                       PCEP Link State Abstraction
                   draft-chen-pce-h-connect-access-07

Abstract

   This document presents extensions to the Path Computation Element
   Communication Protocol (PCEP) for a child PCE to abstract its domain
   information to its parent for supporting a hierarchical PCE system.

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 11, 2021.

Table of Contents

1. Introduction

   A hierarchical PCE architecture is described in RFC 6805, in which a
   parent PCE maintains an abstract domain topology, which contains its
   child domains (seen as vertices in the topology) and the connections
   among them.

   For a domain for which a child PCE is responsible, connections
   attached to the domain may comprise inter-domain links and Area
   Border Routers (ABRs). For a parent PCE to have the abstract domain
   topology, each of its child PCEs needs to advertise its connections
   to the parent PCE.

   In addition to the connections attached to the domain, there may be
   some access points in the domain, which are the addresses in the
   domain to be accessible outside of the domain. For example, an
   address of a server in the domain that provides a number of services
   to users outside of the domain is an access point.

   This document presents extensions to the Path Computation Element
   Communication Protocol (PCEP) for a child PCE to advertise the
   information about its connections and access points to its parent PCE
   and for the parent PCE to build and maintain the abstract domain
   topology based on the information. The extensions may reduce
   configurations, thus simplify operations on a PCE system.

   A child PCE is simply called a child and a parent PCE is called a
   parent in the following sections.

2. Terminology

   ABR: Area Border Router. Router used to connect two IGP areas
      (Areas in OSPF or levels in IS-IS).

   ASBR: Autonomous System (AS) Border Router. Router used to connect
      together ASes via inter-AS links.

   TED: Traffic Engineering Database.

   This document uses terminology 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. Connections and Accesses

   A connection is an inter-domain link between two domains in general.
   An ABR is also a connection, which connects two special domains
   called areas in a same Autonomous System (AS).

   An access point in a domain is an address in the domain to be
   accessible to the outside of the domain. An access point is simply
   called an access.

4.1. Information on Inter-domain Link

   An inter-domain link connects two domains in two different ASes.
   Since there is no IGP running over an inter-domain link, we may not
   obtain the information about the link generated by an IGP. We may
   suppose that IP addresses are configured on inter-domain links.

   For a point-to-point (P2P) link connecting two ABSRs A and B in two
   different domains, from A's point of view, the following information
   about the link may be obtained:

     1) Link Type: P2P
     2) Local IP address
     3) Remote IP address
     4) Traffic engineering metric
     5) Maximum bandwidth
     6) Maximum reservable bandwidth
     7) Unreserved bandwidth
     8) Administrative group
     9) SRLG

   We will have a link ID if it is configured; otherwise no link ID
   (i.e., the Router ID of the neighbor) may be obtained since no IGP
   adjacency over the link is formed.

   For a broadcast link connecting multiple ASBRs in a number of
   domains, on each of the ASBRs X, the same information about the link
   as above may be obtained except for the followings:

     a) Link Type: Multi-access,
     b) Local IP address with mask length, and
     c) No Remote IP address.

   In other words, the information about the broadcast link obtained by
   ASBR X comprises a), b), 4) to 9), but does not include any remote IP
   address or link ID. We will have a link ID if it is configured;
   otherwise no link ID (i.e., the interface address of the designated
   router for the link) may be obtained since no IGP selects it.

   A parent constructs an abstract AS domain topology after receiving
   the information about each of the inter-domain links described above
   from its children.

   RFC 5392 and RFC 5316 describe the distributions of inter-domain
   links in OSPF and IS-IS respectively. For each inter-domain link,

its neighboring AS number and neighboring ASBR Identity (TE Router
ID) need to be configured in IGP (OSPF or IS-IS).

In addition, an IGP adjacency between a network node running IGP and
a PCE running IGP as a component needs to be configured and fully
established if we want the PCE to obtain the inter-domain link
information from IGP.

These configurations and IGP adjacency establishment are not needed
if the extensions in this draft are used.

RFC 7752 (BGP-LS) describes the distributions of TE link state
information including inter-domain link state. A BGP peer between a
network node running BGP and a PCE running BGP as a component needs
to be configured and the peer relation must be established before the
PCE can obtain the inter-domain link information from BGP. However,
some networks may not run BGP.

4.2. Information on ABR

For an AS running IGP and containing multiple areas, an ABR connects
two or more areas. For each area connected to the ABR, the PCE as a
child responsible for the area sends its parent the information about
the ABR, which indicates the identifier (ID) of the ABR.

A parent has the information about each of its children, which
includes the domain such as the area for which the child is
responsible. The parent knows all the areas to which each ABR
connects after receiving the information on the ABR from each of its
children.

4.3. Information on Access Point

For an IP address in a domain to be accessible outside of the domain,
the PCE as a child responsible for the domain sends its parent the
information about the address.

The parent has all the access points (i.e., IP addresses) to be
accessible outside of all its children' domains after receiving the
information on the access points from each of its children.

5. Extensions to PCEP

This section focuses on procedures for abstracting domain information
after briefing messages containing the abstract information.

5.1. Messages for Abstract Information

   A child abstracts its domain to its parent through sending its parent
   a message containing the abstract information on the domain. After
   the relation between the child and the parent is determined, the
   parent has some information on the child, which includes the child's
   ID and domain. The message does not need to contain this
   information. It comprises the followings:

   o For new or updated Connections and Accesses,

      * Indication of Update Connections and Accesses

      * Detail Information about Connections and Accesses

   o For Connections and Accesses down,

      * Indication of Withdraw Connections and Accesses

      * ID Information about Connections and Accesses

   For a P2P link from ASBR A to B and a broadcast link connecting to A,
   the detail information on the links includes A's ID, the information
   on the P2P link and the information on the broadcast link described
   in Section 4. The ID information on the links includes A's ID, 1) to
   3) for the P2P link and a) to b) for the broadcast link described in
   Section 4. A link ID for a link is included if it is configured.

   For an ABR X, the information on X includes X's ID and a flag
   indicating that X is ABR.

   For an Access X (address), the detail information on X includes X and
   a cost associated with it. The ID information on X is X itself.

   There are a few ways to encode the information above into a message.
   For example, one way is to extend an existing Notification message
   for including the information. Another way is to use a new message.
   These are put in Appendix A for your reference.

5.2. Procedures

5.2.1. Child Procedures

5.2.1.1. New or Changed Connections and Accesses

   After a child determines its parent, it sends the parent a message
   containing the information about the connections (i.e., inter-domain

links and ABRs) from its domain to its adjacent domains and the
access points in its domain.

For any new or changed inter-domain links, ABRs and access points in
the domain for which a child is responsible, the child sends its
parent a message containing the information about these links, ABRs
and access points with indication of Update Connections and Accesses.

For example, for a new inter-domain P2P link from ASBR A in a child's
domain to ASBR B in another domain, the child sends its parent a
message containing an indication of Update Connections and Accesses,
A's ID, and the detail information on the link described in section
4.1.

For multiple new or changed inter-domain links from ASBR A, the child
sends its parent a message having an indication of Update Connections
and Accesses, and A's ID followed by the detail information about
each of the links.

In another example, for a new or changed inter-domain broadcast link
connected to ASBR X, an ABR Y and an access point 10.10.10.1/32 with
cost 10 in a child's domain, the child sends its parent a message
containing an indication of Update Connections and Accesses, and X's
ID followed by the detail information about the link attached to X
and the detail information about ABR Y, and the information on access
10.10.10.1/32 with cost 10.

For changes on the attributes (such as bandwidth) of an inter-domain
link, a threshold may be used to control the frequency of updates
that are sent from a child to its parent. At one extreme, the
threshold is set to let a child send its parent a update message for
any change on the attributes of an inter-domain link. At another
extreme, the threshold is set to make a child not to send its parent
any update message for any change on the attributes of an inter-
domain link. Typically, the threshold is set to allow a child to
send its parent a update message for a significant change on the
attributes of an inter-domain link.

5.2.1.2. Connections and Accesses Down

For any inter-domain links, ABRs and access points down in the domain
for which a child is responsible, the child sends its parent a
message containing the information about these links, ABRs and access
points with indication of Withdraw Connections and Accesses.

For example, for the inter-domain P2P link from ASBR A down, the
child sends its parent a message containing an indication of Withdraw

Connections and Accesses, and A's ID, which is followed by the ID
information about the link.

For multiple inter-domain links from ASBR A down, the child sends its
parent a message having an indication of Withdraw Connections and
Accesses, and A's ID, which is followed by the ID information about
each of the links.

5.2.1.3. Child and Parent in Same Organization

If a child and its parent are in a same organization, the child may
send its parent the information inside its domain. For a parent,
after all its children in its organization send their parent the
information in their domains, connections and access points, it has
in its TED the detail information inside each of its children's
domains and the connections among these domains. The parent can
compute a path crossing these domains directly and efficiently
without sending any path computation request to its children.

5.2.1.4. Child as a Parent

There are a few ways in which a child as a parent abstracts its
domain information to its parent.

One way is that the child sends its parent all its domain information
if the child and the parent are in a same organization. The
information includes the detail network topology inside each of the
child's domains, the inter-domain links connecting the domains that
the child's children are responsible and the inter-domain links
connecting these domains to other adjacent domains.

In another way, the child abstracts each of the domains that its
children are responsible as a cloud (or say abstract node) and these
clouds are connected by the inter-domain links attached to the
domains. The child sends its parent all the inter-domain links
attached to any of the domains.

In a third way, the child abstracts all its domains including the
domains for which its children are responsible as a cloud. This
abstraction is described below in details.

If a parent P1 is also a child of another parent P2, P1 as a child
sends its parent P2 a message containing the information about the
connections and access points. P1 as a parent has the connections
among its children's domains. But these connections are hidden from
its parent P2. P1 may have connections from its children's domains
to other domains. P1 as a child sends its parent P2 these
connections.

P1 as a parent has the access points in its children's domains to be accessible outside of the domains. P1 as child may not send all of these to its parent P2. It sends its parent some of these access points according to some local policies.

From P2's point of view, its child P1 is responsible for one domain, which has some connections to its adjacent domains and some access points to be accessible.

5.2.2. Parent Procedures

5.2.2.1. Process Connections and Accesses

A parent stores into its TED the connections and accesses for each of its children according to the messages containing connections and accesses received. For a message containing Update Connections and Accesses, it updates the connections and accesses in the TED accordingly. For a message containing Withdraw Connections and Accesses, it removes the connections and accesses from the TED.

After receiving the messages for connections and accesses from its children, the parent builds and maintains the TED for the topology of its children's domains, in which each of the domains is seen as a cloud or an abstract node. The information inside each of the domains is hidden from the parent. There are connections among the domains and the access points in the domains to be accessible in the topology.

For a new P2P link from node A to B with no link ID configured, when receiving a message containing the link from a child, the parent stores the link from A into its TED, where A is attached to the child's domain as a cloud. It finds the link's remote end B using the remote IP address of the link. After finding B, it associates the link attached to A with B and the link attached to B with A. This creates a bidirectional connection between A and B.

For a new P2P link from node A to B with link ID configured, when receiving a message containing the link, the parent stores the link from A into its TED. It finds the link's remote end B using the link ID (i.e., B's ID).

For a new broadcast link connecting multiple nodes with no link ID configured, when the parent receives a message containing the link attached to node X, it stores the link from X into its TED. It finds the link's remote end P using the link's local IP address with network mask. P is a Pseudo node identified by the local IP address of the designated node selected from the nodes connected to the link. After finding P, it associates the link attached to X with P and the

link connected to P with X. If P is not found, a new Pseudo node P
is created. The parent associates the link attached to X with P and
the link attached to P with X. This creates a bidirectional
connection between X and P.

The first node and second node from which the parent receives a
message containing the link is selected as the designed node and
backup designed node respectively. After the designed node is down,
the backup designed node becomes the designed node and the node other
than the designed node with the largest local IP address connecting
to the link is selected as the backup designed node.

When the old designed node is down and the backup designed node
becomes the new designed node, the parent updates its TED through
removing the link between each of nodes X and old P (the Pseudo node
corresponding to the old designed node) and adding a link between
each of nodes X (still connecting to the broadcast link) and new P
(the Pseudo node corresponding to the new designed node).

5.2.2.2. Detail Topology in a Domain

If a parent is in a same organization as its child, it stores into
its TED the detail information inside the child's domain when
receiving a message containing the information from the child;
otherwise, it discards the information and issues a warning
indicating that the information is sent to a wrong place.

6. Security Considerations

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

7. IANA Considerations

This section specifies requests for IANA allocation.

8. Acknowledgement

The authors would like to thank Jescia Chen, Adrian Farrel, and Eric
Wu for their valuable comments on this draft.

9. References

9.1. 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>.

   [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the
              Path Computation Element Architecture to the Determination
              of a Sequence of Domains in MPLS and GMPLS", RFC 6805,
              DOI 10.17487/RFC6805, November 2012,
              <https://www.rfc-editor.org/info/rfc6805>.

   [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>.

   [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering
              (TE) Extensions to OSPF Version 2", RFC 3630,
              DOI 10.17487/RFC3630, September 2003,
              <https://www.rfc-editor.org/info/rfc3630>.

9.2. Informative References

   [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic
              Engineering", RFC 5305, DOI 10.17487/RFC5305, October
              2008, <https://www.rfc-editor.org/info/rfc5305>.

   [RFC5392] Chen, M., Zhang, R., and X. Duan, "OSPF Extensions in
              Support of Inter-Autonomous System (AS) MPLS and GMPLS
              Traffic Engineering", RFC 5392, DOI 10.17487/RFC5392,
              January 2009, <https://www.rfc-editor.org/info/rfc5392>.

   [RFC5316] Chen, M., Zhang, R., and X. Duan, "ISIS Extensions in
              Support of Inter-Autonomous System (AS) MPLS and GMPLS
              Traffic Engineering", RFC 5316, DOI 10.17487/RFC5316,
              December 2008, <https://www.rfc-editor.org/info/rfc5316>.

   [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and
              S. Ray, "North-Bound Distribution of Link-State and
              Traffic Engineering (TE) Information Using BGP", RFC 7752,
              DOI 10.17487/RFC7752, March 2016,
              <https://www.rfc-editor.org/info/rfc7752>.

Appendix A. Message Encoding

A.1. Extension to Existing Message

   An existing Notification message may be extended to advertise the
   information about connections and access points. The following new
   Notification-type (NT) and Notification-value (NV) of a NOTIFICATION
   object in the message are defined:

   o NT=8 (TBD): Connections and Accesses

      * NV=1: Update Connections and Accesses. A NT=8 and NV=1
         indicates that the child sends its parent updates on the
         information about Connections and Accesses, and TLVs containing
         the information are in the object.

      * NV=2: Withdraw Connections and Accesses. A NT=8 and NV=2
         indicates that the child asks its parent to remove Connections
         and Accesses indicated by TLVs in the object.

A.1.1. TLVs

   Four TLVs are defined for connections and accesses. They are Inter-
   Domain link TLV, Router-ID TLV, Access IPv4/IPv6 Prefix TLV.

   The format of the Inter-Domain link TLV is illustrated 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 (tTBD1)          |            Length             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                 Inter-Domain Link Sub-TLVs                     |
   ~                                                               ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
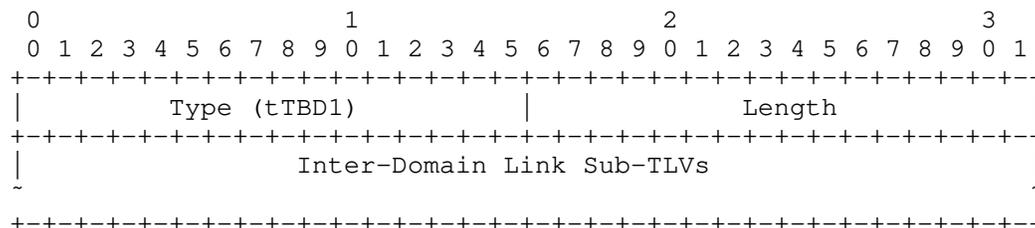
   An Inter-Domain link TLV describes a single inter-domain link. It
   comprises a number of inter-domain link sub-TLVs for the information
   described in section 4, which are the sub-TLVs defined in RFC 3630 or
   their equivalents except for the local IP address with mask length
   defined below.

   The format of the Router-ID TLV is shown below. Undefined flags MUST
   be set to zero. The ID indicates the ID of a router. For a router
   running OSPF, the ID may be the 32-bit OSPF router ID of the router.
   For a router running IS-IS, the ID may be the 48-bit IS-IS router ID

of the router. For a router not running OSPF or IS-IS, the ID may be
the 32-bit ID of the router 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 (tTBD2)        |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|B|E|I|     Flags              |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               +
|                          32-bit/48-bit ID                   ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
Flag B:     Set to 1 indicating ABR (B is for Border)
Flag E:     Set to 1 indicating ASBR (E is for External)
Flag I:     Set to 1 indicating ID of local router (I is for ID)
```
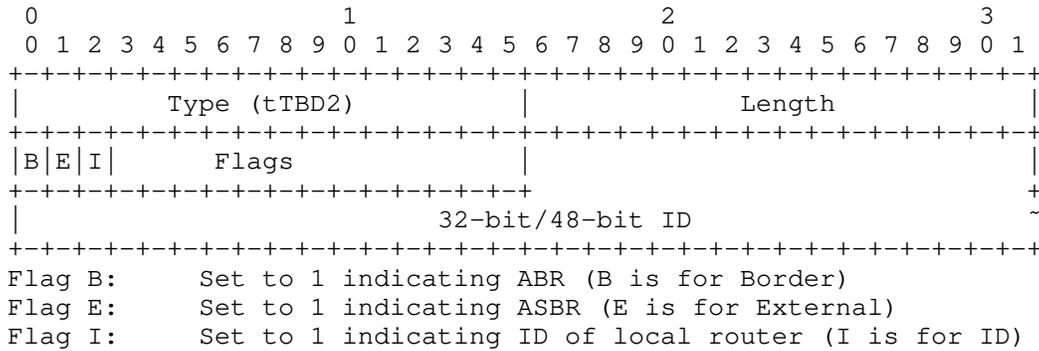
The format of the Access IPv4/IPv6 Prefix TLV is shown as follows.
The cost is the metric to the prefix. The Prefix Length indicates
the length of the prefix. The IPv4/IPv6 Prefix indicates an access
IPv4/IPv6 address prefix.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Type (tTBD3/tTDB4)     |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             cost                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Prefix Length | IPv4/IPv6 Prefix (variable)                 ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
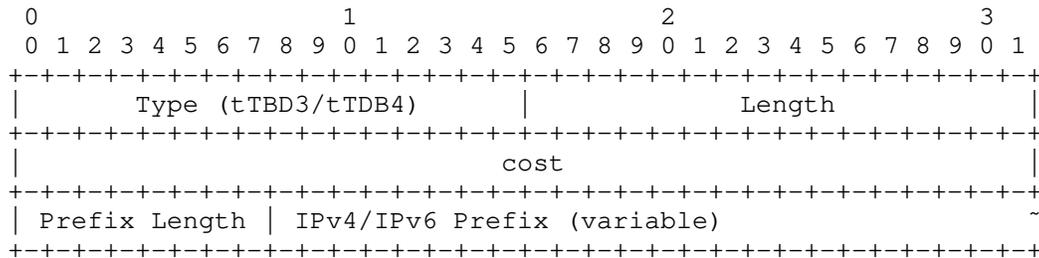
A.1.2. Sub-TLVs

The format of the Sub-TLV for a local IPv4/IPv6 address with mask
length is shown 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 (stTBD1/stTBD2)   |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        IPv4/IPv6 Address                     ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Mask Length  |
+-+-+-+-+-+-+-+-+
```
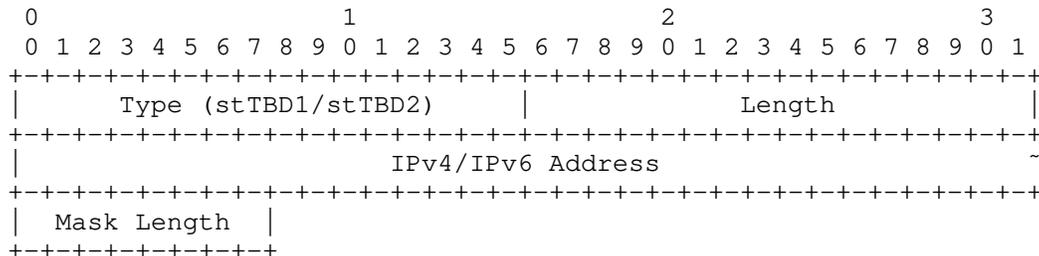
The IPv4/IPv6 Address indicates the local IPv4/IPv6 address of a
link. The Mask Length indicates the length of the IPv4/IPv6 address
mask.

A.2. New Message

A new message may be defined to advertise the connections and
accesses from a child to its parent. The format of the message
containing Connections and Access (AC for short) is as follows:

```
<AC Message> ::= <Common Header> <NRP>
                 <Connection-List> [<Access-List>]
where:
  <Connection-List> ::= <Connection> [<Connection-List>]
  <Connection> ::= [<Inter-Domain-Link> | <ABR>]
  <Access-List> ::= <Access-Address> [<Access-List>]
```

Where the value of the Message-Type in the Common Header indicates
the new message type. The exact value is to be assigned by IANA. A
new RP (NRP) object will be defined, which follows the Common Header.

A new flag W (Withdraw) in the NRP object is defined to indicate
whether the connections and access are withdrawn. When flag W is set
to one, the parent removes the connections and accesses contained in
the message after receiving it. When flag W is set to zero, the
parent adds/updates the connections and accesses in the message after
receiving it.

An alternative to flag W in the NRP object is a similar flag in each
CONNECTION and ACCESS object such as using one bit in Res flags for
flag W. For example, when the flag is set to one in the object, the
parent removes the connections and accesses in the object after
receiving it. When the flag is set to zero in the object, the parent
adds/updates the connections and accesses in the object after
receiving it.

In another option, one byte in a CONNECTION and ACCESS Object is
defined as flags field and one bit is used as flag W. The other
undefined bits in the flags field MUST be set to zero.
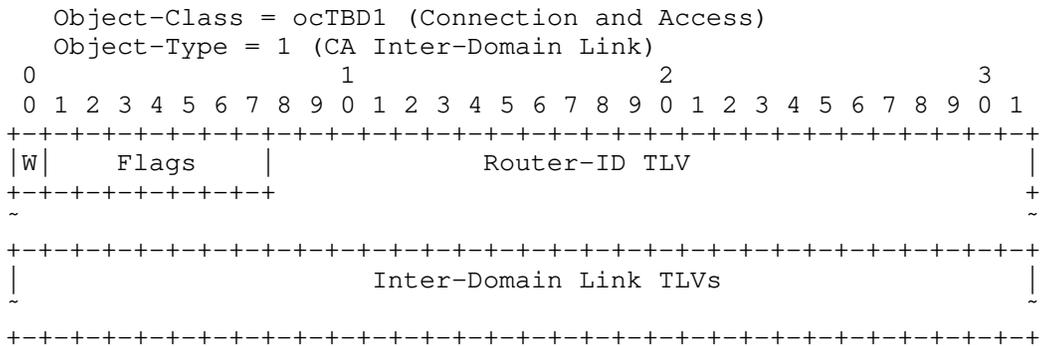
The objects in the new message are defined below.

A.2.1. CONNECTION and ACCESS Object

A new object, called CONNECTION and ACCESS Object (CA for short), is
defined. It has Object-Class ocTBD1. Four Object-Types are defined
under CA object:

o CA Inter-Domain Link: CA Object-Type is 1.

o CA ABR: CA Object-Type is 2.

o CA Access IPv4 Prefix: CA Object-Type is 3.

o CA Access IPv6 Prefix: CA Object-Type is 4.

Each of these objects are described below.

The format of Inter-Domain Link object body is as follows:

```
    Object-Class = ocTBD1 (Connection and Access)
    Object-Type = 1 (CA Inter-Domain Link)
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|W|   Flags    |                Router-ID TLV                  |
+-+-+-+-+-+-+-+-+                                               +
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Inter-Domain Link TLVs                    |
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Router-ID TLV indicates an ASBR in the domain, which is a local
end of inter-domain links. Each of the Inter-Domain Link TLVs
describes an inter-domain link and comprises a number of inter-domain
link Sub-TLVs. Flag W=1 indicates withdraw the links. W=0 indicates
new or changed links.

The format of ABR object body is illustrated below:

```
    Object-Class = ocTBD1 (Connection and Access)
    Object-Type = 2 (CA ABR)
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|W|   Flags    |                Router-ID TLVs                 |
+-+-+-+-+-+-+-+-+                                               +
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
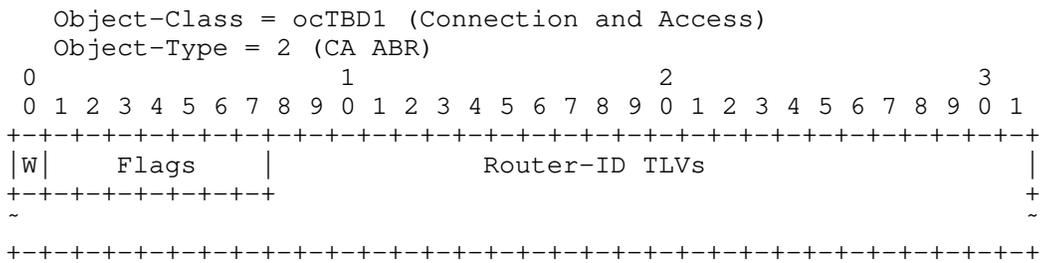
Each of the Router-ID TLVs indicates an ABR in the domain. Flag W=1
indicates withdraw the ABRs. W=0 indicates new ABRs.

The format of Access IPv4/IPv6 Prefix object body is as follows:

```
      Object-Class = ocTBD1 (Connection and Access)
      Object-Type = 3/4 (CA Access IPv4/IPv6 Prefix)
   0                   1                   2                   3
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |W|   Flags     |          Access IPv4/IPv6 Prefix TLVs         |
   +-+-+-+-+-+-+-+-+                                               +
   ~                                                               ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Each of the Access IPv4/IPv6 Prefix TLVs describes an access IPv4/
IPv6 address prefix in the domain, which is accessible to outside of
the domain. Flag W=1 indicates withdraw the address prefixes. W=0
indicates new address prefixes.

The TLVs in the objects are the same as those described above.

Authors' Addresses

Huaimo Chen
Futurewei
Boston, MA
USA

EMail: Huaimo.chen@futurewei.com

Mehmet Toy
Verizon
USA

EMail: mehmet.toy@verizon.com

Xufeng Liu
Volta Networks
McLean, VA
USA

EMail: xufeng.liu.ietf@gmail.com

   Lei Liu
   Fujitsu
   USA

   EMail: liulei.kddi@gmail.com

   Zhenqiang Li
   China Mobile
   No.32 Xuanwumenxi Ave., Xicheng District
   Beijing 100032
   P.R. China

   EMail: li_zhenqiang@hotmail.com

PCE Working Group   D. Dhody, Ed.
Internet-Draft   Huawei Technologies
Intended status: Standards Track   S. Sivabalan, Ed.
Expires: May 1, 2017   Cisco Systems, Inc.
                                                     S. Litkowski
                                                           Orange
                                                      J. Tantsura
                                                       Individual
                                                      J. Hardwick
                                              Metaswitch Networks
                                                October 28, 2016

         Path Computation Element communication Protocol extension for
                        associating Policies and LSPs
                   draft-dhody-pce-association-policy-00

   Abstract

   This document introduces a simple mechanism to associate policies to
   a group of Label Switched Paths (LSPs) via an extension to the Path
   Computation Element (PCE) Communication Protocol (PCEP).

   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 May 1, 2017.

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 communication
   Protocol (PCEP) which enables the communication between a Path
   Computation Client (PCC) and a Path Control Element (PCE), or between
   two PCEs based on the PCE architecture [RFC4655].

   PCEP Extensions for 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 network. Currently, the LSPs
   can either be signaled via RSVP-TE or can be segment routed as
   specified in [I-D.ietf-pce-segment-routing].

[I-D.ietf-pce-association-group] introduces a generic mechanism to
create a grouping of LSPs which can then be used to define
associations 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 or a stateless PCE.

This document specifies a PCEP extension to associate one or more
LSPs with policies using the generic association mechanism.

A PCEP speaker may want to influence the PCEP peer with respect to
path selection and other policies. This document describes a PCEP
extension to associate policies by creating Policy Association Group
(PAG) and encoding this association in PCEP messages. The
specification is applicable to both stateful and stateless PCEP
sessions.

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.

LSR: Label Switch Router.

MPLS: Multiprotocol Label Switching.

PAG: Policy Association Group.

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
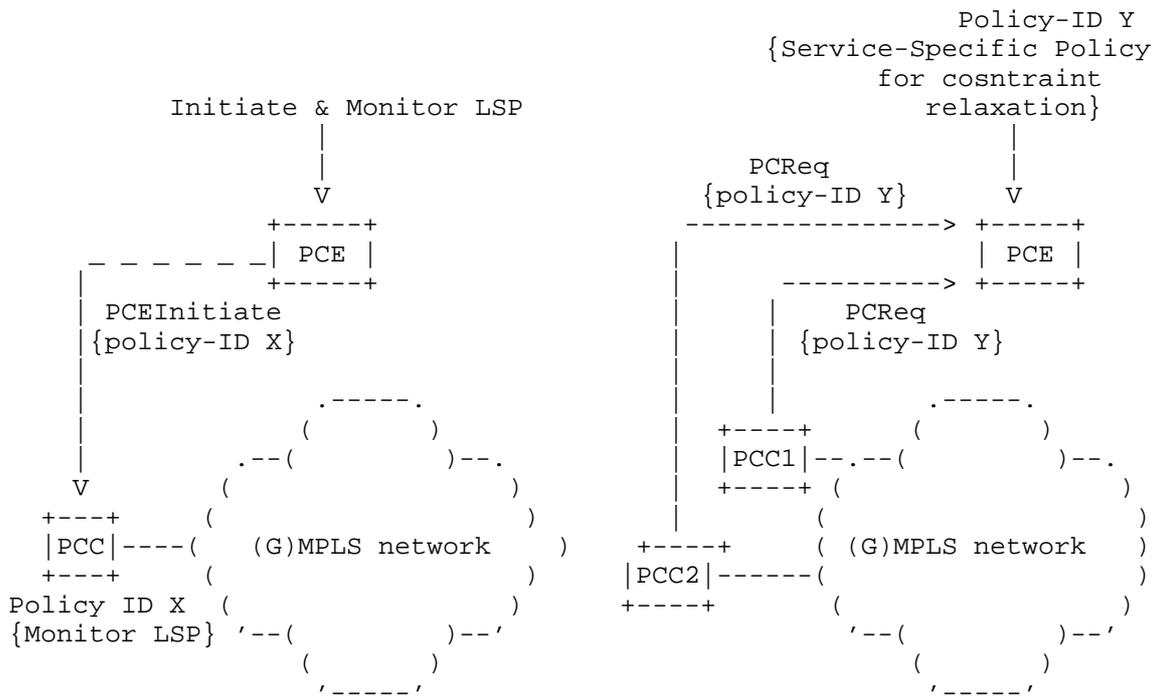   constraints.

PCEP: Path Computation Element Communication Protocol.

3. Motivation

Paths computed using PCE MAY be subjected to various policies on both
PCE and PCC. For example, in a centralized traffic engineering
scenario, network operators may instantiate LSPs and specifies
policies for traffic steering, path monitoring, etc., for some LSPs

via the stateful PCE. Similarly, a PCC may request a user- or
service-specific policy to be applied at the PCE, such as constraints
relaxation to meet optimal QoS and resiliency.

PCEP speaker can use the generic mechanism as per
[I-D.ietf-pce-association-group] to associate a set of LSPs with
policy, without the need to know the details of such policies, which
simplifies network operations, avoids frequent software upgrades, as
well provides an ability to introduce new policy faster.

