitals, as shown here.

2. Terminology

   Terminologies used in this document is same as described in the draft
   [RFC8283] and [I-D.ietf-teas-pcecc-use-cases].

3. Basic PCECC Mode

   In this mode LSPs are provisioned as explicit label instructions at
   each hop on the end-to-end path. Each router along the path must be
   told what label forwarding instructions to program and what resources
   to reserve. The controller uses PCEP to communicate with each router
   along the path of the end-to-end LSP.

   Note that the PCE-based controller will take responsibility for
   managing some part of the MPLS label space for each of the routers
   that it controls, and may taker wider responsibility for partitioning
   the label space for each router and allocating different parts for
   different uses. This is also described in section 3.1.2. of
   [RFC8283]. For the purpose of this document, it is assumed that
   label range to be used by a PCE is known and set on both PCEP peers.
   A future extension could add this capability to advertise the range
   via possible PCEP extensions as well. The rest of processing is
   similar to the existing stateful PCE mechanism.

4. PCEP Requirements

   Following key requirements associated PCECC should be considered when
   designing the PCECC based solution:

   1. PCEP speaker supporting this draft MUST have the capability to
       advertise its PCECC capability to its peers.

   2. PCEP speaker not supporting this draft MUST be able to reject
       PCECC related extensions with a error reason code that indicates
       that this feature is not supported.

   3. PCEP speaker MUST provide a means to identify PCECC based LSP in
       the PCEP messages.

4. PCEP procedures SHOULD provide a means to update (or cleanup) the
    label- download entry to the PCC.

5. PCEP procedures SHOULD provide a means to synchronize the labels
    between PCE to PCC in PCEP messages.

5. Procedures for Using the PCE as the Central Controller (PCECC)

5.1. Stateful PCE Model

   Active stateful PCE is described in [RFC8231]. PCE as a central
   controller (PCECC) reuses existing Active stateful PCE mechanism as
   much as possible to control the LSP.

5.2. New LSP Functions

   This document defines the following new PCEP messages and extends the
   existing messages to support PCECC:

   (PCRpt): a PCEP message described in [RFC8231]. PCRpt message is
      used to send PCECC LSP Reports. It is also extended to report the
      set of Central Controller's Instructions (CCI) (label forwarding
      instructions in the context of this document) received from the
      PCE. See Section 5.4.6 for more details.

   (PCInitiate): a PCEP message described in [RFC8281]. PCInitiate
      message is used to setup PCE-Initiated LSP based on PCECC
      mechanism. It is also extended for Central Controller's
      Instructions (CCI) (download or cleanup the Label forwarding
      instructions in the context of this document) on all nodes along
      the path.

   (PCUpd): a PCEP message described in [RFC8231]. PCUpd message is
      used to send PCECC LSP Update.

   The new LSP functions defined in this document are mapped onto the
   messages as shown in the following table.

```
+-----------------------------------------+------------------------+
| Function                                | Message                |
+-----------------------------------------+------------------------+
| PCECC Capability advertisement          | Open                   |
| Label entry Add                         | PCInitiate             |
| Label entry Cleanup                     | PCInitiate             |
| PCECC Initiated LSP                     | PCInitiate             |
| PCECC LSP Update                        | PCUpd                  |
| PCECC LSP State Report                  | PCRpt                  |
| PCECC LSP Delegation                    | PCRpt                  |
| PCECC Label Report                      | PCRpt                  |
+-----------------------------------------+------------------------+
```

This document specify a new object CCI (see Section 7.3) for the
encoding of central controller's instructions. In the scope of this
document this is limited to Label forwarding instructions. The CC-ID
is the unique identifier for the central controller's instructions in
PCEP. The PCEP messages are extended in this document to handle the
PCECC operations.

5.3. PCECC Capability Advertisement

   During PCEP Initialization Phase, PCEP Speakers (PCE or PCC)
   advertise their support of PCECC extensions.

   This document defines a new Path Setup Type (PST)
   [I-D.ietf-pce-lsp-setup-type] for PCECC, as follows:

   o PST = TBD: Path is setup via PCECC mode.

   A PCEP speaker MUST indicate its support of the function described in
   this document by sending a PATH-SETUP-TYPE-CAPABILITY TLV in the OPEN
   object with this new PST included in the PST list.

   This document also defines the PCECC Capability sub-TLV
   Section 7.1.1. PCEP speakers use this sub-TLV to exchange
   information about their PCECC capability. If a PCEP speaker includes
   PST=TBD in the PST List of the PATH-SETUP-TYPE-CAPABILITY TLV then it
   MUST also include the PCECC Capability sub-TLV inside the PATH-SETUP-
   TYPE-CAPABILITY TLV.

   The presence of the PST and PCECC Capability sub-TLV in PCC's OPEN
   Object indicates that the PCC is willing to function as a PCECC
   client.

The presence of the PST and PCECC Capability sub-TLV in PCE's OPEN message indicates that the PCE is interested in function as a PCECC server.

The PCEP protocol extensions for PCECC MUST NOT be used if one or both PCEP Speakers have not included the PST or the PCECC Capability sub-TLV in their respective OPEN message. If the PCEP Speakers support the extensions of this draft but did not advertise this capability then a PCErr message with Error-Type=19(Invalid Operation) and Error-Value=TBD (Attempted PCECC operations when PCECC capability was not advertised) will be generated and the PCEP session will be terminated.

A PCC or a PCE MUST include both PCECC-CAPABILITY sub-TLV and STATEFUL-PCE-CAPABILITY TLV ([RFC8231]) (with I flag set [RFC8281]) in OPEN Object to support the extensions defined in this document. If PCECC-CAPABILITY sub-TLV is advertised and STATEFUL-PCE-CAPABILITY TLV is not advertised in OPEN Object, it SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD (stateful PCE capability was not advertised) and terminate the session.

5.4. LSP Operations

The PCEP messages pertaining to PCECC MUST include PATH-SETUP-TYPE TLV [I-D.ietf-pce-lsp-setup-type] in the SRP object to clearly identify the PCECC LSP is intended.

5.4.1. Basic PCECC LSP Setup

In order to setup a LSP based on PCECC mechanism, a PCC MUST delegate the LSP by sending a PCRpt message with PST set for PCECC (see Section 7.2) and D (Delegate) flag (see [RFC8231]) set in the LSP object.

LSP-IDENTIFIER TLV MUST be included for PCECC LSP, the tuple uniquely identifies the LSP in the network. The LSP object is included in central controller's instructions (label download) to identify the PCECC LSP for this instruction. The PLSP-ID is the original identifier used by the ingress PCC, so the transit LSR could have multiple central controller instructions that have the same PLSP-ID. The PLSP-ID in combination with the source (in LSP-IDENTIFIER TLV) MUST be unique. The PLSP-ID is included for maintainability reasons. As per [RFC8281], the LSP object could include SPEAKER-ENTITY-ID TLV to identify the PCE that initiated these instructions. Also the CC-ID is unique on the PCEP session as described in Section 7.3.

When a PCE receives PCRpt message with D flags and PST Type set, it calculates the path and assigns labels along the path; and set up the

path by sending PCInitiate message to each node along the path of the
LSP. The PCC generates a Path Computation State Report (PCRpt) and
include the central controller's instruction (CCI) and the identified
LSP. The CC-ID is uniquely identify the central controller's
instruction within PCEP. The PCC further responds with the PCRpt
messages including the CCI and LSP objects.

Once the central controller's instructions (label operations) are
completed, the PCE SHOULD send the PCUpd message to the Ingress PCC.
The PCUpd message is as per [RFC8231] SHOULD include the path
information as calculated by the PCE.

Note that the PCECC LSPs MUST be delegated to a PCE at all times.

LSP deletion operation for PCECC LSP is same as defined in [RFC8231].
If the PCE receives PCRpt message for LSP deletion then it does Label
cleanup operation as described in Section 5.4.2.2 for the
corresponding LSP.

The Basic PCECC LSP setup sequence is as shown below.

```
                  +-------+                        +-------+
                  |PCC    |                        | PCE   |
                  |Ingress|                        +-------+
          +------|       |                         |
          | PCC  +-------+                          |
          | Transit| |                             |
    +------|       | |-- PCRpt,PLSP-ID=1, PST=TBD, D=1---->| PCECC LSP
    |PCC    +-------+ |                             |
    |Egress |  | |                                  |
    +-------+  | |                                  |
       |       | |                                  |
       |<------ PCInitiate,CC-ID=X,PLSP-ID=1  ------------ | Label
       |       | | |                                | download
       |------- PCRpt,CC-ID=X,PLSP-ID=1  ----------------->|
       |       | |                                  |
       |       |<----- PCInitiate,CC-ID=Y,PLSP-ID=1  ----- | Label
       |       | |                                  | download
       |       |----- PCRpt,CC-ID=Y,PLSP-ID=1  ---------->|
       |       | |                                  |
       |       | |<--- PCInitiate,CC-ID=Z,PLSP-ID=1  - | Label
       |       | |                                  | download
       |       | |---- PCRpt,CC-ID=Z,PLSP-ID=1  ------>|
       |       | |                                  |
       |       | |<-- PCUpd,PLSP-ID=1,PST=TBD, D=1-----| PCECC LSP
       |       | |                                  | Update
       |       | |                                  |
```
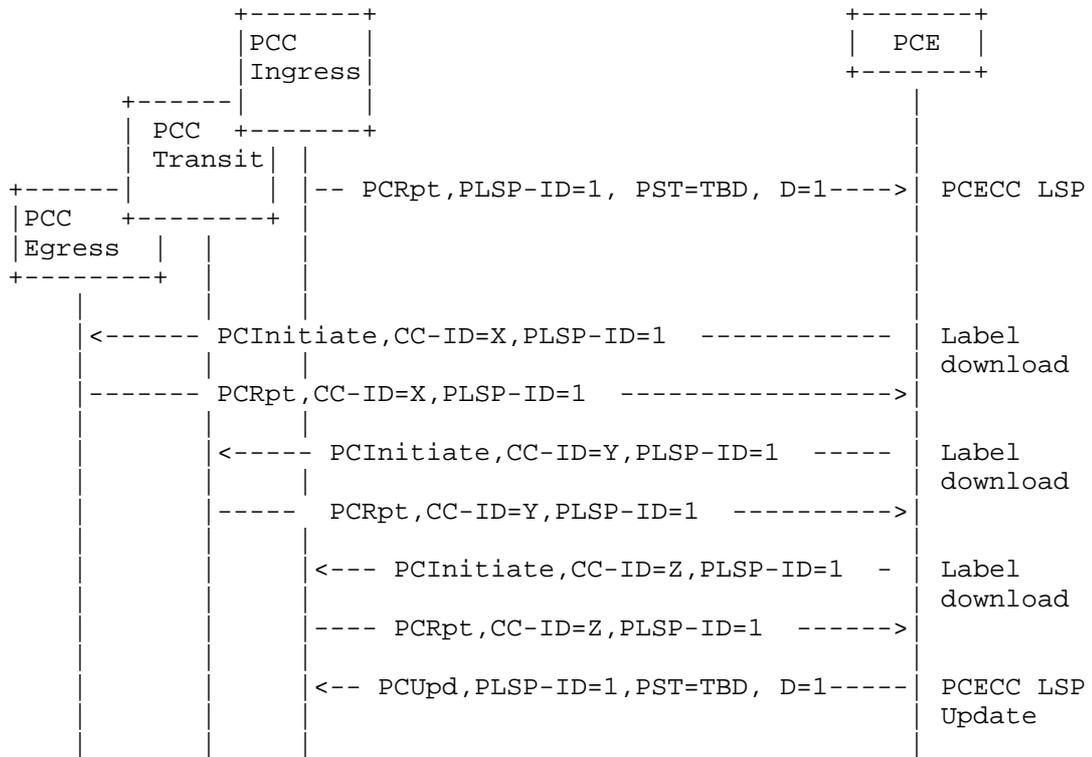
                   Figure 2: Basic PCECC LSP setup

   The PCECC LSP are considered to be 'up' by default (on receipt of
   PCUpd message from PCE). The Ingress MAY further choose to deploy a
   data plane check mechanism and report the status back to the PCE via
   PCRpt message.

5.4.2. Central Control Instructions

   The new central controller's instructions (CCI) for the label
   operations in PCEP is done via the PCInitiate message, by defining a
   new PCEP Objects for CCI operations. Local label range of each PCC
   is assumed to be known at both the PCC and the PCE.

5.4.2.1. Label Download

   In order to setup an LSP based on PCECC, the PCE sends a PCInitiate
   message to each node along the path to download the Label instruction
   as described in Section 5.4.1.

The CCI object MUST be included, along with the LSP object in the
PCInitiate message. The LSP-IDENTIFIER TLV MUST be included in LSP
object. The SPEAKER-ENTITY-ID TLV SHOULD be included in LSP object.

If a node (PCC) receives a PCInitiate message which includes a Label
to download as part of CCI, that is out of the range set aside for
the PCE, it MUST send a PCErr message with Error-type=TBD (PCECC
failure) and Error-value=TBD (Label out of range) and MUST include
the SRP object to specify the error is for the corresponding label
update via PCInitiate message. If a PCC receives a PCInitiate
message but failed to download the Label entry, it MUST send a PCErr
message with Error-type=TBD (PCECC failure) and Error-value=TBD
(instruction failed) and MUST include the SRP object to specify the
error is for the corresponding label update via PCInitiate message.

New PCEP object for central control instructions (CCI) is defined in
Section 7.3.

