Internet Engineering Task Force H. Chen
Internet-Draft Huawei Technologies
Intended status: Standards Track March 12, 2011
Expires: September 13, 2011

Extensions to the Path Computation Element Communication Protocol (PCEP) for Backup Egress of a Traffic Engineering Label Switched Path
draft-chen-pce-compute-backup-egress-01.txt

Abstract

This document presents extensions to the Path Computation Element Communication Protocol (PCEP) for a PCC to send a request for computing a backup egress for an MPLS TE P2MP LSP or an MPLS TE P2P LSP to a PCE and for a PCE to compute the backup egress and reply to the PCC with a computation result for the backup egress.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

"A Path Computation Element-(PCE) Based Architecture" RFC4655 describes a set of building blocks for constructing solutions to compute Point-to-Point (P2P) Traffic Engineering (TE) label switched paths across multiple areas or Autonomous System (AS) domains. A typical PCE-based system comprises one or more path computation servers, traffic engineering databases (TED), and a number of path computation clients (PCC). A routing protocol is used to exchange traffic engineering information from which the TED is constructed. A PCC sends a Point-to-Point traffic engineering Label Switched Path (LSP) computation request to the path computation server, which uses the TED to compute the path and responses to the PCC with the computed path. A path computation server is named as a PCE. The communications between a PCE and a PCC for Point-to-Point label switched path computations follow the PCE communication protocol (PCEP).

"Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths" RFC6006 describes extensions to the PCE communication Protocol (PCEP) to handle requests and responses for the computation of paths for P2MP TE LSPs.

This document defines extensions to the Path Computation Element Communication Protocol (PCEP) for a PCC to send a request for computing a backup egress node for an MPLS TE P2MP LSP or an MPLS TE P2P LSP to a PCE and for a PCE to compute the backup egress node and reply to the PCC with a computation result for the backup egress node.

2. Terminology

This document uses terminologies defined in RFC5440, and RFC4875.

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 RFC 2119.

4. Extensions to PCEP

This section describes the extensions to PCEP for computing a backup egress of an MPLS TE P2MP LSP and an MPLS TE P2P LSP.

4.1. Backup Egress Capability Advertisement

4.1.1. Capability TLV in Existing PCE Discovery Protocol

There are two options for advertising a PCE capability for computing a backup egress for an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

The first option is to define a new flag in the OSPF and IS-IS PCE Capability Flags to indicate the capability that a PCE is capable to compute both a backup egress for an MPLS TE P2MP LSP and a backup egress for an MPLS TE P2P LSP.

The second option is to define two new flags. One new flag in the OSPF and IS-IS PCE Capability Flags indicates the capability that a PCE is capable to compute a backup egress for an MPLS TE P2MP LSP; and another new flag in the OSPF and IS-IS PCE Capability Flags indicates the capability that a PCE is capable to compute a backup egress for an MPLS TE P2P LSP.

This second option is preferred now.

The format of the PCE-CAP-FLAGS sub-TLV is as follows:

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |              Type = 5         |             Length            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
//                   PCE Capability Flags                       //
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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

The following capability bits have been assigned by IANA:

Bit	Capabilities
0	Path computation with GMPLS link constraints
1	Bidirectional path computation
2	Diverse path computation
3	Load-balanced path computation
4	Synchronized path computation
5	Support for multiple objective functions
6	Support for additive path constraints (max hop count, etc.)
7	Support for request prioritization
8	Support for multiple requests per message
9	Global Concurrent Optimization (GCO)
10	P2MP path computation
11-31	Reserved for future assignments by IANA.

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

For the second option, one bit such as bit 13 may be assigned to indicate that a PCE is capable to compute a backup egress for an MPLS TE P2MP LSP and another bit such as bit 14 may be assigned to indicate that a PCE is capable to compute a backup egress for an MPLS TE P2P LSP.

Bit	Capabilities
13	Backup egress computation for P2MP LSP
14	Backup egress computation for P2P LSP
15-31	Reserved for future assignments by IANA.