```
                                                     Policy-ID Y
                                                 {Service-Specific Policy
                                                     for cosntraint
           Initiate & Monitor LSP                     relaxation}
                    |                                      |
                    |                      PCReq           |
                    V                    {policy-ID Y}     V
               +-----+                 ---------------> +-----+
      _ _ _ _ _| PCE |                 |                | PCE |
      |        +-----+                 |   ----------> +-----+
      | PCEInitiate                    |   |    PCReq
      |{policy-ID X}                   |   |  {policy-ID Y}
      |                                |   |
      |              .-----.           |   |         .-----.
      |            (       )           | +----+     (       )
      |          .--(       )--.       | |PCC1|--.--(       )--.
      V         (             )        | +----+ (             )
    +---+      (               )       |       (               )
    |PCC|----( (G)MPLS network  )   +----+    ( (G)MPLS network  )
    +---+      (               )   |PCC2|------(               )
 Policy ID X  (               )    +----+     (               )
 {Monitor LSP} '--(       )--'               '--(       )--'
                 (       )                       (       )
                  '-----'                         '-----'

    Case 1: Policy initiated by PCE    Case 2: Policy initiated by
            and enforced by PCC                PCC and enforced by
                                               PCE


        Sample use-cases for carrying policies over PCEP session
```

3.1. Policy based Constraints

   In the context of policy-enabled path computation [RFC5394], path
   computation policies may be applied at both a PCC and a PCE.
   Consider an Label Switch Router (LSR) with a policy enabled PCC, it
   receives a service request via signaling, including over a Network-
   Network Interface (NNI) or User Network Interface (UNI) reference
   point, or receives a configuration request over a management
   interface to establish a service. The PCC may also apply user- or
   service-specific policies to decide how the path selection process
   should be constrained, that is, which constraints, diversities,
   optimization criterion, and constraint relaxation strategies should
   be applied in order for the service LSP(s) to have a likelihood to be
   successfully established and provide necessary QoS and resilience
   against network failures. The user- or service-specific policies
   applied to PCC and are then passed to the PCE along with the Path
   computation request, in the form of constraints [RFC5394].

   PCEP speaker can use the generic mechanism as per
   [I-D.ietf-pce-association-group] to associate a set of LSPs with
   policy and its resulting path computation constraints. This
   simplified the path computation message exchanges.

4. Overview

   As per [I-D.ietf-pce-association-group], LSPs are associated with
   other LSPs with which they interact by adding them to a common
   association group. Grouping can also be used to define association
   between LSPs and policies associated to them. One new Association
   Type is defined in this document, based on the generic Association
   object -

   o Association type = TBD1 ("Policy Association Type") for Policy
      Association Group (PAG)

   A PAG can have one or more LSPs and its associated policy(s). The
   Association ID defined in [I-D.ietf-pce-association-group] is used to
   identify the PAG.

5. Policy Association Group

   Association groups and their memberships are defined using the
   ASSOCIATION object defined in [I-D.ietf-pce-association-group]. Two
   object types for IPv4 and IPv6 are defined. The ASSOCIATION object
   includes "Association type" indicating the type of the association
   group. This document add a new Association type -

   Association type = TBD1 ("Policy Association Type") for PAG.

PAG may carry optional TLVs including but not limited to -

o VENDOR-INFORMATION-TLV: Used to communicate arbitrary vendor
   specific behavioral information, described in [RFC7470].

6. Security Considerations

   This document defines one new type for association, which do not add
   any new security concerns beyond those discussed in [RFC5440],
   [I-D.ietf-pce-stateful-pce] and [I-D.ietf-pce-association-group] in
   itself.

   Some deployments may find policy associations and their implications
   as extra sensitive and thus should employ suitable PCEP security
   mechanisms like TCP-AO or [I-D.ietf-pce-pceps].

7. IANA Considerations

7.1. Association object Type Indicators

   This document defines the following new association type originally
   defined in [I-D.ietf-pce-association-group].

   Value   Name   Reference

   TBD1   Policy Association Type   [This I.D.]

8. Manageability Considerations

8.1. Control of Function and Policy

   An operator MUST BE allowed to configure the policy associations at
   PCEP peers and associate it with the LSPs.

8.2. Information and Data Models

   [RFC7420] describes the PCEP MIB, there are no new MIB Objects for
   this document.

8.3. 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].

8.4. Verify Correct Operations

   Mechanisms defined in this document do not imply any new operation
   verification requirements in addition to those already listed in
   [RFC5440].

8.5. Requirements On Other Protocols

   Mechanisms defined in this document do not imply any new requirements
   on other protocols.

8.6. 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].

9. Acknowledgments

   A special thanks to author of [I-D.ietf-pce-association-group], this
   document borrow some of the text from it.

10. References

10.1. 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>.

   [I-D.ietf-pce-association-group]
              Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H.,
              Zhang, X., and Y. Tanaka, "PCEP Extensions for
              Establishing Relationships Between Sets of LSPs", draft-
              ietf-pce-association-group-01 (work in progress), July
              2016.

   [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-16 (work in progress), September 2016.

10.2. 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>.

   [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>.

   [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>.

   [RFC7470] Zhang, F. and A. Farrel, "Conveying Vendor-Specific
              Constraints in the Path Computation Element Communication
              Protocol", RFC 7470, DOI 10.17487/RFC7470, March 2015,
              <http://www.rfc-editor.org/info/rfc7470>.

   [I-D.ietf-pce-pceps]
              Lopez, D., Dios, O., Wu, W., and D. Dhody, "Secure
              Transport for PCEP", draft-ietf-pce-pceps-10 (work in
              progress), July 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.

   [I-D.ietf-pce-segment-routing]
              Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E.,
              Raszuk, R., Lopez, V., Tantsura, J., Henderickx, W., and
              J. Hardwick, "PCEP Extensions for Segment Routing", draft-
              ietf-pce-segment-routing-08 (work in progress), October
              2016.

Appendix A. Contributor Addresses

    Qin Wu
    Huawei Technologies
    101 Software Avenue, Yuhua District
    Nanjing, Jiangsu 210012
    China

    EMail: sunseawq@huawei.com

    Clarence Filsfils
    Cisco Systems, Inc.
    Pegasus Parc
    De kleetlaan 6a, DIEGEM BRABANT 1831
    BELGIUM

    Email: cfilsfil@cisco.com

    Xian Zhang
    Huawei Technologies
    Bantian, Longgang District
    Shenzhen 518129
    P.R.China

    EMail: zhang.xian@huawei.com

    Udayasree Palle
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: udayasree.palle@huawei.com

Authors' Addresses

    Dhruv Dhody (editor)
    Huawei Technologies
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka 560066
    India

    EMail: dhruv.ietf@gmail.com

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

    EMail: msiva@cisco.com

    Stephane Litkowski
    Orange

    EMail: stephane.litkowski@orange.com

    Jeff Tantsura
    Individual

    EMail: jefftant.ietf@gmail.com

    Jonathan Hardwick
    Metaswitch Networks
    100 Church Street
    Enfield, Middlesex
    UK

    EMail: Jonathan.Hardwick@metaswitch.com

PCE Working Group   Young Lee
Internet Draft   Samsung
Intended Status: Standard track   Haomian Zheng
Expires: March 2021   Huawei
                                                 Daniele Ceccarelli
                                                          Ericsson
                                                          Wei Wang
                                      Beijing Univ. of Posts and Telecom
                                                        Peter Park
                                                               KT
                                                   Bin Young Yoon
                                                             ETRI

                                                September 9, 2020

      PCEP Extension for Distribution of Link-State and TE information for
                              Optical Networks

                         draft-lee-pce-pcep-ls-optical-10

Abstract

   In order to compute and provide optimal paths, Path Computation
   Elements (PCEs) require an accurate and timely Traffic Engineering
   Database (TED). Traditionally this Link State and TE information has
   been obtained from a link state routing protocol (supporting traffic
   engineering extensions).

   This document extends the Path Communication Element Communication
   Protocol (PCEP) with Link-State and TE information for optical
   networks.

Status of this Memo

   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."

   The list of current Internet-Drafts can be accessed at
   http://www.ietf.org/ietf/1id-abstracts.txt

   The list of Internet-Draft Shadow Directories can be accessed at
   http://www.ietf.org/shadow.html

   This Internet-Draft will expire on September 9, 2020.

Copyright Notice

   Copyright (c) 2020 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

   In Multiprotocol Label Switching (MPLS) and Generalized MPLS
   (GMPLS), a Traffic Engineering Database (TED) is used in computing
   paths for connection oriented packet services and for circuits. The
   TED contains all relevant information that a Path Computation
   Element (PCE) needs to perform its computations. It is important
   that the TED should be complete and accurate anytime so that the PCE
   can perform path computations.

   In MPLS and GMPLS networks, Interior Gateway routing Protocols
   (IGPs) have been used to create and maintain a copy of the TED at
   each node. One of the benefits of the PCE architecture [RFC4655] is
   the use of computationally more sophisticated path computation
   algorithms and the realization that these may need enhanced
   processing power not necessarily available at each node
   participating in an IGP.

   Section 4.3 of [RFC4655] describes the potential load of the TED on
   a network node and proposes an architecture where the TED is
   maintained by the PCE rather than the network nodes. However it does
   not describe how a PCE would obtain the information needed to
   populate its TED. PCE may construct its TED by participating in the
   IGP ([RFC3630] and [RFC5305] for MPLS-TE; [RFC4203] and [RFC5307]
   for GMPLS). An alternative is offered by [RFC7752].

   [RFC7399] touches upon this issue:"It has also been proposed that
   the PCE Communication Protocol (PCEP) [RFC5440] could be extended to
   serve as an information collection protocol to supply information
   from network devices to a PCE. The logic is that the network devices
   may already speak PCEP and so the protocol could easily be used to
   report details about the resources and state in the network,
   including the LSP state discussed in Sections 14 and 15."

   [RFC8231] describes a set of extensions to PCEP to provide stateful
   control. A stateful PCE has access to not only the information
   carried by the network's Interior Gateway Protocol (IGP), but also
   the set of active paths and their reserved resources for its
   computations. PCC can delegate the rights to modify the LSP
   parameters to an Active Stateful PCE. This requires PCE to quickly
   be updated on any changes in the Topology and TEDB, so that PCE can
   meet the need for updating LSPs effectively and in a timely manner.
   The fastest way for a PCE to be updated on TED changes is via a
   direct interface with each network node and with incremental update

from each network node. [S-PCE-GMPLS] specified the extensions to apply stateful PCE to GMPLS-based networks.

[RFC8281] describes the setup, maintenance and teardown of PCE-initiated LSPs under the stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic network that is centrally controlled and deployed. This model requires timely topology and TED update at the PCE.

[PCEP-LS] proposes alternative architecture approaches for learning and maintaining the Link State (and TE) information directly on a PCE from network nodes as an alternative to IGPs and BGP transport and investigate the impact from the PCE, routing protocol, and network node perspectives.

[RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs). Within the Hierarchical PCE (H-PCE) architecture [RFC6805], the Parent PCE (P-PCE) is used to compute a multi-domain path based on the domain connectivity information. A Child PCE (C-PCE) may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its domain topology information.

[RFC8751] presents general considerations for stateful PCE(s) in hierarchical PCE architecture. In particular, the behavior changes and additions to the existing stateful PCE mechanisms (including PCE-initiated LSP setup and active PCE usage) in the context of networks using the H-PCE architecture.

[PCEP-LS] describes a mechanism by which Link State and TE information can be collected from packet networks and shared with PCE with the PCEP itself. This is achieved using a new PCEP message format.

This draft describes an optical extension of [PCEP-LS] and explains how encodings suggested by [PCEP-LS] can be used in the optical network contexts.

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. Applicability

   There are three main applicability of this alternative proposed by
   this draft:

      - Case 1: Where there is IGP running in optical network but
        there is a need for a faster link-state and TE resource
        collection at the PCE directly from an optical node (PCC) via
        a PCC-PCE interface.

           o A PCE may receive an incremental update (as opposed to
             the entire TE information of the node/link).

        Note: A PCE may receive full information from IGP using
        existing mechanism. In some cases, the convergence of full
        link-state and TE resource information of the entire network
        may not be appropriate for certain applications. Incremental
        update capability will enhance the accuracy of the TE
        information at a given time.

      - Case 2: Where there is no IGP running in the optical network
        and there is a need for link-state and TE resource collections
        at the PCE directly from an optical node (PCC) via a PCC-PCE
        interface.

      - Case 3: Where there is a need for transporting abstract
        optical link-state and TE information from child PCE and to a
        parent PCE in H-PCE [RFC6805] and [RFC8751] as well as for
        Physical Network Controller (PNC) to Multi-Domain Service
        Coordinator (MDSC) in Abstraction and Control of TE Networks
        (ACTN) [RFC8453].

        Note: The applicability for Case 3 may arise as a consequence
        of Case 1 and Case 2. When TE information changes occur in the
        optical network, this may also affect abstracted TE
        information and thus needs to be updated to Parent PCE/MSDC
        from each child PCE/PNC.

3. Requirements for PCEP extension

   The key requirements associated with link-state (and TE)
   distribution are identified for PCEP and listed in Section 4 of
   [PCEP-LS]. These new functions required in PCEP to support
   distribution of link-state (and TE) information are described in
   Section 5 of [PCEP-LS]. Details of PCEP messages and related
   Objects/TLVs are specified in Sections 8 and 9 of [PCEP-LS]. The key
   requirements and new functions specified in [PCEP-LS] are equally
   applicable to optical networks.

   Besides the generic requirements specified in [PCEP-LS], optical
   specific features also need to be considered in this document. As
   connection-based network, there are specific parameters such as
   reachable table, optical latency, wavelength consistency and some
   other parameters that need to be included during the topology
   collection. Without these restrictions, the path computation may be
   inaccurate or infeasible for deployment, therefore these information
   MUST be included in the PCEP.

   The procedure on how the optical parameters are used is described in
   following sections.

      3.1. Reachable source-destination

   The reachable source-destination node pair indicates that there are
   a few number of OCh paths between two nodes. The reachability is
   restricted by impairment, wavelength consistency and so on. This
   information is necessary at PCE to promise the path computed between
   source node and destination node is reachable. In this scenario, the
   PCE should be responsible to compute how many OCh paths are
   available to set up connections between source and destination node.
   Moreover, if a set of optical wavelength is indicated in the path
   computation request, the PCE should also determine whether a
   wavelength of the set of preselected optical wavelength is available
   for the source-destination pair connection.

   To enable PCE to complete the above functions, the reachable
   relationship and OMS link information need to be reported to PCE.
   Once PCE detect that any wavelength is available, the corresponding
   OMS link should be included in a lambda plane. Then this link can be
   used for path computation in future.

   Moreover, in a hierarchical PCE architecture, the information above
   need to be reported from child PCE to parent PCE, who acts as a
   service coordinator.

3.2. Optical latency

It is a usual case that the PCC indicates the latency when requesting the path computation. In optical networks the latency is a very sensitive parameter and there is stricter requirement on latency. Given the maximal number of OCh paths between source-destination nodes, the PCE also need to determine how many OCh path satisfies the indicated latency threshold.

There is usually high-performance algorithm running on the PCE to guarantee the performance of the computed path. During the computation, the delay factor may be converted into a kind of link weight. After the algorithm provides a few candidate paths between the source and destination nodes, the PCE SHOULD be capable to selecting one shortest path by computing the total path propagation delay.

Optical PCEs are embedded with optimization algorithm, e.g., shortest path algorithm, to improve the performance of computed path.

4. PCEP-LS extension for Optical Networks

This section provides additional PCEP-LS extension necessary to support optical networks. All Objects/TLVs defined in [PCEP-LS] are applicable to optical networks.

4.1. Node Attributes TLV

Node-Attributed TLV is defined in Section 9.2.10.1 in [PCEP-LS] as follows. This TLV is applicable for LS Node Object-Type as defined in [PCEP-LS].

```
 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            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//             Node Attributes Sub-TLVs (variable)             //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
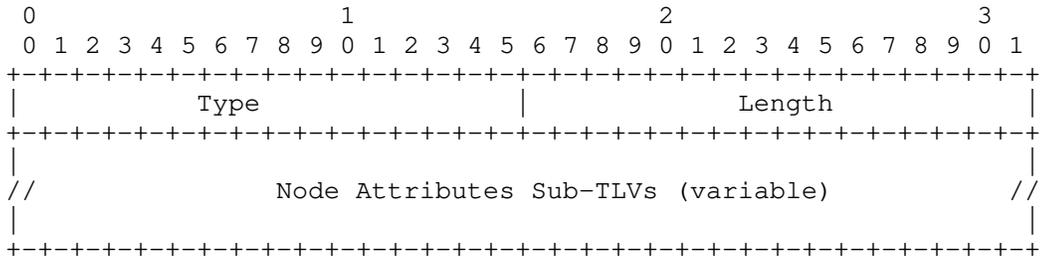
The following 'Node Attribute' sub-TLVs are valid for optical
networks:

```
+----------+------------------+-------------+---------+------------+
|  Sub-TLV | Description      | TLV/Sub-TLV | Length  |Reference   |
+----------+------------------+-------------+---------+------------+
|   TBD    | Connectivity     |    5/14     | variable|[RFC7579]   |
|          | Matrix           |             |         |[RFC7580]   |
|   TBD    | Resource Block   |    6/1      | variable|[RFC7688]   |
|          | Information      |             |         |            |
|   TBD    | Resource Block   |    6/2      | variable|[RFC7688]   |
|          | Accessibility    |             |         |            |
|   TBD    | Resource Block   |    6/3      | variable|[RFC7688]   |
|          | Wavelength Const |             |         |            |
|   TBD    | Resource Block   |    6/4      | variable|[RFC7688]   |
|          | Pool State       |             |         |            |
|   TBD    | Resource Block   |    6/5      | variable|[RFC7688]   |
|          | Shared Access    |             |         |            |
|          | Wavelength Avail.|             |         |            |
+-----------------------------------------------------=----------+
```

4.2. Link Attributes TLV

Link-Attributes TLV is defined in Section 9.2.10.2 in [PCEP-LS] as
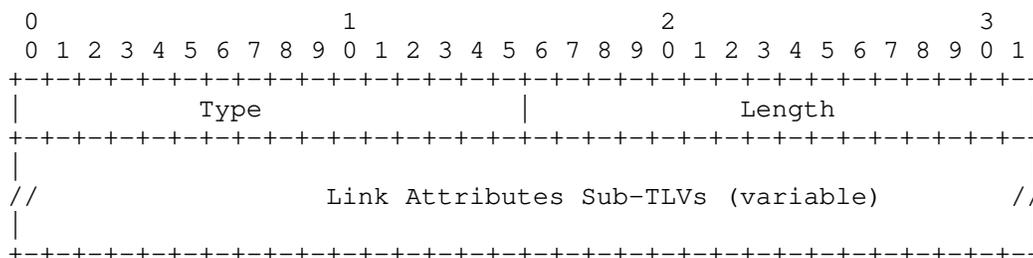follows. This TLV is applicable for LS Link Object-Type as defined
in [PCEP-LS].

```
    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             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   //               Link Attributes Sub-TLVs (variable)        //
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The following 'Link Attribute' sub-TLVs are valid for optical
networks:

```
+----------+----------------+-------------+--------+----------+
|  Sub-TLV | Description    | TLV/Sub-TLV | Length |Reference |
|          |                |             |        |          |
+----------+----------------+-------------+--------+----------+
|   TBD    | ISCD           |    15       |Variable|[RFC4203] |
|          |                |             |        |          |
|   TBD    | OTN-TDM SCSI   |    15/1,2   |Variable|[RFC4203] |
|          |                |             |        |[RFC7138] |
|   TBD    | WSON-LSC SCSI  |    15/1,2   |Variable|[RFC4203] |
|          |                |             |        |[RFC7688] |
|   TBD    | Flexi-grid SCSI|    15/1     |Variable|[RFC8363] |
|          |                |             |        |          |
|   TBD    | Port Label     |    34       |Variable|[RFC7579] |
|          | Restriction    |             |        |[RFC7580] |
|          |                |             |        |[RFC8363] |
+----------+----------------+-------------+--------+----------+
```

   4.3. PCEP-LS for Optical Network Extension

This section provides additional PCEP-LS extension necessary to
support optical networks parameters discussed in Sections 3.1 and
3.2.

The link state information collection is usually done before the
path computation processing. The procedure can be divided into 1)
link state collection by receiving the corresponding topology
information in a periodical style; 2) path computation on PCE,
triggered by receiving the path computation request message from
PCC, and completed by transmitting a path computation reply with the
path computation result, per [RFC4655].

For OTN networks, max bandwidth available may be per ODU 0/1/2/3
switching level or aggregated across all ODU switching levels (i.e.,
ODUj/k).

For WSON networks, RWA information collected from NEs would be
utilized to compute light paths. The list of information can be
found in [RFC7688]. More specifically, the max bandwidth available
may be per lambda/frequency level (OCh) or aggregated across all
lambda/frequency level. Per OCh level abstraction gives more
detailed data to the P-PCE at the expense of more information
processing. Either the OCh-level or the aggregated level abstraction
in the RWA constraint (i.e., wavelength continuity) needs to be
taken into account by the PCE during path computation. Resource
Block Accessibility (i.e., wavelength conversion information) in
[RFC7688] needs to be taken into account in order to guarantee the
reliability for optical path computation.

   [Editor's Note: Encoding will be provided in the revision, including
   RFC7688 RWA information]

5. Security Considerations

   This document extends PCEP for LS (and TE) distribution including a
   set of TLVs. Procedures and protocol extensions defined in this
   document do not affect the overall PCEP security model. See
   [RFC5440], [RFC8253]. The PCE implementation SHOULD provide
   mechanisms to prevent strains created by network flaps and amount of
   LS (and TE) information. Thus it is suggested that any mechanism
   used for securing the transmission of other PCEP message be applied
   here as well. As a general precaution, it is RECOMMENDED that these
   PCEP extensions only be activated on authenticated and encrypted
   sessions belonging to the same administrative authority.

6. IANA Considerations

   This document requests IANA actions to allocate code points for the
   protocol elements defined in this document.

      6.1. PCEP-LS Sub-TLV Type Indicators

   This document specifies a set of PCEP-LS Sub-TLVs. IANA is requested
   to create a "PCEP-LS Sub-TLV Types" sub-registry in the "PCEP TLV
   Type Indicators" for the sub-TLVs carried in the PCEP-LS TLV (Node
   Attributes TLV and Link Attributes TLV).

   +----------+-----------------+-------------+----------+
   | Sub-TLV | Description | Ref Sub-TLV | Reference|
   +----------+-----------------+-------------+----------+
TBD	Connectivity	5/14	[RFC7579]
	Matrix		[RFC7580]
TBD	Resource Block	6/1	[RFC7688]
	Information		
TBD	Resource Block	6/2	[RFC7688]
	Accessibility		
TBD	Resource Block	6/3	[RFC7688]
	Wavelength Const		
TBD	Resource Block	6/4	[RFC7688]
	Pool State		
TBD	Resource Block	6/5	[RFC7688]
	Shared Access		
	Wavelength Avail.		

TBD	ISCD	15	[RFC4203]
TBD	OTN-TDM SCSI	15/1,2	[RFC4203] [RFC7138]
TBD	WSON-LSC SCSI	15/1,2	[RFC4203] [RFC7688]
TBD	Flexi-grid SCSI	15/1	[RFC8363]
TBD	Port Label Restriction	34	[RFC7579] [RFC7580] [RFC8363]

7. References

   7.1. Normative References

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

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

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

[RFC7688] Lee, Y., Ed., and G. Bernstein, Ed., "GMPLS 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.

   7.2. Informative References

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

   [RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in
             Support of Generalized Multi-Protocol Label Switching
             (GMPLS)", RFC 4203, October 2005.

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

   [RFC5307] Kompella, K., Ed., and Y. Rekhter, Ed., "IS-IS Extensions
             in Support of Generalized Multi-Protocol Label Switching
             (GMPLS)", RFC 5307, October 2008.

   [RFC7752] Gredler, H., Medved, J., Previdi, S., Farrel, A., and
             S.Ray, "North-Bound Distribution of Link-State and TE
             information using BGP", RFC 7752, March 2016.

   [S-PCE-GMPLS] X. Zhang, et. al, "Path Computation Element (PCE)
             Protocol Extensions for Stateful PCE Usage in GMPLS-
             controlled Networks", draft-ietf-pce-pcep-stateful-pce-
             gmpls, work in progress.

   [RFC7399] A. Farrel and D. king, "Unanswered Questions in the Path
             Computation Element Architecture", RFC 7399, October 2015.

   [RFC8453] D.Ceccarelli, and Y. Lee (Editors), "Framework for
             Abstraction and Control of TE Networks", RFC453, August,
             2018.

   [RFC6805] A. Farrel and D. King, "The Application of the Path
             Computation Element Architecture to the Determination of a
             Sequence of Domains in MPLS and GMPLS", RFC 6805, November
             2012.

   [PCEP-LS] D. Dhody, Y. Lee and D. Ceccarelli "PCEP Extension for
             Distribution of Link-State and TE Information.", draft-
             dhodylee-pce-pcep-ls, work in progress, July, 2020

   [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP
             Extensions for Stateful PCE", RFC8231, September 2017.

   [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., Dhody, D.,
             "PCEPS: Usage of TLS to Provide a Secure Transport for the
             Path Computation Element Communication Protocol (PCEP)",
             RFC8253, October 2017.

   [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP
             Extensions for PCE-initiated LSP Setup in a Stateful PCE
             Model", RFC8281, December 2017.

   [RFC8751] D. Dhody, Y. Lee and D. Ceccarelli, "Hierarchical Stateful
             Path Computation Element (PCE)", RFC8751, March 2020.

   [RFC8363] X. Zhang, H. Zheng, R. Casellas, O. Gonzalez de Dios, D.
             Ceccarelli, "GMPLS OSPF Extensions in support of Flexi-
             grid DWDM networks", RFC8363, May 2018.

Appendix A. Contributor Addresses

   Dhruv Dhody
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India
   Email: dhruv.ietf@gmail.com

Author's Addresses

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

   Haomian Zheng
   Huawei Technologies Co., Ltd.
   Email: zhenghaomian@huawei.com

   Daniele Ceccarelli
   Ericsson
   Torshamnsgatan,48
   Stockholm
   Sweden

   EMail: daniele.ceccarelli@ericsson.com

   Wei Wang
   Beijing University of Posts and Telecom
   No. 10, Xitucheng Rd. Haidian District, Beijing 100876, P.R.China

   Email: weiw@bupt.edu.cn

     Peter Park
   KT
   Email: peter.park@kt.com

   Bin Yeong Yoon
   ETRI
   Email: byyun@etri.re.kr

              PCEP Link-State extensions for Segment Routing
                  draft-li-pce-pcep-ls-sr-extension-01

Abstract

   Segment Routing leverages source routing. A node steers a packet
   through a controlled set of instructions, called segments, by
   prepending the packet with an SR header. A segment can represent any
   instruction, topological or service-based. SR allows to enforce a
   flow through any topological path and service chain while maintaining
   per-flow state only at the ingress node of the SR domain.

   IGP protocols have been extended to advertise the segments. Because
   of IGP's propagation scope limitation, it is not suited for IGP to
   signal paths that span across AS borders. This document introduces
   extensions of PCEP-LS to solve the problem without the similar
   limitation.

Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [RFC2119].

Copyright Notice

Table of Contents

1. Introduction

   Segment Routing [I-D.ietf-spring-segment-routing] leverages source
   routing. A node steers a packet through a controlled set of
   instructions, called segments, by prepending the packet with an SR
   header. A segment can represent any instruction, topological or
   service-based. SR allows to enforce a flow through any topological
   path and service chain while maintaining per-flow state only at the
   ingress node of the SR domain.

   IGP protocols have been extended to advertise the segments. Because
   of IGP's propagation scope limitation, it is not suited for IGP to
   signal paths that span across AS borders.

In order to fulfill the need for applications that require visibility of SR paths across IGP areas or even across ASes, this document defines extensions for the mechanism introduced in [I-D.dhodylee-pce-pcep-ls] to propagate SR information in those scenarios that have no IGP SR extension or BGP-LS running.

2. PCEP extensions for Segment Routing

   PCEP-LS [I-D.dhodylee-pce-pcep-ls] introduces new message type and new object to accommodate link-state information in PCEP. This document defines new additional TLVs to map segment routing information. The value portion of these new TLVs can reuse the structure defined in [I-D.ietf-isis-segment-routing-extensions].

2.1. Node Attribute TLVs

   Some new optional, non-transitive node attribute TLVs are defined for carrying segment routing information and are listed below:

TLV Code Point	Description	Length	Value defined
TBD1	SID/Label Binding	variable	[ISIS-SR]#section2.4
TBD2	SR-Capabilities	variable	[ISIS-SR]#section3.1
TBD3	SR-Algorithm	variable	[ISIS-SR]#section3.2

   [ISIS-SR]: https://datatracker.ietf.org/doc/draft-ietf-isis-segment-routing-extensions/

   Table 1: Node Attribute TLVs

2.2. Link Attribute TLVs

   Some new optional, non-transitive link attribute TLVs are defined for carrying segment routing information and are listed below:

TLV Code Point	Description	Length	Value defined
TBD4	Adjacency Segment	variable	[ISIS-SR]#section2.2.1
TBD5	LAN Adjacency Segment	variable	[ISIS-SR]#section2.2.2
TBD6	Tunnel Segment	variable	

[ISIS-SR]: https://datatracker.ietf.org/doc/draft-ietf-isis-segment-routing-extensions/

Table 2: Link Attribute TLVs

2.3. Prefix Attribute TLVs

A new optional, non-transitive link attribute TLVs are defined for carrying segment routing information and are listed below:

```
+--------------+----------------------+----------+----------------------+
|TLV Code Point| Description          | Length   |    Value defined     |
+--------------+----------------------+----------+----------------------+
|     TBD7     | Prefix Segment       | variable | [ISIS-SR]#section2.1.2|
+--------------+----------------------+----------+----------------------+
```

[ISIS-SR]: https://datatracker.ietf.org/doc/draft-ietf-isis-segment-routing-extensions/

Table 3: Prefix Attribute TLVs

3. Operational Considerations

3.1. Segment Routing report

The procedure for segment routing information reporting from PCC to PCE will follow those defined in [I-D.dhodylee-pce-pcep-ls].

3.2. Tunnel Segment Identifier

Tunnel Segment introduced in [I-D.li-spring-tunnel-segment] is used to identify a tunnel of any kind in a segment routing network. It is originated by the tunnel ingress node and one SID is allocated and attached to it either locally or globally.

4. IANA Considerations

TBD.

5. Security Considerations

TBD.

6. Acknowledgements

TBD.

7. References

7.1. Normative References

   [I-D.dhodylee-pce-pcep-ls]
              Dhody, D., Lee, Y., and D. Ceccarelli, "PCEP Extension for
              Distribution of Link-State and TE Information.", draft-
              dhodylee-pce-pcep-ls-06 (work in progress), September
              2016.

   [I-D.ietf-isis-segment-routing-extensions]
              Previdi, S., Filsfils, C., Bashandy, A., Gredler, H.,
              Litkowski, S., Decraene, B., and j. jefftant@gmail.com,
              "IS-IS Extensions for Segment Routing", draft-ietf-isis-
              segment-routing-extensions-08 (work in progress), October
              2016.

   [I-D.ietf-spring-segment-routing]
              Filsfils, C., Previdi, S., Decraene, B., Litkowski, S.,
              and R. Shakir, "Segment Routing Architecture", draft-ietf-
              spring-segment-routing-09 (work in progress), July 2016.

   [I-D.li-spring-tunnel-segment]
              Li, Z. and N. Wu, "Tunnel Segment in Segment Routing",
              draft-li-spring-tunnel-segment-01 (work in progress),
              March 2016.

   [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>.

7.2. Informative References

   [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>.

Authors' Addresses

   Zhenbin Li
   Huawei Technologies
   Huawei Bld., No.156 Beiqing Rd.
   Beijing 100095
   China

   Email: lizhenbin@huawei.com

Xia Chen
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: jescia.chenxia@huawei.com

Nan Wu
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: eric.wu@huawei.com

       Path Computation Element communication Protocol extension for signaling
                         LSP diversity constraint
               draft-litkowski-pce-association-diversity-01

   Abstract

      This document introduces a simple mechanism to signal path diversity
      for a group of Label Switched Paths (LSPs) via an extension to the
      Path Computation Element Communication Protocol (PCEP).

   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 May 27, 2017.

   Copyright Notice

the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

1. Introduction

   [I-D.ietf-pce-association-group] introduces a generic mechanism to
   create a grouping of LSPs which can then be used to define
   associations between a set of LSPs and a set of attributes (such as
   configuration parameters or behaviours) and is equally applicable to
   the active and passive modes of a stateful PCE
   [I-D.ietf-pce-stateful-pce] or a stateless PCE [RFC5440].

   This document specifies a PCEP extension to signal that a particular
   group of LSPs should use diverse paths including the requested type
   of diversity.

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.

   LSR: Label Switch Router.

   MPLS: Multiprotocol Label Switching.

   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.

   PCEP: Path Computation Element Communication Protocol.

   SRLG: Shared Risk Link Group.

3. Motivation

   Path diversity is a very common use case today in IP/MPLS networks
   especially for layer 2 transport over MPLS. A customer may request
   that the operator provide two end-to-end disjoint paths across the
   IP/MPLS core. The customer may use those paths as primary/backup or
   active/active.

   Different level of disjointness may be offered:

   o Link disjointness: the paths of the associated LSPs should transit
      different links (but may use common nodes or different links that
      may have some shared fate).

   o Node disjointness: the paths of the associated LSPs should transit
      different nodes (but may use different links that may have some
      shared fate).

   o SRLG disjointness: the paths of the associated LSPs should transit
      different links that do not share fate (but may use common transit
      nodes).

   o Node+SRLG disjointness: the paths of the associated LSPs should
      transit different links that do not have any common shared fate
      and should transit different nodes.

   The associated LSPs may originate from the same or from different
   head-end(s) and may terminate at the same or different tail-end(s).

```
          _____
         /                                                 \
        /            +------+                                \
       |             | PCE  |                                |
       |             +------+                                |
       |                                                     |
       |      ************************>                      |
       |  +------+           10               +------+       |
   CE1 ****| PE 1 | ----- R1 ---- R2 ------- | PE 2 |**** CE2
       |  +------+         |        |          +------+       |
       |                   |        |                        |
       |                   |        |                        |
       |  +------+         |        |          +------+       |
   CE3 ****| PE 3 | ----- R3 ---- R4 ------- | PE 4 |**** CE4
       |  +------+ ************************> +------+         |
       |                                                     |
        \                                                   /
         _____/
```
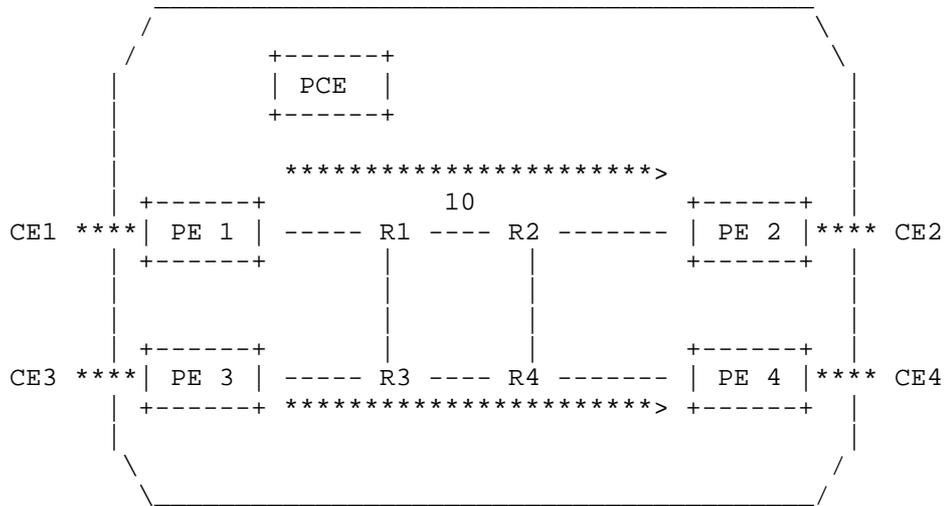
                Figure 1 - Disjoint paths with different head-ends

   In the figure above, the customer wants to have two disjoint paths
   between CE1/CE2 and CE3/CE4. From an IP/MPLS network point view, in
   this example, the CEs are connected to different PEs to maximize
   their disjointness. When LSPs originate from different head-ends,
   distributed computation of diverse paths can be difficult. Whereas,
   computation via a centralized PCE ensures path disjointness
   correctness and simplicity.

   Using PCEP, the PCC MUST indicate that disjoint path computation is
   required, such indication SHOULD include disjointness parameters such
   as the type of disjointness, the disjoint group-id, and any
   customization parameters according to local policy.

   The PCC can use the generic mechanism as per
   [I-D.ietf-pce-association-group] to associate a set of LSPs with a
   particular disjoint-group.

   The management of the disjoint group-ids will be a key point for the
   operator as the Association ID field is limited to 65535. The local
   configuration of IPv4/IPv6 association source, or Global Association
   Source/Extended Association ID should allow to overcome this
   limitation. For example, when a PCC or PCE initiates all the LSPs in
   a particular disjoint-group, it can set the IPv4/IPv6 association
   source can be set to one of its IP address. When disjoint LSPs are
   initiated from different head-ends, a unique association identifier
   SHOULD be used for those LSPs: this can be achieved by setting the

IPv4/IPv6 source address to a common value (zero value can be used)
as well as the Association ID.
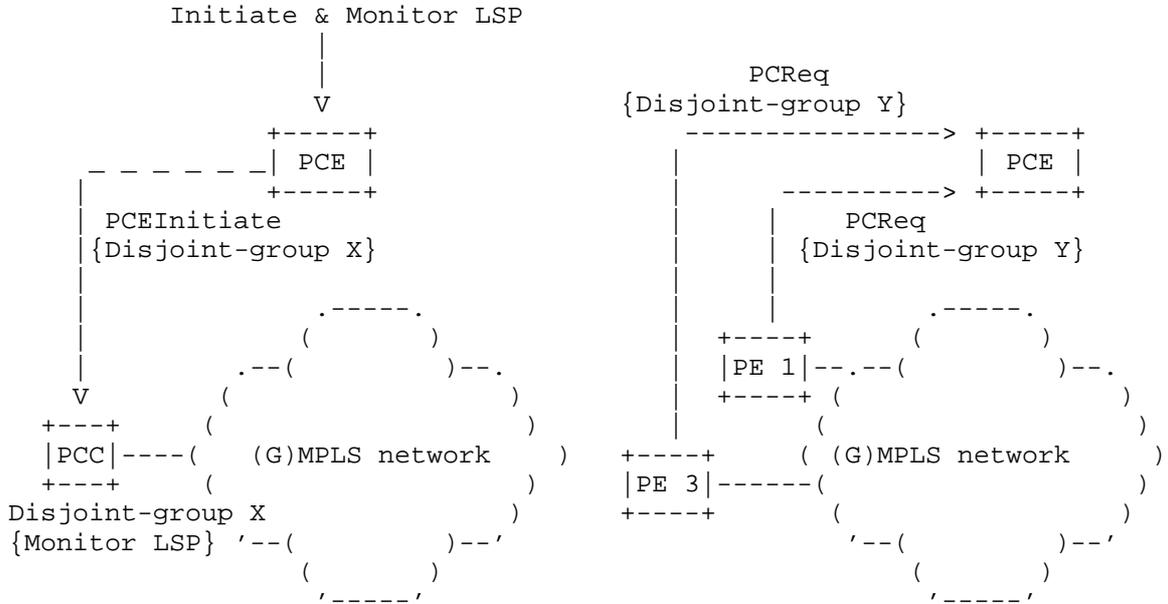
```
            Initiate & Monitor LSP
                    |
                    |                        PCReq
                    V              {Disjoint-group Y}
                 +-----+          ---------------> +-----+
 _ _ _ _ _ _ _ _| PCE |          |                | PCE |
|                +-----+          |   ----------> +-----+
|  PCEInitiate                    |   |   PCReq
| {Disjoint-group X}              |   | {Disjoint-group Y}
|                                 |   |
|                    .-----.      |   |       .-----.
|                   (       )     |   +----+ (       )
|              .--(           )--.| |PE 1|--.--(       )--.
 V            (                 ) | +----+ (               )
+---+        (                   )|        (               )
|PCC|----(    (G)MPLS network    )|+----+  ( (G)MPLS network    )
+---+        (                   )||PE 3|--------(               )
Disjoint-group X                 )|+----+      (               )
{Monitor LSP} '--(         )--'              '--(         )--'
                ( _____ )                        ( _____ )
                 '-----'                          '-----'

   Case 1: Disjointness initiated by   Case 2: Disjointness initiated by
        PCE and enforced by PCC             PCC and enforced by PCE
```

      Figure 2 - Sample use-cases for carrying disjoint-group over PCEP
                                   session

   Using the disjoint-group within a PCUpdate or PCInit may have two
   purposes:

   o Information: in case the PCE is performing the path computation,
      it may communicate to the PCC the locally used configured
      parameters in the attribute-list of the LSP.

   o Configuration: in case the PCC is performing the path computation
      but the PCE (without computation engine) is managing the LSP
      parameters, the PCE should add the disjoint-group within the
      PCUpdate message to communicate to the PCC the disjointness
      constraint.

4. Protocol extension

4.1. Association group

As per [I-D.ietf-pce-association-group], LSPs are associated with
other LSPs with which they interact by adding them to a common
association group. The Association ID will be used to identify the
disjoint group a set of LSPs belongs to. This document defines four
new Association types, based on the generic Association object -

   o Association type = TBD1 ("Disjointness Association Type") for link
      disjoint group.

   o Association type = TBD2 ("Disjointness Association Type") for node
      disjoint group.

   o Association type = TBD3 ("Disjointness Association Type") for srlg
      disjoint group.

   o Association type = TBD4 ("Disjointness Association Type") for
      node+srlg disjoint group.

A disjoint group can have two or more LSPs. But a PCE may be limited
in how many LSPs it can take into account when computing
disjointness: usually PCEs are able to compute a pair of disjoint
paths. If a PCE receives more LSPs in the group than it can handle
in its computation algorithm, it SHOULD apply disjointness
computation to only a subset of LSPs in the group. The subset of
disjoint LSPs will be decided by the implementation.

Local polices on the PCC or PCE MAY define the computational behavior
for the other LSPs in the group. For example, the PCE may provide no
path, a shortest path, or a constrained path based on relaxing
disjointness, etc.

Associating a particular LSP to multiple disjoint groups is
authorized from a protocol perspective, however there is no insurance
that the PCE will be able to compute properly the multi-disjointness
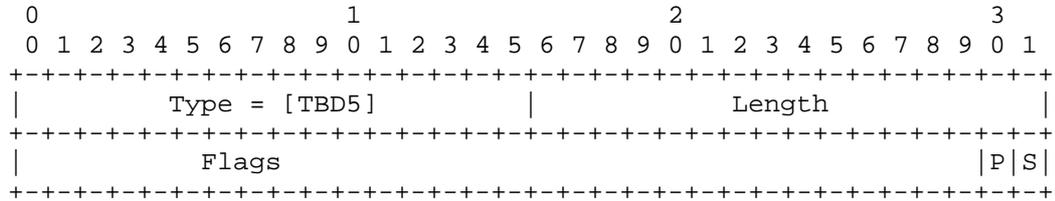constraint.

4.2. Optional TLVs

The disjoint group MAY carry some optional TLVs including but not
limited to:

   o VENDOR-INFORMATION-TLV: Used to communicate arbitrary vendor
      specific behavioral information, described in [RFC7150].

o DISJOINTNESS-INFORMATION-TLV: Used to communicate some
   disjointness specific parameters.

The DISJOINTNESS-INFORMATION-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 = [TBD5]        |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Flags                                          |P|S|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Flags:

* P: shortest path, this particular LSP in the group SHOULD use
   the shortest constrained path and others MAY use a non shortest
   constrained path. The shortest constrained path is the
   shortest path (from the requested metric point of view) that
   fills all the LSP constraints without taking into account the
   disjointness constraint. This means that an LSP with P flag
   set should be placed as if the disjointness constraint has not
   been configured, while the other LSP in the association with P
   flag unset should be placed by taking into account the
   disjointness constraint. Setting P flag changes the
   relationship between LSPs to a unidirectional relationship (LSP
   1 with P=0 depends of LSP 2 with P=1, but LSP 2 with P=1 does
   not depend of LSP 1 with P=0).

* S: strict disjointness, when set if disjoint paths cannot be
   found, PCE should return no path for LSPs that could not be be
   disjoint. When unset, PCE is allowed to relax disjointness by
   using either using a lower disjoint type (link instead of node)
   or relaxing disjointness constraint at all.

If a PCEP speaker receives a disjoint-group without DISJOINTNESS-
INFORMATION-TLV, it SHOULD use its locally configured parameters or
use the following default parameters:

o Strict disjointness is assumed.

o LSP can use a non shortest-path.

If a PCEP speaker receives two LSPs with the same disjoint-group but
with a different S flag value, it SHOULD apply a strict disjointness
path computation for this disjoint-group (it considers S flag set for
all LSPs).

5. Security Considerations

   This document defines one new type for association, which do not add
   any new security concerns beyond those discussed in [RFC5440],
   [I-D.ietf-pce-stateful-pce] and [I-D.ietf-pce-association-group] in
   itself.

6. IANA Considerations

6.1. Association object Type Indicators

   This document defines the following new association type originally
   defined in [I-D.ietf-pce-association-group].

   Value   Name   Reference

   TBD1   Link Disjoint-group
                               Association Type   [This I.D.]
   TBD2   Node Disjoint-group
                               Association Type   [This I.D.]
   TBD3   SRLG Disjoint-group
                               Association Type   [This I.D.]
   TBD4   Node+SRLG Disjoint-group
                               Association Type   [This I.D.]

   This document defines the following new PCEP TLV:

   Value   Name   Reference

   TBD5   DISJOINTNESS-INFORMATION-TLV   [This I.D.]

   IANA is requested to manage the space of flags carried in the
   DISJOINTNESS-INFORMATION TLV defined in this document, numbering them
   from 0 as the least significant bit.

   New bit numbers may be allocated in future.

   IANA is requested to allocate the following bit numbers in the
   DISJOINTNESS-INFORMATION-TLV flag space:

   Bit Number   Name   Reference
      0   Strict disjointness   [This I.D.]
      1   Shortest-path   [This I.D.]

7. Manageability Considerations

7.1. Control of Function and Policy

   An operator MUST be allowed to configure the disjointness
   associations and parameters at PCEP peers and associate it with the
   LSPs.

7.2. Information and Data Models

   [RFC7420] describes the PCEP MIB, there are no new MIB Objects for
   this document.

7.3. 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].

7.4. Verify Correct Operations

   Mechanisms defined in this document do not imply any new operation
   verification requirements in addition to those already listed in
   [RFC5440].

7.5. Requirements On Other Protocols

   Mechanisms defined in this document do not imply any new requirements
   on other protocols.

7.6. 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].

8. Acknowledgments

   A special thanks to author of [I-D.ietf-pce-association-group], this
   document borrow some of the text from it.

9. References

9.1. 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>.

   [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>.

   [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-association-group]
              Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H.,
              Zhang, X., and Y. Tanaka, "PCEP Extensions for
              Establishing Relationships Between Sets of LSPs", draft-
              ietf-pce-association-group-01 (work in progress), July
              2016.

   [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-17 (work in progress), November 2016.

9.2. Informative References

   [RFC7150] Zhang, F. and A. Farrel, "Conveying Vendor-Specific
              Constraints in the Path Computation Element Communication
              Protocol", RFC 7150, DOI 10.17487/RFC7150, March 2014,
              <http://www.rfc-editor.org/info/rfc7150>.

   [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-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.

Authors' Addresses

   Stephane Litkowski
   Orange

   EMail: stephane.litkowski@orange.com

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

EMail: msiva@cisco.com

Colby Barth
Juniper Networks

EMail: cbarth@juniper.net

Inter Stateful Path Computation Element (PCE) Communication Procedures.
                   draft-litkowski-pce-state-sync-09

Abstract

   The Path Computation Element Communication Protocol (PCEP) provides
   mechanisms for Path Computation Elements (PCEs) to perform path
   computation in response to a Path Computation Client (PCC) request.
   The Stateful PCE extensions allow stateful control of Multi-Protocol
   Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths
   (LSPs) using PCEP.

   A Path Computation Client (PCC) can synchronize an LSP state
   information to a Stateful Path Computation Element (PCE). The
   stateful PCE extension allows a redundancy scenario where a PCC can
   have redundant PCEP sessions towards multiple PCEs. In such a case,
   a PCC gives control of a LSP to only a single PCE, and only one PCE
   is responsible for path computation for this delegated LSP.

   There are some use cases, where an inter-PCE stateful communication
   can bring additional resiliency in the design, for instance when some
   PCC-PCE session fails. The inter-PCE stateful communication may also
   provide a faster update of the LSP states when such an event occurs.
   Finally, when, in a redundant PCE scenario, there is a need to
   compute a set of paths that are part of a group (so there is a
   dependency between the paths), there may be some cases where the
   computation of all paths in the group is not handled by the same PCE:
   this situation is called a split-brain. This split-brain scenario
   may lead to computation loops between PCEs or suboptimal path
   computation.

   This document describes the procedures to allow a stateful
   communication between PCEs for various use-cases and also the
   procedures to prevent computations loops.

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 May 5, 2021.

Copyright Notice

   Copyright (c) 2020 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
   (https://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 and Problem Statement

   The Path Computation Element communication Protocol (PCEP) [RFC5440]
   provides mechanisms for Path Computation Elements (PCEs) to perform
   path computations in response to Path Computation Clients' (PCCs)
   requests.

   A stateful PCE [RFC8231] is capable of considering, for the purposes
   of path computation, not only the network state in terms of links and
   nodes (referred to as the Traffic Engineering Database or TED) but
   also the status of active services (previously computed paths, and
   currently reserved resources, stored in the Label Switched Paths
   Database (LSP-DB).

   [RFC8051] describes general considerations for a stateful PCE
   deployment and examines its applicability and benefits, as well as
   its challenges and limitations through a number of use cases.

The examples in this section are for illustrative purpose to showcase
the need for inter-PCE stateful PCEP sessions.

1.1. Reporting LSP Changes

When using a stateful PCE ([RFC8231]), a PCC can synchronize an LSP
state information to the stateful PCE. If the PCC grants the control
of the LSP to the PCE (called delegation [RFC8231]), the PCE can
update the LSP parameters at any time.

In a multi PCE deployment (redundancy, loadbalancing...), with the
current specification defined in [RFC8231], when a PCE makes an
update, it is the PCC that is in charge of reporting the LSP status
to all PCEs with LSP parameter change which brings additional hops
and delays in notifying the overall network of the LSP parameter
change.

This delay may affect the reaction time of the other PCEs if they
need to take action after being notified of the LSP parameter change.

Apart from the synchronization from the PCC, it is also useful if
there is a synchronization mechanism between the stateful PCEs. As
stateful PCE make changes to its delegated LSPs, these changes
(pending LSPs and the sticky resources [RFC7399]) can be synchronized
immediately to the other PCEs.

```
              +----------+
              |   PCC1   |  LSP1
              +----------+
                 /     \
                /       \
    +---------+          +---------+
    |  PCE1   |          |  PCE2   |
    +---------+          +---------+
            \            /
             \          /
              +----------+
              |   PCC2   |  LSP2
              +----------+
```
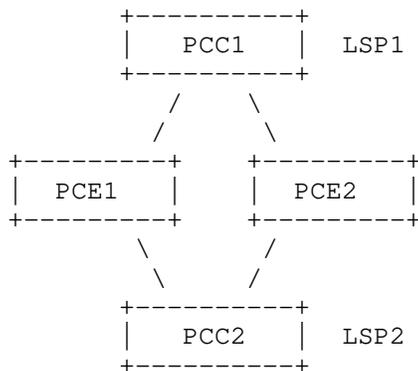
In the figure above, we consider a load-balanced PCE architecture, so
PCE1 is responsible to compute paths for PCC1 and PCE2 is responsible
to compute paths for PCC2. When PCE1 triggers an LSP update for
LSP1, it sends a PCUpd message to PCC1 containing the new parameters
for LSP1. PCC1 will take the parameters into account and will send a
PCRpt message to PCE1 and PCE2 reflecting the changes. PCE2 will so

be notified of the change only after receiving the PCRpt message from
PCC1.

Let's consider that the LSP1 parameters changed in such a way that
LSP1 will take over resources from LSP2 with a higher priority.
After receiving the report from PCC1, PCE2 will therefore try to find
a new path for LSP2. If we consider that there is a round trip delay
of about 150 milliseconds (ms) between the PCEs and PCC1 and a round
trip delay of 10 ms between the two PCEs if will take more than 150
ms for PCE2 to be notified of the change.

Adding a PCEP session between PCE1 and PCE2 may allow to reduce the
synchronization time, so PCE2 can react more quickly by taking the
pending LSPs and attached resources into account during path
computation and re-optimization.

1.2. Split-Brain

In a resiliency case, a PCC has redundant PCEP sessions towards
multiple PCEs. In such a case, a PCC gives control on an LSP to a
single PCE only, and only this PCE is responsible for the path
computation for the delegated LSP: the PCC achieves this by setting
the D flag only towards the active PCE [RFC8231] selected for
delegation. The election of the active PCE to delegate an LSP is
controlled by each PCC. The PCC usually elects the active PCE by a
local configured policy (by setting a priority). Upon PCEP session
failure, or active PCE failure, PCC may decide to elect a new active
PCE by sending new PCRpt message with D flag set to this new active
PCE. When the failed PCE or PCEP session comes back online, it will
be up to the implementation to do preemption. Doing preemption may
lead to some disruption on the existing path if path results from
both PCEs are not exactly the same. By considering a network with
multiple PCCs and implementing multiple stateful PCEs for redundancy
purpose, there is no guarantee that at any time all the PCCs delegate
their LSPs to the same PCE.