5.4.2.2. Label Cleanup

In order to delete an LSP based on PCECC, the PCE sends a central
controller instructions via a PCInitiate message to each node along
the path of the LSP to cleanup the Label forwarding instruction.

If the PCC receives a PCInitiate message but does not recognize the
label in the CCI, the PCC MUST generate a PCErr message with Error-
Type 19(Invalid operation) and Error-Value=TBD, "Unknown Label" and
MUST include the SRP object to specify the error is for the
corresponding label cleanup (via PCInitiate message).

The R flag in the SRP object defined in [RFC8281] specifies the
deletion of Label Entry in the PCInitiate message.

```
                    +-------+                        +-------+
                    |PCC    |                        | PCE   |
                    |Ingress|                        +-------+
            +------|       |                             |
            | PCC  +-------+                             |
            | Transit| |                                |
    +-------|       | |-- PCRpt,PLSP-ID=1,PST=TBD,D=1,R=1--->| PCECC LSP
    |PCC    +--------+ |                                |  remove
    |Egress |  |     |                                 |
    +--------+  |     |                                 |
         |      |     |                                 |
         |<------ PCInitiate,CC-ID=X,PLSP-ID=1 ------------  | Label
         |      |     |                       R=1        | cleanup
         |------- PCRpt,CC-ID=X,PLSP-ID=1  ----------------->|
         |      |     |                                 |
         |      |<----- PCInitiate,CC-ID=Y,PLSP-ID=1  ------ | Label
         |      |     |                       R=1        | cleanup
         |      |----- PCRpt,CC-ID=Y,PLSP-ID=1  ----------->|
         |      |     |                                 |
         |      |     |<--- PCInitiate,CC-ID=Z,PLSP-ID=1  -- | Label
         |      |     |                       R=1        | cleanup
         |      |     |---- PCRpt,CC-ID=Z,PLSP-ID=1  ------->|
         |      |     |                                 |
```
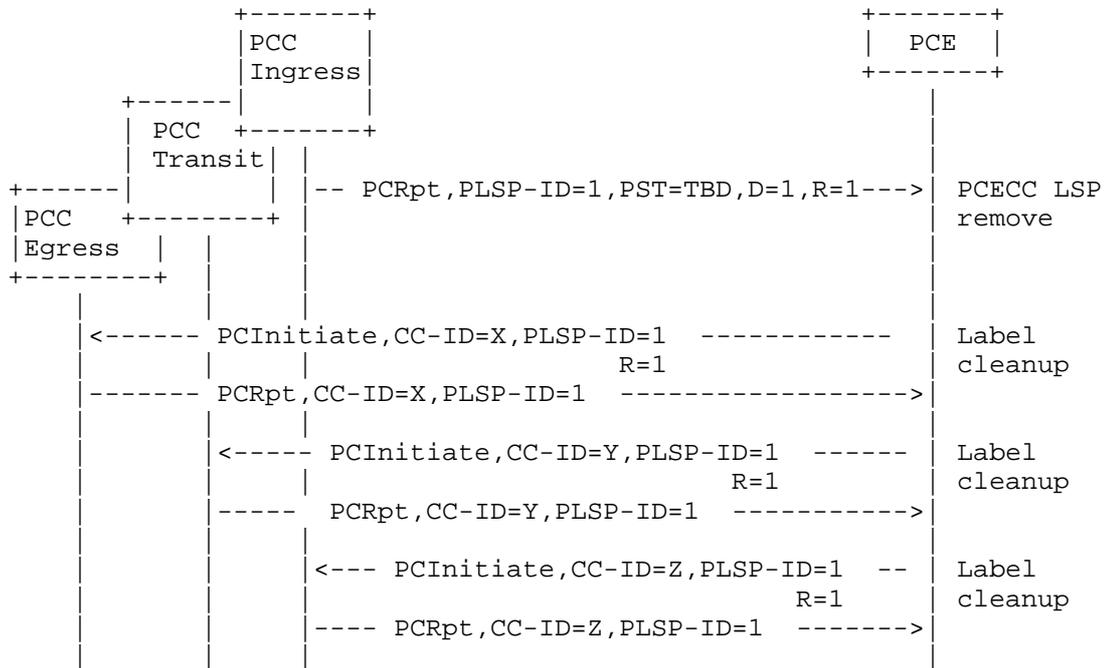
As per [RFC8281], following the removal of the Label forwarding
instruction, the PCC MUST send a PCRpt message. The SRP object in
the PCRpt MUST include the SRP-ID-number from the PCInitiate message
that triggered the removal. The R flag in the SRP object MUST be
set.

5.4.3. PCE Initiated PCECC LSP

The LSP Instantiation operation is same as defined in [RFC8281].

In order to setup a PCE Initiated LSP based on the PCECC mechanism, a
PCE sends PCInitiate message with Path Setup Type set for PCECC (see
Section 7.2) to the Ingress PCC.

The Ingress PCC MUST also set D (Delegate) flag (see [RFC8231]) and C
(Create) flag (see [RFC8281]) in LSP object of PCRpt message. The
PCC responds with first PCRpt message with the status as "GOING-UP"
and assigned PLSP-ID.

Note that the label forwarding instructions from PCECC are send after
the initial PCInitiate and PCRpt exchange. This is done so that the
PLSP-ID and other LSP identifiers can be obtained from the ingress
and can be included in the label forwarding instruction in the next

PCInitiate message. The rest of the PCECC LSP setup operations are
same as those described in Section 5.4.1.

The LSP deletion operation for PCE Initiated PCECC LSP is same as
defined in [RFC8281]. The PCE should further perform Label entry
cleanup operation as described in Section 5.4.2.2 for the
corresponding LSP.

The PCE Initiated PCECC LSP setup sequence is shown below -

```
                    +-------+                        +-------+
                    |PCC    |                        | PCE   |
                    |Ingress|                        +-------+
             +------|       |                        |       |
             | PCC  +-------+                        |       |
             | Transit|    |                         |       |
  +------|   Transit|   | |<--PCInitiate,PLSP-ID=0,PST=TBD,D=1---|  PCECC LSP
  |PCC   +--------+ |   |                            |  Initiate
  |Egress |  |   |   |--- PCRpt,PLSP-ID=2,P=1,D=1,C=1--->  |  PCECC LSP
  +--------+ |   |   |          (GOING-UP)           |
    |        |   |   |                               |
    |<------ PCInitiate,CC-ID=X,PLSP-ID=2 -------------- |  Label
    |        |   |   |                               |  download
    |------- PCRpt,CC-ID=X,PLSP-ID=2  ----------------->|
    |        |   |   |                               |
    |        |   |<----- PCInitiate,CC-ID=Y,PLSP-ID=2 ------- |  Label
    |        |   |   |                               |  download
    |        |   |----- PCRpt,CC-ID=Y,PLSP-ID=2  ----------->|
    |        |   |   |                               |
    |        |   |   |<--- PCInitiate,CC-ID=Z,PLSP-ID=2 --- |  Label
    |        |   |   |                               |  download
    |        |   |   |---- PCRpt,CC-ID=Z,PLSP-ID=2  ------->|
    |        |   |   |                               |
    |        |   |   |<-- PCUpd, PLSP-ID=2, PST=TBD, D=1--- |  PCECC LSP
    |        |   |   |       (UP)                    |  Update
    |        |   |   |--- PCRpt,PLSP-ID=2,P=1,D=1,C=1--->   |
    |        |   |   |       (UP)                    |
```
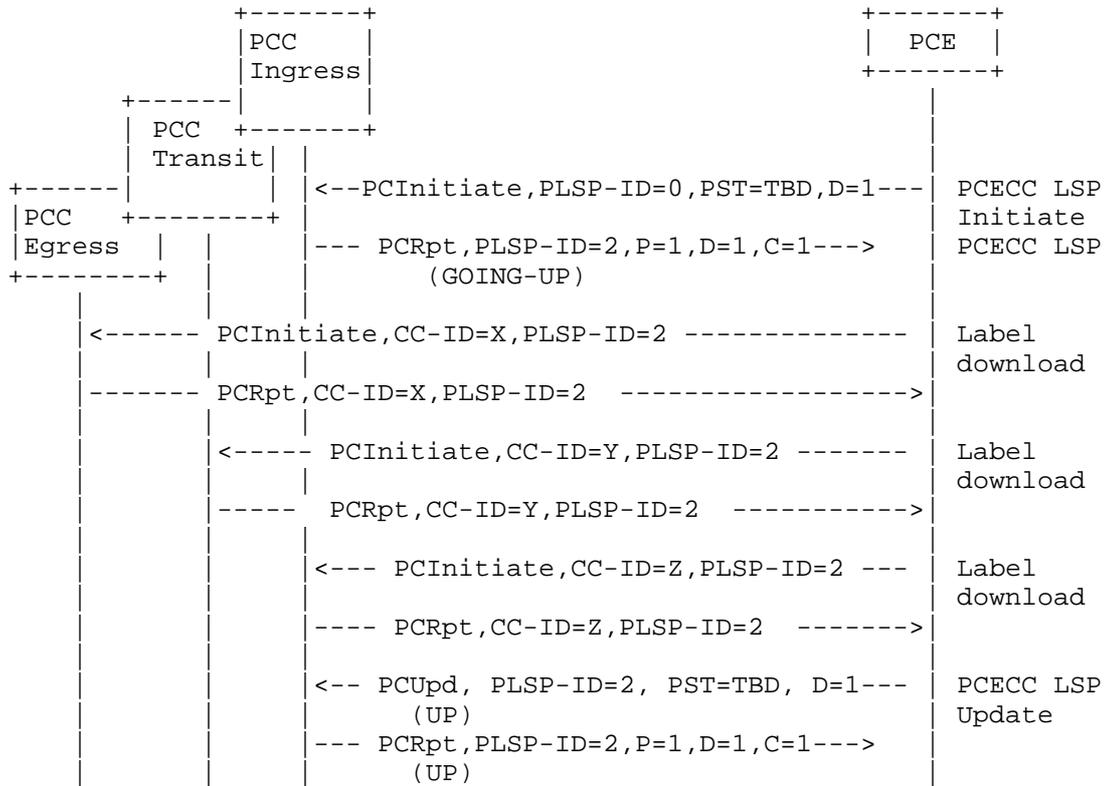
Once the label operations are completed, the PCE SHOULD send the
PCUpd message to the Ingress PCC. The PCUpd message is as per
[RFC8231].

5.4.4. PCECC LSP Update

   In case of a modification of PCECC LSP with a new path, a PCE sends a
   PCUpd message to the Ingress PCC. But to follow the make-before-
   break procedures, the PCECC first update new instructions based on
   the updated LSP and then update to ingress to switch traffic, before
   cleaning up the old instructions. A new CC-ID is used to identify
   the updated instruction, the existing identifiers in the LSP object
   identify the existing LSP. Once new instructions are downloaded, the
   PCE further updates the new path at the ingress which triggers the
   traffic switch on the updated path. The Ingress PCC acknowledges
   with a PCRpt message, on receipt of PCRpt message, the PCE does
   cleanup operation for the old LSP as described in Section 5.4.2.2.

   The PCECC LSP Update sequence is shown below -

```
                    +-------+                      +-------+
                    |PCC    |                      |  PCE  |
                    |Ingress|                      +-------+
            +------|       |                       |
            | PCC  +-------+                        |
            | Transit|     | |                      |
    +------|        |     | |                       |
    |PCC    +-------+ |     |                        |
    |Egress |       | |     |                        |
    +-------+ |     | |     |                        |
      |       |     | |     |                        | New Path for
      |<------ PCInitiate,CC-ID=XX,PLSP-ID=1 ----------- | LSP trigger
      |       |     | |     |                        | new instruct
      |------- PCRpt,CC-ID=XX,PLSP-ID=1  --------------->|
      |       |     | |                               |
      |       |<----- PCInitiate,CC-ID=YY,PLSP-ID=1------ | Label
      |       |     | |                               | download
      |       |----- PCRpt,CC-ID=YY,PLSP-ID=1  --------->|
      |       |     | |                               |
      |       |     |<--- PCInitiate,CC-ID=ZZ,PLSP-ID=1 - | Label
      |       |     | |                               | download
      |       |     |---- PCRpt,CC-ID=ZZ,PLSP-ID=1  ----->|
      |       |     | |                               |
      |       |     |<-- PCUpd, PLSP-ID=1, PST=TBD, D=1-- | PCECC
      |       |     |      SRP=S                       | LSP Update
      |       |     | |                               |
      |       |     |-- PCRpt,PLSP-ID=1,PST=TBD,D=1    -->| Trigger
      |       |     |         (SRP=S)                  | Delete old
      |       |     | |                               | instruct
      |       |     | |                               |
      |<------ PCInitiate,CC-ID=X, PLSP-ID=1 ----------- | Label
      |       |     | |                    R=1          | cleanup
      |------- PCRpt,CC-ID=X, PLSP-ID=1  --------------->|
      |       |     | |                               |
      |       |<----- PCInitiate,CC-ID=Y, PLSP-ID=1 ----- | Label
      |       |     | |                    R=1          | cleanup
      |       |----- PCRpt,CC-ID=Y, PLSP-ID=1  --------->|
      |       |     | |                               |
      |       |     |<--- PCInitiate,CC-ID=Z, PLSP-ID=1 - | Label
      |       |     | |                    R=1          | cleanup
      |       |     |---- PCRpt,CC-ID=Z, PLSP-ID=1  ----->|
      |       |     | |                               |
```

   The modified PCECC LSP are considered to be 'up' by default. The
   Ingress MAY further choose to deploy a data plane check mechanism and
   report the status back to the PCE via PCRpt message.