4.1.2. Open Message Extension

If a PCE does not advertise its backup egress compution capability during discovery, PCEP should be used to allow a PCC to discover, during the Open Message Exchange, which PCEs are capable of supporting backup egress computation.

To achieve this, we extend the PCEP OPEN object by defining a new optional TLV to indicate the PCE's capability to perform backup egress computation for an MPLS TE P2MP LSP and an MPLS TE P2P LSP.

We request IANA to allocate a value such as 8 from the "PCEP TLV Type Indicators" subregistry, as documented in Section below ("Backup Egress Capability TLV"). The description is "backup egress capable", and the length value is 2 bytes. The value field is set to indicate the capability of a PCE for backup egress computation for an MPLS TE LSP in details.

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

The inclusion of this TLV in an OPEN object indicates that the sender can perform backup egress computation for an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

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

4.2. Request and Reply Message Extension

This section describes extensions to the existing RP (Request Parameters) object to allow a PCC to request a PCE for computing a backup egress of an MPLS TE P2MP LSP or an MPLS TE P2P LSP when the PCE receives the PCEP request.

4.2.1. RP Object Extension

The following flags are added into the RP Object:

The T bit is added in the flag bits field of the RP object to tell the receiver of the message that the request/reply is for computing a bcakup egress of an MPLS TE P2MP LSP and an MPLS TE P2P LSP.

- o T (Backup Egress bit - 1 bit):

  0: This indicates that this is not PCReq/PCRep for backup egress.

  1: This indicates that this is PCReq or PCRep message for backup egress.

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

This T bit with the N bit defined in RFC 6006 can indicate whether a request/reply is for a bcakup egress of an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

- o T = 1 and N = 1: This indicates that this is a PCReq/PCRep message for backup egress of an MPLS TE P2MP LSP.

- o T = 1 and N = 0: This indicates that this is a PCReq/PCRep message for backup egress of an MPLS TE P2P LSP.

4.2.2. External Destination Nodes Object

In addition to the information about the path that an MPLS TE P2MP LSP or an MPLS TE P2P LSP traverses, a request message may comprise other information that may be used for computing the backup egress for the P2MP LSP or P2P LSP. For example, the information about an external destination node, to which data traffic is delivered from an egress node of the P2MP LSP or P2P LSP, is useful for computing a backup egress node.

The PCC can specify an external destination nodes (EDN) Object. In order to represent the external destination nodes efficiently, we define two types of encodes for the external destination nodes in the object.

One encode indicates that the EDN object contains an external destination node for every egress node of an MPLS TE P2MP LSP or an MPLS TE P2P LSP. The order of the external destination nodes in the object is the same as the egress node(s) of the P2MP LSP or P2P LSP contained in the PCE messages.

Another encode indicates that the EDN object contains a list of egress node and external destination node pairs. For an egress node and external destination node pair, the data traffic is delivered to the external destination node from the egress node of the LSP.

The format of the external destination nodes (EDN) object boby for IPv4 with the first type of encodes is illustrated 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Encode of External Destination Nodes (1)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              External  Destination  IPv4 address              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              External  Destination  IPv4 address              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                            ...                                ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              External  Destination  IPv4 address              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Format of EDN Object with one Encode for IPv4

The format of the external destination nodes (EDN) object boby for IPv4 with the second type of encodes 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
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |            Encode of External Destination Nodes (2)           |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                         Egress IPv4 address                   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                External Destination IPv4 address              |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                         Egress IPv4 address                   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                External Destination IPv4 address              |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       ~                              ...                              ~
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                         Egress IPv4 address                   |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                External Destination IPv4 address              |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Format of EDN Object with another Encode for IPv4

The format of the external destination nodes (EDN) object boby for IPv6 with the first type of encodes is illustrated 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
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |            Encode of External Destination Nodes (1)           |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                               |
       |           External Destination IPv6 address (16 bytes)        |
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                               |
       |           External Destination IPv6 address (16 bytes)        |
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       ~                              ...                              ~
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                               |
       |           External Destination IPv6 address (16 bytes)        |
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: Format of EDN Object with one Encode for IPv6