```
                  +----------+
                  |   PCC1   |  LSP1
                  +----------+
                    /      \
                   /        \
         +---------+      +---------+
         |  PCE1   |      |  PCE2   |
         +---------+      +---------+
               \            /
        *fail*   \        /
                  +----------+
                  |   PCC2   |  LSP2
                  +----------+
```
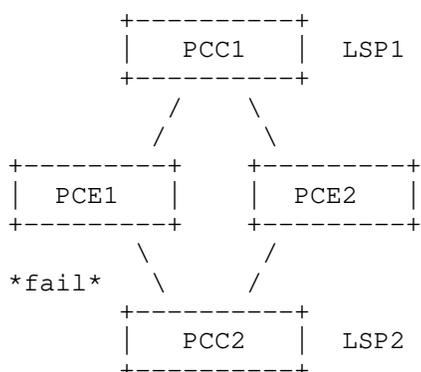
In the example above, we consider that by configuration, both PCCs will firstly delegate their LSPs to PCE1. So, PCE1 is responsible for computing a path for both LSP1 and LSP2. If the PCEP session between PCC2 and PCE1 fails, PCC2 will delegate LSP2 to PCE2. So PCE1 becomes responsible only for LSP1 path computation while PCE2 is responsible for the path computation of LSP2. When the PCC2-PCE1 session is back online, PCC2 will keep using PCE2 as active PCE (consider no preemption in this example). So the result is a permanent situation where each PCE is responsible for a subset of path computation.

This situation is called a split-brain scenario, as there are multiple computation brains running at the same time while a central computation unit was required in some deployments/use cases.

Further, there are use cases where a particular LSP path computation is linked to another LSP path computation: the most common use case is path disjointness (see [RFC8800]). The set of LSPs that are dependent to each other may start from a different head-end.

```
       _____
      /                                           \
     /         +------+              +------+       \
     |         | PCE1 |              | PCE2 |       |
     |         +------+              +------+       |
     |                                             |
     | +------+                       +------+     |
     | | PCC1 | --------------------> | PCC2 |     |
     | +------+                       +------+     |
     |                                             |
     |                                             |
     | +------+                       +------+     |
     | | PCC3 | --------------------> | PCC4 |     |
     | +------+                       +------+     |
     |                                             |
      \                                           /
       _____/


       _____
      /                                           \
     /         +------+              +------+       \
     |         | PCE1 |              | PCE2 |       |
     |         +------+              +------+       |
     |                                             |
     | +------+         10           +------+      |
     | | PCC1 | ----- R1 ---- R2 ------- | PCC2 |  |
     | +------+         |         |      +------+  |
     |                  |         |                |
     | +------+         |         |      +------+  |
     | | PCC3 | ----- R3 ---- R4 ------- | PCC4 |  |
     | +------+                          +------+  |
     |                                             |
      \                                           /
       _____/
```
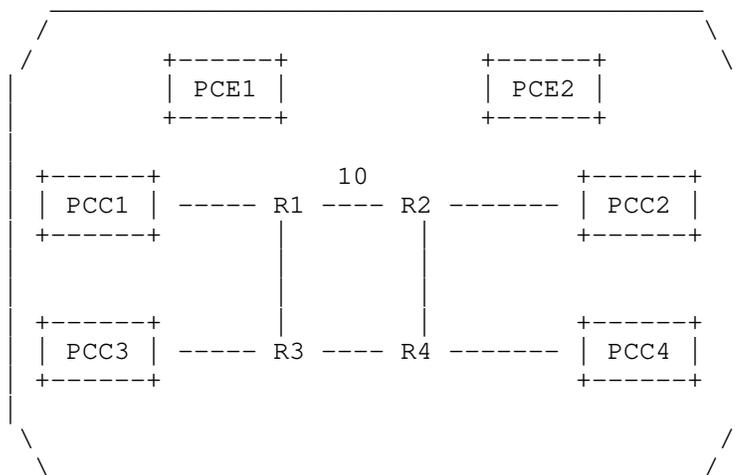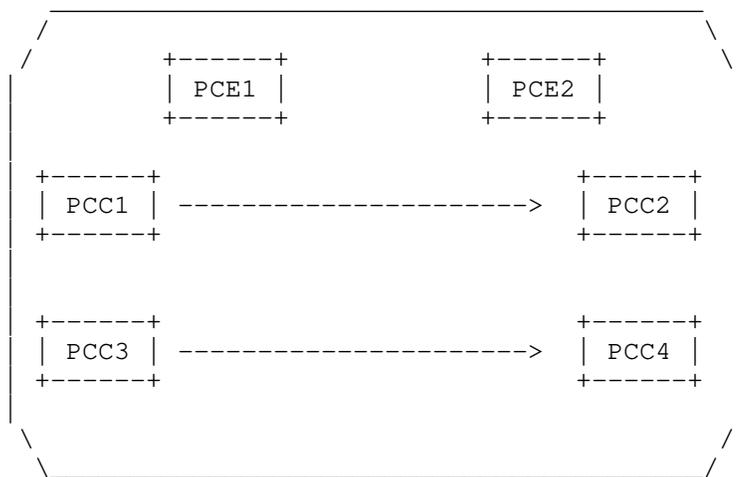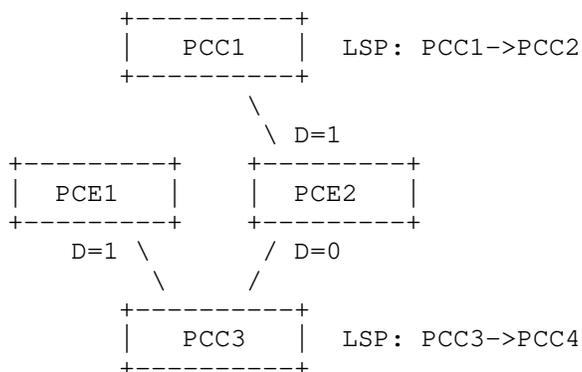
   In the figure above, the requirement is to create two link-disjoint
   LSPs: PCC1->PCC2 and PCC3->PCC4. In the topology, all links cost
   metric is set to 1 except for the link 'R1-R2' which has a metric of
   10. The PCEs are responsible for the path computation and PCE1 is
   the active primary PCE for all PCCs in the nominal case.

   Scenario 1:

In the normal case (PCE1 as active primary PCE), consider that
PCC1->PCC2 LSP is configured first with the link disjointness
constraint, PCE1 sends a PCUpd message to PCC1 with the ERO:
R1->R3->R4->R2->PCC2 (shortest path). PCC1 signals and installs the
path. When PCC3->PCC4 is configured, the PCEs already knows the path
of PCC1->PCC2 and can compute a link-disjoint path: the solution
requires to move PCC1->PCC2 onto a new path to let room for the new
LSP. PCE1 sends a PCUpd message to PCC1 with the new ERO:
R1->R2->PCC2 and a PCUpd to PCC3 with the following ERO:
R3->R4->PCC4. In the normal case, there is no issue for PCE1 to
compute a link-disjoint path.

Scenario 2:

Consider that PCC1 lost its PCEP session with PCE1 (all other PCEP
sessions are UP). PCC1 delegates its LSP to PCE2.


```
                +----------+
                |   PCC1   |   LSP: PCC1->PCC2
                +----------+
                      \
                       \ D=1
       +---------+    +---------+
       |  PCE1   |    |  PCE2   |
       +---------+    +---------+
          D=1 \         / D=0
               \       /
                +----------+
                |   PCC3   |   LSP: PCC3->PCC4
                +----------+
```


Consider that the PCC1->PCC2 LSP is configured first with the link
disjointness constraint, PCE2 (which is the new active primary PCE
for PCC1) sends a PCUpd message to PCC1 with the ERO:
R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured,
PCE1 is not aware of LSPs from PCC1 any more, so it cannot compute a
disjoint path for PCC3->PCC4 and will send a PCUpd message to PCC3
with the shortest path ERO: R3->R4->PCC4. When PCC3->PCC4 LSP will
be reported to PCE2 by PCC3, PCE2 will ensure disjointness
computation and will correctly move PCC1->PCC2 (as it owns delegation
for this LSP) on the following path: R1->R2->PCC2. With this
sequence of event and these PCEP sessions, disjointness is ensured.

Scenario 3:

```
           +----------+
           |   PCC1   |   LSP: PCC1->PCC2
           +----------+
              /     \
       D=1   /       \  D=0
    +---------+    +---------+
    |  PCE1   |    |  PCE2   |
    +---------+    +---------+
                   / D=1
                  /
           +----------+
           |   PCC3   |   LSP: PCC3->PCC4
           +----------+
```
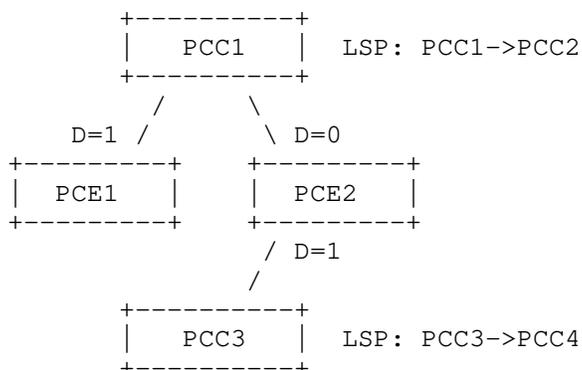
Consider the above PCEP sessions and the PCC1->PCC2 LSP is configured
first with the link disjointness constraint, PCE1 computes the
shortest path as it is the only LSP in the disjoint association group
that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When
PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this
LSP. The only solution found is to move PCC1->PCC2 LSP on another
path, but PCE2 cannot do it as it does not have delegation for this
LSP. In this set-up, PCEs are not able to find a disjoint path.

Scenario 4:

```
           +----------+
           |   PCC1   |   LSP: PCC1->PCC2
           +----------+
              /     \
       D=1   /       \  D=0
    +---------+    +---------+
    |  PCE1   |    |  PCE2   |
    +---------+    +---------+
       D=0  \        / D=1
             \      /
           +----------+
           |   PCC3   |   LSP: PCC3->PCC4
           +----------+
```
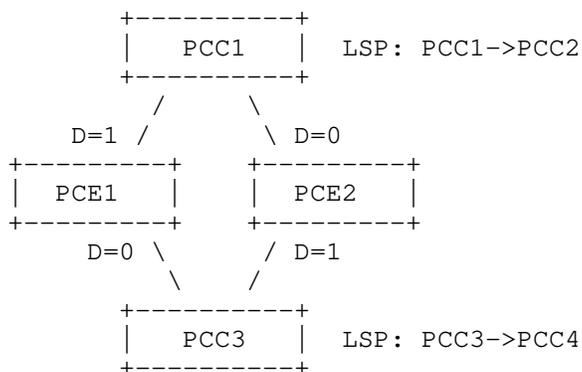
Consider the above PCEP sessions and that PCEs are configured to
fall-back to the shortest path if disjointness cannot be found as
described in [RFC8800]. The PCC1->PCC2 LSP is configured first, PCE1
computes the shortest path as it is the only LSP in the disjoint
association group that it is aware of: R1->R3->R4->R2->PCC2 (shortest
path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint
path for this LSP. The only solution found is to move PCC1->PCC2 LSP
on another path, but PCE2 cannot do it as it does not have delegation

for this LSP. PCE2 then provides the shortest path for PCC3->PCC4:
R3->R4->PCC4. When PCC3 receives the ERO, it reports it back to both
PCEs. When PCE1 becomes aware of the PCC3->PCC4 path, it recomputes
the constrained shortest path first (CSPF) algorithm and provides a
new path for PCC1->PCC2: R1->R2->PCC2. The new path is reported back
to all PCEs by PCC1. PCE2 recomputes also CSPF to take into account
the new reported path. The new computation does not lead to any path
update.

Scenario 5:

```
           _____
          /                                        \
         /       +------+          +------+         \
        |        | PCE1 |          | PCE2 |          |
        |        +------+          +------+          |
        |                                            |
        | +------+          100         +------+     |
        | |      |   _____  |      |     |
        | | PCC1 |   ----- R1 ----------| PCC2 |     |
        | +------+          |           +------+     |
        |                   |                        |
        |   6 |           2 |                   2 |  |
        |     |             |                     |  |
        | +------+          |           +------+     |
        | | PCC3 |  ----- R3 -----------| PCC4 |     |
        | +------+          10          +------+     |
        |                                            |
         \                                          /
          _____/
```
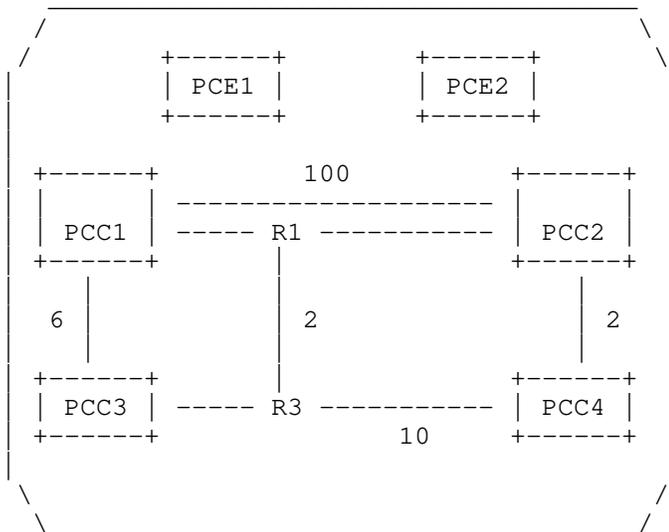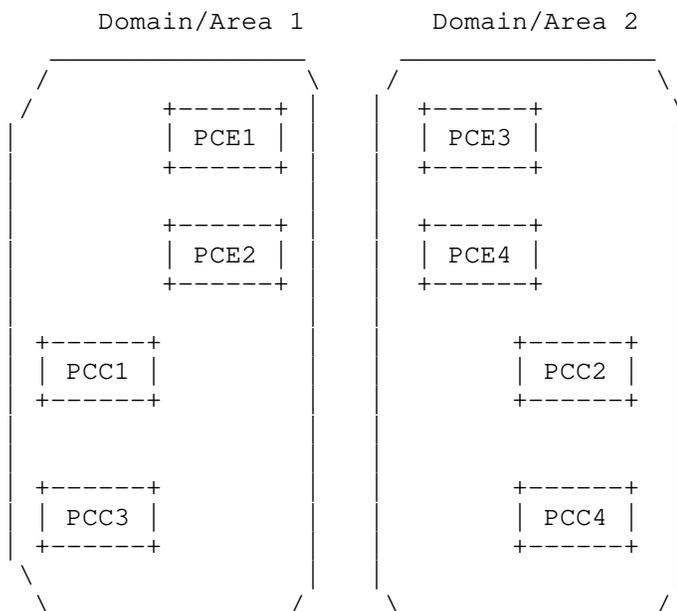
Now, consider a new network topology with the same PCEP sessions as
the previous example. Suppose that both LSPs are configured almost
at the same time. PCE1 will compute a path for PCC1->PCC2 while PCE2
will compute a path for PCC3->PCC4. As each PCE is not aware of the
path of the second LSP in the association group (not reported yet),
each PCE is computing the shortest path for the LSP. PCE1 computes
ERO: R1->PCC2 for PCC1->PCC2 and PCE2 computes ERO:
R3->R1->PCC2->PCC4 for PCC3->PCC4. When these shortest paths will be
reported to each PCE. Each PCE will recompute disjointness. PCE1
will provide a new path for PCC1->PCC2 with ERO: PCC1->PCC2. PCE2
will provide also a new path for PCC3->PCC4 with ERO: R3->PCC4. When
those new paths will be reported to both PCEs, this will trigger CSPF
again. PCE1 will provide a new more optimal path for PCC1->PCC2 with
ERO: R1->PCC2 and PCE2 will also provide a more optimal path for
PCC3->PCC4 with ERO: R3->R1->PCC2->PCC4. So we come back to the

initial state. When those paths will be reported to both PCEs, this
will trigger CSPF again. An infinite loop of CSPF computation is
then happening with a permanent flap of paths because of the split-
brain situation.

This permanent computation loop comes from the inconsistency between
the state of the LSPs as seen by each PCE due to the split-brain:
each PCE is trying to modify at the same time its delegated path
based on the last received path information which de facto
invalidates this received path information.

Scenario 6: multi-domain

```
          Domain/Area 1          Domain/Area 2
        _____       _____
      /                  \   /                  \
     /        +------+    | /      +------+       \
     |        | PCE1 |    | |      | PCE3 |       |
     |        +------+    | |      +------+       |
     |                    | |                     |
     |        +------+    | |      +------+       |
     |        | PCE2 |    | |      | PCE4 |       |
     |        +------+    | |      +------+       |
     |                    | |                     |
   +------+               | |            +------+ |
   | PCC1 |               | |            | PCC2 | |
   +------+               | |            +------+ |
     |                    | |                     |
     |                    | |                     |
   +------+               | |            +------+ |
   | PCC3 |               | |            | PCC4 | |
   +------+               | |            +------+ |
     \                    | |                     |
      _____/   _____/
```

In the example above, suppose that the disjoint LSPs from PCC1 to
PCC2 and from PCC4 to PCC3 are created. All the PCEs have the
knowledge of both domain topologies (e.g. using BGP-LS [RFC7752]).
For operation/management reasons, each domain uses its own group of
redundant PCEs. PCE1/PCE2 in domain 1 have PCEP sessions with PCC1
and PCC3 while PCE3/PCE4 in domain 2 have PCEP sessions with PCC2 and
PCC4. As PCE1/2 does not know about LSPs from PCC2/4 and PCE3/4 do
not know about LSPs from PCC1/3, there is no possibility to compute
the disjointness constraint. This scenario can also be seen as a
split-brain scenario. This multi-domain architecture (with multiple
groups of PCEs) can also be used in a single domain, where an

operator wants to limit the failure domain by creating multiple
groups of PCEs maintaining a subset of PCCs. As for the multi-domain
example, there will be no possibility to compute the disjoint path
starting from head-ends managed by different PCE groups.

In this document, we propose a solution that addresses the
possibility to compute LSP association based constraints (like
disjointness) in split-brain scenarios while preventing computation
loops.

1.3. Applicability to H-PCE

[RFC8751] describes general considerations and use cases for the
deployment of Stateful PCE(s) using the Hierarchical PCE [RFC6805]
architecture. In this architecture, there is a clear need to
communicate between a child stateful PCE and a parent stateful PCE.
The procedures and extensions as described in Section 3 are equally
applicable to the H-PCE scenario.

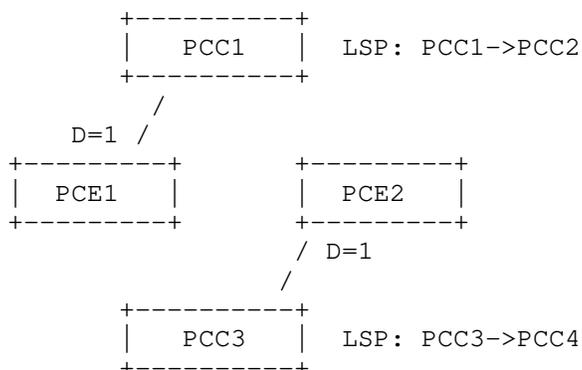## 2. Proposed solution

Our solution is based on :

o The creation of the inter-PCE stateful PCEP session with specific
   procedures.

o A Primary/Secondary relationship between PCEs.
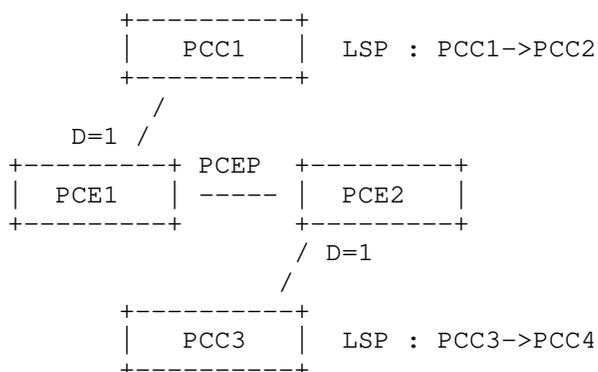
2.1. State-sync session

This document proposes to set-up a PCEP session between the stateful
PCEs. Creating such a session is already authorized by multiple
scenarios like the one described in [RFC4655] (multiple PCEs that are
handling part of the path computation) and [RFC6805] (hierarchical
PCE) but was only focused on the stateless PCEP sessions. As
stateful PCE brings additional features (LSP state synchronization,
path update, delegation, ...), thus some new behaviors need to be
defined.

This inter-PCE PCEP session will allow the exchange of LSP states
between PCEs that would help some scenarios where PCEP sessions are
lost between PCC and PCE. This inter-PCE PCEP session is henceforth
called a state-sync session.

For example, in the scenario below, there is no possibility to
compute disjointness as there is no PCE that is aware of both LSPs.

```
             +----------+
             |   PCC1   |  LSP: PCC1->PCC2
             +----------+
               /
        D=1  /
    +---------+        +---------+
    |  PCE1   |        |  PCE2   |
    +--------+         +--------+
                        / D=1
                       /
             +----------+
             |   PCC3   |  LSP: PCC3->PCC4
             +----------+
```

   If we add a state-sync session, PCE1 will be able to do state
   synchronization via PCRpt messages for its LSP to PCE2 and PCE2 will
   do the same. All the PCEs will be aware of all LSPs even if a
   PCC->PCE session is down. PCEs will then be able to compute disjoint
   paths.

```
             +----------+
             |   PCC1   |  LSP : PCC1->PCC2
             +----------+
               /
        D=1  /
    +---------+ PCEP  +---------+
    |  PCE1   | ----- |  PCE2   |
    +--------+         +--------+
                        / D=1
                       /
             +----------+
             |   PCC3   |  LSP : PCC3->PCC4
             +----------+
```

   The procedures associated with this state-sync session are defined in
   Section 3.

   By just adding this state-sync session, it does not ensure that a
   path with LSP association based constraints can always be computed
   and does not prevent the computation loop, but it increases
   resiliency and ensures that PCEs will have the state information for
   all LSPs. Also, this session will allow for a PCE to update the
   other PCEs providing a faster synchronization mechanism than relying
   on PCCs only.

2.2. Primary/Secondary relationship between PCE

   As seen in Section 1, performing a path computation in a split-brain
   scenario (multiple PCEs responsible for computation) may provide a
   non-optimal LSP placement, no path, or computation loops. To provide
   the best efficiency, an LSP association constraint-based computation
   requires that a single PCE performs the path computation for all LSPs
   in the association group. Note that, it could be all LSPs belonging
   to a particular association group, or all LSPs from a particular PCC,
   or all LSPs in the network that need to be delegated to a single PCE
   based on the deployment scenarios.

   This document proposes to add a priority mechanism between PCEs to
   elect a single computing PCE. Using this priority mechanism, PCEs
   can agree on the PCE that will be responsible for the computation for
   a particular association group, or set of LSPs. The priority could
   be set per association, per PCC, or for all LSPs. How this priority
   is set or advertised is out of the scope of this document. The rest
   of the text considers the association group as an example.

   When a single PCE is performing the computation for a particular
   association group, no computation loop can happen and an optimal
   placement will be provided. The other PCEs will only act as state
   collectors and forwarders.

   In the scenario described in Section 2.1, PCE1 and PCE2 will decide
   that PCE1 will be responsible for the path computation of both LSPs.
   If we first configure PCC1->PCC2, PCE1 computes the shortest path at
   it is the only LSP in the disjoint-group that it is aware of:
   R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured,
   PCE2 will not perform computation even if it has delegation but
   forwards the delegation via PCRpt message to PCE1 through the state-
   sync session. PCE1 will then perform disjointness computation and
   will move PCC1->PCC2 onto R1->R2->PCC2 and provides an ERO to PCE2
   for PCC3->PCC4: R3->R4->PCC4. The PCE2 will further update the PCC3
   with the new path.

3. Procedures and Protocol Extensions

3.1. Opening a state-sync session

3.1.1. Capability Advertisement

   A PCE indicates its support of state-sync procedures during the PCEP
   Initialization phase [RFC5440]. The OPEN object in the Open message
   MUST contains the "Stateful PCE Capability" TLV defined in [RFC8231].
   A new P (INTER-PCE-CAPABILITY) flag is introduced to indicate the
   support of state-sync.

This document adds a new bit in the Flags field with :

   P (INTER-PCE-CAPABILITY - 1 bit): If set to 1 by a PCEP Speaker,
   the PCEP speaker indicates that the session MUST follow the state-
   sync procedures as described in this document. The P bit MUST be
   set by both speakers: if a PCEP Speaker receives a STATEFUL-PCE-
   CAPABILITY TLV with P=0 while it advertised P=1 or if both set P
   flag to 0, the session SHOULD be set-up but the state-sync
   procedures MUST NOT be applied on this session.

The U flag [RFC8231] MUST be set when sending the STATEFUL-PCE-
CAPABILITY TLV with the P flag set. In case the U flag is not set
along with the P flag, the state sync capability is not enabled and
it is considered as if the P flag is not set. The S flag MAY be set
if optimized synchronization is required as per [RFC8232].

3.2. State synchronization

When the state sync capability has been negotiated between stateful
PCEs, each PCEP speaker will behave as a PCE and as a PCC at the same
time regarding the state synchronization as defined in [RFC8231].
This means that each PCEP Speaker:

o MUST send a PCRpt message towards its neighbor with S flag set for
   each LSP in its LSP database learned from a PCC. (PCC role)

o MUST send the End Of Synchronization Marker towards its neighbor
   when all LSPs have been reported. (PCC role)

o MUST wait for the LSP synchronization from its neighbor to end
   (receiving an End Of Synchronization Marker). (PCE role)

The process of synchronization runs in parallel on each PCE (with no
defined order).

The optimized state synchronization procedures MAY be used, as
defined in [RFC8232].

When a PCEP Speaker sends a PCRpt on a state-sync session, it MUST
add the SPEAKER-IDENTITY-TLV (defined in [RFC8232]) in the LSP
Object, the value used will refer to the 'owner' PCC of the LSP. If
a PCEP Speaker receives a PCRpt on a state-sync session without this
TLV, it MUST discard the PCRpt message and it MUST reply with a PCErr
message using error-type=6 (Mandatory Object missing) and error-
value=TBD1 (SPEAKER-IDENTITY-TLV missing).

3.3. Incremental updates and report forwarding rules

   During the life of an LSP, its state may change (path, constraints,
   operational state...) and a PCC will advertise a new PCRpt to the PCE
   for each such change.

   When propagating LSP state changes from a PCE to other PCEs, it is
   mandatory to ensure that a PCE always uses the freshest state coming
   from the PCC.

   When a PCE receives a new PCRpt from a PCC with the LSP-DB-VERSION,
   the PCE MUST forward the PCRpt to all its state-sync sessions and
   MUST add the appropriate SPEAKER-IDENTITY-TLV in the PCRpt. In
   addition, it MUST add a new ORIGINAL-LSP-DB-VERSION TLV (described
   below). The ORIGINAL-LSP-DB-VERSION contains the LSP-DB-VERSION
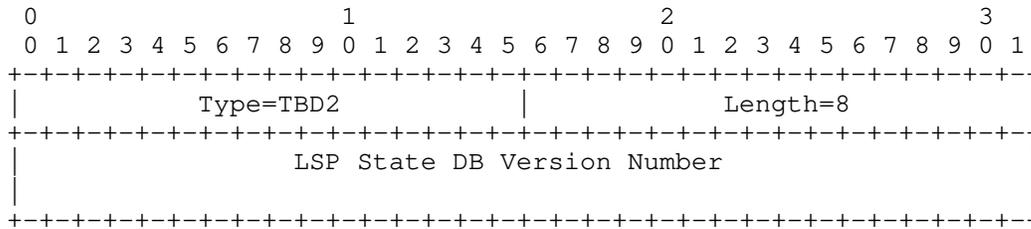   coming from the PCC.

   When a PCE receives a new PCRpt from a PCC without the LSP-DB-
   VERSION, it SHOULD NOT forward the PCRpt on any state-sync sessions
   and log such an event on the first occurrence.

   When a PCE receives a new PCRpt from a PCC with the R flag (Remove)
   set and an LSP-DB-VERSION TLV, the PCE MUST forward the PCRpt to all
   its state-sync sessions keeping the R flag set (Remove) and MUST add
   the appropriate SPEAKER-IDENTITY-TLV and ORIGINAL-LSP-DB-VERSION TLV
   in the PCRpt message.

   When a PCE receives a PCRpt from a state-sync session, it MUST NOT
   forward the PCRpt to other state-sync sessions. This helps to
   prevent message loops between PCEs. As a consequence, a full mesh of
   PCEP sessions between PCEs are REQUIRED.

   When a PCRpt is forwarded, all the original objects and values are
   kept. As an example, the PLSP-ID used in the forwarded PCRpt will be
   the same as the original one used by the PCC. Thus an implementation
   supporting this document MUST consider SPEAKER-IDENTITY-TLV and PLSP-
   ID together to uniquely identify an LSP on the state-sync session.

   The ORIGINAL-LSP-DB-VERSION TLV is encoded as follows and MUST always
   contain the LSP-DB-VERSION received from the owner PCC of the 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type=TBD2           |            Length=8           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  LSP State DB Version Number                  |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Using the ORIGINAL-LSP-DB-VERSION TLV allows a PCE to keep using
   optimized synchronization ([RFC8232]) with another PCE. In such a
   case, the PCE will send a PCRpt to another PCE with both ORIGINAL-
   LSP-DB-VERSION TLV and LSP-DB-VERSION TLV. The ORIGINAL-LSP-DB-
   VERSION TLV will contain the version number as allocated by the PCC
   while the LSP-DB-VERSION will contain the version number allocated by
   the local PCE.

3.4. Maintaining LSP states from different sources

   When a PCE receives a PCRpt on a state-sync session, it stores the
   LSP information into the original PCC address context (as the LSP
   belongs to the PCC). A PCE SHOULD maintain a single state for a
   particular LSP and SHOULD maintain the list of sources it learned a
   particular state from.

   A PCEP speaker may receive state information for a particular LSP
   from different sources: the PCC that owns the LSP (through a regular
   PCEP session) and some PCEs (through PCEP state-sync sessions). A
   PCEP speaker MUST always keep the freshest state in its LSP database,
   overriding the previously received information.

   A PCE, receiving a PCRpt from a PCC, updates the state of the LSP in
   its LSP-DB with the newly received information. When receiving a
   PCRpt from another PCE, a PCE SHOULD update the LSP state only if the
   ORIGINAL-LSP-DB-VERSION present in the PCRpt is greater than the
   current ORIGINAL-LSP-DB-VERSION of the stored LSP state. This
   ensures that a PCE never tries to update its stored LSP state with an
   old information. Each time a PCE updates an LSP state in its LSP-DB,
   it SHOULD reset the source list associated with the LSP state and
   SHOULD add the source speaker address in the source list. When a PCE
   receives a PCRpt which has an ORIGINAL-LSP-DB-VERSION (if coming from
   a PCE) or an LSP-DB-VERSION (if coming from the PCC) equals to the
   current ORIGINAL-LSP-DB-VERSION of the stored LSP state, it SHOULD
   add the source speaker address in the source list.

   When a PCE receives a PCRpt requesting an LSP deletion from a
   particular source, it SHOULD remove this particular source from the
   list of sources associated with this LSP.

When the list of sources becomes empty for a particular LSP, the LSP state MUST be removed. This means that all the sources must send a PCRpt with R=1 for an LSP to make the PCE remove the LSP state.

3.5. Computation priority between PCEs and sub-delegation

A computation priority is necessary to ensure that a single PCE will perform the computation for all the LSPs in an association group: this will allow for a more optimized LSP placement and will prevent computation loops.

All PCEs in the network that are handling LSPs in a common LSP association group SHOULD be aware of each other including the computation priority of each PCE. Note that there is no need for PCC to be aware of this. The computation priority is a number and the PCE having the highest priority SHOULD be responsible for the computation. If several PCEs have the same priority value, their IP address SHOULD be used as a tie-breaker to provide a rank: the highest IP address has more priority. How PCEs are aware of the priority of each other is out of the scope of this document, but as example learning priorities could be done through PCE discovery or local configuration.

The definition of the priority could be global so the highest priority PCE will handle all path computations or more granular, so a PCE may have the highest priority for only a subset of LSPs or association-groups.

A PCEP Speaker receiving a PCRpt from a PCC with the D flag set that does not have the highest computation priority, SHOULD forward the PCRpt on all state-sync sessions (as per Section 3.3) and SHOULD set D flag on the state-sync session towards the highest priority PCE, D flag will be unset to all other state-sync sessions. This behavior is similar to the delegation behavior handled at the PCC side and is called a sub-delegation (the PCE sub-delegates the control of the LSP to another PCE). When a PCEP Speaker sub-delegates an LSP to another PCE, it loose control of the LSP and cannot update it anymore by its own decision. When a PCE receives a PCRpt with D flag set on a state-sync session, as a regular PCE, it is granted control over the LSP.

If the highest priority PCE is failing or if the state-sync session between the local PCE and the highest priority PCE failed, the local PCE MAY decide to delegate the LSP to the next highest priority PCE or to take back control of the LSP. It is a local policy decision.

When a PCE has the delegation for an LSP and needs to update this LSP, it MUST send a PCUpd message to all state-sync sessions and to

the PCC session on which it received the delegation. The D-Flag
would be unset in the PCUpd for state-sync sessions whereas the
D-Flag would be set for the PCC. In the case of sub-delegation, the
computing PCE will send the PCUpd only to all state-sync sessions (as
it has no direct delegation from a PCC). The D-Flag would be set for
the state-sync session to the PCE that sub-delegated this LSP and the
D-Flag would be unset for other state-sync sessions.

The PCUpd sent over a state-sync session MUST contain the SPEAKER-
IDENTITY-TLV in the LSP Object (the value used must identify the
target PCC). The PLSP-ID used is the original PLSP-ID generated by
the PCC and learned from the forwarded PCRpt. If a PCE receives a
PCUpd on a state-sync session without the SPEAKER-IDENTITY-TLV, it
MUST discard the PCUpd and MUST reply with a PCErr message using
error-type=6 (Mandatory Object missing) and error-value=TBD1
(SPEAKER-IDENTITY-TLV missing).

When a PCE receives a valid PCUpd on a state-sync session, it SHOULD
forward the PCUpd to the appropriate PCC (identified based on the
SPEAKER-IDENTITY-TLV value) that delegated the LSP originally and
SHOULD remove the SPEAKER-IDENTITY-TLV from the LSP Object. The
acknowledgment of the PCUpd is done through a cascaded mechanism, and
the PCC is the only responsible for triggering the acknowledgment:
when the PCC receives the PCUpd from the local PCE, it acknowledges
it with a PCRpt as per [RFC8231]. When receiving the new PCRpt from
the PCC, the local PCE uses the defined forwarding rules on the
state-sync session so the acknowledgment is relayed to the computing
PCE.

A PCE SHOULD NOT compute a path using an association-group constraint
if it has delegation for only a subset of LSPs in the group. In this
case, an implementation MAY use a local policy on PCE to decide if
PCE does not compute path at all for this set of LSP or if it can
compute a path by relaxing the association-group constraint.

3.6. Passive stateful procedures

   In the passive stateful PCE architecture, the PCC is responsible for
   triggering a path computation request using a PCReq message to its
   PCE. Similarly to PCRpt Message, which remains unchanged for passive
   mode, if a PCE receives a PCReq for an LSP and if this PCE finds that
   it does not have the highest computation priority of this LSP, or
   groups..., it MUST forward the PCReq message to the highest priority
   PCE over the state-sync session. When the highest priority PCE
   receives the PCReq, it computes the path and generates a PCRep
   message towards the PCE that made the request. This PCE will then
   forward the PCRep to the requesting PCC. The handling of LSP object

and the SPEAKER-IDENTITY-TLV in PCReq and PCRep is similar to PCRpt/
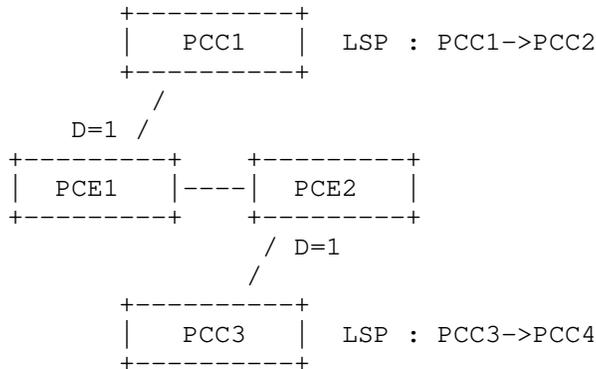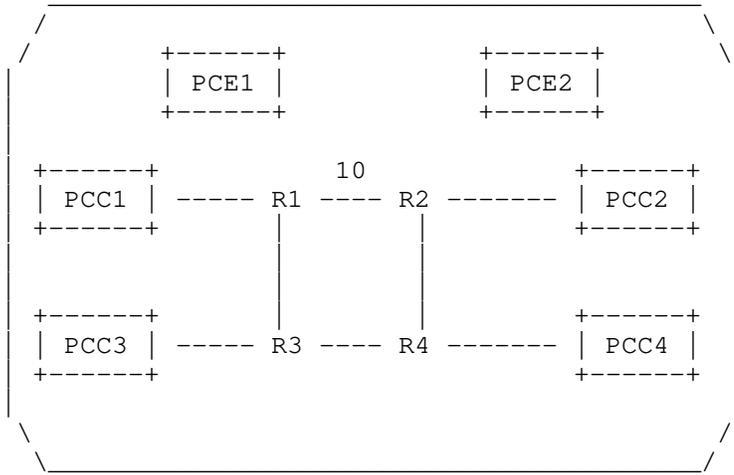PCUpd messages.

3.7. PCE initiation procedures

   TBD

4. Examples

   The examples in this section are for illustrative purpose to show how
   the behavior of the state sync inter-PCE sessions.

4.1. Example 1