5.4.5. Re Delegation and Cleanup

   As described in [RFC8281], a new PCE can gain control over the
   orphaned LSP. In case of PCECC LSP, the new PCE MUST also gain
   control over the central controllers instructions in the same way by
   sending a PCInitiate message that includes the SRP, LSP and CCI
   objects and carries the CC-ID and PLSP-ID identifying the
   instruction, it wants to take control of.

   Further, as described in [RFC8281], the State Timeout Interval timer
   ensures that a PCE crash does not result in automatic and immediate
   disruption for the services using PCE-initiated LSPs. Similarly the
   central controller instructions are not removed immediately upon PCE
   failure. Instead, they are cleaned up on the expiration of this
   timer. This allows for network cleanup without manual intervention.
   The PCC MUST support removal of CCI as one of the behaviors applied
   on expiration of the State Timeout Interval timer.

5.4.6. Synchronization of Central Controllers Instructions

   The purpose of Central Controllers Instructions synchronization
   (labels in the context of this document) is to make sure that the
   PCE's view of CCI (Labels) matches with the PCC's Label allocation.
   This synchronization is performed as part of the LSP state
   synchronization as described in [RFC8231] and [RFC8233].

   As per LSP State Synchronization [RFC8231], a PCC reports the state
   of its LSPs to the PCE using PCRpt messages and as per [RFC8281], PCE
   would initiate any missing LSPs and/or remove any LSPs that are not
   wanted. The same PCEP messages and procedure is also used for the
   Central Controllers Instructions synchronization. The PCRpt message
   includes the CCI and the LSP object to report the label forwarding
   instructions. The PCE would further remove any unwanted instructions
   or initiate any missing instructions.

5.4.7. PCECC LSP State Report

   As mentioned before, an Ingress PCC MAY choose to apply any OAM
   mechanism to check the status of LSP in the Data plane and MAY
   further send its status in PCRpt message to the PCE.

6. PCEP messages

   As defined in [RFC5440], a PCEP message consists of a common header
   followed by a variable-length body made of a set of objects that can
   be either mandatory or optional. An object is said to be mandatory
   in a PCEP message when the object must be included for the message to
   be considered valid. For each PCEP message type, a set of rules is

defined that specify the set of objects that the message can carry.
An implementation MUST form the PCEP messages using the object
ordering specified in this document.

LSP-IDENTIFIERS TLV MUST be included in the LSP object for PCECC LSP.

6.1. The PCInitiate message

The PCInitiate message [RFC8281] can be used to download or remove
the labels, the message has been extended as shown below -


```
    <PCInitiate Message> ::= <Common Header>
                            <PCE-initiated-lsp-list>
Where:
    <Common Header> is defined in [RFC5440]

    <PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                                [<PCE-initiated-lsp-list>]

    <PCE-initiated-lsp-request> ::=
                        (<PCE-initiated-lsp-instantiation>|
                         <PCE-initiated-lsp-deletion>|
                         <PCE-initiated-lsp-central-control>)

    <PCE-initiated-lsp-central-control> ::= <SRP>
                                            <LSP>
                                            <cci-list>

    <cci-list> ::=  <CCI>
                    [<cci-list>]

Where:
    <PCE-initiated-lsp-instantiation> and
    <PCE-initiated-lsp-deletion> are as per
     [RFC8281].

    The LSP and SRP object is defined in [RFC8231].
```


When PCInitiate message is used for central controller's instructions
(labels), the SRP, LSP and CCI objects MUST be present. The SRP
object is defined in [RFC8231] and if the SRP object is missing, the
receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory
Object missing) and Error-value=10 (SRP object missing). The LSP
object is defined in [RFC8231] and if the LSP object is missing, the
receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory
Object missing) and Error-value=8 (LSP object missing). The CCI

object is defined in Section 7.3 and if the CCI object is missing,
the receiving PCC MUST send a PCErr message with Error-type=6
(Mandatory Object missing) and Error-value=TBD (CCI object missing).
More than one CCI object MAY be included in the PCInitiate message
for the transit LSR.

To cleanup the SRP object must set the R (remove) bit.

At max two instances of CCI object would be included in case of
transit LSR to encode both in-coming and out-going label forwarding
instructions. Other instances MUST be ignored.

6.2. The PCRpt message

The PCRpt message can be used to report the labels that were
allocated by the PCE, to be used during the state synchronization
phase.

```
    <PCRpt Message> ::= <Common Header>
                          <state-report-list>
Where:

    <state-report-list> ::= <state-report>[<state-report-list>]

    <state-report> ::= (<lsp-state-report>|
                        <central-control-report>)

    <lsp-state-report> ::= [<SRP>]
                            <LSP>
                            <path>

    <central-control-report> ::= [<SRP>]
                                  <LSP>
                                  <cci-list>

    <cci-list> ::=  <CCI>
                    [<cci-list>]

  Where:
    <path> is as per [RFC8231] and the LSP and SRP object are
    also defined in [RFC8231].
```

When PCRpt message is used to report the central controller's
instructions (labels), the LSP and CCI objects MUST be present. The
LSP object is defined in [RFC8231] and if the LSP object is missing,
the receiving PCE MUST send a PCErr message with Error-type=6
(Mandatory Object missing) and Error-value=8 (LSP object missing).

The CCI object is defined in Section 7.3 and if the CCI object is
missing, the receiving PCC MUST send a PCErr message with Error-
type=6 (Mandatory Object missing) and Error-value=TBD (CCI object
missing). Two CCI object can be included in the PCRpt message for
the transit LSR.

7. PCEP Objects

   The PCEP objects defined in this document are compliant with the PCEP
   object format defined in [RFC5440].

7.1. OPEN Object

   This document defines a new optional TLVs for use in the OPEN Object.

7.1.1. PCECC Capability sub-TLV

   The PCECC-CAPABILITY sub-TLV is an optional TLV for use in the OPEN
   Object for PCECC capability advertisement in PATH-SETUP-TYPE-
   CAPABILITY TLV. Advertisement of the PCECC capability implies
   support of LSPs that are setup through PCECC as per PCEP extensions
   defined in this document.

   Its format is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=TBD          |           Length=4            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   The type of the TLV is TBD and it has a fixed length of 4 octets.

   The value comprises a single field - Flags (32 bits).

   No flags are assigned right now.

   Unassigned bits are considered reserved. They MUST be set to 0 on
   transmission and MUST be ignored on receipt.

7.2. PATH-SETUP-TYPE TLV

   The PATH-SETUP-TYPE TLV is defined in [I-D.ietf-pce-lsp-setup-type];
   this document defines a new PST value:

   o PST = TBD: Path is setup via PCECC mode.

   On a PCRpt/PCUpd/PCInitiate message, the PST=TBD in PATH-SETUP-TYPE
   TLV in SRP object indicates that this LSP was setup via a PCECC based
   mechanism.

7.3. CCI Object

   The Central Control Instructions (CCI) Object is used by the PCE to
   specify the forwarding instructions (Label information in the context
   of this document) to the PCC, and MAY be carried within PCInitiate or
   PCRpt message for label download.

   CCI Object-Class is TBD.

   CCI Object-Type is 1 for the MPLS Label.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           CC-ID                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           Reserved            |           Flags             |O|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |            Label              |           Reserved           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                              |
   //                     Optional TLV                           //
   |                                                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
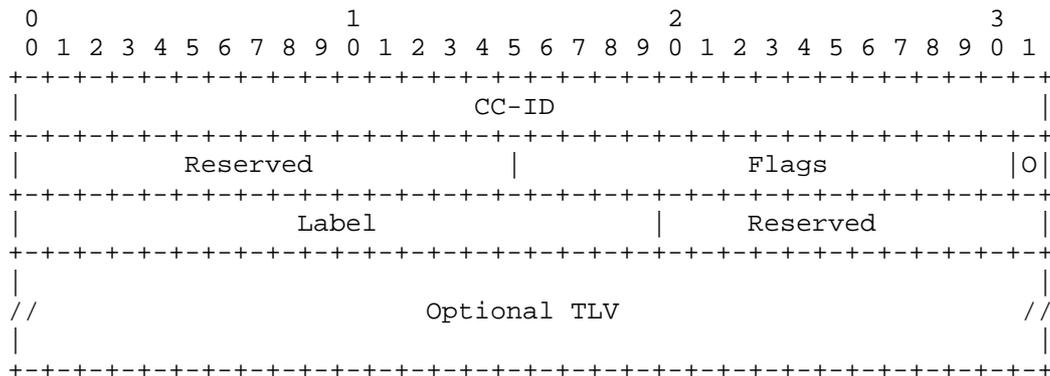
   The fields in the CCI object are as follows:

   CC-ID: A PCEP-specific identifier for the CCI information. A PCE
      creates an CC-ID for each instruction, the value is unique within
      the scope of the PCE and is constant for the lifetime of a PCEP
      session. The values 0 and 0xFFFFFFFF are reserved and MUST NOT be
      used.

   Flags: is used to carry any additional information pertaining to the
      CCI. Currently, the following flag bit is defined:

   * O bit(Out-label) : If the bit is set, it specifies the label is
      the OUT label and it is mandatory to encode the next-hop
      information (via IPV4-ADDRESS TLV or IPV6-ADDRESS TLV or
      UNNUMBERED-IPV4-ID-ADDRESS TLV in the CCI object). If the bit
      is not set, it specifies the label is the IN label and it is
      optional to encode the local interface information (via
      IPV4-ADDRESS TLV or IPV6-ADDRESS TLV or UNNUMBERED-IPV4-ID-
      ADDRESS TLV in the CCI object).

   Label (20-bit): The Label information.

   Reserved (12 bit): Set to zero while sending, ignored on receive.

7.3.1. Address TLVs

   This document defines the following TLVs for the CCI object to
   associate the next-hop information in case of an outgoing label and
   local interface information in case of an incoming label.

IPV4-ADDRESS TLV:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=TBD          |      Length = 4               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         IPv4 address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IPV6-ADDRESS TLV:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=TBD          |      Length = 16              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//               IPv6 address (16 bytes)                       //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

UNNUMBERED-IPV4-ID-ADDRESS TLV:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=TBD          |      Length = 8               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Node-ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Interface ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
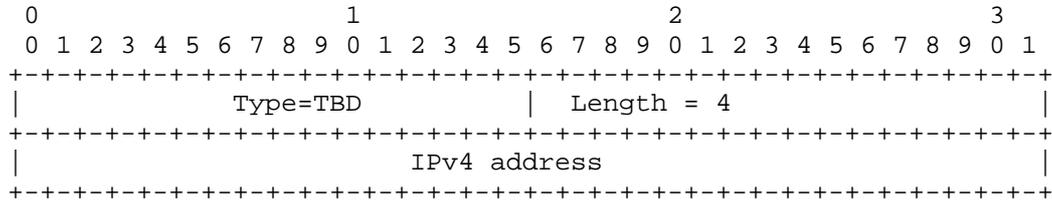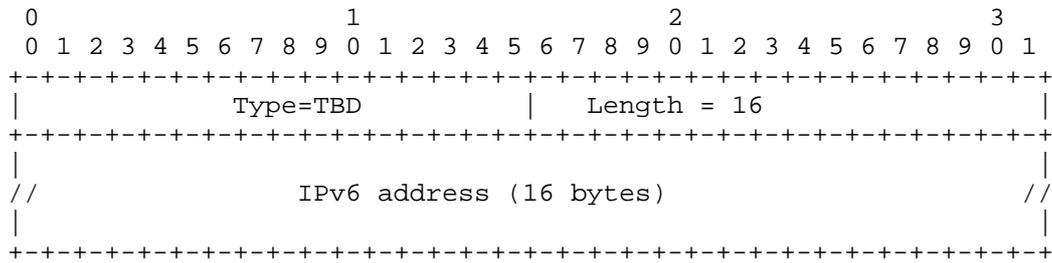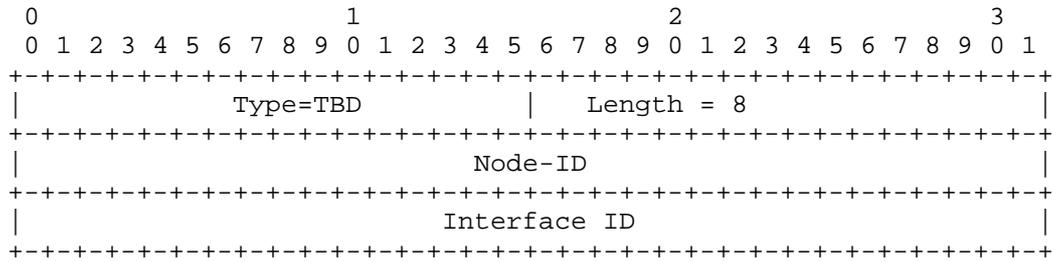
The address TLVs are as follows:

IPV4-ADDRESS TLV: an IPv4 address.

IPV6-ADDRESS TLV: an IPv6 address.

UNNUMBERED-IPV4-ID-ADDRESS TLV: a pair of Node ID / Interface ID
   tuples.

8. Security Considerations

   The security considerations described in [RFC8231] and [RFC8281]
   apply to the extensions described in this document. Additional
   considerations related to a malicious PCE are introduced.