The format of the external destination nodes (EDN) object boby for IPv6 with the second type of encodes 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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |           Encode of External Destination Nodes (2)            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |                       Egress IPv6 address                     |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |          External Destination IPv6 address (16 bytes)         |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |                       Egress IPv6 address                     |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |          External Destination IPv6 address (16 bytes)         |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    ~                              ...                              ~
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |                       Egress IPv6 address                     |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    |          External Destination IPv6 address (16 bytes)         |
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Format of EDN Object with another Encode for IPv6

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

The Object-Class and Object-types will need to be allocated by IANA. The IANA request is documented in Section below (PCEP Objects).

4.2.3. Constraints between Egress and Backup Egress

A request message sent to a PCE from a PCC for computing a backup egress of an MPLS TE P2MP LSP or an MPLS TE P2P LSP may comprise a constraint indicating that there must be a path from the backup egress node to be computed to the egress node of the P2MP LSP or P2P LSP and that the length of the path is within a given hop limit such as one hop.

This constraint can be considered as default by a PCE or explicitly sent to the PCE by a PCC [TBD].

4.2.4. Constraints for Backup Path

A request message sent to a PCE from a PCC for computing a backup egress of a P2MP LSP or P2P LSP may comprise a constraint indicating that the backup egress node to be computed may not be a node on the P2MP LSP or P2P LSP. In addition, the request message may comprise a list of nodes, each of which is a candidate for the backup egress node.

A request message sent to a PCE from a PCC for computing a backup egress of a P2MP LSP or P2P LSP may comprise a constraint indicating that there must be a path from the previous hop node of the egress node of the P2MP LSP or P2P LSP to the backup egress node to be computed and that there is not an internal node of the path from the previous hop node of the egress node of the P2MP LSP or P2P LSP to the backup egress that is on the path of the P2MP LSP or P2P LSP.

Most of these constraints for the backup path can be considered as default by a PCE. The constraints for the backup path may be explicitly sent to the PCE by a PCC [TBD].

4.2.5. Backup Egress Node

The PCE may send a reply message to the PCC in return to the request message for computing a new backup egress node or a number of backup egress nodes. The reply message may comprise information about the computed backup egress node(s), which is contained in the path(s) from the previous-hop node of the egress node of the P2MP LSP or P2P LSP to the backup egress node(s) computed.

4.2.6. Backup Egress PCEP Error Objects and Types

In some cases, the PCE may not complete the backup egress computation as requested, for example based on a set of constraints. As such, the PCE may send a reply message to the PCC that indicates an unsuccessful backup egress computation attempt. The reply message may comprise a PCEP-error object, which may comprise an error-value, error-type and some detail information.

4.2.7. Request Message Format

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

Below is the message format for a request message:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request>
<request>::= <RP>
             <end-point-rro-pair-list>
             [<OF>]
             [<LSPA>]
             [<BANDWIDTH>]
             [<metric-list>]
             [<EDNO>]
             [<IRO>]
             [<LOAD-BALANCING>]
    where:
           <EDNO> is an external destination nodes object.
```

Figure 5: The Format for a Request Message

The definitions for svec-list, RP, end-point-rro-pair-list, OF, LSPA, BANDWIDTH, metric-list, IRO, and LOAD-BALANCING are described in RFC5440 and RFC6006.

4.2.8. Reply Message Format

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

Below is the message format for a reply message:

```
               <PCRep Message>::= <Common Header>
                                  <response>
               <response>::= <RP>
                             <end-point-path-pair-list>
                             [<NO-PATH>]
                             [<attribute-list>]
       where:

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


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

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

Figure 6: The Format for a Reply Message

The definitions for RP, NO-PATH, END-POINTS, OF, LSPA, BANDWIDTH, metric-list, IRO, and SERO are described in RFC5440, RFC6006 and RFC4875.

5. Security Considerations

The mechanism described in this document does not raise any new security issues for the PCEP, OSPF or IS-IS protocols.