```
          _____
         /                                         \
        /       +------+                 +------+    \
        |       | PCE1 |                 | PCE2 |     |
        |       +------+                 +------+     |
        |                                            |
        | +------+          10           +------+    |
        | | PCC1 | ----- R1 ---- R2 ------- | PCC2 | |
        | +------+          |       |       +------+ |
        |                   |       |                |
        |                   |       |                |
        | +------+          |       |       +------+ |
        | | PCC3 | ----- R3 ---- R4 ------- | PCC4 | |
        | +------+                         +------+  |
        |                                            |
         \                                          /
          _____/
```

```
        +----------+
        |   PCC1   |  LSP : PCC1->PCC2
        +----------+
              /
       D=1   /
   +---------+     +---------+
   |  PCE1   |----|  PCE2   |
   +---------+     +---------+
                    / D=1
                   /
        +----------+
        |   PCC3   |  LSP : PCC3->PCC4
        +----------+
```

PCE1 computation priority 100
PCE2 computation priority 200

Consider the PCEP sessions as shown above, where computation priority
is global for all the LSPs and link disjoint between LSPs PCC1->PCC2
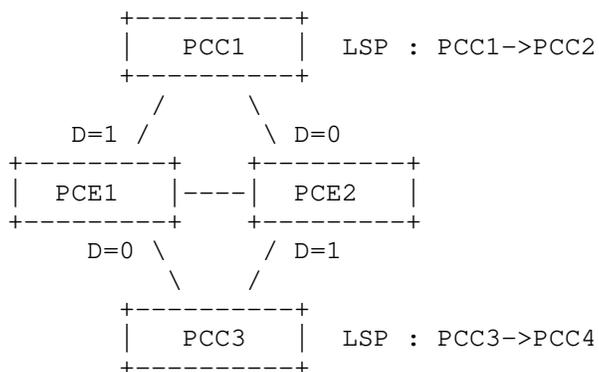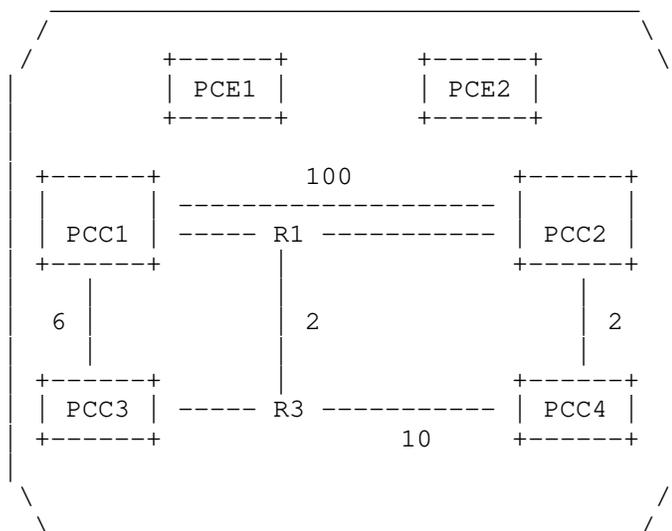and PCC3->PCC4 is required.

Consider the PCC1->PCC2 is configured first and PCC1 delegates the
LSP to PCE1, but as PCE1 does not have the highest computation
priority, it sub-delegates the LSP to PCE2 by sending a PCRpt with
D=1 and including the SPEAKER-IDENTITY-TLV over the state-sync
session. PCE2 receives the PCRpt and as it has delegation for this
LSP, it computes the shortest path: R1->R3->R4->R2->PCC2. It then
sends a PCUpd to PCE1 (including the SPEAKER-IDENTITY-TLV) with the
computed ERO. PCE1 forwards the PCUpd to PCC1 (removing the SPEAKER-

IDENTITY-TLV). PCC1 acknowledges the PCUpd by a PCRpt to PCE1. PCE1
forwards the PCRpt to PCE2.

When PCC3->PCC4 is configured, PCC3 delegates the LSP to PCE2, PCE2
can compute a disjoint path as it has knowledge of both LSPs and has
delegation also for both. The only solution found is to move
PCC1->PCC2 LSP on another path, PCE2 can move PCC1->PCC2 as it has
sub-delegation for it. It creates a new PCUpd with a new ERO:
R1->R2-PCC2 towards PCE1 which forwards to PCC1. PCE2 sends a PCUpd
to PCC3 with the path: R3->R4->PCC4.

In this set-up, PCEs are able to find a disjoint path while without
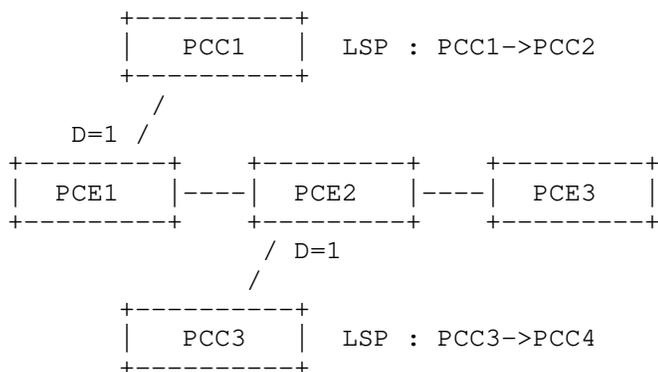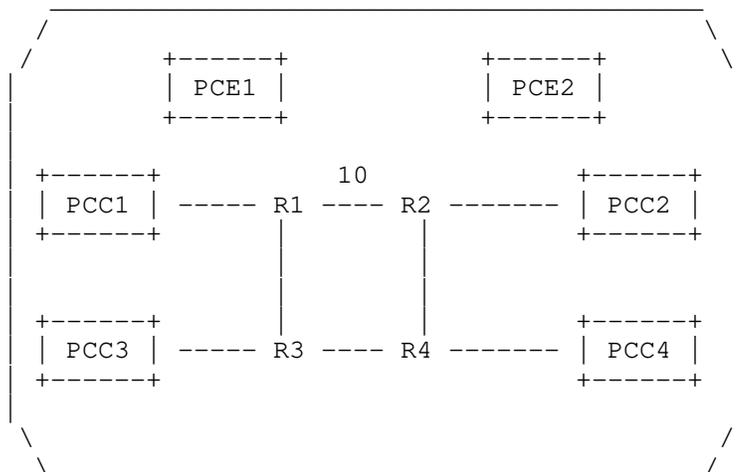state-sync and computation priority they could not.

4.2. Example 2

```
       _____
      /                                        \
     /        +------+            +------+       \
     |        | PCE1 |            | PCE2 |       |
     |        +------+            +------+       |
     |                                          |
     | +------+          100          +------+  |
     | |      | ------------------- |      |  |
     | | PCC1 | ----- R1 ---------- | PCC2 |  |
     | +------+                     +------+  |
     |                                        |
     |   6 |         | 2              | 2     |
     |     |         |                |       |
     | +------+      |            +------+    |
     | | PCC3 | ----- R3 ---------- | PCC4 |  |
     | +------+              10     +------+  |
     |  \                                  /  |
      \  _____/  /
       _____/
```

```
        +----------+
        |   PCC1   |  LSP : PCC1->PCC2
        +----------+
         /      \
    D=1 /        \ D=0
  +---------+    +---------+
  | PCE1    |----| PCE2    |
  +---------+    +---------+
    D=0 \         / D=1
         \       /
        +----------+
        |   PCC3   |  LSP : PCC3->PCC4
        +----------+
```

    PCE1 computation priority 200
    PCE2 computation priority 100

    In this example, suppose both LSPs are configured almost at the same
    time. PCE1 sub-delegates PCC1->PCC2 to PCE2 while PCE2 keeps
    delegation for PCC3->PCC4, PCE2 computes a path for PCC1->PCC2 and
    PCC3->PCC4 and can achieve disjointness computation easily. No
    computation loop happens in this case.

4.3. Example 3

```
        _____
      /                                          \
     /        +------+              +------+       \
    |         | PCE1 |              | PCE2 |        |
    |         +------+              +------+        |
    |                                              |
    |  +------+         10          +------+        |
    |  | PCC1 | ----- R1 ---- R2 ------- | PCC2 |   |
    |  +------+         |        |        +------+  |
    |                   |        |                  |
    |                   |        |                  |
    |  +------+         |        |        +------+  |
    |  | PCC3 | ----- R3 ---- R4 ------- | PCC4 |   |
    |  +------+                          +------+   |
    |                                              |
     \                                            /
      _____/


           +----------+
           |   PCC1   |  LSP : PCC1->PCC2
           +----------+
                /
       D=1 /
  +---------+    +---------+    +---------+
  |  PCE1   |----|  PCE2   |----|  PCE3   |
  +---------+    +---------+    +---------+
                 / D=1
                /
           +----------+
           |   PCC3   |  LSP : PCC3->PCC4
           +----------+
```

    PCE1 computation priority 100
    PCE2 computation priority 200
    PCE3 computation priority 300

    With the PCEP sessions as shown above, consider the need to have link
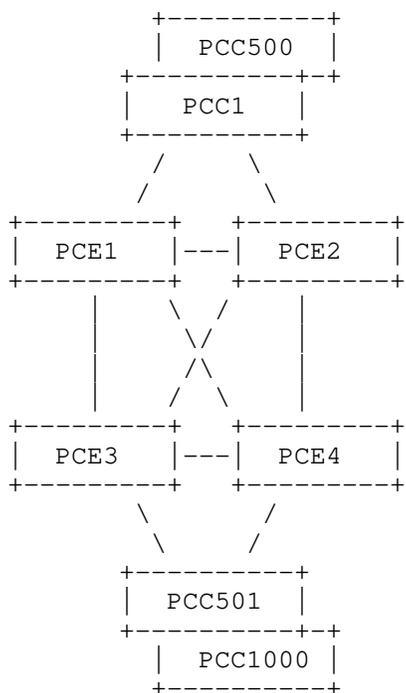   disjoint LSPs PCC1->PCC2 and PCC3->PCC4.

    Suppose PCC1->PCC2 is configured first, PCC1 delegates the LSP to
    PCE1, but as PCE1 does not have the highest computation priority, it
    will sub-delegate the LSP to PCE2 (as it not aware of PCE3 and has no
    way to reach it). PCE2 cannot compute a path for PCC1->PCC2 as it
    does not have the highest priority and is not allowed to sub-delegate
    the LSP again towards PCE3 as per Section 3.

When PCC3->PCC4 is configured, PCC3 delegates the LSP to PCE2 that
performs sub-delegation to PCE3. As PCE3 will have knowledge of only
one LSP in the group, it cannot compute disjointness and can decide
to fall-back to a less constrained computation to provide a path for
PCC3->PCC4. In this case, it will send a PCUpd to PCE2 that will be
forwarded to PCC3.

Disjointness cannot be achieved in this scenario because of lack of
state-sync session between PCE1 and PCE3, but no computation loop
happens. Thus it is advised for all PCEs that support state-sync to
have a full mesh sessions between each other.

5. Using Primary/Secondary Computation and State-sync Sessions to
    increase Scaling

The Primary/Secondary computation and state-sync sessions
architecture can be used to increase the scaling of the PCE
architecture. If the number of PCCs is really high, it may be too
resource consuming for a single PCE to maintain all the PCEP sessions
while at the same time performing all path computations. Using
primary/secondary computation and state-sync sessions may allow to
create groups of PCEs that manage a subset of the PCCs and perform
some or no path computations. Decoupling PCEP session maintenance
and computation will allow increasing scaling of the PCE
architecture.

```
                 +----------+
                 |  PCC500  |
                 +----------+-+
                 |  PCC1    |
                 +----------+
                   /      \
                  /        \
          +---------+  +---------+
          |  PCE1   |---|  PCE2   |
          +---------+  +---------+
              |    \  /    |
              |     \/     |
              |     /\     |
              |    /  \    |
          +---------+  +---------+
          |  PCE3   |---|  PCE4   |
          +---------+  +---------+
                 \      /
                  \    /
               +----------+
               |  PCC501  |
               +----------+-+
                 |  PCC1000 |
                 +----------+
```

   In the figure above, two groups of PCEs are created: PCE1/2 maintain
   PCEP sessions with PCC1 up to PCC500, while PCE3/4 maintain PCEP
   sessions with PCC501 up to PCC1000. A granular primary/secondary
   policy is set-up as follows to load-share computation between PCEs:

   o PCE1 has priority 200 for association ID 1 up to 300, association
      source 0.0.0.0. All other PCEs have a decreasing priority for
      those associations.

   o PCE3 has priority 200 for association ID 301 up to 500,
      association source 0.0.0.0. All other PCEs have a decreasing
      priority for those associations.

   If some PCCs delegate LSPs with association ID 1 up to 300 and
   association source 0.0.0.0, the receiving PCE (if not PCE1) will sub-
   delegate the LSPs to PCE1. PCE1 becomes responsible for the
   computation of these LSP associations while PCE3 is responsible for
   the computation of another set of associations.

   The procedures described in this document could help greatly in load-
   sharing between a group of stateful PCEs.
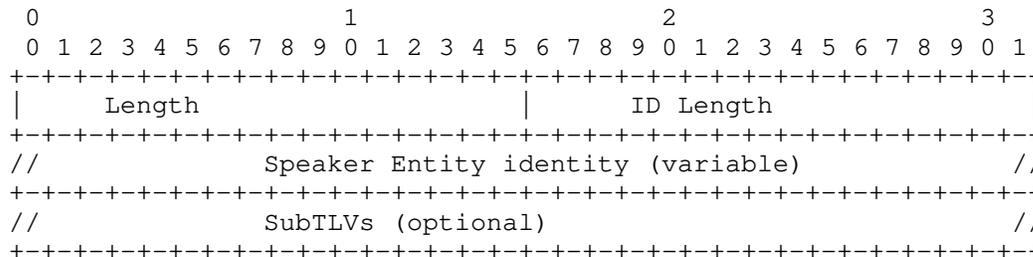
6. PCEP-PATH-VECTOR TLV

   This document allows PCEP messages to be propagated among PCEP
   speaker. It may be useful to track information about the propagation
   of the messages. One of the use cases is a message loop detection
   mechanism, but other use cases like hop by hop information recording
   may also be implemented.

   This document introduces the PCEP-PATH-VECTOR TLV (type TBD3) with
   the following format:

```
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=TBD3          |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 PCEP-SPEAKER-INFORMATION#1                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          ...                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          ...                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 PCEP-SPEAKER-INFORMATION#n                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
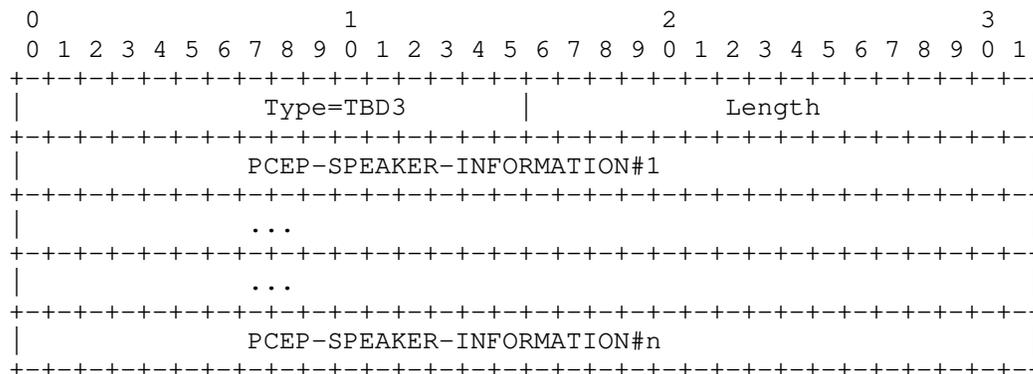
   The TLV format and padding rules are as per [RFC5440].

   The PCEP-SPEAKER-INFORMATION field has the following format:

```
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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Length               |           ID Length           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//             Speaker Entity identity (variable)             //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//             SubTLVs (optional)                             //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

      Length: defines the total length of the PCEP-SPEAKER-INFORMATION
      field.

      ID Length: defines the length of the Speaker identity actual field
      (non-padded).

      Speaker Entity identity: same possible values as the SPEAKER-
      IDENTIFIER-TLV. Padded with trailing zeros to a 4-byte boundary.

The PCEP-SPEAKER-INFORMATION may also carry some optional subTLVs
so each PCEP speaker can add local information that could be
recorded. This document does not define any sub-TLV.

The PCEP-PATH-VECTOR TLV MAY be carried in the LSP Object. Its usage
is purely optional.

The list of speakers within the PCEP-PATH-VECTOR TLV MUST be ordered.
When sending a PCEP message (PCRpt, PCUpd, or PCInitiate), a PCEP
Speaker MAY add the PCEP-PATH-VECTOR TLV with a PCEP-SPEAKER-
INFORMATION containing its own information. If the PCEP message sent
is the result of a previously received PCEP message, and if the PCEP-
PATH-VECTOR TLV was already present in the initial message, the PCEP
speaker MAY append a new PCEP-SPEAKER-INFORMATION containing its own
information.

7. Security Considerations

The security considerations described in [RFC8231] and [RFC5440]
apply to the extensions described in this document as well.
Additional considerations related to state syncronization and sub-
delegation between stateful PCEs are introduced, as it could be
spoofed and could be used as an attack vector. An attacker could
attempt to create too much state in an attempt to load the PCEP peer.
The PCEP peer responds with a PCErr message as described in
[RFC8231]. An attacker could impact LSP operations by creating bogus
state. Further, state syncronization between stateful PCEs could
provide an adversary with the opportunity to eavesdrop on the
network. Thus, securing the PCEP session using Transport Layer
Security (TLS) [RFC8253], as per the recommendations and best current
practices in [RFC7525], is RECOMMENDED.

8. Acknowledgements

Thanks to [I-D.knodel-terminology] urging for better use of terms.

9. IANA Considerations

This document requests IANA actions to allocate code points for the
protocol elements defined in this document.

9.1. PCEP-Error Object

IANA is requested to allocate a new Error Value for the Error Type 9.

```
   Error-Type Meaning                                      Reference
       6        Mandatory Object Missing                   [RFC5440]
                Error-value=TBD1: SPEAKER-IDENTITY-TLV     This document
                missing
```

9.2. PCEP TLV Type Indicators

IANA is requested to allocate new TLV Type Indicator values within
the "PCEP TLV Type Indicators" sub-registry of the PCEP Numbers
registry, as follows:

```
   Value          Meaning                   Reference
    TBD2 ORIGINAL-LSP-DB-VERSION TLV This document
    TBD3      PCEP-PATH-VECTOR TLV    This document
```

9.3. STATEFUL-PCE-CAPABILITY TLV

IANA is requested to allocate a new bit value in the STATEFUL-PCE-
CAPABILITY TLV Flag Field sub-registry.