8.1. Malicious PCE

   PCE has complete control over PCC to update the labels and can cause
   the LSP's to behave inappropriate and cause cause major impact to the
   network. As a general precaution, it is RECOMMENDED that these PCEP
   extensions only be activated on authenticated and encrypted sessions
   across PCEs and PCCs belonging to the same administrative authority,
   using Transport Layer Security (TLS) [RFC8253], as per the
   recommendations and best current practices in [RFC7525].

9. Manageability Considerations

9.1. Control of Function and Policy

   A PCE or PCC implementation SHOULD allow to configure to enable/
   disable PCECC capability as a global configuration.

9.2. Information and Data Models

   [RFC7420] describes the PCEP MIB, this MIB can be extended to get the
   PCECC capability status.

   The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to
   enable/disable PCECC capability.

9.3. Liveness Detection and Monitoring

   Mechanisms defined in this document do not imply any new liveness
   detection and monitoring requirements in addition to those already
   listed in [RFC5440].

9.4. Verify Correct Operations

   Mechanisms defined in this document do not imply any new operation
   verification requirements in addition to those already listed in
   [RFC5440] and [RFC8231].

9.5. Requirements On Other Protocols

   PCEP extensions defined in this document do not put new requirements
   on other protocols.

9.6. Impact On Network Operations

   PCEP extensions defined in this document do not put new requirements
   on network operations.

10. IANA Considerations

10.1. PCEP TLV Type Indicators

   IANA is requested to confirm the early allocation of the following
   TLV Type Indicator values within the "PCEP TLV Type Indicators" sub-
   registry of the PCEP Numbers registry, and to update the reference in
   the registry to point to this document, when it is an RFC:

Value	Meaning	Reference
TBD	PCECC-CAPABILITY	This document
TBD	IPV4-ADDRESS TLV	This document
TBD	IPV6-ADDRESS TLV	This document
TBD	UNNUMBERED-IPV4-ID-ADDRESS TLV	This document

10.2. New Path Setup Type Registry

   IANA is requested to allocate new PST Field in PATH- SETUP-TYPE TLV.
   The allocation policy for this new registry should be by IETF
   Consensus. The new registry should contain the following value:

Value	Description	Reference
TBD	Traffic engineering path is setup using PCECC mode	This document

10.3. PCEP Object

   IANA is requested to allocate new registry for CCI PCEP object.

Object-Class Value	Name	Reference
TBD	CCI Object-Type	This document
	1	MPLS Label

10.4. CCI Object Flag Field

   IANA is requested to create a registry to manage the Flag field of
   the CCI object.

   One bit to be defined for the CCI Object flag field in this document:

   Codespace of the Flag field (CCI Object)

```
        Bit              Description           Reference
        7                Specifies label       This document
                         is out label
```

10.5. PCEP-Error Object

   IANA is requested to allocate new error types and error values within
   the "PCEP-ERROR Object Error Types and Values" sub-registry of the
   PCEP Numbers registry for the following errors:

```
   Error-Type   Meaning
   ----------   -------
   19           Invalid operation.

                  Error-value = TBD :              Attempted PCECC
                                                   operations when
                                                   PCECC capability
                                                   was not advertised
                  Error-value = TBD :              Stateful PCE
                                                   capability was not
                                                   advertised
                  Error-value = TBD :              Unknown Label
   6            Mandatory Object missing.

                  Error-value = TBD :              CCI object missing
   TBD          PCECC failure.

                  Error-value = TBD :              Label out of range.
                  Error-value = TBD :              Instruction failed.
```

11. Acknowledgments

   We would like to thank Robert Tao, Changjing Yan, Tieying Huang and
   Avantika for their useful comments and suggestions.

12. References

12.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <https://www.rfc-editor.org/info/rfc5440>.

   [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
              Hardwick, "Path Computation Element Communication Protocol
              (PCEP) Management Information Base (MIB) Module",
              RFC 7420, DOI 10.17487/RFC7420, December 2014,
              <https://www.rfc-editor.org/info/rfc7420>.

   [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre,
              "Recommendations for Secure Use of Transport Layer
              Security (TLS) and Datagram Transport Layer Security
              (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May
              2015, <https://www.rfc-editor.org/info/rfc7525>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for Stateful PCE", RFC 8231,
              DOI 10.17487/RFC8231, September 2017,
              <https://www.rfc-editor.org/info/rfc8231>.

   [RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki,
              "Extensions to the Path Computation Element Communication
              Protocol (PCEP) to Compute Service-Aware Label Switched
              Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September
              2017, <https://www.rfc-editor.org/info/rfc8233>.

   [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path
              Computation Element Communication Protocol (PCEP)
              Extensions for PCE-Initiated LSP Setup in a Stateful PCE
              Model", RFC 8281, DOI 10.17487/RFC8281, December 2017,
              <https://www.rfc-editor.org/info/rfc8281>.

12.2. Informative References

   [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation
              Element (PCE)-Based Architecture", RFC 4655,
              DOI 10.17487/RFC4655, August 2006,
              <https://www.rfc-editor.org/info/rfc4655>.

   [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C.
              Margaria, "Requirements for GMPLS Applications of PCE",
              RFC 7025, DOI 10.17487/RFC7025, September 2013,
              <https://www.rfc-editor.org/info/rfc7025>.

   [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path
              Computation Element Architecture", RFC 7399,
              DOI 10.17487/RFC7399, October 2014,
              <https://www.rfc-editor.org/info/rfc7399>.

   [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for
              Application-Based Network Operations", RFC 7491,
              DOI 10.17487/RFC7491, March 2015,
              <https://www.rfc-editor.org/info/rfc7491>.

   [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody,
              "PCEPS: Usage of TLS to Provide a Secure Transport for the
              Path Computation Element Communication Protocol (PCEP)",
              RFC 8253, DOI 10.17487/RFC8253, October 2017,
              <https://www.rfc-editor.org/info/rfc8253>.

   [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An
              Architecture for Use of PCE and the PCE Communication
              Protocol (PCEP) in a Network with Central Control",
              RFC 8283, DOI 10.17487/RFC8283, December 2017,
              <https://www.rfc-editor.org/info/rfc8283>.

   [I-D.ietf-teas-pcecc-use-cases]
              Zhao, Q., Li, Z., Khasanov, B., Ke, Z., Fang, L., Zhou,
              C., Communications, T., and A. Rachitskiy, "The Use Cases
              for Using PCE as the Central Controller(PCECC) of LSPs",
              draft-ietf-teas-pcecc-use-cases-01 (work in progress), May
              2017.

   [I-D.ietf-pce-lsp-setup-type]
              Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J.
              Hardwick, "Conveying path setup type in PCEP messages",
              draft-ietf-pce-lsp-setup-type-10 (work in progress), May
              2018.

   [I-D.ietf-pce-pcep-yang]
              Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A
              YANG Data Model for Path Computation Element
              Communications Protocol (PCEP)", draft-ietf-pce-pcep-
              yang-07 (work in progress), March 2018.

   [I-D.zhao-pce-pcep-extension-pce-controller-sr]
              Zhao, Q., Li, Z., Dhody, D., Karunanithi, S., Farrel, A.,
              and C. Zhou, "PCEP Procedures and Protocol Extensions for
              Using PCE as a Central Controller (PCECC) of SR-LSPs",
              draft-zhao-pce-pcep-extension-pce-controller-sr-02 (work
              in progress), March 2018.

Appendix A. Contributor Addresses

   Udayasree Palle
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   EMail: udayasreereddy@gmail.com

   Mahendra Singh Negi
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   EMail: mahendrasingh@huawei.com

   Katherine Zhao
   Huawei Technologies
   2330 Central Expressway
   Santa Clara, CA 95050
   USA

   EMail: katherine.zhao@huawei.com

   Boris Zhang
   Telus Ltd.
   Toronto
   Canada

   EMail: boris.zhang@telus.com

Authors' Addresses

   Quintin Zhao
   Huawei Technologies
   125 Nagog Technology Park
   Acton, MA 01719
   USA

   EMail: quintin.zhao@huawei.com

   Zhenbin Li
   Huawei Technologies
   Huawei Bld., No.156 Beiqing Rd.
   Beijing   100095
   China

   EMail: lizhenbin@huawei.com

   Dhruv Dhody
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   EMail: dhruv.ietf@gmail.com

   Satish Karunanithi
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   EMail: satishk@huawei.com

   Adrian Farrel
   Juniper Networks, Inc
   UK

   EMail: adrian@olddog.co.uk

   Chao Zhou
   Cisco Systems

   EMail: choa.zhou@cisco.com

              PCEP Extensions for LSP scheduling with stateful PCE
                draft-zhuang-pce-stateful-pce-lsp-scheduling-00

Abstract

   This document proposes a set of extensions needed to the stateful
   Path Computation Element (PCE) communication Protocol (PCEP), so as
   to enable Labeled Switched Path (LSP) scheduling for path computation
   and LSP setup/deletion based on the actual network resource usage
   duration of a traffic service in a centralized network environment.
   A scheduled LSP can be setup at the its starting time and deleted
   after its usage duration such that LSPs for the other traffic
   services can take over these network resources beyond that period.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   The Path Computation Element Protocol (PCEP) defined in [RFC5440] is
   used between a Path Computation Element (PCE) and a Path Computation
   Client (PCC) (or other PCE) to enable computation of Multi-protocol
   Label Switching (MPLS) for Traffic Engineering Label Switched Path
   (TE LSP).

   Further, in order to support use cases described in [I-D.ietf-pce-
   stateful-pce-app], [I-D.ietf-pce-stateful-pce] specifies a set of
   extensions to PCEP to enable stateful control of MPLS-TE and GMPLS
   LSPs via PCEP.

   Traditionally, the network resources, especially bandwidth, usage and
   allocation can be supported by a Network Management System operation
   such as path pre-establishment. However, this does not provide
   efficient network usage since the established paths exclude the
   possibility of being used by other services even when they are not
   used for undertaking any service.

   With LSP scheduling, it allows network operators to reserve resources
   in advance according to the agreements with their customers, and
   allow them to transmit data with specified starting time and
   duration, for example for a scheduled bulk data replication between
   data centers. It enables the activation of bandwidth usage at the
   time the service really being used while letting other services
   obtain it in spare time. The requirement of scheduled LSP provision
   is mentioned in [I-D.ietf-pce-stateful-pce-app] and [RFC7399], so as
   to provide more efficient network resource usage for traffic
   engineering, which hasn't been solved yet.

   This document proposes a set of extensions needed to the stateful
   PCE, so as to enable LSP scheduling for path computation and LSP
   setup/deletion based on the actual network resource usage duration of
   a traffic service. A scheduled LSP can be setup at the its starting
   time and deleted after its usage duration such that LSPs for the
   other traffic services can take over these network resources beyond
   that period.

2. 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 [RFC2119].

2.1. Terminology

   The following terminologies are used in this document:

   Active Stateful PCE: PCE that uses LSP state information learned
      from PCCs to optimize path computations. Additionally, it
      actively updates delegated LSP states in those PCCs that delegated
      the control.

   Delegation: An operation to grant a PCE temporary rights to modify a
      subset of LSP parameters on one or more PCC's LSPs. LSPs are
      delegated from a PCC to a PCE.

   PCC: Path Computation Client: any client application requesting a
      path computation to be performed by a Path Computation Element.

   PCE: Path Computation Element. An entity (component, application,
      or network node) that is capable of computing a network path or
      route based on a network graph and applying computational
      constraints.

   TE LSP: Traffic Engineering Label Switched Path.

   Scheduled TE LSP: a LSP that carries traffic flow demand at an
      activation time and last for a certain duration. The PCE operates
      path computation per LSP availability at the required time and
      duration.

3. Motivation and Objectives

   A stateful PCE can support better efficiency by using LSP scheduling
   described in the use case of [I-D.ietf-pce-stateful-pce]. This
   requires the PCE to maintain the scheduled LSPs and their associated
   resource usage, e.g. bandwidth, as well as the ability for PCC to
   trigger signaling for the LSP setup/tear-down at the correct time.

   Note that existing configuration tools can be used for LSP
   scheduling, but as highlighted in section 3.1.3 of [I-D.ietf-pce-
   stateful-pce], doing this as a part of PCEP, has obvious advantages.

   The objective of this document is to provide a set of extensions to
   PCEP to enable LSP scheduling for LSPs creation/deletion under the
   stateful PCE control, according to traffic services from customers,
   so as to improve the usage of network resources.

4. Architecture Overview

4.1. LSP scheduling Overview

   The LSP scheduling allows PCEs and PCCs to provide scheduled LSP for
   customers' traffic services at its actual usage time, so as to
   improve the network resource efficient utilization.

   For stateful PCE supporting LSP scheduling, there are two types of
   LSP databases used in this document. One is the LSP-DB defined in
   PCEP [I-D.ietf-pce-stateful-pce], while the other is the scheduled
   LSP database (SLSP- DB, see section 6). The SLSP-DB records
   scheduled LSPs and is used as a complementary to the TED and LSP-DB
   to show the network resource availability for path computation. Note
   that the two types of LSP databases can be implemented in one
   physical database or two different databases. This document does not
   state any preference here.

   In case of implementing PCC-initiated scheduled LSPs, at the time of
   delegation,a PCC can send a path computation LSP State Report message
   (PCRpt message) with LSP information of its starting time and the
   duration. Upon receiving the PCRpt message with the scheduled LSP
   delegation, a stateful PCE SHALL not only check the current network
   resource availability recorded in the Traffic Engineering Database
   (TED) and LSP-DB, but also consider scheduled resource reservation
   for scheduled LSPs in the SLSP-DB and then the stateful PCE sends the
   path for the scheduled LSP in an LSP Update Request carried in a
   PCUpd message to the PCC. Note that PCE can either calculate the
   path for scheduled LSP based on current information and update it
   later at any time based on network events or PCE MAY chooses to
   calculate the path closer to the activation time.

   In case of implementing PCE-Initiated Scheduling LSP, the stateful
   PCE can send a path computation LSP Initiate (PCInitiate message)
   with LSP information at its starting time and duration to reserve a
   path. In addition, the stateful PCE may send PCUpd message at the
   time of activation to activate the path.

   In case of PCE Initiated LSP, it is recommended that PCE send
   PCInitiate at creation time so that these scheduled LSP is known at
   PCC and it can be further syncronized to other PCE as well. . At any
   time, stateful PCE may change the attribute of a scheduled LSP by
   sending the PCUpd message.

   Based on PCUpd or PCInitiate message, a PCC creates a LSP with
   scheduled LSP information. This scheduled LSP MUST be added into the
   SLSP-DB and synchronized among PCEs and PCC via PCRpt message. For a

scheduled LSP, a PCC MUST not trigger signaling for LSP setup at its
creation time but wait until its starting time.

For setup/activation of scheduled LSPs, PCC MAY activate the LSP at
the starting time or PCE MAY control the activation, with active
stateful PCE notifies the PCC that the status of the scheduled LSP
has been changed and it SHOULD trigger signaling for the LSP setup.
When the requested usage duration expires, PCC or active stateful PCE
removes the LSP from the data base.

The following terms are used in this document:

o Scheduled LSP: A LSP with the scheduling attributes, that is
   activated in future and last for a duration. The PCE operates
   path computation per resource availability at the required time
   and duration.

o Starting time: This value indicates when the scheduled LSP is used
   and the corresponding LSP must be setup and active. In other
   time, the LSP can be inactive to include the possibility of the
   resources being used by other services.

o Duration: the value indicates the time duration that the LSP is
   undertaken by a traffic flow and the corresponding LSP must be
   setup and active. At the end of which, the LSP is teardown and
   removed from the data base.

4.2. Support of LSP Scheduling

4.2.1. The Open Message

   After a TCP connection for a PCEP session has been established, the
   PCE or the PCC can send an Open Message with the B flag in Stateful
   PCE Capability TLV set to 1 described in [section 4.2] to indicate
   that it supports LSP scheduling to its peer. The definition of the
   Open message (see [RFC5440]) is unchanged.

4.2.2. Stateful PCE Capability TLV

   A PCC and a PCE indicates its ability to support LSP scheduling
   during the PCEP session establishment phase. The Open Object in the
   Open message contains the STATEFUL-PCE-CAPABILITY TLV defined in [I-
   D.ietf-pce-stateful-pce]. A new flag is defined for the STATEFUL-
   PCE-CAPABILITY TLV defined in [I-D.ietf-pce-stateful-pce] and updated
   in [I-D.ietf-pce-pce-initiated-lsp] and [I-D.ietf-pce-stateful-sync-
   optimizations].

A new bit B (SCHED-LSP-CAPABLITY) flag is added in this document to
indicate the support of LSP scheduling.

B (LSP-SCHEDULING-CAPABILITY - 1 bit): If set to 1 by a PCC, the B
   Flag indicates that the PCC allows LSP scheduling; if set to 1 by
   a PCE, the B Flag indicates that the PCE is capable of LSP
   scheduling. The B bit MUST be set by both PECP peers in order to
   support LSP scheduling for path computation.

4.3. Scheduled LSP creation

   In order to realize scheduled LSP in a centralized network
   environment, a PCC has to separate the setup of a LSP into two steps.
   The first step is to create a LSP but not signal it over the network.
   The second step is to signal the scheduled LSP over the LSRs (Labeled
   switched Router) at its starting time.

   For PCC Initiated scheduled LSPs, a PCC can send a path computation
   LSP report (PCRpt) message (see section 4.3.1) including its demanded
   resources with the starting time and its usage duration and
   delegation to a stateful PCE.

   Upon receiving the delegation via PCRpt message, the stateful PCE
   computes the path for the scheduled LSP per its starting time and
   duration based on the network resource availability from traffic
   engineering database (TED) (defined in [RFC5440]) and LSP-DB (defined
   in [I-D.ietf-pce-stateful-pce]), as well as scheduled resource
   reservation in the SLSP-DB (see section 6).

   If a resultant path is found, the stateful PCE will send a PCUpd
   message (see section 5.x) with path information back to the PCC as
   defined in [I-D.ietf-pce-stateful-pce].

   For PCE-Initiated Scheduled LSP, the stateful PCE can send a path
   computation LSP Initiate (PCInitiate message) with LSP information at
   its starting time and duration to reserve a path.

   Upon receiving the PCInitiate or PCUpd message for scheduled LSP from
   PCEs, tThe PCC then creates a scheduled LSP including the scheduled
   LSP information for the traffic but not trigger signaling for the LSP
   setup on LSRs.

   Note that PCE can either calculate the path for scheduled LSP based
   on current information and update it later at any time based on
   network events or PCE MAY chooses to calculate the path closer to the
   activation time. In any case, stateful PCE can update the Sheduled
   LSP parameters on any network events using the PCUpd message.

4.3.1. The PCRpt Message

   After scheduled LSP capability negotiation and for PCC Initiated
   scheduled LSPs, PCC can send a PCRpt message including the SCHED-LSP-
   ATTRIBUTE TLV (see section 4.3.3.1) carried in the LSP Object (see
   section 4.3.3) body to indicate the requested LSP scheduling
   parameters for a customer's traffic service with the delegation bit
   set to 1 in LSP Object. The value of requested bandwidth is taken
   via the existing 'Requested Bandwidth with BANDWIDTH Object- Type as
   1' defined in [RFC5440].

   The definition of the PCRpt message to carry LSP objects (see [I-
   D.ietf-pce-stateful-pce]) remains unchanged.

4.3.2. The PCUpd Message

   To provide scheduled LSP for TE-LSPs, the stateful PCE SHALL compute
   the path for the scheduled LSP carried on PCRpt message based on
   network resource availability recorded in TED, LSP-DB and SLSP-DB.

   If the request can be satisfied and an available path is found, the
   stateful PCE SHALL send a PCUpd Message including the SCHED- LSP-
   ATTRIBUTE TLV in the LSP Object body.

   Note that, the stateful PCE can update the Sheduled LSP parameters
   later as well based on any network events using the same PCUpd
   message.

4.3.3. The PCInitiate Message

   To provide scheduled LSP for TE-LSPs, the stateful PCE SHALL compute
   the path for the requesting traffic based on network resource
   availability recorded in TED, LSP-DB and SLSP-DB.

   If the request can be satisfied the stateful PCE SHALL send a
   PCInitiate Message including the SCHED-LSP-ATTRIBUTE TLV in the LSP
   Object body to request PCC to create a scheduled LSP.

   PCE can either calculate the path at initiation and update it later
   at any time based on network events or PCE MAY chooses to calculate
   the path closer to the activation time.

4.3.4. LSP Object

   The LSP object is defined in [I-D.ietf-pce-stateful-pce]. This
   document add an optional SCHED-LSP-ATTRIBUTE TLV.

The presence of SCHED-LSP-ATTRIBUTE TLV in the LSP object indicates that this LSP is requesting scheduled parameters. The TLV MUST be present in LSP Object for each scheduled LSP carried in the PCInitiate message, the PCRpt message and the PCUpd message.

4.3.4.1. SCHED-LSP-ATTRIBUTE TLV

The SCHED-LSP-ATTRIBUTE TLV can be included as an optional TLV within the LSP object for LSP scheduling for the requesting traffic service.

This TLV SHOULD be included only if both PCEP peers have set the B (LSP-SCHEDULING-CAPABILITY bit) in STATEFUL-PCE-CAPABILITY TLV carried in open message.

The format of the SCHED-LSP-ATTRIBUTE TLV is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Type             |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Start Time (minutes)                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Duration (minutes)                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The type of the TLV is [TBD] and it has a fixed length of 8 octets.

The fields in the format are:

Start Time (32 bits): This value in minutes, indicates when the scheduled LSP is used and the corresponding LSP must be setup and activated. At the expiry of this time, the LSP is setup. Otherwise, the LSP is inactive to include the possibility of the resources to be used by other services. The

Duration (32 bits): The value in minutes, indicates the duration that the LSP is undertaken by a traffic flow and the corresponding LSP must be up to carry traffic. At the expiry of this time after setup, the LSP is tear down and deleted.

Note, that the values of start time and duration is from the perspective of the PCEP peer that is sending the message, also note the unit of time is minutes, and thus the time spent on transmission on wire can be easily ignored.

4.4. Scheduled LSP information synchronization

   As for a stateful PCE, it maintains a database of LSPs (LSP-DB) that
   are active in the network, so as to reveal the available network
   resources and place new LSPs more cleverly.

   With the scheduled LSPs, they are not activated while creation, but
   should be considered when operating future path computation. Hence,
   a scheduled LSP Database (SLSP-DB) is suggested to maintain all
   scheduled LSP information.

   The information of SLSP-DB MUST be shared and synchronized among all
   PCEs within the centralized network. In order to synchronize the
   scheduled LSP information in SLSP-DB among PCEs and PCCs, the PCRpt
   Message is used as before.

4.4.1. The PCRpt Message

   It is the responsibility of PCC to report the scheduled LSPs to all
   PCEs via a PCRpt message. The message shall include the SCHED-LSP-
   ATTRIBUTE TLV within the LSP Object.

   Since the scheduled LSP is not signaled over the path yet, the
   mandatory LSP identifiers TLV should be all zero as defined in [I-
   D.ietf-pce-stateful-pce] but with the PLSP-ID for the LSP specified
   in the LSP object.

   Upon receiving the PCRpt message with scheduled LSP information, the
   PCE SHALL update the scheduled LSP information with its PLSP-ID into
   the SLSP-DB for further path computation.

4.5. PCC initiated scheduled LSP

   In case of PCC initiated scheduled LSP, the PCC MAY delegate the
   scheduled LSP to an active stateful PCE via the PCRpt message with
   the D Flag set to 1 as stated in [I-D.ietf-pce-stateful-pce]. The
   scheduled LSP is created but not signaled over the LSRs.

   The stateful PCE MAY send a PCUpd Message including the SCHED- LSP-
   ATTRIBUTE TLV in the LSP Object body with the path now or later
   closer to the setup time.

   Note that, the stateful PCE can update the Sheduled LSP parameters at
   any time based on any network events using the same PCUpd message.

4.6. Scheduled LSP activation and deletion

   The PCC itself MAY activate the scheduled LSP at the starting time
   indicated in the SCHED-LSP-ATTRIBUTE TLV carried on PCUpd message or
   PCInitiate message by signaling the LSP over LSRs. Alternatively,
   the active stateful PCE MAY activate the scheduled LSP immediately by
   using PCUpd message with A flag set (see section 4.5.2) to request
   the PCC to setup/activate the LSP.

   After the scheduled duration expires, the PCC itself MAY delete the
   LSP and release the resources and report the same to the PCEs. Or,
   the active stateful PCE SHALL notify the PCC to delete the LSP and
   release the resources immediately via a PCUpd message with the R Flag
   set to 1 and the A Flag set to zero in the SRP object(see section
   4.5.2). Upon receiving this message, the PCC shall trigger tear down
   to delete the LSP over the network. Moreover, it SHALL notify all
   PCEs of deletion of this LSP via a PCRpt Message.

   Note that, the stateful PCE can update the Sheduled LSP parameters at
   any time based on any network events using the PCUpd message
   including SCHED-LSP-ATTRIBUTE TLV in the LSP Object body.

4.6.1. The PCUpd Message

   PCC can activate and delete the scheduled LSP on its own based on the
   parameters in the SCHED-LSP-ATTRIBUTE TLV, but in some case PCE may
   override it and request PCC to activate or remove the LSP
   immediately.

   When the scheduled LSP needs to be activated, the active stateful PCE
   MAY send a PCUpd message with the A Flag set to 1 in the SRP
   object(see section 8.1) as well as ERO of the path for the LSP. Upon
   receiving this PCUpd message, the PCC MUST trigger signaling to setup
   the LSP over the network nodes immediately.

   When the scheduled LSP needs to be removed, the active stateful PCE
   SHALL request the PCC to delete the LSP and release the resources for
   it via a PCUpd message with the R Flag set to 1 and the A Flag set to
   zero in the SRP object(see section 4.4.2).

4.6.1.1. SRP Object

   The SRP Object is defined in [I-D.ietf-pce-stateful-pce] and a new
   flag is added to indicate activation of a scheduled LSP:

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                            Flags                          |A|R|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                        SRP-ID-number                          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    //                       Optional TLVs                         //
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
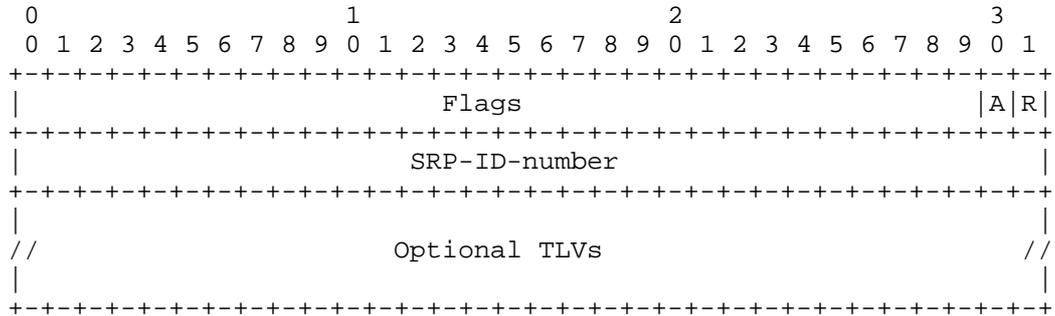
   The R bit in flags field is defined in [I-D. ietf-pce-pce-initiated-
   lsp].