```
   Bit      Description         Reference
   TBD INTER-PCE-CAPABILITY This document
```

10. References

10.1. 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>.

   [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>.

   [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>.

10.2. Informative References

   [I-D.knodel-terminology]
              Knodel, M. and N. Oever, "Terminology, Power, and
              Inclusive Language in Internet-Drafts and RFCs", draft-
              knodel-terminology-04 (work in progress), August 2020.

   [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>.

   [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the
              Path Computation Element Architecture to the Determination
              of a Sequence of Domains in MPLS and GMPLS", RFC 6805,
              DOI 10.17487/RFC6805, November 2012,
              <https://www.rfc-editor.org/info/rfc6805>.

   [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>.

   [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>.

   [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and
              S. Ray, "North-Bound Distribution of Link-State and
              Traffic Engineering (TE) Information Using BGP", RFC 7752,
              DOI 10.17487/RFC7752, March 2016,
              <https://www.rfc-editor.org/info/rfc7752>.

   [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>.

   [RFC8751]   Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., and D. King,
               "Hierarchical Stateful Path Computation Element (PCE)",
               RFC 8751, DOI 10.17487/RFC8751, March 2020,
               <https://www.rfc-editor.org/info/rfc8751>.

   [RFC8800]   Litkowski, S., Sivabalan, S., Barth, C., and M. Negi,
               "Path Computation Element Communication Protocol (PCEP)
               Extension for Label Switched Path (LSP) Diversity
               Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800,
               July 2020, <https://www.rfc-editor.org/info/rfc8800>.

Appendix A. Contributors

   Dhruv Dhody
   Huawei Technologies
   Divyashree Techno Park, Whitefield
   Bangalore, Karnataka 560066
   India

   Email: dhruv.ietf@gmail.com

Authors' Addresses

   Stephane Litkowski
   Cisco

   Email: slitkows.ietf@gmail.com

   Siva Sivabalan
   Ciena Corporation

   Email: msiva282@gmail.com

   Cheng Li
   Huawei Technologies
   Huawei Campus, No. 156 Beiqing Rd.
   Beijing 100095
   China

   Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan, Guangdong 523808
China

Email: zhenghaomian@huawei.com

                             Extensions to
          the Path Computation Element Communication Protocol (PCEP)
         for Point-to-Multipoint Traffic Engineering Label Switched Paths

                     draft-palleti-pce-rfc6006bis-01

Abstract

   Point-to-point Multiprotocol Label Switching (MPLS) and Generalized
   MPLS (GMPLS) Traffic Engineering Label Switched Paths (TE LSPs) may
   be established using signaling techniques, but their paths may first
   need to be determined. The Path Computation Element (PCE) has been
   identified as an appropriate technology for the determination of the
   paths of point-to-multipoint (P2MP) TE LSPs.

   This document describes extensions to the PCE communication Protocol
   (PCEP) to handle requests and responses for the computation of paths
   for P2MP TE LSPs.

   This document obsoletes RFC 6006.

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 11, 2017.

Copyright Notice

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.

   [RFC4875] describes how to set up point-to-multipoint (P2MP) Traffic
   Engineering Label Switched Paths (TE LSPs) for use in Multiprotocol
   Label Switching (MPLS) and Generalized MPLS (GMPLS) networks.

The PCE has been identified as a suitable application for the
computation of paths for P2MP TE LSPs [RFC5671].

The PCE communication Protocol (PCEP) is designed as a communication
protocol between PCCs and PCEs for point-to-point (P2P) path
computations and is defined in [RFC5440]. However, that
specification does not provide a mechanism to request path
computation of P2MP TE LSPs.

A P2MP LSP is comprised of multiple source-to-leaf (S2L) sub-LSPs.
These S2L sub-LSPs are set up between ingress and egress Label
Switching Routers (LSRs) and are appropriately overlaid to construct
a P2MP TE LSP. During path computation, the P2MP TE LSP may be
determined as a set of S2L sub-LSPs that are computed separately and
combined to give the path of the P2MP LSP, or the entire P2MP TE LSP
may be determined as a P2MP tree in a single computation.

This document relies on the mechanisms of PCEP to request path
computation for P2MP TE LSPs. One path computation request message
from a PCC may request the computation of the whole P2MP TE LSP, or
the request may be limited to a sub-set of the S2L sub-LSPs. In the
extreme case, the PCC may request the S2L sub-LSPs to be computed
individually with it being the PCC's responsibility to decide whether
to signal individual S2L sub-LSPs or combine the computation results
to signal the entire P2MP TE LSP. Hence the PCC may use one path
computation request message or may split the request across multiple
path computation messages.

This document obsoletes RFC 6006 and incorporates all outstanding
Errata:

o Erratum with IDs: 3819, 3830, 3836, 4867, and 4868.

1.1. Terminology

Terminology used in this document:

TE LSP: Traffic Engineering Label Switched Path.

LSR: Label Switching Router.

OF: Objective Function: A set of one or more optimization criteria
used for the computation of a single path (e.g., path cost
minimization), or for the synchronized computation of a set of
paths (e.g., aggregate bandwidth consumption minimization).

P2MP: Point-to-Multipoint.

P2P: Point-to-Point.

This document also uses the terminology defined in [RFC4655],
[RFC4875], and [RFC5440].

1.2. 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 RFC 2119 [RFC2119].

2. PCC-PCE Communication Requirements

   This section summarizes the PCC-PCE communication requirements for
   P2MP MPLS-TE LSPs described in [RFC5862]. The numbering system
   corresponds to the requirement numbers used in [RFC5862].

   1.   The PCC MUST be able to specify that the request is a P2MP path
        computation request.

   2.   The PCC MUST be able to specify that objective functions are to
        be applied to the P2MP path computation request.

   3.   The PCE MUST have the capability to reject a P2MP path request
        and indicate non-support of P2MP path computation.

   4.   The PCE MUST provide an indication of non-support of P2MP path
        computation by back-level PCE implementations.

   5.   A P2MP path computation request MUST be able to list multiple
        destinations.

   6.   A P2MP path computation response MUST be able to carry the path
        of a P2MP LSP.

   7.   By default, the path returned by the PCE SHOULD use the
        compressed format.

   8.   It MUST be possible for a single P2MP path computation request or
        response to be conveyed by a sequence of messages.

   9.   It MUST NOT be possible for a single P2MP path computation
        request to specify a set of different constraints, traffic
        parameters, or quality-of-service requirements for different
        destinations of a P2MP LSP.

   10. P2MP path modification and P2MP path diversity MUST be supported.

   11. It MUST be possible to reoptimize existing P2MP TE LSPs.

   12. It MUST be possible to add and remove P2MP destinations from
        existing paths.

13. It MUST be possible to specify a list of applicable branch nodes
    to use when computing the P2MP path.

14. It MUST be possible for a PCC to discover P2MP path computation
    capability.

15. The PCC MUST be able to request diverse paths when requesting a
    P2MP path.

3. Protocol Procedures and Extensions

   The following section describes the protocol extensions required to
   satisfy the requirements specified in Section 2 ("PCC-PCE
   Communication Requirements") of this document.

3.1. P2MP Capability Advertisement

3.1.1. P2MP Computation TLV in the Existing PCE Discovery Protocol

   [RFC5088] defines a PCE Discovery (PCED) TLV carried in an OSPF
   Router Information Link State Advertisement (LSA) defined in
   [RFC7770] to facilitate PCE discovery using OSPF. [RFC5088]
   specifies that no new sub-TLVs may be added to the PCED TLV. This
   document defines a new flag in the OSPF PCE Capability Flags to
   indicate the capability of P2MP computation.

   Similarly, [RFC5089] defines the PCED sub-TLV for use in PCE
   Discovery using IS-IS. This document will use the same flag
   requested for the OSPF PCE Capability Flags sub-TLV to allow IS-IS to
   indicate the capability of P2MP computation.

   The IANA assignment for a shared OSPF and IS-IS P2MP Capability Flag
   is documented in Section 6.9 ("OSPF PCE Capability Flag") of this
   document.

   PCEs wishing to advertise that they support P2MP path computation
   would set the bit (10) accordingly. PCCs that do not understand this
   bit will ignore it (per [RFC5088] and [RFC5089]). PCEs that do not
   support P2MP will leave the bit clear (per the default behavior
   defined in [RFC5088] and [RFC5089]).

   PCEs that set the bit to indicate support of P2MP path computation
   MUST follow the procedures in Section 3.3.2 ("The New P2MP END-POINTS
   Object") to further qualify the level of support.

3.1.2. Open Message Extension

   Based on the Capabilities Exchange requirement described in
   [RFC5862], if a PCE does not advertise its P2MP capability during
   discovery, PCEP should be used to allow a PCC to discover, during the
   Open Message Exchange, which PCEs are capable of supporting P2MP path
   computation.

   To satisfy this requirement, we extend the PCEP OPEN object by
   defining a new optional TLV to indicate the PCE's capability to
   perform P2MP path computations.

   IANA has allocated value 6 from the "PCEP TLV Type Indicators" sub-
   registry, as documented in Section 6.1 ("PCEP TLV Type Indicators").
   The description is "P2MP capable", and the length value is 2 bytes.
   The value field is set to default value 0.

   The inclusion of this TLV in an OPEN object indicates that the sender
   can perform P2MP path computations.

   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 path computation capabilities.

3.2. Efficient Presentation of P2MP LSPs

   When specifying additional leaves, or optimizing existing P2MP TE
   LSPs as specified in [RFC5862], it may be necessary to pass existing
   P2MP LSP route information between the PCC and PCE in the request and
   reply messages. In each of these scenarios, we need new path objects
   for efficiently passing the existing P2MP LSP between the PCE and
   PCC.

   We specify the use of the Resource Reservation Protocol Traffic
   Engineering (RSVP-TE) extensions Explicit Route Object (ERO) to
   encode the explicit route of a TE LSP through the network. PCEP ERO
   sub-object types correspond to RSVP-TE ERO sub-object types. The
   format and content of the ERO object are defined in [RFC3209] and
   [RFC3473].

   The Secondary Explicit Route Object (SERO) is used to specify the
   explicit route of a S2L sub-LSP. The path of each subsequent S2L
   sub-LSP is encoded in a P2MP_SECONDARY_EXPLICIT_ROUTE object SERO.
   The format of the SERO is the same as an ERO defined in [RFC3209] and
   [RFC3473].

The Secondary Record Route Object (SRRO) is used to record the
explicit route of the S2L sub-LSP. The class of the P2MP SRRO is the
same as the SRRO defined in [RFC4873].

The SERO and SRRO are used to report the route of an existing TE LSP
for which a reoptimization is desired. The format and content of the
SERO and SRRO are defined in [RFC4875].

A new PCEP object class and type are requested for SERO and SRRO.

```
Object-Class Value      29
Name                    SERO
Object-Type             1: SERO
                        2-15: Unassigned
Reference               RFC 6006


Object-Class Value      30
Name                    SRRO
Object-Type             1: SRRO
                        2-15: Unassigned
Reference               RFC 6006
```

The IANA assignment is documented in Section 6.5 ("PCEP Objects").

Since the explicit path is available for immediate signaling by the
MPLS or GMPLS control plane, the meanings of all of the sub-objects
and fields in this object are identical to those defined for the ERO.

3.3. P2MP Path Computation Request/Reply Message Extensions

This document extends the existing P2P RP (Request Parameters) object
so that a PCC can signal a P2MP path computation request to the PCE
receiving the PCEP request. The END-POINTS object is also extended
to improve the efficiency of the message exchange between PCC and PCE
in the case of P2MP path computation.

3.3.1. The Extension of the RP Object

The PCE path computation request and reply messages will need the
following additional parameters to indicate to the receiving PCE that
the request and reply messages have been fragmented across multiple
messages, that they have been requested for a P2MP path, and whether
the route is represented in the compressed or uncompressed format.

This document adds the following flags to the RP Object:

The F-bit is added to the flag bits of the RP object to indicate to
the receiver that the request is part of a fragmented request, or is
not a fragmented request.

o F (RP fragmentation bit - 1 bit):

   0: This indicates that the RP is not fragmented or it is the last
      piece of the fragmented RP.

   1: This indicates that the RP is fragmented and this is not the
      last piece of the fragmented RP. The receiver needs to wait
      for additional fragments until it receives an RP with the same
      RP-ID and with the F-bit set to 0.

The N-bit is added in the flag bits field of the RP object to signal
the receiver of the message that the request/reply is for P2MP or is
not for P2MP.

o N (P2MP bit - 1 bit):

   0: This indicates that this is not a PCReq or PCRep message for
      P2MP.

   1: This indicates that this is a PCReq or PCRep message for P2MP.

The E-bit is added in the flag bits field of the RP object to signal
the receiver of the message that the route is in the compressed
format or is not in the compressed format. By default, the path
returned by the PCE SHOULD use the compressed format.

o E (ERO-compression bit - 1 bit):

   0: This indicates that the route is not in the compressed format.

   1: This indicates that the route is in the compressed format.

The IANA assignment is documented in Section 6.2 ("Request Parameter
Bit Flags") of this document.

3.3.2. The New P2MP END-POINTS Object

The END-POINTS object is used in a PCReq message to specify the
source IP address and the destination IP address of the path for
which a path computation is requested. To represent the end points
for a P2MP path efficiently, we define two new types of END-POINTS
objects for the P2MP path:

o Old leaves whose path can be modified/reoptimized;

o Old leaves whose path must be left unchanged.

With the new END-POINTS object, the PCE path computation request
message is expanded in a way that allows a single request message to
list multiple destinations.

In total, there are now 4 possible types of leaves in a P2MP request:

o New leaves to add (leaf type = 1)

o Old leaves to remove (leaf type = 2)

o Old leaves whose path can be modified/reoptimized (leaf type = 3)

o Old leaves whose path must be left unchanged (leaf type = 4)

A given END-POINTS object gathers the leaves of a given type. The
type of leaf in a given END-POINTS object is identified by the END-
POINTS object leaf type field.

Using the new END-POINTS object, the END-POINTS portion of a request
message for the multiple destinations can be reduced by up to 50% for
a P2MP path where a single source address has a very large number of
destinations.

Note that a P2MP path computation request can mix the different types
of leaves by including several END-POINTS objects per RP object as
shown in the PCReq Routing Backus-Naur Form (RBNF) [RFC5511] format
in Section 3.4 ("Request Message Format").

The format of the new END-POINTS object body for IPv4 (Object-Type 3)
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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Leaf type                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Source IPv4 address                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination IPv4 address                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                             ...                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination IPv4 address                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
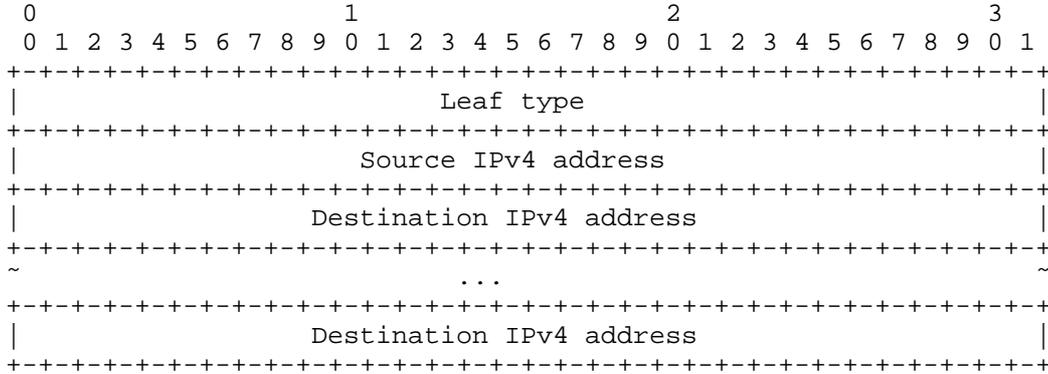
   Figure 1. The New P2MP END-POINTS Object Body Format for IPv4

The format of the END-POINTS object body for IPv6 (Object-Type 4) 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Leaf type                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                 Source IPv6 address (16 bytes)                |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|               Destination IPv6 address (16 bytes)             |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                             ...                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|               Destination IPv6 address (16 bytes)             |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
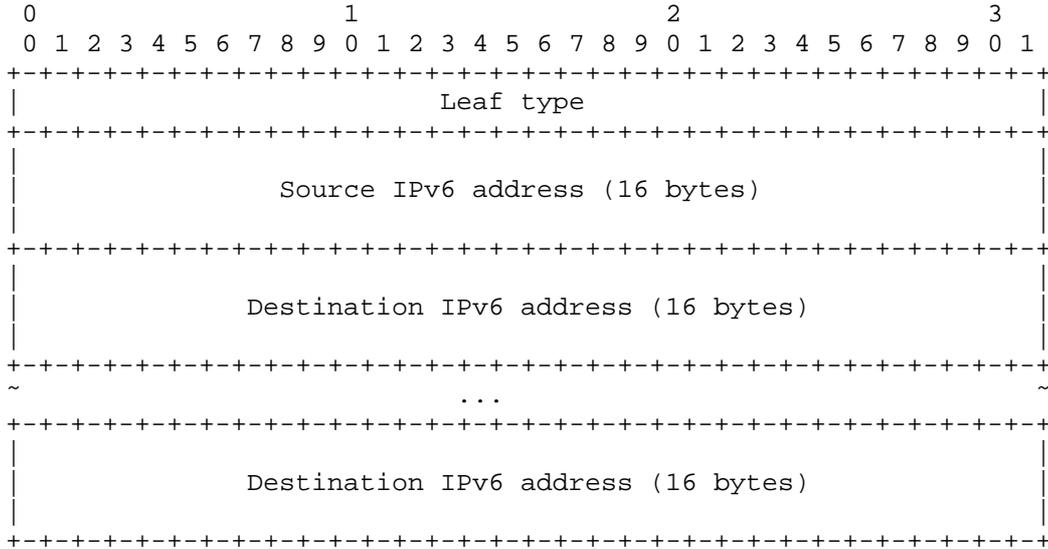
   Figure 2. The New P2MP END-POINTS Object Body Format for IPv6

The END-POINTS object body has a variable length. These are
multiples of 4 bytes for IPv4, and multiples of 16 bytes, plus 4
bytes, for IPv6.

3.4. Request Message Format

   As per [RFC5440], a Path Computation Request message (also referred
   to as a PCReq message) is a PCEP message sent by a PCC to a PCE to
   request a path computation. A PCReq message may carry more than one
   path computation request.

   As per [RFC5541], the OF object MAY be carried within a PCReq
   message. If an objective function is to be applied to a set of
   synchronized path computation requests, the OF object MUST be carried
   just after the corresponding SVEC (Synchronization VECtor) object and
   MUST NOT be repeated for each elementary request.

   The PCReq message is encoded as follows using RBNF as defined in
   [RFC5511].

   Below is the message format for the request message:

           <PCReq Message>::= <Common Header>
                          [<svec-list>]
                          <request-list>

        where:

           <svec-list>::=<SVEC>
                          [<OF>]
                          [<metric-list>]
                          [<svec-list>]

           <request-list>::=<request>[<request-list>]

           <request>::= <RP>
                          <end-point-rro-pair-list>
                          [<OF>]
                          [<LSPA>]
                          [<BANDWIDTH>]
                          [<metric-list>]
                          [<IRO>|<BNC>]
                          [<LOAD-BALANCING>]

        where:

           <end-point-rro-pair-list>::=
                            <END-POINTS>[<RRO-List>[<BANDWIDTH>]]
                          [<end-point-rro-pair-list>]

```
<RRO-List>::=(<RRO>|<SRRO>)[<RRO-List>]
<metric-list>::=<METRIC>[<metric-list>]
```

Figure 3.   The Message Format for the Request Message

Note that we preserve compatibility with the [RFC5440] definition of
<request>. At least one instance of <endpoints> MUST be present in
this message.

We have documented the IANA assignment of additional END-POINTS
Object-Types in Section 6.5 ("PCEP Objects") of this document.

3.5. Reply Message Format

The PCEP Path Computation Reply message (also referred to as a PCRep
message) is a PCEP message sent by a PCE to a requesting PCC in
response to a previously received PCReq message. PCEP supports the
bundling of multiple replies to a set of path computation requests
within a single PCRep message.

The PCRep message is encoded as follows using RBNF as defined in
[RFC5511].

Below is the message format for the reply message:

```
<PCRep Message>::= <Common Header>
                   <response-list>
```

where:

```
<response-list>::=<response>[<response-list>]

<response>::=<RP>
        [<end-point-path-pair-list>]
        [<NO-PATH>]
        [<UNREACH-DESTINATION>]
        [<attribute-list>]

<end-point-path-pair-list>::=
        [<END-POINTS>]<path>
        [<end-point-path-pair-list>]

<path> ::= (<ERO>|<SERO>) [<path>]
```

where:

```
<attribute-list>::=[<OF>]
                   [<LSPA>]
                   [<BANDWIDTH>]
                   [<metric-list>]
                   [<IRO>]
```

Figure 4. The Message Format for the Reply Message

The optional END-POINTS object in the reply message is used to specify which paths are removed, changed, not changed, or added for the request. The path is only needed for the end points that are added or changed.

If the E-bit (ERO-Compress bit) was set to 1 in the request, then the path will be formed by an ERO followed by a list of SEROs.

Note that we preserve compatibility with the [RFC5440] definition of <response> and the optional <end-point-path-pair-list> and <path>.

3.6. P2MP Objective Functions and Metric Types

3.6.1. New Objective Functions

Six objective functions have been defined in [RFC5541] for P2P path

computation.

This document defines two additional objective functions -- namely,
SPT (Shortest Path Tree) and MCT (Minimum Cost Tree) that apply to
P2MP path computation. Hence two new objective function codes have
to be defined.

The description of the two new objective functions is as follows.
Objective Function Code: 7

   Name: Shortest Path Tree (SPT)

   Description: Minimize the maximum source-to-leaf cost with respect
   to a specific metric or to the TE metric used as the default
   metric when the metric is not specified (e.g., TE or IGP metric).

Objective Function Code: 8

   Name: Minimum Cost Tree (MCT)

   Description: Minimize the total cost of the tree, that is the sum
   of the costs of tree links, with respect to a specific metric or
   to the TE metric used as the default metric when the metric is not
   specified.

Processing these two new objective functions is subject to the rules
defined in [RFC5541].

3.6.2. New Metric Object Types

There are three types defined for the <METRIC> object in [RFC5440] --
namely, the IGP metric, the TE metric, and the hop count metric. This
document defines three additional types for the <METRIC> object: the
P2MP IGP metric, the P2MP TE metric, and the P2MP hop count metric.
They encode the sum of the metrics of all links of the tree. We
propose the following values for these new metric types:

o P2MP IGP metric: T=8

o P2MP TE metric: T=9

o P2MP hop count metric: T=10

3.7. Non-Support of P2MP Path Computation

   o If a PCE receives a P2MP path request and it understands the P2MP
      flag in the RP object, but the PCE is not capable of P2MP
      computation, the PCE MUST send a PCErr message with a PCEP-ERROR

object and corresponding Error-Value. The request MUST then be
cancelled at the PCC. New Error-Types and Error-Values are
requested in Section 6 ("IANA Considerations") of this document.

o If the PCE does not understand the P2MP flag in the RP object,
then the PCE MUST send a PCErr message with Error-value=2
(capability not supported).

3.8. Non-Support by Back-Level PCE Implementations

   If a PCE receives a P2MP request and the PCE does not understand the
   P2MP flag in the RP object, and therefore the PCEP P2MP extensions,
   then the PCE SHOULD reject the request.

3.9. P2MP TE Path Reoptimization Request

   A reoptimization request for a P2MP TE path is specified by the use
   of the R-bit within the RP object as defined in [RFC5440] and is
   similar to the reoptimization request for a P2P TE path. The only
   difference is that the user MUST insert the list of RROs and SRROs
   after each type of END-POINTS in the PCReq message, as described in
   the "Request Message Format" section (Section 3.4) of this document.

   An example of a reoptimization request and subsequent PCReq message
   is described below:

            Common Header
            RP with P2MP flag/R-bit set
            END-POINTS for leaf type 3
              RRO list
            OF (optional)

            Figure 5. PCReq Message Example 1 for Optimization

   In this example, we request reoptimization of the path to all leaves
   without adding or pruning leaves. The reoptimization request would
   use an END-POINT type 3. The RRO list would represent the P2MP LSP
   before the optimization, and the modifiable path leaves would be
   indicated in the END-POINTS object.

   It is also possible to specify distinct leaves whose path cannot be
   modified. An example of the PCReq message in this scenario would be:

            Common Header
            RP with P2MP flag/R-bit set
            END-POINTS for leaf type 3
              RRO list
            END-POINTS for leaf type 4
              RRO list
            OF (optional)

            Figure 6. PCReq Message Example 2 for Optimization

3.10. Adding and Pruning Leaves to/from the P2MP Tree

   When adding new leaves to or removing old leaves from the existing
   P2MP tree, by supplying a list of existing leaves, it SHOULD be
   possible to optimize the existing P2MP tree. This section explains
   the methods for adding new leaves to or removing old leaves from the
   existing P2MP tree.

   To add new leaves, the user MUST build a P2MP request using END-
   POINTS with leaf type 1.

   To remove old leaves, the user must build a P2MP request using END-
   POINTS with leaf type 2. If no type-2 END-POINTS exist, then the PCE
   MUST send an error type 17, value=1: The PCE is not capable of
   satisfying the request due to no END-POINTS with leaf type 2.

   When adding new leaves to or removing old leaves from the existing
   P2MP tree, the PCC must also provide the list of old leaves, if any,
   including END-POINTS with leaf type 3, leaf type 4, or both. New
   PCEP-ERROR objects and types are necessary for reporting when certain
   conditions are not satisfied (i.e., when there are no END-POINTS with
   leaf type 3 or 4, or in the presence of END-POINTS with leaf type 1
   or 2). A generic "Inconsistent END-POINT" error will be used if a
   PCC receives a request that has an inconsistent END-POINT (i.e., if a
   leaf specified as type 1 already exists). These IANA assignments are
   documented in Section 6.6 ("PCEP-ERROR Objects and Types") of this
   document.

   For old leaves, the user MUST provide the old path as a list of RROs
   that immediately follows each END-POINTS object. This document
   specifies error values when specific conditions are not satisfied.

   The following examples demonstrate full and partial reoptimization of
   existing P2MP LSPs:

   Case 1: Adding leaves with full reoptimization of existing paths

           Common Header
           RP with P2MP flag/R-bit set
           END-POINTS for leaf type 1
             RRO list
           END-POINTS for leaf type 3
             RRO list
           OF (optional)

   Case 2: Adding leaves with partial reoptimization of existing paths

```
        Common Header
        RP with P2MP flag/R-bit set
        END-POINTS for leaf type 1
        END-POINTS for leaf type 3
          RRO list
        END-POINTS for leaf type 4
          RRO list
        OF (optional)
```

   Case 3: Adding leaves without reoptimization of existing paths

```
        Common Header
        RP with P2MP flag/R-bit set
        END-POINTS for leaf type 1
          RRO list
        END-POINTS for leaf type 4
          RRO list
        OF (optional)
```

   Case 4: Pruning Leaves with full reoptimization of existing paths

```
        Common Header
        RP with P2MP flag/R-bit set
        END-POINTS for leaf type 2
          RRO list
        END-POINTS for leaf type 3
          RRO list
        OF (optional)
```

   Case 5: Pruning leaves with partial reoptimization of existing paths

```
        Common Header
        RP with P2MP flag/R-bit set
        END-POINTS for leaf type 2
          RRO list
        END-POINTS for leaf type 3
          RRO list
        END-POINTS for leaf type 4
          RRO list
        OF (optional)
```

Case 6: Pruning leaves without reoptimization of existing paths

```
Common Header
RP with P2MP flag/R-bit set
END-POINTS for leaf type 2
  RRO list
END-POINTS for leaf type 4
  RRO list
OF (optional)
```

Case 7: Adding and pruning leaves with full reoptimization of
existing paths

```
Common Header
RP with P2MP flag/R-bit set
END-POINTS for leaf type 1
END-POINTS for leaf type 2
  RRO list
END-POINTS for leaf type 3
  RRO list
OF (optional)
```

Case 8: Adding and pruning leaves with partial reoptimization of
existing paths

```
Common Header
RP with P2MP flag/R-bit set
END-POINTS for leaf type 1
END-POINTS for leaf type 2
  RRO list
END-POINTS for leaf type 3
  RRO list
END-POINTS for leaf type 4
  RRO list
OF (optional)
```

Case 9: Adding and pruning leaves without reoptimization of existing
paths

```
Common Header
RP with P2MP flag/R-bit set
END-POINTS for leaf type 1
END-POINTS for leaf type 2
  RRO list
END-POINTS for leaf type 4
  RRO list
OF (optional)
```

3.11. Discovering Branch Nodes

   Before computing the P2MP path, a PCE may need to be provided means
   to know which nodes in the network are capable of acting as branch
   LSRs. A PCE can discover such capabilities by using the mechanisms
   defined in [RFC5073].

3.11.1. Branch Node Object

   The PCC can specify a list of nodes that can be used as branch nodes
   or a list of nodes that cannot be used as branch nodes by using the
   Branch Node Capability (BNC) Object. The BNC Object has the same
   format as the Include Route Object (IRO) defined in [RFC5440], except
   that it only supports IPv4 and IPv6 prefix sub-objects. Two Object-
   types are also defined:

   o Branch node list: List of nodes that can be used as branch nodes.

   o Non-branch node list: List of nodes that cannot be used as branch
      nodes.

   The object can only be carried in a PCReq message. A Path Request
   may carry at most one Branch Node Object.

   The Object-Class and Object-types have been allocated by IANA. The
   IANA assignment is documented in Section 6.5 ("PCEP Objects").

3.12. Synchronization of P2MP TE Path Computation Requests

   There are cases when multiple P2MP LSPs' computations need to be
   synchronized. For example, one P2MP LSP is the designated backup of
   another P2MP LSP. In this case, path diversity for these dependent
   LSPs may need to be considered during the path computation.

   The synchronization can be done by using the existing Synchronization
   VECtor (SVEC) functionality defined in [RFC5440].

An example of synchronizing two P2MP LSPs, each having two leaves for
Path Computation Request Messages, is illustrated below:

```
        Common Header
        SVEC for sync of LSP1 and LSP2
        OF (optional)
        RP for LSP1
          END-POINTS1 for LSP1
          RRO1 list
        RP for LSP2
          END-POINTS2 for LSP2
          RRO2 list
```

        Figure 7. PCReq Message Example for Synchronization

This specification also defines two new flags to the SVEC Object Flag
Field for P2MP path dependent computation requests. The first new
flag is to allow the PCC to request that the PCE should compute a
secondary P2MP path tree with partial path diversity for specific
leaves or a specific S2L sub-path to the primary P2MP path tree. The
second flag, would allow the PCC to request that partial paths should
be link direction diverse.

The following flags are added to the SVEC object body in this
document:

o P (Partial Path Diverse bit - 1 bit):

   When set, this would indicate a request for path diversity for a
   specific leaf, a set of leaves, or all leaves.

o D (Link Direction Diverse bit - 1 bit):

   When set, this would indicate a request that a partial path or
   paths should be link direction diverse.

   The IANA assignment is referenced in Section 6.8 of this document.

3.13. Request and Response Fragmentation

   The total PCEP message length, including the common header, is
   16 bytes. In certain scenarios the P2MP computation request may not
   fit into a single request or response message. For example, if a
   tree has many hundreds or thousands of leaves, then the request or
   response may need to be fragmented into multiple messages.

The F-bit has been outlined in "The Extension of the RP Object"
(Section 3.3.1) of this document. The F-bit is used in the RP object
to signal that the initial request or response was too large to fit
into a single message and will be fragmented into multiple messages.
In order to identify the single request or response, each message
will use the same request ID.

3.13.1. Request Fragmentation Procedure

If the initial request is too large to fit into a single request
message, the PCC will split the request over multiple messages. Each
message sent to the PCE, except the last one, will have the F-bit set
in the RP object to signify that the request has been fragmented into
multiple messages. In order to identify that a series of request
messages represents a single request, each message will use the same
request ID.

The assumption is that request messages are reliably delivered and in
sequence, since PCEP relies on TCP.

3.13.2. Response Fragmentation Procedure

Once the PCE computes a path based on the initial request, a response
is sent back to the PCC. If the response is too large to fit into a
single response message, the PCE will split the response over
multiple messages. Each message sent by the PCE, except the last
one, will have the F-bit set in the RP object to signify that the
response has been fragmented into multiple messages. In order to
identify that a series of response messages represents a single
response, each message will use the same response ID.

Again, the assumption is that response messages are reliably
delivered and in sequence, since PCEP relies on TCP.

3.13.3. Fragmentation Examples

The following example illustrates the PCC sending a request message
with Req-ID1 to the PCE, in order to add one leaf to an existing tree
with 1200 leaves. The assumption used for this example is that one
request message can hold up to 800 leaves. In this scenario, the
original single message needs to be fragmented and sent using two
smaller messages, which have the Req-ID1 specified in the RP object,
and with the F-bit set on the first message, and cleared on the
second message.