   A (ACTIVATING-LSP - 1 bit): On a PCUpd message , the A Flag set to 1
      indicates that this scheduled LSP SHALL be activated, which means
      it shall be up and ready to carry traffic. The A Flag set to zero
      indicates no operation for this LSP. For non-scheduled LSPs, this
      A flag shall set to zero.

5. Security Considerations

   This document defines LSP-SCHEDULING-CAPABILITY TLV and SCHED- LSP-
   ATTRIBUTE TLV which does not add any new security concerns beyond
   those discussed in [RFC5440] and [I-D.ietf-pce-stateful-pce].

6. Manageability Consideration

6.1. Control of Function and Policy

   The LSP-Scheduling feature MUST BE controlled per tunnel by the
   active stateful PCE, the values for parameters like starting time,
   duration SHOULD BE configurable by customer applications and based on
   the local policy at PCE.

6.2. Information and Data Models

   [RFC7420] describes the PCEP MIB, there are no new MIB Objects for
   this document.

6.3. Liveness Detection and Monitoring

   Mechanisms defined in this document do not imply any new liveness
   detection and monitoring requirements in addition to those already
   listed in [RFC5440].

6.4. Verify Correct Operations

   Mechanisms defined in this document do not imply any new operation
   verification requirements in addition to those already listed in
   [RFC5440].

6.5. Requirements On Other Protocols

   Mechanisms defined in this document do not imply any new requirements
   on other protocols.

6.6. Impact On Network Operations

   Mechanisms defined in this document do not have any impact on network
   operations in addition to those already listed in [RFC5440].

7. IANA Considerations

7.1. PCEP TLV Type Indicators

   This document defines the following new PCEP TLV; IANA is requested
   to make the following allocations from this registry.

   Value   Meaning   Reference
    TBD   SCHED-LSP-ATTRIBUTE   This document

7.2. LSP-SCHEDULING-CAPABLITY

   This document requests that a registry is created to manage the Flags
   field in the STATEFUL-PCE-CAPABILITY TLV in the OPEN object. New
   values are to be assigned by Standards Action [RFC5226]. Each bit
   should be tracked with the following qualities:

   o Bit number (counting from bit 0 as the most significant bit)

   o Capability description

   o Defining RFC

   The following values are defined in this document:

   Bit   Description   Reference
    28   LSP-SCHEDULING-CAPABILITY (B-bit)   This document

7.3. ACTIVATING-LSP

   This document requests that a registry is created to manage the Flags
   field in the SRP object. New values are to be assigned by Standards
   Action [RFC5226]. Each bit should be tracked with the following
   qualities:

   o Bit number (counting from bit 0 as the most significant bit)

   o Capability description

   o Defining RFC

   The following values are defined in this document:

            Bit   Description   Reference
            30   ACTIVATING-LSP   This document

8. References

8.1. Normative References

   [I-D.ietf-pce-pce-initiated-lsp]
            Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
            Extensions for PCE-initiated LSP Setup in a Stateful PCE
            Model", draft-ietf-pce-pce-initiated-lsp-04 (work in
            progress), April 2015.

   [I-D.ietf-pce-stateful-pce]
            Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
            Extensions for Stateful PCE", draft-ietf-pce-stateful-
            pce-11 (work in progress), April 2015.

   [I-D.ietf-pce-stateful-sync-optimizations]
            Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X.,
            and D. Dhody, "Optimizations of Label Switched Path State
            Synchronization Procedures for a Stateful PCE", draft-
            ietf-pce-stateful-sync-optimizations-02 (work in
            progress), January 2015.

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

8.2. Informative References

   [I-D.ietf-pce-stateful-pce-app]
              Zhang, X. and I. Minei, "Applicability of a Stateful Path
              Computation Element (PCE)", draft-ietf-pce-stateful-pce-
              app-04 (work in progress), April 2015.

Appendix A. Appendix A. Contributor Addresses

   Zitao Wang
   Huawei
   101 Software Avenue, Yuhua District
   Nanjing, Jiangsu 210012
   China

   Email: wangzitao@huawei.com

   Xian Zhang
   Huawei Technologies
   Research Area F3-1B,
   Huawei Industrial Base,
   Shenzhen, 518129, China

   Email: zhang.xian@huawei.com

Authors' Addresses

   Yan Zhuang
   Huawei
   101 Software Avenue, Yuhua District
   Nanjing, Jiangsu 210012
   China

   Email: zhuangyan.zhuang@huawei.com

   Qin Wu
   Huawei
   101 Software Avenue, Yuhua District
   Nanjing, Jiangsu 210012
   China

   Email: bill.wu@huawei.com

Dhruv Dhody
Huawei
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560037
India

Email: dhruv.ietf@gmail.com

PCE Working Group   H. Chen, Ed.
Internet-Draft   Y. Zhuang, Ed.
Intended status: Standards Track   Q. Wu
Expires: September 28, 2017   D. Dhody
                                                              Huawei
                                                        D. Ceccarelli
                                                            Ericsson
                                                      March 27, 2017

PCEP Extensions for LSP scheduling with stateful PCE
draft-zhuang-pce-stateful-pce-lsp-scheduling-05

Abstract

   This document proposes a set of extensions needed to the stateful
   Path Computation Element (PCE) communication Protocol (PCEP), so as
   to enable Labeled Switched Path (LSP) scheduling for path computation
   and LSP setup/deletion based on the actual network resource usage
   duration of a traffic service in a centralized network environment as
   stated in [I.D.ietf-teas-scheduled-resources].

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on September 28, 2017.

carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   The Path Computation Element Protocol (PCEP) defined in [RFC5440] is
   used between a Path Computation Element (PCE) and a Path Computation
   Client (PCC) (or other PCE) to enable computation of Multi-protocol
   Label Switching (MPLS) for Traffic Engineering Label Switched Path
   (TE LSP).

   Further, in order to support use cases described in [I-D.ietf-pce-
   stateful-pce-app], [I-D.ietf-pce-stateful-pce] specifies a set of
   extensions to PCEP to enable stateful control of MPLS-TE and GMPLS
   LSPs via PCEP.

   Traditionally, the usage and allocation of network resources,
   especially bandwidth, can be supported by a Network Management System
   operation such as path pre-establishment. However, this does not
   provide efficient network usage since the established paths exclude
   the possibility of being used by other services even when they are
   not used for undertaking any service. [I-D.ietf-teas-scheduled-
   resources] then provides a framework that describes and discusses the
   problem and propose an appropriate architecture for the scheduled
   reservation of TE resources.

   With the scheduled reservation of TE resources, it allows network
   operators to reserve resources in advance according to the agreements
   with their customers, and allow them to transmit data with scheduling
   such as specified starting time and duration, for example for a
   scheduled bulk data replication between data centers. It enables the
   activation of bandwidth usage at the time the service really being
   used while letting other services obtain it in spare time. The
   requirement of scheduled LSP provision is mentioned in [I-D.ietf-pce-
   stateful-pce-app] and [RFC7399], so as to provide more efficient
   network resource usage for traffic engineering, which hasn't been
   solved yet. Also, for deterministic networks, the scheduled LSP can
   provide a better network resource usage for guaranteed links. This
   idea can also be applied in segment routing to schedule the network
   resources over the whole network in a centralized manner as well.

   With this in mind, this document proposes a set of extensions needed
   to the stateful PCE, so as to enable LSP scheduling for path
   computation and LSP setup/deletion based on the actual network
   resource usage duration of a traffic service. A scheduled LSP is
   characterized by a starting time and a duration. When the end of the
   LSP life is reached, it is deleted to free up the resources for other
   LSP (scheduled or not).

2. 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 [RFC2119].

2.1. Terminology

   The following terminologies are re-used from existing PCE documents.

   o Active Stateful PCE [I-D.ietf-pce-stateful-pce];

   o Delegation [I-D.ietf-pce-pce-initiated-lsp];

   o PCC [RFC5440], [I-D.ietf-pce-stateful-pce];

   o PCE [RFC5440], [I-D.ietf-pce-stateful-pce];

   o TE LSP [RFC5440], [I-D.ietf-pce-stateful-pce];

   o TED [RFC5440], [I-D.ietf-pce-stateful-pce];

   o LSP DB [RFC5440], [I-D.ietf-pce-stateful-pce];

   In addition, this document defines the following terminologies.

   Scheduled TE LSP: a LSP with the scheduling attributes,that carries
      traffic flow demand at an starting time and last for a certain
      duration. The PCE operates path computation per LSP availability
      at the required time and duration.

   Scheduled LSP DB: a database of scheduled LSPs

   Scheduled TED: Traffic engineering database with the awareness of
      scheduled resources for TE. This database is generated by the PCE
      from the information in TED and scheduled LSP DB and allows
      knowing, at any time, the amount of available resources (does not
      include failures in the future).

   Starting time(start-time): This value indicates when the scheduled
      LSP is used and the corresponding LSP must be setup and active.
      In other time(i.e., before the starting time or after the starting
      time plus Duration), the LSP can be inactive to include the
      possibility of the resources being used by other services.

   Duration: The value indicates the time duration that the LSP is
      undertaken by a traffic flow and the corresponding LSP must be

setup and active. At the end of which, the LSP is teardown and
removed from the data base.

3. Motivation and Objectives

A stateful PCE can support better efficiency by using LSP scheduling
described in the use case of [I-D.ietf-pce-stateful-pce]. This
requires the PCE to maintain the scheduled LSPs and their associated
resource usage, e.g. bandwidth for Packet-switched network, as well
as the ability to trigger signaling for the LSP setup/tear-down at
the correct time.

Note that existing configuration tools can be used for LSP
scheduling, but as highlighted in section 3.1.3 of [I-D.ietf-pce-
stateful-pce] as well as discussions in [I-D.ietf-teas-scheduled-
resources], doing this as a part of PCEP in a centralized manner, has
obvious advantages.

The objective of this document is to provide a set of extensions to
PCEP to enable LSP scheduling for LSPs creation/deletion under the
stateful PCE control, according to traffic services from customers,
so as to improve the usage of network resources.

4. Architecture Overview

4.1. LSP scheduling Overview

The LSP scheduling allows PCEs and PCCs to provide scheduled LSP for
customers' traffic services at its actual usage time, so as to
improve the network resource efficient utilization.

For stateful PCE supporting LSP scheduling, there are two types of
LSP databases used in this document. One is the LSP-DB defined in
PCEP [I-D.ietf-pce-stateful-pce], while the other is the scheduled
LSP database (SLSP- DB, see section 6). The SLSP-DB records
scheduled LSPs and is used as a complementary to the TED and LSP-DB.
Note that the two types of LSP databases can be implemented in one
physical database or two different databases. This document does not
state any preference here.

Furthermore, a scheduled TED can be generated from the scheduled LSP
DB, LSP DB and TED to indicate the network links and nodes with
resource availability information for now and future. The scheduled
TED should be maintained by all PCEs within the network environment.

In case of implementing PCC-initiated scheduled LSPs, a PCC can
request a path computation with LSP information of its scheduling
parameters, including the starting time and the duration. Upon

receiving the request with the scheduled LSP delegation, a stateful
PCE SHALL check the scheduled TED for the network resource
availability on network nodes and computes a path for the LSP with
the scheduling information.

For a multiple PCE environment, in order to coordinate the scheduling
request of the LSP path over the network, the PCE needs to send a
requestmessage with the path information as well as the scheduled
resource for the scheduled LSP to other PCEs within the network, so
as to coordinate with their scheduled LSP DBs and scheduled TEDs.
Once other PCEs receive the request message with the scheduled LSPs
information, if not conflicting with their scheduled LSP DBs, they
reply to the requesting PCE with a response message carrying the
scheduled LSP and update their scheduled LSP DBs and scheduled TEDs.
After the requesting PCE confirms with all PCEs, the PCE SHALL add
the scheduled LSP into its scheduled LSP Database and update its
scheduled TED.

Then the stateful PCE can response to the PCC with the path for the
scheduled LSP to notify the result of the computation. However, the
PCC should not signal the LSP over the path once receiving these
messages since the path is not activated yet until its starting time.

Alternatively, the service can also be initiated by PCE itself. In
case of implementing PCE-initiated scheduled LSP, the stateful PCE
shall check the network resource availability for the traffic and
computes a path for the scheduled LSP per request in the same way as
in PCC- Initiated mode and then for a multiple PCE network
environment, coordinate the scheduled LSP with other PCEs in the
network in the same way as in the PCC-Initiated mode.

In both modes, for activation of scheduled LSPs, the stateful PCE can
send a path computation LSP Initiate (PCInitiate message) with LSP
information at its starting time to the PCC for signaling the LSP
over the network nodes as defined in [I-D.ietf-pce-pce- initiated-
lsp]. Also, in the PCC-initiated mode, with scheduling information
,the PCC can activate the LSP itself by triggering over the path at
its starting time as well. When the scheduling usage expires, active
stateful PCE SHALL remove the LSP from the network , as well as
notify other PCEs to delete the scheduled LSP from the scheduled LSP
database.

4.2. Support of LSP Scheduling

4.2.1. LSP Scheduling

   For a scheduled LSP, a user configures it with an arbitrary
   scheduling duration time Ta to time Tb, which may be represented as
   [Ta, Tb].

   When an LSP is configured with arbitrary scheduling duration [Ta,
   Tb], a path satisfying the constraints for the LSP in the scheduling
   duration is computed and the LSP along the path is set up to carry
   traffic from time Ta to time Tb.

4.2.2. Periodical LSP Scheduling

   In addition to LSP Scheduling at an arbitrary time period, there are
   also periodical LSP Scheduling.

   A periodical LSP Scheduling represents Scheduling LSP every time
   interval. It has a scheduling duration such as [Ta, Tb], a number of
   repeats such as 10 (repeats 10 times), and a repeat cycle/time
   interval such as a week (repeats every week). The scheduling
   interval: "[Ta, Tb] repeats n times with repeat cycle C" represents
   n+1 scheduling intervals as follows:

   [Ta, Tb], [Ta+C, Tb+C], [Ta+2C, Tb+2C], ..., [Ta+nC, Tb+nC]

   When an LSP is configured with a scheduling interval such as "[Ta,
   Tb] repeats 10 times with a repeat cycle a week" (representing 11
   scheduling intervals), a path satisfying the constraints for the LSP
   in each of the scheduling intervals represented by the periodical
   scheduling interval is computed and the LSP along the path is set up
   to carry traffic in each of the scheduling intervals.

4.2.2.1. Elastic Time LSP Scheduling

   In addition to the basic LSP scheduling at an arbitrary time period,
   another option is elastic time intervals, which is represented as
   within -P and Q, where P and Q is an amount of time such as 300
   seconds. P is called elastic range lower bound and Q is called
   elastic range upper bound.

   For a simple time interval such as [Ta, Tb] with an elastic range,
   elastic time interval: "[Ta, Tb] within -P and Q" means a time period
   from (Ta+X) to (Tb+X), where -P <= X <= Q. Note that both Ta and Tb
   may be shifted the same X.

   When an LSP is configured with elastic time interval "[Ta, Tb] within
   -P and Q", a path is computed such that the path satisfies the
   constraints for the LSP in the time period from (Ta+X) to (Tb+X)

and |X| is the minimum value from 0 to max(P, Q). That is that
[Ta+X, Tb+X] is the time interval closest to time interval [Ta, Tb]
within the elastic range. The LSP along the path is set up to carry
traffic in the time period from (Ta+X) to (Tb+X).

Similarly, for a recurrent time interval with an elastic range,
elastic time interval: "[Ta, Tb] repeats n times with repeat cycle C
within -P and Q" represents n+1 simple elastic time intervals as
follows:

[Ta+X0, Tb+X0], [Ta+C+X1, Tb+C+X1], ..., [Ta+nC+Xn, Tb+nC+Xn]
                         where -P <= Xi <= Q, i = 0, 1, 2, ..., n.

If a user wants to keep the same repeat cycle between any two
adjacent time intervals, elastic time interval: "[Ta, Tb] repeats n
times with repeat cycle C within -P and Q SYNC" may be used, which
represents n+1 simple elastic time intervals as follows:

[Ta+X, Tb+X], [Ta+C+X, Tb+C+X], ..., [Ta+nC+X, Tb+nC+X]
                         where -P <= X <= Q.

4.2.2.2. Graceful Periods

Besides the stated time scheduling, a user may want to have some
graceful periods for each or some of the time intervals for the LSP.
Two graceful periods may be configured for a time interval. One is
the graceful period before the time interval, called grace-before,
which extends the lifetime of the LSP for grace-before (such as 30
seconds) before the time interval. The other is the one after the
time interval, called grace-after, which extends the lifetime of the
LSP for grace-after (such as 60 seconds) after the time interval.

When an LSP is configured with a simple time interval such as [Ta,
Tb] with graceful periods such as grace-before GB and grace-after GA,
a path is computed such that the path satisfies the constraints for
the LSP in the time period from Ta to Tb. The LSP along the path is
set up to carry traffic in the time period from (Ta-GB) to (Tb+GA).
During graceful periods from (Ta-GB) to Ta and from Tb to (Tb+GA),
the LSP is up to carry traffic (maybe in best effort).

4.2.3. Stateful PCE Capability TLV

After a TCP connection for a PCEP session has been established, a PCC
and a PCE indicates its ability to support LSP scheduling during the
PCEP session establishment phase. For a multiple-PCE environment,
the PCEs should also establish PCEP session and indicate its ability
to support LSP scheduling among PCEP peers. The Open Object in the
Open message contains the STATEFUL-PCE-CAPABILITY TLV defined in [I-

D.ietf-pce-stateful-pce]. Note that the STATEFUL- PCE-CAPABILITY TLV
is defined in [I-D.ietf-pce-stateful- pce] and updated in [I-D.ietf-
pce-pce-initiated-lsp] and [I-D.ietf- pce-stateful-sync-
optimizations]. In this document, we define a new flag bit B (SCHED-
LSP-CAPABLITY) flag for the STATEFUL- PCE-CAPABILITY TLV to indicate
the support of LSP scheduling and another flag bit PD (PD-LSP-
CAPABLITY) to indicate the support of LSP periodical scheduling.

   B (LSP-SCHEDULING-CAPABILITY - 1 bit): If set to 1 by a PCC, the B
      Flag indicates that the PCC allows LSP scheduling; if set to 1 by
      a PCE, the B Flag indicates that the PCE is capable of LSP
      scheduling. The B bit MUST be set by both PCEP peers in order to
      support LSP scheduling for path computation.

   PD (PD-LSP-CAPABLITY - 1 bit): If set to 1 by a PCC, the PD Flag
      indicates that the PCC allows LSP scheduling periodically; if set
      to 1 by a PCE, the PD Flag indicates that the PCE is capable of
      periodical LSP scheduling. The PD bit MUST be set by both PCEP
      peers in order to support periodical LSP scheduling for path
      computation.

4.3. Scheduled LSP creation

   In order to realize PCC-Initiated scheduled LSP in a centralized
   network environment, a PCC has to separate the setup of a LSP into
   two steps. The first step is to request and get a LSP but not signal
   it over the network. The second step is to signal the scheduled LSP
   over the LSRs (Labeled switched Router) at its starting time.

   For PCC-Initiated scheduled LSPs, a PCC can send a path computation
   request (PCReq) message (see section 4.3.1) or a path computation LSP
   report (PCRpt) message (see section 4.3.1) including its demanded
   resources with the scheduling information and delegation to a
   stateful PCE.

   Upon receiving the delegation via PCRpt message, the stateful PCE
   computes the path for the scheduled LSP per its starting time and
   duration based on the network resource availability stored in
   scheduled TED (see section 4.1).

   If a resultant path is found, the stateful PCE will send a PCReq
   message with the path information as well as the scheduled resource
   information for the scheduled LSP to other PCEs within the network if
   there is any, so as to keep their scheduling information
   synchronized.

   Once other PCEs receive the PCReq message with the scheduled LSP, if
   not conflicts with their scheduled LSP DBs, they will reply to the

requesting PCE with a PCRep message carrying the scheduled LSP and
update their scheduled LSP DBs and scheduled TEDs. After the
requesting PCE confirms with all PCEs, the PCE SHALL add the
scheduled LSP into its scheduled LSP DB and update its scheduled TED.
If conflicts happen or no path available is found, the requesting PCE
SHALL return a PCRep message with NO PATH back to the PCC.
Otherwise, the stateful PCE will send a PCRep message or PCUpd
message (see section 4.3.3) with the path information back to the PCC
as confirmation.

For PCE-Initiated Scheduled LSP, the stateful PCE can compute a path
for the scheduled LSP per requests from network management systems
automatically based on the network resource availability in the
scheduled TED and coordinate with other PCEs on the scheduled LSP in
the same way as in the PCC- Initiated mode.

In both modes:

o the stateful PCE is required to update its local scheduled LSP DB
   and scheduled TED with the scheduled LSP. Besides, it shall send
   a PCReq message with the scheduled LSP to other PCEs within the
   network, so as to achieve the scheduling traffic engineering
   information synchronization.

o Upon receiving the PCRep message or PCUpd message for scheduled
   LSP from PCEs with a found path, the PCC knows that it gets a
   scheduled path for the LSP but not trigger signaling for the LSP
   setup on LSRs.

o In any case, stateful PCE can update the Scheduled LSP parameters
   on any network events using the PCUpd message to PCC as well as
   other PCEs.

o When it is time (i.e., at the start time) for the LSP to be set
   up, the delegated PCE sends a PCEP Initiate request to the head
   end LSR providing the path to be signaled.

4.3.1. The PCReq message and PCRpt Message

   After scheduled LSP capability negotiation, for PCC-Initiated mode, a
   PCC can send a PCReq message or a PCRpt message including the SCHED-
   LSP- ATTRIBUTE TLV (see section 4.3.4.1) or SCHED-PD-LSP-ATTRIBUTE
   TLV (see section 4.3.4.2) carried in the LSP Object (see section
   4.3.4) body to indicate the requested LSP scheduling parameters for a
   customer's traffic service with the delegation bit set to 1 in LSP
   Object. The value of requested bandwidth is taken via the existing
   'Requested Bandwidth with BANDWIDTH Object- Type as 1' defined in
   [RFC5440].

Meanwhile, for both modes (PCC-Initiated and PCE-Initiated), the
delegated PCE shall distribute the scheduling information to other
PCEs in the environment by sending a PCReq message with the SCHED-
LSP-ATTRIBUTE TLV or SCHED-PD-LSP-ATTRIBUTE TLV, as well as the
Bandwith Object and RRO for the found path.

The definition of the PCReq message and PCRpt message to carry LSP
objects (see [I- D.ietf-pce-stateful-pce]) remains unchanged.

4.3.2. The PCRep Message

To provide scheduled LSP for TE-LSPs, the stateful PCE SHALL compute
the path for the scheduled LSP carried on PCReq message based on
network resource availability recorded in scheduled TED which is
generated from the scheduled LSP-DB and TED and also synchronize the
scheduling with other PCEs in the environment by using PCReq message
with path and resource information for the scheduled LSP.

If no conflict exists, other PCEs SHALL send a PCRep message with the
SCHED-LSP-ATTRIBUTE TLV or SCHED-PD-LSP-ATTRIBUTE TLV, as well as the
Bandwith Object and RRO back to the requesting PCE.

If the LSP request can be satisfied and an available path is found,
the stateful PCE SHALL send a PCRep Message including the SCHED- LSP-
ATTRIBUTE TLV or SCHED-PD-LSP-ATTRIBUTE TLV in the LSP Object body,
as well as the Bandwith Object and RRO for the found path back to the
PCC as a successful acknowledge.

If conflicts happen or no path available is found, the requesting PCE
SHALL return a PCRep message with NO PATH back to the PCC.

4.3.3. The PCUpd Message

To provide scheduled LSP for TE-LSPs, the stateful PCE SHALL compute
the path for the scheduled LSP carried on PCRpt message based on
network resource availability recorded in scheduled TED which is
generated from the scheduled LSP-DB, LSP DB and TED.

If the request can be satisfied and an available path is found, the
stateful PCE SHALL send a PCUpd Message including the SCHED- LSP-
ATTRIBUTE TLV or SCHED-PD-LSP-ATTRIBUTE TLV in the LSP Object body to
the PCC Note that, the stateful PCE can update the Scheduled LSP
parameters later as well based on any network events using the same
PCUpd message.

If conflicts happen or no path available is found, the requesting PCE
SHALL return a PCUpd message with ERO empty.

4.3.4. LSP Object

   The LSP object is defined in [I-D.ietf-pce-stateful-pce]. This
   document add an optional SCHED-LSP-ATTRIBUTE TLV for normal LSP
   scheduling and an optional SCHED-PD-LSP-ATTRIBUTE TLV for periodical
   LSP scheduling.

   The presence of SCHED-LSP-ATTRIBUTE TLV in the LSP object indicates
   that this LSP is requesting scheduled parameters while the SCHED-PD-
   LSP-ATTRIBUTE TLV indicates that this scheduled LSP is periodical.
   The scheduled LSP attribute TLV MUST be present in LSP Object for
   each scheduled LSP carried in the PCReq message, the PCRpt message
   and the PCUpd message. For periodical LSPs, the SCHED-PD-LSP-
   ATTRIBUTE TLV can be used in LSP Object.

4.3.4.1. SCHED-LSP-ATTRIBUTE TLV

   The SCHED-LSP-ATTRIBUTE TLV can be included as an optional TLV within
   the LSP object for LSP scheduling for the requesting traffic service.

   This TLV SHOULD be included only if both PCEP peers have set the B
   (LSP-SCHEDULING-CAPABILITY bit) in STATEFUL-PCE-CAPABILITY TLV
   carried in open message.

   The format of the SCHED-LSP-ATTRIBUTE TLV is shown in the following
   figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type              |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Start-Time (minutes)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Duration (minutes)                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   The type of the TLV is [TBD] and it has a fixed length of 8 octets.

   The fields in the format are:

   Start-Time (32 bits): This value in minutes, indicates when the
      scheduled LSP is used to carry traffic and the corresponding LSP
      must be setup and activated.

   Duration (32 bits): The value in minutes, indicates the duration
      that the LSP is undertaken by a traffic flow and the corresponding

LSP must be up to carry traffic. At the expiry of this duration, the LSP is tear down and deleted.

Note, that the values of starting time and duration is from the perspective of the PCEP peer that is sending the message, also note the unit of time is minutes, and thus the time spent on transmission on wire can be easily ignored.