```
        Common Header
        RP1 with Req-ID1 and P2MP=1 and F-bit=1
        OF (optional)
        END-POINTS1 for P2MP
          RRO1 list

        Common Header
        RP2 with Req-ID1 and P2MP=1 and F-bit=0
        OF (optional)
        END-POINTS1 for P2MP
          RRO1 list
```

              Figure 8. PCReq Message Fragmentation Example

   To handle a scenario where the last fragmented message piece is lost,
   the receiver side of the fragmented message may start a timer once it
   receives the first piece of the fragmented message. When the timer
   expires and it has not received the last piece of the fragmented
   message, it should send an error message to the sender to signal that
   it has received an incomplete message. The relevant error message is
   documented in Section 3.15 ("P2MP PCEP-ERROR Objects and Types").

3.14. UNREACH-DESTINATION Object

   The PCE path computation request may fail because all or a subset of
   the destinations are unreachable.

   In such a case, the UNREACH-DESTINATION object allows the PCE to
   optionally specify the list of unreachable destinations.

   This object can be present in PCRep messages. There can be up to one
   such object per RP.

   The following UNREACH-DESTINATION objects will be required:

   UNREACH-DESTINATION Object-Class is 28.
   UNREACH-DESTINATION Object-Type for IPv4 is 1.
   UNREACH-DESTINATION Object-Type for IPv6 is 2.

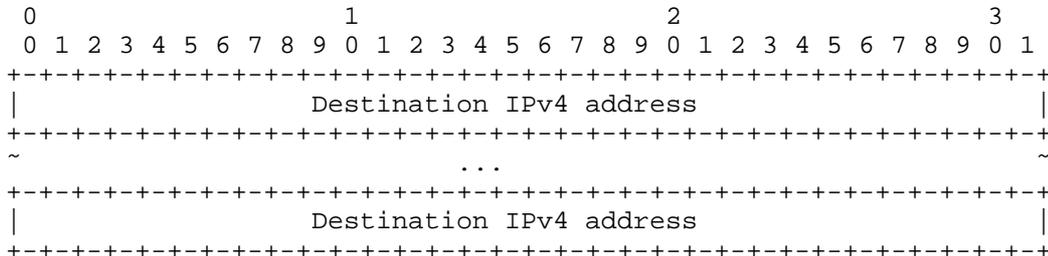The format of the UNREACH-DESTINATION object body for IPv4 (Object-
Type=1) is as follows:

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                   Destination IPv4 address                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~                            ...                                ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                   Destination IPv4 address                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

           Figure 9. UNREACH-DESTINATION Object Body for IPv4

The format of the UNREACH-DESTINATION object body for IPv6 (Object-
Type=2) 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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                              |
   |                Destination IPv6 address (16 bytes)           |
   |                                                              |
   |                                                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~                            ...                                ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                              |
   |                Destination IPv6 address (16 bytes)           |
   |                                                              |
   |                                                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
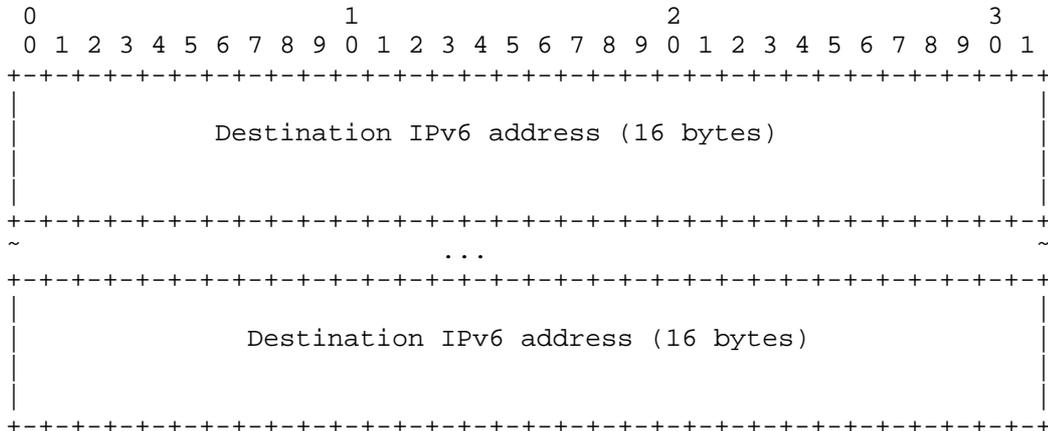
           Figure 10. UNREACH-DESTINATION Object Body for IPv6

3.15. P2MP PCEP-ERROR Objects and Types

   To indicate an error associated with policy violation, a new error
   value "P2MP Path computation not allowed" should be added to the
   existing error code for policy violation (Error-Type=5) as defined in
   [RFC5440]:

Error-Type=5; Error-Value=7: if a PCE receives a P2MP path
computation request that is not compliant with administrative
privileges (i.e., "The PCE policy does not support P2MP path
computation"), the PCE MUST send a PCErr message with a PCEP-ERROR
object (Error-Type=5) and an Error-Value (Error-Value=7). The
corresponding P2MP path computation request MUST also be cancelled.

To indicate capability errors associated with the P2MP path request,
a new Error-Type (16) and subsequent error-values are defined as
follows for inclusion in the PCEP-ERROR object:

Error-Type=16; Error-Value=1: if a PCE receives a P2MP path request
and the PCE is not capable of satisfying the request due to
insufficient memory, the PCE MUST send a PCErr message with a PCEP-
ERROR object (Error-Type=16) and an Error-Value (Error-Value=1). The
corresponding P2MP path computation request MUST also be cancelled.

Error-Type=16; Error-Value=2: if a PCE receives a P2MP path request
and the PCE is not capable of P2MP computation, the PCE MUST send a
PCErr message with a PCEP-ERROR object (Error-Type=16) and an Error-
Value (Error-Value=2). The corresponding P2MP path computation
request MUST also be cancelled.

To indicate P2MP message fragmentation errors associated with a P2MP
path request, a new Error-Type (18) and subsequent error-values are
defined as follows for inclusion in the PCEP-ERROR object:

Error-Type=18; Error-Value=1: if a PCE has not received the last
piece of the fragmented message, it should send an error message to
the sender to signal that it has received an incomplete message
(i.e., "Fragmented request failure"). The PCE MUST send a PCErr
message with a PCEP-ERROR object (Error-Type=18) and an Error-Value
(Error-Value=1).

3.16. PCEP NO-PATH Indicator

To communicate the reasons for not being able to find P2MP path
computation, the NO-PATH object can be used in the PCRep message.

One new bit is defined in the NO-PATH-VECTOR TLV carried in the
NO-PATH Object:

bit 24: when set, the PCE indicates that there is a reachability
problem with all or a subset of the P2MP destinations. Optionally,
the PCE can specify the destination or list of destinations that are
not reachable using the new UNREACH-DESTINATION object defined in
Section 3.14.

4. Manageability Considerations

   [RFC5862] describes various manageability requirements in support of
   P2MP path computation when applying PCEP. This section describes how
   manageability requirements mentioned in [RFC5862] are supported in
   the context of PCEP extensions specified in this document.

   Note that [RFC5440] describes various manageability considerations in
   PCEP, and most of the manageability requirements mentioned in
   [RFC5862] are already covered there.

4.1. Control of Function and Policy

   In addition to PCE configuration parameters listed in [RFC5440], the
   following additional parameters might be required:

   o The ability to enable or disable P2MP path computations on the
      PCE.

   o The PCE may be configured to enable or disable the advertisement
      of its P2MP path computation capability. A PCE can advertise its
      P2MP capability via the IGP discovery mechanism discussed in
      Section 3.1.1 ("P2MP Computation TLV in the Existing PCE Discovery
      Protocol"), or during the Open Message Exchange discussed in
      Section 3.1.2 ("Open Message Extension").

4.2. Information and Data Models

   A number of MIB objects have been defined for general PCEP control
   and monitoring of P2P computations in [RFC7420]. [RFC5862] specifies
   that MIB objects will be required to support the control and
   monitoring of the protocol extensions defined in this document. A new
   document will be required to define MIB objects for PCEP control and
   monitoring of P2MP computations.

4.3. Liveness Detection and Monitoring

   There are no additional considerations beyond those expressed in
   [RFC5440], since [RFC5862] does not address any additional
   requirements.

4.4. Verifying Correct Operation

   There are no additional requirements beyond those expressed in
   [RFC4657] for verifying the correct operation of the PCEP sessions.
   It is expected that future MIB objects will facilitate verification
   of correct operation and reporting of P2MP PCEP requests, responses,
   and errors.

4.5. Requirements for Other Protocols and Functional Components

   The method for the PCE to obtain information about a PCE capable of
   P2MP path computations via OSPF and IS-IS is discussed in
   Section 3.1.1 ("P2MP Computation TLV in the Existing PCE Discovery
   Protocol") of this document.

   The subsequent IANA assignments are documented in Section 6.9 ("OSPF
   PCE Capability Flag") of this document.

4.6. Impact on Network Operation

   It is expected that the use of PCEP extensions specified in this
   document will not significantly increase the level of operational
   traffic. However, computing a P2MP tree may require more PCE state
   compared to a P2P computation. In the event of a major network
   failure and multiple recovery P2MP tree computation requests being
   sent to the PCE, the load on the PCE may also be significantly
   increased.

5. Security Considerations

   As described in [RFC5862], P2MP path computation requests are more
   CPU-intensive and also utilize more link bandwidth. In the event of
   an unauthorized P2MP path computation request, or a denial of service
   attack, the subsequent PCEP requests and processing may be disruptive
   to the network. Consequently, it is important that implementations
   conform to the relevant security requirements of [RFC5440] that
   specifically help to minimize or negate unauthorized P2MP path
   computation requests and denial of service attacks. These mechanisms
   include:

   o Securing the PCEP session requests and responses using TCP
      security techniques (Section 10.2 of [RFC5440]).

   o Authenticating the PCEP requests and responses to ensure the
      message is intact and sent from an authorized node (Section 10.3
      of [RFC5440]).

   o Providing policy control by explicitly defining which PCCs, via IP
      access-lists, are allowed to send P2MP path requests to the PCE
      (Section 10.6 of [RFC5440]).

   PCEP operates over TCP, so it is also important to secure the PCE and
   PCC against TCP denial of service attacks. Section 10.7.1 of
   [RFC5440] outlines a number of mechanisms for minimizing the risk of
   TCP based denial of service attacks against PCEs and PCCs.

   PCEP implementations SHOULD consider the additional security provided
   by the TCP Authentication Option (TCP-AO) [RFC5925].

6. IANA Considerations

   IANA maintains a registry of PCEP parameters. A number of IANA
   considerations have been highlighted in previous sections of this
   document. IANA made the allocations as per [RFC6006].

6.1. PCEP TLV Type Indicators

   As described in Section 3.1.2., the P2MP capability TLV allows the
   PCE to advertise its P2MP path computation capability.

   IANA had made an allocation from the "PCEP TLV Type Indicators"
   subregistry, where RFC 6006 was the reference. IANA is requested to
   update the reference as follows to point to this document.

        Value   Description   Reference

        6   P2MP capable   [This I-D]

6.2. Request Parameter Bit Flags

   As described in Section 3.3.1, three RP Object Flags have been
   defined.

   IANA has made an allocations from the PCEP "RP Object Flag Field"
   sub-registry, where RFC 6006 was the reference. IANA is requested to
   update the reference as follows to point to this document.

        Bit   Description   Reference

        18   Fragmentation (F-bit)   [This I-D]
        19   P2MP (N-bit)   [This I-D]
        20   ERO-compression (E-bit)   [This I-D]

6.3. Objective Functions

   As described in Section 3.6.1, two Objective Functions have been
   defined.

   IANA has made an allocations from the PCEP "Objective Function" sub-
   registry, where RFC 6006 was the reference.IANA is requested to
   update the reference as follows to point to this document.

```
   Code Point          Name         Reference

   7                   SPT          [This I-D]
   8                   MCT          [This I-D]
```

6.4. Metric Object Types

   As described in Section 3.6.2, three metric object T fields have been
   defined.

   IANA has made an allocations from the PCEP "METRIC Object T Field"
   sub-registry, where RFC 6006 was the reference. IANA is requested to
   update the reference as follows to point to this document.


```
     Value              Description              Reference

     8                  P2MP IGP metric          [This I-D]
     9                  P2MP TE metric           [This I-D]
     10                 P2MP hop count metric     [This I-D]
```

6.5. PCEP Objects

   As discussed in Section 3.3.2, two END-POINTS Object-Types are
   defined.

   IANA has made the Object-Type allocations from the "PCEP Objects"
   sub-registry, where RFC 6006 was the reference. IANA is requested to
   update the reference as follows to point to this document.

```
     Object-Class Value   4
     Name                 END-POINTS
     Object-Type          3: IPv4
                          4: IPv6
                          5-15: Unassigned
     Reference            [This I-D]
```

   As described in Section 3.2, Section 3.11.1, and Section 3.14, four
   PCEP Object-Classes and six PCEP Object-Types have been defined.

   IANA has made an allocations from the "PCEP Objects" sub- registry,
   where RFC 6006 was the reference. IANA is requested to update the
   reference as follows to point to this document.

```
     Object-Class Value   28
     Name                 UNREACH-DESTINATION
     Object-Type          0: Reserved
                          1: IPv4
```

```
                        2: IPv6
                        3-15: Unassigned
      Reference         [This I-D]


      Object-Class Value  29
      Name                SERO
      Object-Type         0: Reserved
                          1: SERO
                          2-15: Unassigned
      Reference           [This I-D]

      Object-Class Value  30
      Name                SRRO
      Object-Type         0: Reserved
                          1: SRRO
                          2-15: Unassigned
      Reference           [This I-D]

      Object-Class Value  31
      Name                Branch Node Capability Object
      Object-Type         0: Reserved
                          1: Branch node list
                          2: Non-branch node list
                          3-15: Unassigned
      Reference           [This I-D]
```

6.6. PCEP-ERROR Objects and Types

   As described in Section 3.15, number of PCEP-ERROR Object Error Types
   and Values have been defined.

   IANA has made an allocations from the PCEP "PCEP-ERROR Object Error
   Types and Values" sub-registry, where RFC 6006 was the reference.
   IANA is requested to update the reference as follows to point to this
   document.

```
      Error
      Type  Meaning                                         Reference

      5     Policy violation
               Error-value=7:                               [This I-D]
                 P2MP Path computation is not allowed

      16    P2MP Capability Error
               Error-Value=0: Unassigned                    [This I-D]
               Error-Value=1:                               [This I-D]
                 The PCE is not capable to satisfy the request
                 due to insufficient memory
```

```
             Error-Value=2:                          [This I-D]
               The PCE is not capable of P2MP computation

       17    P2MP END-POINTS Error
             Error-Value=0: Unassigned               [This I-D]
             Error-Value=1:                          [This I-D]
               The PCE is not capable to satisfy the request
               due to no END-POINTS with leaf type 2
             Error-Value=2:                          [This I-D]
               The PCE is not capable to satisfy the request
               due to no END-POINTS with leaf type 3
             Error-Value=3:                          [This I-D]
               The PCE is not capable to satisfy the request
               due to no END-POINTS with leaf type 4
             Error-Value=4:                          [This I-D]
               The PCE is not capable to satisfy the request
               due to inconsistent END-POINTS

       18    P2MP Fragmentation Error
             Error-Value=0: Unassigned               [This I-D]
             Error-Value=1:                          [This I-D]
               Fragmented request failure
```

6.7. PCEP NO-PATH Indicator

   As discussed in Section 3.16, NO-PATH-VECTOR TLV Flag Field has been
   defined.

   IANA has made an allocation from the PCEP "NO-PATH-VECTOR TLV Flag
   Field" sub-registry, where RFC 6006 was the reference. IANA is
   requested to update the reference as follows to point to this
   document.

```
       Bit    Description                        Reference

       24     P2MP Reachability Problem          [This I-D]
```

6.8. SVEC Object Flag

   As discussed in Section 3.12, two SVEC Object Flags are defined.

   IANA has made an allocation from the PCEP "SVEC Object Flag Field"
   sub-registry, where RFC 6006 was the reference. IANA is requested to
   update the reference as follows to point to this document.

```
       Bit     Description                        Reference

       19      Partial Path Diverse               [This I-D]
```

         20   Link Direction Diverse   [This I-D]

6.9. OSPF PCE Capability Flag

   As discussed in Section 3.1.1, OSPF Capability Flag is defined to
   indicate P2MP path computation capability.

   IANA has made an assignment from the OSPF Parameters "Path
   Computation Element (PCE) Capability Flags" registry, where RFC 6006
   was the reference. IANA is requested to update the reference as
   follows to point to this document.

         Bit   Description   Reference

         10   P2MP path computation   [This I-D]

7. Acknowledgements

   Authors would like to thank Jonathan Hardwick for providing review
   comments with suggested text.

   Thanks to Deborah Brungard for handling of related errata.

8. References

8.1. Normative References

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

   [RFC3209]   Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V.,
               and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP
               Tunnels", RFC 3209, December 2001.

   [RFC3473]   Berger, L., Ed., "Generalized Multi-Protocol Label
               Switching (GMPLS) Signaling Resource ReserVation
               Protocol-Traffic Engineering (RSVP-TE) Extensions",
               RFC 3473, January 2003.

   [RFC4873]   Berger, L., Bryskin, I., Papadimitriou, D., and A.
               Farrel, "GMPLS Segment Recovery", RFC 4873, May 2007.

   [RFC4875]   Aggarwal, R., Ed., Papadimitriou, D., Ed., and S.
               Yasukawa, Ed., "Extensions to Resource Reservation
               Protocol - Traffic Engineering (RSVP-TE) for Point-to-
               Multipoint TE Label Switched Paths (LSPs)", RFC 4875, May
               2007.

   [RFC5073]   Vasseur, J., Ed., and J. Le Roux, Ed., "IGP Routing
               Protocol Extensions for Discovery of Traffic Engineering
               Node Capabilities", RFC 5073, December 2007.

   [RFC5088]   Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R.
               Zhang, "OSPF Protocol Extensions for Path Computation
               Element (PCE) Discovery", RFC 5088, January 2008.

   [RFC5089]   Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R.
               Zhang, "IS-IS Protocol Extensions for Path Computation
               Element (PCE) Discovery", RFC 5089, January 2008.

   [RFC5511]   Farrel, A., "Routing Backus-Naur Form (RBNF): A Syntax
               Used to Form Encoding Rules in Various Routing Protocol
               Specifications", RFC 5511, April 2009.

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

   [RFC7770] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and
              S. Shaffer, "Extensions to OSPF for Advertising Optional
              Router Capabilities", RFC 7770, February 2016.

8.2. Informative References

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

   [RFC4657] Ash, J., Ed., and J. Le Roux, Ed., "Path Computation
              Element (PCE) Communication Protocol Generic
              Requirements", RFC 4657, September 2006.

   [RFC5671] Yasukawa, S. and A. Farrel, Ed., "Applicability of the
              Path Computation Element (PCE) to Point-to-Multipoint
              (P2MP) MPLS and GMPLS Traffic Engineering (TE)",
              RFC 5671, October 2009.

   [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.

   [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP
              Authentication Option", RFC 5925, June 2010.

   [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, September 2010.

   [RFC7420] Koushik, K., Stephan, E., Zhao, Q., King D., and J.
              Hardwick "PCE communication protocol (PCEP) Management
              Information Base (MIB) Module", RFC 7420, December 2014.

Appendix A. Summary of the RBNF Changes from RFC 6006

   o Update to RBNF for Request message format:

      * Update to the request message to allow for the bundling of
      multiple path computation requests within a single Path
      Computation Request (PCReq) message.

      * Addition of <svec-list> in PCReq message. This object was missed
      in [RFC6006].

      * Addition of BNC object in PCReq message. This object is required
      to support P2MP. It shares the same format as Include Route Object
      (IRO) but it is a different object.

      * Update to the <RRO-List> format, to also allow Secondary Record
      Route object (SRRO). This object was missed in [RFC6006].

      * Removed the BANDWIDTH Object followed by Record Route Object
      (RRO) from <RRO-List>. As BANDWIDTH object doesn't need to follow
      for each RRO in the <RRO-List>, there already exist BANDWIDTH
      object follow <RRO-List> and is backward compatible with
      [RFC5440].

      * Update to the <end-point-rro-pair-list>, to allow optional
      BANDWIDTH object only if <RRO-List> is included.

   o Update the RBNF for Reply message format:

      * Update to the reply message to allow for bundling of multiple
      path computation requests within a single Path Computation Reply
      (PCRep) message.

      * Addition of the UNREACH-DESTINATION in PCRep message. This
      object was missed in [RFC6006].

Contributors

   Jean-Louis Le Roux
   Orange
   2, Avenue Pierre-Marzin
   22307 Lannion Cedex
   France
   EMail: jeanlouis.leroux@orange.com

   Mohamad Chaitou

      France
      EMail: mohamad.chaitou@gmail.com

      Udayasree Palle
      Huawei Technologies
      Divyashree Techno Park, Whitefield
      Bangalore, Karnataka 560066
      India
      EMail: udayasree.palle@huawei.com

Authors' Addresses
    Quintin Zhao
    Huawei Technology
    125 Nagog Technology Park
    Acton, MA   01719
    US
    EMail: quintin.zhao@huawei.com

    Dhruv Dhody
    Huawei Technology
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka   560066
    India
    EMail: dhruv.ietf@gmail.com

    Ramanjaneya Reddy Palleti
    Huawei Technology
    Divyashree Techno Park, Whitefield
    Bangalore, Karnataka   560066
    India
    EMail: ramanjaneya.palleti@huawei.com

    Daniel King
    Old Dog Consulting
    UK
    EMail: daniel@olddog.co.uk

    Fabien Verhaeghe
    Thales Communication France
    160 Bd Valmy 92700 Colombes
    France
    EMail: fabien.verhaeghe@gmail.com

    Tomonori Takeda
    NTT Corporation
    3-9-11, Midori-Cho
    Musashino-Shi, Tokyo 180-8585
    Japan
    EMail: takeda.tomonori@lab.ntt.co.jp

    Julien Meuric
    Orange
    2, Avenue Pierre-Marzin
    22307 Lannion Cedex
    France
    EMail: julien.meuric@orange.com

PCE Working Group   A. Raghuram
Internet-Draft   A. Goddard
Intended status: Standards Track   C. Yadlapalli
Expires: May 3, 2018   AT&T
                                                         J. Karthik
                                                       S. Sivabalan
                                                          J. Parker
                                                 Cisco Systems, Inc.
                                                           D. Dhody
                                                 Huawei Technologies
                                                   October 30, 2017

        Ability for a stateful PCE to request and obtain control of a LSP
                 draft-raghu-pce-lsp-control-request-05

Abstract

   The stateful Path Computation Element (PCE) communication Protocol
   (PCEP) extensions provide stateful control of Multiprotocol Label
   Switching (MPLS) Traffic Engineering Label Switched Paths (TE LSP)
   via PCEP, for a model where a Path Computation Client (PCC) delegates
   control over one or more locally configured LSPs to a stateful PCE.
   There are use-cases in which a stateful PCE may wish to request and
   obtain control of one or more LSPs from a PCC. This document
   describes a simple extension to stateful PCEP to achieve such an
   objective.

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.

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 May 3, 2018.

Copyright Notice

Table of Contents

1. Introduction

   Stateful PCEP extensions [RFC8231] specifies a set of extensions to
   PCEP [RFC5440] to enable stateful control of TE LSPs between and
   across PCEP sessions in compliance with [RFC4657]. It includes
   mechanisms to effect LSP state synchronization between PCCs and PCEs,

delegation of control of LSPs to PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions. The stateful PCEP defines the following two useful network operations:

o Delegation: As per [RFC8051], an operation to grant a PCE temporary rights to modify a subset of LSP parameters on one or more LSPs of a PCC. LSPs are delegated from a PCC to a PCE and are referred to as "delegated" LSPs.

o Revocation: As per [RFC8231], an operation performed by a PCC on a previously delegated LSP. Revocation revokes the rights granted to the PCE in the delegation operation.

For Redundant Stateful PCEs (section 5.7.4. of [RFC8231]), during a PCE failure, one of the redundant PCE could request to take control over an LSP. The redundant PCEs MAY use a local policy or a proprietary election mechanism to decide which PCE would take control. In this case, a mechanism is needed for a stateful PCE to request control of one or more LSPs from a PCC, so that a newly elected primary PCE can request to take over control.

In case of virtualized PCEs (vPCE) running as virtual network function (VNF), as the computation load in the network increases, a new instance of vPCE could be instantiated to balance the current load. The PCEs could use proprietary algorithm to decide which LSPs to be assigned to the new vPCE. Thus having a mechanism for the PCE to request control of some LSPs is needed.

In some deployments, the operator would like to use stateful PCE for global optimization algorithms but would still like to keep the control of the LSP at the PCC. In such cases, a stateful PCE could request to take control during the global optimization and return the delegation once done.

This specification provides a simple extension, by using this a PCE can request control of one or more LSPs from any PCC over the stateful PCEP channel. The procedures for granting and relinquishing control of the LSPs are specified in accordance with the specification [RFC8231].

2. Terminology

   The following terminologies are used in this document:

   PCC: Path Computation Client.

   PCE: Path Computation Element

PCEP: Path Computation Element communication Protocol.

PCRpt: Path Computation State Report message.

PCUpd: Path Computation Update Request message.

PLSP-ID: A PCEP-specific identifier for the LSP.

3. LSP Control Request Flag

The Stateful PCE Request Parameters (SRP) object is defined in
[RFC8231], it includes a Flags field.
[I-D.ietf-pce-pce-initiated-lsp] defines a R (LSP-REMOVE) flag.

A new flag, the "LSP Control Request Flag" (C), is introduced in the
SRP object. On a PCUpd message, a PCE sets the C Flag to 1 to
indicate that, it wishes to gain control of LSP(s). The LSP is
identified by the LSP object. A PLSP-ID of value other than 0 and
0xFFFFF is used to identify the LSP for which the PCE requests
control. The PLSP-ID value of 0 indicates that the PCE is requesting
control of all LSPs originating from the PCC that it wishes to
delegate. The flag has no meaning in the PCRpt and PCInitiate
message and SHOULD be set to 0 on transmission and MUST be ignored on
receipt.

4. Operation

During normal operation, a PCC that wishes to delegate the control of
an LSP sets the D Flag (delegate) to 1 in all PCRpt messages
pertaining to the LSP. The PCE confirms the delegation by setting D
Flag to 1 in all PCUpd messages pertaining to the LSP. The PCC
revokes the control of the LSP from the PCE by setting D Flag to 0 in
PCRpt messages pertaining to the LSP. If the PCE wishes to
relinquish the control of the LSP, it sets D Flag to 0 in all PCUpd
messages pertaining to the LSP.

If a PCE wishes to gain control over an LSP, it sends a PCUpd message
with C Flag set to 1 in SRP object. The LSP for which the PCE
requests control is identified by the PLSP-ID. The PLSP-ID of 0
indicates that the PCE wants control over all LSPs originating from
the PCC. If the LSP(s) is/are already delegated to the PCE making
the request, the PCC ignores the C Flag. A PCC can decide to
delegate the control of the LSP at its own discretion. If the PCC
grants or denies the control, it sends PCRpt message with D Flag set
to 1 and 0 respectively in accordance with according with stateful
PCEP [RFC8231] . If the PCC does not grant the control, it MAY choose
to not respond, and the PCE may choose to retry requesting the

control preferably using exponentially increasing timer. A PCE
ignores the C Flag on the PCRpt message.

In case multiple PCEs request control over an LSP, and if the PCC is
willing to grant the control, the LSP MUST be delegated to only one
PCE chosen by the PCC based on its local policy.

It should be noted that a legacy implementation of PCC, that does not
understand the C flag in PCUpd message, would simply ignore the flag
and the request to grant control over the LSP.

[I-D.ietf-pce-pce-initiated-lsp] describes the setup, maintenance and
teardown of PCE-initiated LSPs under the stateful PCE model. It also
specify how a PCE MAY obtain control over an orphaned LSP that was
PCE-initiated. A PCE implementation can apply the mechanism
described in this document in conjunction with those in
[I-D.ietf-pce-pce-initiated-lsp].

5. Security Considerations

The security considerations listed in [RFC8231] apply to this
document as well. However, this document also introduces a new
attack vectors. An attacker may flood the PCC with request to
delegate all its LSPs at a rate which exceeds the PCC's ability to
process them, either by spoofing messages or by compromising the PCE
itself. The PCC can simply ignore these messages with no extra
actions. Securing the PCEP session using mechanism like Transport
Layer Security (TLS) [RFC8253] is RECOMMENDED.

6. IANA Considerations

This document requests IANA actions to allocate code points for the
protocol elements defined in this document.

6.1. SRP Object Flags

The SRP object is defined in [RFC8231] and the registry to manage the
Flag field of the SRP object is requested in
[I-D.ietf-pce-pce-initiated-lsp]. IANA is requested to make the
following allocation in the aforementioned registry.

Bit	Description	Reference
TBD	LSP Control Request Flag (c-bit)	This document

7. Manageability Considerations

   All manageability requirements and considerations listed in [RFC5440]
   and [RFC8231] apply to PCEP protocol extensions defined in this
   document. In addition, requirements and considerations listed in
   this section apply.

7.1. Control of Function and Policy

   A PCE or PCC implementation SHOULD allow the operator to configure
   the policy based on which it honor the request to control the LSPs.
   Further, the operator MAY be to be allowed to trigger the LSP control
   request at the PCE.

7.2. Information and Data Models

   The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to
   include mechanism to trigger the LSP control request.

7.3. 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].

7.4. 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].

7.5. Requirements On Other Protocols

   Mechanisms defined in this document do not imply any new requirements
   on other protocols.

7.6. Impact On Network Operations

   Mechanisms defined in [RFC5440] and [RFC8231] 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.

8. Acknowledgements

   Thanks to Jonathan Hardwick to remind the authors to not use
   suggested values in IANA section.

9. References

9.1. 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>.

9.2. Informative 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-11 (work in
              progress), October 2017.

   [I-D.ietf-pce-pcep-yang]
              Dhody, D., Hardwick, J., Beeram, V., and j.
              jefftant@gmail.com, "A YANG Data Model for Path
              Computation Element Communications Protocol (PCEP)",
              draft-ietf-pce-pcep-yang-05 (work in progress), June 2017.

   [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>.

   [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>.

   [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>.

Authors' Addresses

   Aswatnarayan Raghuram
   AT&T
   200 S Laurel Aevenue
   Middletown, NJ 07748
   USA

   Email: ar2521@att.com

   Al Goddard
   AT&T
   200 S Laurel Aevenue
   Middletown, NJ 07748
   USA

   Email: ag6941@att.com

   Chaitanya Yadlapalli
   AT&T
   200 S Laurel Aevenue
   Middletown, NJ 07748
   USA

   Email: cy098d@att.com

   Jay Karthik
   Cisco Systems, Inc.
   125 High Street
   Boston, Massachusetts 02110
   USA

   Email: jakarthi@cisco.com

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

      Email: msiva@cisco.com

      Jon Parker
      Cisco Systems, Inc.
      2000 Innovation Drive
      Kanata, Ontario K2K 3E8
      Canada

      Email: jdparker@cisco.com

      Dhruv Dhody
      Huawei Technologies
      Divyashree Techno Park, Whitefield
      Bangalore, Karnataka 560066
      India

      Email: dhruv.ietf@gmail.com

TEAS Working Group   A.Wang
Internet Draft   China Telecom
                                                   Boris Khasanov
                                               Huawei Technologies
                                                Sudhir Cheruathur
                                                 Juniper Networks
                                                         Chun Zhu
                                                      ZTE Company

Intended status: Standard Track   March 9,2017
Expires: September 8, 2017

                  PCEP Extension for Native IP Network
                draft-wang-pce-extension-native-ip-01.txt

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 September8, 201717.

Copyright Notice

Abstract

    This document defines the PCEP extension for PCE application in
    Native IP network. The scenario and architecture of PCE in native IP
    is described in [I-D.draft-wang-teas-pce-native-ip]. This draft
    describes the key information that is transferred between PCE and
    PCC to accomplish the end2end traffic assurance in Native IP network
    under central control mode.

Table of Contents

1. Introduction

    Traditionally, MPLS-TE traffic assurance requires the corresponding
    network devices support MPLS or the complex RSVP/LDP/Segment Routing

etc. technologies to assure the end-to-end traffic performance. But in native IP network, there will be no such signaling protocol to synchronize the action among different network devices. It is necessary to use the central control mode that described in [I-D.draft-ietf-teas-pce-control-function] to correlate the forwarding behavior among different network devices. Draft [I-D.draft-wang-teas-pce-native-ip] describes the architecture and solution philosophy for the end2end traffic assurance in Native IP network via Dual/Multi BGP solution. This draft describes the corresponding PCEP extension to transfer the key information about peer address list, peer prefix association and the explicit peer route on on-path router.

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 RFC 2119 [RFC2119].

3. New Objects Extension

   Three new objects are defined in this draft; they are Peer Address List Object (PAL Object), Peer Prefix Association Object (PPA Object) and Explicit Peer Route object (EPR Object).

   Peer Address List object is used to tell the network device which peer it should be peered with dynamically, Peer Prefix Association is used to tell which prefixes should be advertised via the corresponding peer and Explicit Peer Route object is used to point out which route should be to taken to arrive to the peer.

4. Object Formats.

   Each extension object takes the similar format, that is to say, it began with the common object header defined in [RFC5440] as the following:

```
    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-Class  |   OT  |Res|P|I|     Object Length (bytes)     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
```

```
         //                      (Object body)                      //

         |                                                          |

         +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
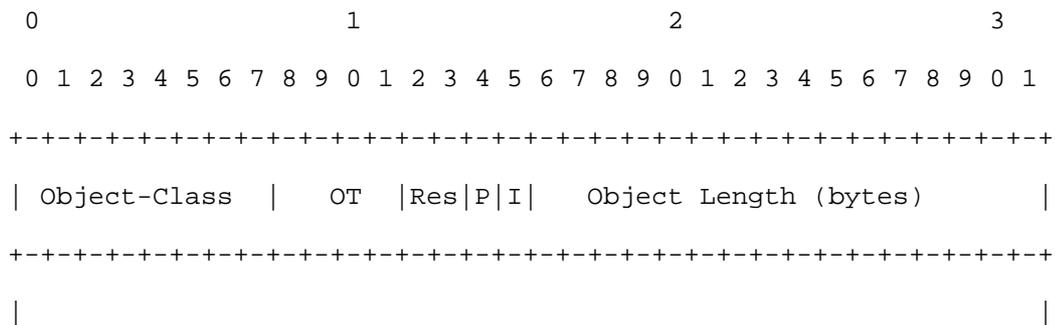
   Different object-class, object type and the corresponding object
   body is defined separated in the following section.

4.1. Peer Address List object.

   The Peer Address List object is used in a PCE Initiate message
   [draft-ietf-pce-pce-initiated-lsp] to specify the ip address of peer
   that the received network device should establish the BGP
   relationship with.

   This Object should only be sent to the head and end router of the
   end2end path in case there is no RR involved. If the RR is used
   between the head end routers, then such information should be sent
   to head router/RR and end router/RR respectively.

   Peer Address List object Object-Class is **

   Peer Address List object Object-Type is **

```
    0                   1                   2                   3

    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

   |  Peer Num    |   Peer-Id    |     AT       |      Resv.

   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

   |               Local IP Address(4/16 Bytes)                   |

   //              Peer IP Address(4/16 Bytes)                    //

   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Peer Num (8 bits): Peer Address Number on the advertised router.

   Peer-Id(8 bits): To distinguish the different peer pair, will be
      referenced in Peer Prefix Association, if the PCE use multi-BGP
      solution for different QoS assurance requirement.

   AT(8 bits): Address Type. To indicate the address type of Peer.
      Equal to 4, if the following IP address of peer is belong to IPv4;
      Equal to 6 if the following IP address of peer is belong to IPv6.

   Resv(8 bits): Reserved for future use.

   Local IP Address(4/16 Bytes): IPv4 address of the local router, used
      to peer with other end router. When AT equal to 4, length is 32bit;
      when AT equal to 16, length is 128bit;

   Peer IP Address(4/16 Bytes): IPv4 address of the peer router, used
      to peer with the local router. When AT equal to 4, length is 32bit;
      IPv6 address of the peer when AT equal to 16, length is 128bit;

4.2. Peer Prefix Association

   THE Peer Prefix Association object is carried within in a PCE
   Initiate message [draft-ietf-pce-pce-initiated-lsp] to specify the
   IP prefixes that should be advertised by the corresponding Peer.

   This Object should only be sent to the head and end router of the
   end2end path in case there is no RR involved. If the RR is used
   between the head end routers, then such information should be sent
   to head router/RR and end router/RR respectively.

   Peer Prefix Association object Object-Class is **

   Peer Prefix Association object Object-Type is **

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1

   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    Peer-Id     |      AT       |     Resv.     | Prefixes Num.
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           Peer Associated IP Prefix TLV                      |

   //          Peer Associated IP Prefix TLV                     //

   |           Peer Associated IP Prefix TLV                      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Peer-Id(8 bits): To indicate which peer should be used to advertise the following IP Prefix TLV. This value is assigned in the Peer Address List object and is referred in this object.

AT(8 bits): Address Type. To indicate the address type of Peer. Equal to 4, if the following IP address of peer is belong to IPv4; Equal to 6 if the following IP address of peer is belong to IPv6.

Resv(8 bits): Reserved for future use.

Prefixes Num(8 bits): Number of prefixes that advertised by the corresponding Peer. It should be equal to num of the following IP prefix TLV.

Peer Associated IP Prefix TLV: Variable Length, use the TLV format to indicate the advertised IP Prefix.

4.3. EXPLICIT PEER ROUTE Object

THE EXPLICIT PEER ROUTE Object is carried in a PCE Initiate message [draft-ietf-pce-pce-initiated-lsp] to specify the explicit peer route to the corresponding peer address on each device that is on the end2end assurance path.

This Object should be sent to all the devices that locates on the end2end assurance path that calculated by PCE.

EXPLICIT PEER ROUTE Object Object-Class is **

EXPLICIT PEER ROUTE Object Object-Type is **

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Peer-Id      |      AT       |         Resv.                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Next Hop Address to the Peer (IPv4/IPv6)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Peer-Id(8 bits): To indicate the peer that the following next hop
    address point to. This value is assigned in the Peer Address List
    object and is referred in this object.

AT(8 bits): Address Type. To indicate the address type of explicit
    peer route. Equal to 4, if the following next hop address to the
    peer is belong to IPv4; Equal to 6 if the following next hop
    address to the peer is belong to IPv6.

Resv(16 bits): Reserved for future use.

Next Hop Address to the Peer TLV: Variable Length, use the TLV
    format to indicate the next hop address to the corresponding peer
    that indicated by the Peer-Id.

5. Management Consideration.

6. Security Considerations

TBD

7. IANA Considerations

TBD

8. Conclusions

TBD

9. References

9.1. Normative References

[RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path

        Computation Element (PCE)-Based Architecture", RFC

        4655, August 2006,<http://www.rfc-editor.org/info/rfc4655>.

[RFC5440]Vasseur, JP., Ed., and JL. Le Roux, Ed., "Path

        Computation Element (PCE) Communication Protocol

        (PCEP)", RFC 5440, March 2009,

Internet-Draft   PCE Extension for Native IP Network   March 8, 2017
                   <http://www.rfc-editor.org/info/rfc5440>.

9.2. Informative References

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

   [I-D.draft-wang-teas-pce-native-ip]
   Aijun Wang, Quintin Zhao, Boris Khasanov, Raghavendra Mallya, Shaofu
   Peng "PCE in Native IP Network", https://tools.ietf.org/html/draft-
   wang-teas-pce-native-ip-02(work in progress), March, 2017

   [I-D.draft-ietf-teas-pce-control-function]
   Farrel, Q.Zhao "An Architecture for use of PCE and PCEP in a Network
   with Central Control"
   https://tools.ietf.org/html/draft-ietf-teas-pce-central-control-01

   (work in progress),December, 2016

10. Acknowledgments

   TBD

Authors' Addresses

   Aijun Wang
   China Telecom
   Beiqijia Town, Changping District
   Beijing,China

   Email: wangaj.bri@chinatelecom.cn

   Boris Khasanov
   Huawei Technologies
   Moskovskiy Prospekt 97A
   St.Petersburg 196084
   Russia

   EMail: khasanov.boris@huawei.com

    Sudhir Cheruathur
    Juniper Networks
    1133 Innovation Way
    Sunnyvale, California 94089 USA

    Email: scheruathur@juniper.net

     Chun Zhu
    ZTE Corporation
    50 Software Avenue, Yuhua District
    Nanjing, Jiangsu 210012
    China
    Email:zhu.chun1@zte.com.cn

TEAS Working Group   A.Wang
Internet Draft   China Telecom
                                                        Quintin Zhao
                                                       Boris Khasanov
                                                          HuaiMo Chen
                                                  Huawei Technologies
                                                          Penghui Mi
                                                      Tencent Company

Intended status: Experimental Track   January 25, 2018
Expires: July 24, 2018

                       PCE in Native IP Network
                   draft-wang-teas-pce-native-ip-07.txt

Status of this Memo

Copyright Notice

Abstract

   This document defines the framework for CCDR traffic engineering
   within Native IP network, using Dual/Multi-BGP session strategy and
   PCE-based central control architecture.

    The proposed central mode control framework conforms to the concept
    that defined in RFC " An Architecture for Use of PCE and the PCE
    Communication Protocol (PCEP) in a Network with Central Control".

    The scenario and simulation results of CCDR traffic engineering is
    described in draft "CCDR Scenario, Simulation and Suggestion".

Table of Contents

1. Introduction

    Draft [I-D.draft-wang-teas-ccdr] describes the scenario and simulation
    results for the CCDR traffic engineering. In summary, the requirements for
    CCDR traffic engineering in Native IP network are the following:
    1) No complex MPLS signaling procedure.
    2) End to End traffic assurance, determined QoS behavior.
    3) Identical deployment method for intra- and inter- domain.
    4) No influence to existing router forward behavior.
    5) Can utilize the power of centrally control(PCE) and
       flexibility/robustness of distributed control protocol.
    6) Coping with the differentiation requirements for large amount
       traffic and prefixes.
    7) Flexible deployment and automation control.

    This document defines the framework for CCDR traffic engineering
    within Native IP network, using Dual/Multi-BGP session strategy and
    CCDR architecture, to meet the above requirements in dynamical and
    central control mode. Future PCEP protocol extensions to transfer the
    key parameters between PCE and the underlying network devices(PCC)
    are provided in draft [draft-wang-pcep-extension-native-IP]

2. Dual-BGP framework for simple topology.

   Dual-BGP framework for simple topology is illustrated in Fig.1, which
   is comprised by SW1, SW2, R1, R2. There are multiple physical links
   between R1 and R2. Traffic between IP11 and IP21 is normal traffic,
   traffic between IP12 and IP22 is priority traffic that should be
   treated differently.

   Only Native IGP/BGP protocol is deployed between R1 and R2. The traffic
   between each address pair may change timely and the corresponding
   source/destination addresses of the traffic may also change dynamically.

   The key idea of the Dual-BGP framework for this simple topology is
   the following:
   1) Build two BGP sessions between R1 and R2, via the different loopback
      address lo0, lo1 on these routers.
   2) Send different prefixes via the two BGP sessions. (For example,
      IP11/IP21 via the BGP pair 1 and IP12/IP22 via the BGP pair 2).
   3) Set the explicit peer route on R1 and R2 respectively for BGP next
      hop of lo0, lo1 to different physical link address between R1 and
      R2.

   So, the traffic between the IP11 and IP21, and the traffic between
   IP12 and IP22 will go through different physical links between R1 and
   R2, each type of traffic occupy the different dedicated physical
   links.

   If there is more traffic between IP12 and IP22 that needs to be
   assured , one can add more physical links on R1 and R2 to reach the
   loopback address lo1(also the next hop for BGP Peer pair2). In this
   cases the prefixes that advertised by two BGP peer need not be
   changed.

   If, for example, there is traffic from another address pair that
   needs to be assured (for example IP13/IP23), but the total volume of
   assured traffic does not exceed the capacity of the previous
   appointed physical links, then one need only to advertise the newly
   added source/destination prefixes via the BGP peer pair2, then the
   traffic between IP13/IP23 will go through the assigned dedicated
   physical links as the traffic between IP12/IP22.

Such decouple philosophy gives the network operator more flexible
control ability on the network traffic, get the determined QoS
assurance effect to meet the application's requirement. No complex
MPLS signal procedures is introduced, the router need only support
native IP protocol.

```
                        |  BGP Peer Pair2  |
                        +------------------+
                        |lo1           lo1 |
                        |                  |
                        |  BGP Peer Pair1  |
                        +------------------+
          IP12          |lo0           lo0 |          IP22
          IP11          |                  |          IP21
          SW1-------R1----------------R2-------SW2
                        Links Group
```

Fig.1 Design Philosophy for Dual-BGP Framework

3. Dual-BGP in large Scale Topology

When the assured traffic spans across one large scale network, as
that illustrated in Fig.2, the dual BGP sessions cannot be
established hop by hop especially for the iBGP within one AS. For
such scenario, we should consider to use the Route Reflector (RR) to
achieve the similar Dual-BGP effect, select one router which performs
the role of RR (for example R3 in Fig.2), every other edge router
will establish two BGP peer sessions with the RR, using their
different loopback addresses respectively. The other two steps for
traffic differentiation are same as one described in the Dual-BGP
simple topology usage case.

For the example shown in Fig.2, if we select the R1-R2-R4-R7 as the
dedicated path, then we should set the explicit peer routes on these
routers respectively, pointing to the BGP next hop (loopback
addresses of R1 and R7, which are used to send the prefix of the
assured traffic) to the actual address of the physical link

```
            +------------R3--------------+
            |                            |
```

```
          SW1-------R1-------R5---------R6-------R7--------SW2
              |         |           |          |
              +-------R2---------R4--------+
```

                 Fig.2 Dual-BGP Framework for large scale network

4. Multi-BGP for Extended Traffic Differentiation

   In general situation, several additional traffic differentiation
   criteria exist, including:
   o Traffic that requires low latency links and is not sensitive to
   packet loss
   o Traffic that requires low packet loss but can endure higher latency
   o Traffic that requires lowest jitter path
   o Traffic that requires high bandwidth links

   These different traffic requirements can be summarized in the
   following table:

```
      +----------+------------+--------------+----------------+
      | Flow No. |   Latency  |  Packet Loss |   Jitter       |
      +----------+------------+--------------+----------------+
      |    1     |    Low     |   Normal     |  Don't care    |
      +----------+------------+--------------+----------------+
      |    2     |   Normal   |    Low       |  Dont't care   |
      +----------+------------+--------------+----------------+
      |    3     |   Normal   |   Normal     |   Low          |
      +----------+------------+--------------+----------------+
```
                 Table 1. Traffic Requirement Criteria

   For Flow No.1, we can select the shortest distance path to carry the
   traffic; for Flow No.2, we can select the idle links to form its end
   to end path; for Flow No.3, we can let all the traffic pass one
   single path, no ECMP distribution on the parallel links is required.

   It is difficult and almost impossible to provide an end-to-end (E2E)
   path with latency, latency variation, packet loss, and bandwidth
   utilization constraints to meet the above requirements in large scale
   IP-based network via the traditional distributed routing protocol,
   but these requirements can be solved using the CCDR architecture
   since the PCE has the overall network view, can collect real network
   topology and network performance information about the underlying

network, select the appropriate path to meet the various network
performance requirements of different traffic type.

5. CCDR based framework for Multi-BGP strategy deployment.

   With the advent of SDN concepts towards pure IP networks, it is
   possible now to accomplish the central and dynamic control of network
   traffic according to the application's various requirements.

   The procedure to implement the dynamic deployment of Multi-BGP
   strategy is the following:
   1) PCE gets topology and link utilization information from the
      underlying network, calculate the appropriate link path upon
      application's requirements.
   2) PCE sends the key parameters to edge/RR routers(R1, R7 and R3 in
      Fig.3) to build multi-BGP peer relations and advertise different
      prefixes via them.
   3) PCE sends the route information to the routers (R1,R2,R4,R7 in
      Fig.3) on forwarding path via PCEP, to build the path to the BGP
      next-hop of the advertised prefixes.
   4) If the assured traffic prefixes were changed but the total volume
      of assured traffic does not exceed the physical capacity of the
      previous end-to-end path, then PCE needs only change the related
      information on edge routers (R1,R7 in Fig.3).
   5) If volume of the assured traffic exceeds the capacity of previous
      calculated path, PCE must recalculate the appropriate path to
      accommodate the exceeding traffic via some new end-to-end physical
      link. After that PCE needs to update on-path routers to build such
      path hop by hop.

```
                          +----+
                 ***********+PCE +*************
                 *         +--*-+           *
                 *          / * \           *
                 *           *             *
             PCEP*           *BGP-LS/SNMP   *PCEP
                 *           *             *
                 *           *        \ * /
            \ * /            *         \ */
             \*/-----------R3--------------*
              |                        |
              |                        |
         SW1-------R1-------R5---------R6-------R7--------SW2
```

```
                   |        |         |        |
                   |        |         |        |
                   +-------R2---------R4--------+
```
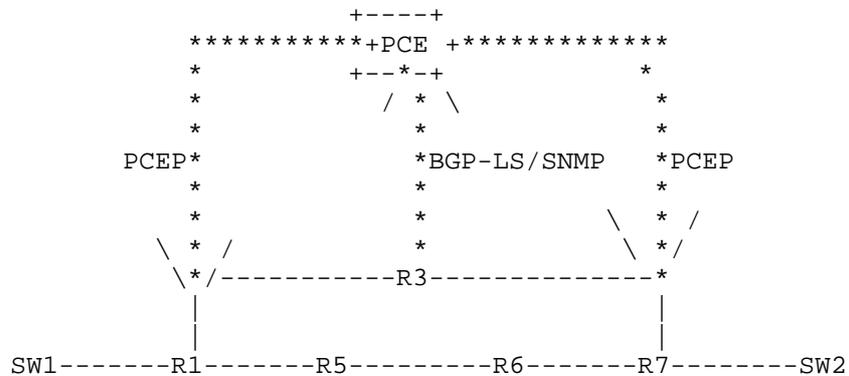
              Fig.3 PCE based framework for Multi-BGP deployment

6. PCEP extension for key parameters delivery.

   The PCEP protocol needs to be extended to transfer the following key
   parameters:
   1) BGP peer address and advertised prefixes.
   2) Explicit route information to BGP next hop of advertised prefixes.

   Once the router receives such information, it should establish the
   BGP session with the peer appointed in the PCEP message, advertise
   the prefixes that contained in the corresponding PCEP message, and
   build the end to end dedicated path hop by hop. Details of
   communications between PCEP and BGP subsystems in router's control
   plane are out of scope of this draft and will be described in
   separate draft.[draft-wang-pce-extension for native IP]

   The reason why we selected PCEP as the southbound protocol instead of
   OpenFlow, is that PCEP is suitable for the changes in control plane
   of the network devices, there OpenFlow dramatically changes the
   forwarding plane. We also think that the level of centralization that
   requires by OpenFlow is hardly achievable in many today's SP networks
   so hybrid BGP+PCEP approach looks much more interesting.

7. CCDR Deployment Consideration

   CCDR framework requires the parallel work of 2 subsystems in router's
   control plane: PCE (PCEP) and BGP as well as coordination between
   them, so it might require additional planning work before deployment.

8.1 Scalability

   In CCDR framework, PCE needs only to influence the edge routers for
   the prefixes differentiation via the multi-BGP deployment. The route
   information for these prefixes within the on-path routers were
   distributed via the traditional BGP protocol. Unlike the solution
   from BGP Flowspec, the on-path router need only keep the specific
   policy routes to the BGP next-hop of the differentiate prefixes, not

   the specific routes to the prefixes themselves. This can lessen the
   burden from the table size of policy based routes for the on-path
   routers, and has more scalability when comparing with the solution
   from BGP flowspec or Openflow.

8.2 High Availability

   CCDR framework is based on the traditional distributed IP protocol.
   If the PCE failed, the forwarding plane will not be impacted, as the
   BGP session between all devices will not flap, and the forwarding
   table will remain the same. If one node on the optimal path is failed,
   the assurance traffic will fall over to the best-effort forwarding
   path. One can even design several assurance paths to load balance/hot
   standby the assurance traffic to meet the path failure situation, as
   done in MPLS FRR.
   From PCE/SDN-controller HA side we will rely on existing HA solutions
   of SDN controllers such as clustering.

8.3 Incremental deployment

   Not every router within the network support will support the PCEP
   extension that defined in [draft-wang-pce-extension-native-IP]
   simultaneously. For such situations, router on the edge of sub domain
   can be upgraded first, and then the traffic can be assured between
   different sub domains. Within each sub domain, the traffic will be
   forwarded along the best-effort path. Service provider can
   selectively upgrade the routers on each sub-domain in sequence.

8. Security Considerations

   TBD

9. IANA Considerations

   TBD

10. Conclusions

   TBD

11. References

11.1. Normative References

   [RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path

             Computation Element (PCE)-Based Architecture", RFC

             4655, August 2006,<http://www.rfc-editor.org/info/rfc4655>.

   [RFC5440]Vasseur, JP., Ed., and JL. Le Roux, Ed., "Path

             Computation Element (PCE) Communication Protocol

             (PCEP)", RFC 5440, March 2009,

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

   [RFC8283] A.Farrel, Q.Zhao et al.," An Architecture for Use of PCE
   and the PCE Communication Protocol (PCEP) in a Network with Central
   Control", [RFC8283], December 2017

11.2. Informative References

   [I-D.draft-wang-teas-ccdr]

   A.Wang, X.Huang et al. "CCDR Scenario, Simulation and Suggestion"

   https://datatracker.ietf.org/doc/draft-wang-teas-ccdr/

   [I-D. draft-ietf-teas-pcecc-use-cases]

   Quintin Zhao, Robin Li, Boris Khasanov et al. "The Use Cases for
   Using PCE as the Central Controller(PCECC) of LSPs

   https://tools.ietf.org/html/draft-ietf-teas-pcecc-use-cases-00

   March,2017

   [draft-wang-pcep-extension for native IP]

    Aijun Wang, Boris Khasanov et al. "PCEP Extension for Native IP
    Network" https://datatracker.ietf.org/doc/draft-wang-pce-extension-
    native-ip/

12. Acknowledgments

    The authors would like to thank George Swallow, Xia Chen, Jeff
    Tantsura,Scharf Michael,Daniele Ceccarelli and Dhruv Dhody for their
    valuable comments and suggestions.

    The authors would also like to thank Lou Berger, Adrian Farrel,
    Vishnu Pavan Beeram, Deborah Brungard and King Daniel for their
    suggestions to put forward this draft.

Authors' Addresses

    Aijun Wang
    China Telecom
    Beiqijia Town, Changping District
    Beijing,China

    Email: wangaj.bri@chinatelecom.cn

Quintin Zhao
Huawei Technologies
125 Nagog Technology Park
Acton, MA 01719
USA

EMail: quintin.zhao@huawei.com

Boris Khasanov
Huawei Technologies
Moskovskiy Prospekt 97A
St.Petersburg 196084
Russia

EMail: khasanov.boris@huawei.com

Huaimo Chen
Huawei Technologies
Boston, MA,
USA

EMail: huaimo.chen@huawei.com

Penghui Mi
Tencent
Tencent Building, Kejizhongyi Avenue,
Hi-techPark, Nanshan District,Shenzhen 518057, P.R.China

Email kevinmi@tencent.com

Raghavendra Mallya
Juniper Networks
1133 Innovation Way
Sunnyvale, California 94089 USA

Email: rmallya@juniper.net

Shaofu Peng
ZTE Corporation
No.68 Zijinghua Road,Yuhuatai District
Nanjing 210012
China

Email: peng.shaofu@zte.com.cn