Editor Note 1: As described in [I-D.zhuang-teas-scheduled-resources],the encoding of the resource state information could also be expressed as a start time and end time. Multiple periods, possibly of different lengths, may be associated with one reservation request, and a reservation might repeat on a regular cycle.

Editor Notes2: The time stated in this section and in section 4.3.4.2 may be a relative time or an absolute time, which need more discussions.

Editor Note3: the elastic interval and graceful interval may also be applied to the random LSP scheduling which need more discussion.

4.3.4.2. SCHED-PD-LSP-ATTRIBUTE TLV

The periodical LSP is a special case of LSP scheduling. The traffic service happens in a series of repeated time intervals. The SCHED-PD-LSP-ATTRIBUTE TLV can be included as an optional TLV within the LSP object for this periodical LSP scheduling.

This TLV SHOULD be included only if both PCEP peers have set the B (LSP-SCHEDULING-CAPABILITY bit) and PD (PD-LSP-CAPABLITY bit) in STATEFUL-PCE-CAPABILITY TLV carried in open message.

The format of the SCHED-PD-LSP-ATTRIBUTE TLV is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type (3)           |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Start-Time                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Duration                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Repeat-time-length                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Options |         Number-repeats        |     Reserved (0)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              GrB              |              GrA              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Elastic-Lower-Bound      |      Elastic-Upper-Bound      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
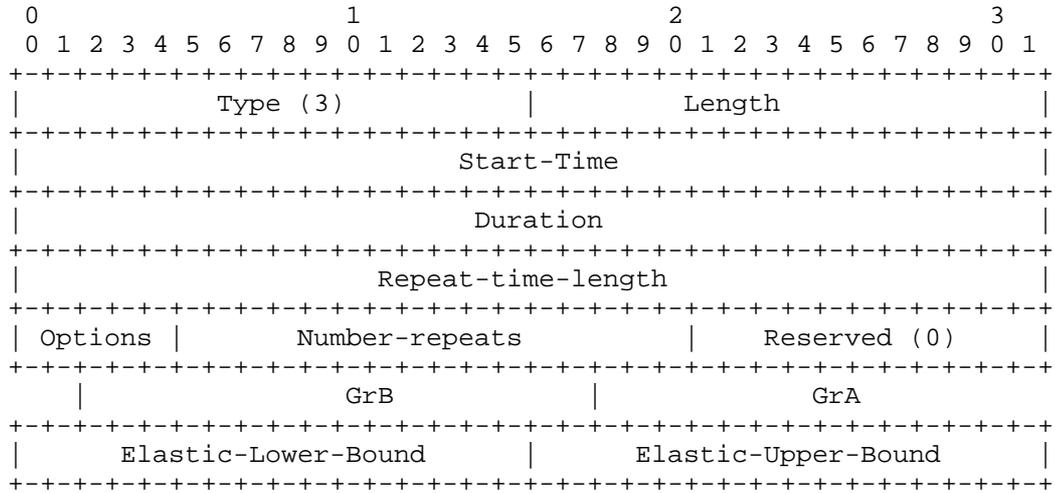
Start-Time (32 bits): This value in minutes, indicates the time when
   the scheduled LSP is used to carry traffic and the corresponding
   LSP must be setup and activated.

Duration (32 bits): The value in minutes, indicates the duration
   that the LSP is undertaken by a traffic flow and the corresponding
   LSP must be up to carry traffic.

Repeat-time-length: The time length in minutes after which LSP
   starts to carry traffic again for (Start Time-End Time).

Options: Indicates a way to repeat.

     Options = 1: repeat every day;

     Options = 2: repeat every week;

     Options = 3: repeat every month;

     Options = 4: repeat every year;

     Options = 5: repeat every Repeat-time-length.

Number-repeats: The number of repeats. In each of repeats, LSP
   carries traffic.

In addition, it contains an non zero grace-before and grace-after if
graceful periods are configured. It includes an non zero elastic
range lower bound and upper bound if there is an elastic range
configured.

    o GrB (Grace-Before): The graceful period time length in seconds
       before the starting time.

    o GrA (Grace-After): The graceful period time length in seconds
       after time interval [starting time, starting time + duration].

    o Elastic-Lower-Bound: The maximum amount of time in seconds that
       time interval can shift to lower/left.

    o Elastic-Upper-Bound: The maximum amount of time in seconds that
       time interval can shift to upper/right.

4.4. Scheduled LSP Updates

After a scheduled LSP is configured, a user may change its parameters
including the requested time as well as the bandwidth.

In PCC-Initiated case, the PCC can send a PCRpt message for the
scheduled LSP with updated bandwidth as well as scheduled information
included in the SCHED-LSP-ATTRIBUTE TLV (see section 4.3.4.1) or
SCHED-PD-LSP-ATTRIBUTE TLV carried in the LSP Object. The PCE should
calculate the updated resources and synchronized with other PCEs. If
the updates can be satisfied, PCE shall return a PCUpd message to PCC
as described in section 4.3.3. If the requested updates cannot be
met, PCE shall return a PCUpd message with the original reserved
attributes carried in the LSP Object.

The stateful PCE can update the Scheduled LSP parameters to other
PCEs and the requested PCC at any time based on any network events
using the PCUpd message including SCHED-LSP-ATTRIBUTE TLV or SCHED-
PD-LSP-ATTRIBUTE TLV in the LSP Object body.

4.5. Scheduled LSP activation and deletion

In PCC-Initiated LSP scheduling, the PCC itself MAY activate the
scheduled LSP at the starting time. Alternatively, the stateful PCE
MAY activate the scheduled LSP at its scheduled time by send a
PCInitiated message.

After the scheduled duration expires, the PCE shall send a PCUpd
message with R flag set to the PCC to delete the LSP over the path,
as well as to other PCEs to remove the scheduled LSP in the
databases. Additionally, it shall update its scheduled LSP DB and
scheduled TED.

Note that, the stateful PCE can update the Scheduled LSP parameters
at any time based on any network events using the PCUpd message
including SCHED-LSP-ATTRIBUTE TLV in the LSP Object body.

5. Security Considerations

   This document defines LSP-SCHEDULING-CAPABILITY TLV and SCHED- LSP-
   ATTRIBUTE TLV which does not add any new security concerns beyond
   those discussed in [RFC5440] and [I-D.ietf-pce-stateful-pce].

6. Manageability Consideration

6.1. Control of Function and Policy

   The LSP-Scheduling feature MUST BE controlled per tunnel by the
   active stateful PCE, the values for parameters like starting time,
   duration SHOULD BE configurable by customer applications and based on
   the local policy at PCE.

6.2. Information and Data Models

   [RFC7420] describes the PCEP MIB, there are no new MIB Objects for
   this document.

6.3. Liveness Detection and Monitoring

   Mechanisms defined in this document do not imply any new liveness
   detection and monitoring requirements in addition to those already
   listed in [RFC5440].

6.4. Verify Correct Operations

   Mechanisms defined in this document do not imply any new operation
   verification requirements in addition to those already listed in
   [RFC5440].

6.5. Requirements On Other Protocols

   Mechanisms defined in this document do not imply any new requirements
   on other protocols.

6.6. Impact On Network Operations

   Mechanisms defined in this document do not have any impact on network
   operations in addition to those already listed in [RFC5440].

7. IANA Considerations

7.1. PCEP TLV Type Indicators

   This document defines the following new PCEP TLV; IANA is requested
   to make the following allocations from this registry.

   Value   Meaning   Reference
    TBD   SCHED-LSP-ATTRIBUTE   This document
    TBD   SCHED-PD-LSP-ATTRIBUTE   This document

7.2. LSP-SCHEDULING-CAPABLITY

   This document requests that a registry is created to manage the Flags
   field in the STATEFUL-PCE-CAPABILITY TLV in the OPEN object. New
   values are to be assigned by Standards Action [RFC5226]. Each bit
   should be tracked with the following qualities:

   o Bit number (counting from bit 0 as the most significant bit)

   o Capability description

   o Defining RFC

   The following values are defined in this document:

   Bit   Description   Reference
    28   LSP-SCHEDULING-CAPABILITY (B-bit)   This document
    29   PD-LSP-CAPABLITY (PD-bit)   This document

8. Acknowledgments

   This work has benefited from the discussions of resource scheduling
   on the mailing list and with Huaimo chen, author of [I-D.chen-pce-
   tts] since Prague meeting. We gratefully acknowledge the
   contributions of Huaimo Chen. The authors of this document would
   also like to thank Rafal Szarecki,Adrian Farrel, Cyril Margaria, Xian
   Zhang for the review and comments.

9. References

9.1. Normative References

   [I-D.dhody-pce-stateful-pce-auto-bandwidth]
             Dhody, D., Palle, U., Singh, R., Gandhi, R., and L. Fang,
             "PCEP Extensions for MPLS-TE LSP Automatic Bandwidth
             Adjustment with Stateful PCE", draft-dhody-pce-stateful-
             pce-auto-bandwidth-09 (work in progress), November 2016.

    [I-D.ietf-pce-pce-initiated-lsp]
              Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
              Extensions for PCE-initiated LSP Setup in a Stateful PCE
              Model", draft-ietf-pce-pce-initiated-lsp-09 (work in
              progress), March 2017.

    [I-D.ietf-pce-stateful-pce]
              Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
              Extensions for Stateful PCE", draft-ietf-pce-stateful-
              pce-18 (work in progress), December 2016.

    [I-D.ietf-pce-stateful-sync-optimizations]
              Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X.,
              and D. Dhody, "Optimizations of Label Switched Path State
              Synchronization Procedures for a Stateful PCE", draft-
              ietf-pce-stateful-sync-optimizations-10 (work in
              progress), March 2017.

    [I-D.ietf-teas-scheduled-resources]
              Zhuangyan, Z., Wu, Q., Chen, H., and A. Farrel,
              "Architecture for Scheduled Use of Resources", draft-ietf-
              teas-scheduled-resources-02 (work in progress), January
              2017.

    [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

    [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol (PCEP)", RFC 5440,
              DOI 10.17487/RFC5440, March 2009,
              <http://www.rfc-editor.org/info/rfc5440>.

9.2. Informative References

    [I-D.ietf-pce-stateful-pce-app]
              Zhang, X. and I. Minei, "Applicability of a Stateful Path
              Computation Element (PCE)", draft-ietf-pce-stateful-pce-
              app-08 (work in progress), October 2016.

    [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an
              IANA Considerations Section in RFCs", BCP 26, RFC 5226,
              DOI 10.17487/RFC5226, May 2008,
              <http://www.rfc-editor.org/info/rfc5226>.

   [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
              Hardwick, "Path Computation Element Communication Protocol
              (PCEP) Management Information Base (MIB) Module",
              RFC 7420, DOI 10.17487/RFC7420, December 2014,
              <http://www.rfc-editor.org/info/rfc7420>.

Appendix A. Scheduled LSP information synchronization

   As for a stateful PCE, it maintains a database of LSPs (LSP-DB) that
   are active in the network, so as to reveal the available network
   resources and place new LSPs more cleverly.

   With the scheduled LSPs, they are not activated while creation, but
   should be considered when operating future path computation. Hence,
   a scheduled LSP Database (SLSP-DB) is suggested to maintain all
   scheduled LSP information.

   The information of SLSP-DB MUST be shared and synchronized among all
   PCEs within the centralized network by using PCReq message, PCRep
   message with scheduled LSP information. In order to synchronize the
   scheduled LSP information in SLSP-DB among PCEs, the PCReq message
   and PCRep Message is used as described in section 4.3.1 and section
   4.3.2.

   To achieve the synchronization, the PCE should generate and maintain
   a scheduled TED based on LSP DB, scheduled LSP DB and TED, which is
   used to indicate the network resource availability on network nodes
   for LSP path computation.

Appendix B. Contributor Addresses

         Xufeng Liu
         Ericsson
         USA
         Email: xliu@kuatrotech.com

         Mehmet Toy
         Verizon
         USA
         Email: mehmet.toy@verizon.com

         Vic Liu
         China Mobile
         No.32 Xuanwumen West Street, Xicheng District
         Beijing,   100053
         China
         Email: liu.cmri@gmail.com

         Lei Liu
         Fujitsu
         USA
         Email: lliu@us.fujitsu.com

         Khuzema Pithewan
         Infinera
         Email: kpithewan@infinera.com

         Zitao Wang
         Huawei
         101 Software Avenue, Yuhua District
         Nanjing, Jiangsu 210012
         China

         Email: wangzitao@huawei.com

         Xian Zhang
         Huawei Technologies
         Research Area F3-1B,
         Huawei Industrial Base,
         Shenzhen, 518129, China

         Email: zhang.xian@huawei.com

Authors' Addresses

   Huaimo Chen (editor)
   Huawei
   Boston, MA
   USA

   Email: huaimo.chen@huawei.com

   Yan Zhuang (editor)
   Huawei
   101 Software Avenue, Yuhua District
   Nanjing, Jiangsu 210012
   China

   Email: zhuangyan.zhuang@huawei.com

   Qin Wu
   Huawei
   101 Software Avenue, Yuhua District
   Nanjing, Jiangsu 210012
   China

   Email: bill.wu@huawei.com

   Dhruv Dhody
   Huawei
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   Email: dhruv.ietf@gmail.com

   Daniele Ceccarelli
   Ericsson
   Via A. Negrone 1/A
   Genova - Sestri Ponente
   Italy

   Email: daniele.ceccarelli@ericsson.com