6. IANA Considerations

This section specifies requests for IANA allocation.

6.1. Backup Egress Capability Flag

Two new OSPF Capability Flags are defined in this document to indicate the capabilities for computing a backup egress for an MPLS TE P2MP LSP and an MPLS TE P2P LSP. IANA is requested to make the assignment from the "OSPF Parameters Path Computation Element (PCE) Capability Flags" registry:

Bit	Description	Reference
13	Backup egress for P2MP LSP	This I-D
14	Backup egress for P2P LSP	This I-D

6.2. Backup Egress Capability TLV

A new backup egress capability TLV is defined in this document to allow a PCE to advertize its backup egress computation capability. IANA is requested to make the following allocation from the "PCEP TLV Type Indicators" sub-registry.

Value	Description	Reference
8	Backup egress capable	This I-D

6.3. Request Parameter Bit Flags

A new RP Object Flag has been defined in this document. IANA is requested to make the following allocation from the "PCEP RP Object Flag Field" Sub-Registry:

Bit	Description	Reference
15	Backup egress (T-bit)	This I-D

6.4. PCEP Objects

An External Destination Nodes Object-Type is defined in this document. IANA is requested to make the following Object-Type allocation from the "PCEP Objects" sub-registry:

Object-Class Value	34
Name	External Destination Nodes
Object-Type	1: IPv4
	2: IPv6
	3-15: Unassigned
Reference	This I-D

7. Acknowledgement

The author would like to thank Quintin Zhao and others for their valuable comments on this draft.

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.

[RFC4090] Pan, P., Swallow, G., and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, May 2005.

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

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

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

8.2. Informative References

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

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

Author's Address

Huaimo Chen
Huawei Technologies
Boston, MA
US

Email: Huaimochen@huawei.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: September 13, 2011

H. Chen
Huawei Technologies
March 12, 2011

Extensions to the Path Computation Element Communication Protocol (PCEP) for Backup Ingress of a Traffic Engineering Label Switched Path

draft-chen-pce-compute-backup-ingress-01.txt

Abstract

This document presents extensions to the Path Computation Element Communication Protocol (PCEP) for a PCC to send a request for computing a backup ingress for an MPLS TE P2MP LSP or an MPLS TE P2P LSP to a PCE and for a PCE to compute the backup ingress and reply to the PCC with a computation result for the backup ingress.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

"Fast Reroute Extensions to RSVP-TE for LSP Tunnels" RFC4090 describes two methods to backup P2P LSP tunnels or paths at local repair points. The local repair points may comprise a number of intermediate nodes between an ingress node and an egress node along the path. The first method is a one-to-one backup method, where a detour backup P2P LSP for each protected P2P LSP is created at each potential point of local repair. The second method is a facility bypass backup protection method, where a bypass backup P2P LSP tunnel is created using MPLS label stacking to protect a potential failure point for a set of P2P LSP tunnels. The bypass backup tunnel can protect a set of P2P LSPs that have similar backup constraints.

"Extensions to RSVP-TE for P2MP TE LSPs" RFC4875 describes how to use the one-to-one backup method and facility bypass backup method to protect a link or intermediate node failure on the path of a P2MP LSP.

However, there is no mention of locally protecting an ingress node failure in a protected P2MP LSP or P2P LSP.

The methods for protecting an ingress node of a P2MP LSP or P2P LSP may be classified into two categories.

A first category uses a backup P2MP LSP that is from a backup ingress node to the number of destination nodes for the P2MP LSP, and a backup P2P LSP that is from a backup ingress node to the destination node for the P2P LSP. The disadvantages of this class of methods include more network resource such as computer power and link bandwidth consumption since the backup P2MP LSP or P2P LSP is from the backup ingress node to the number of destination nodes or the destination respectively.

A second category uses a local P2MP LSP or P2P LSP for protecting the ingress of a P2MP LSP or P2P LSP locally. The local P2MP LSP is from a backup ingress node to the next hop nodes of the ingress of the P2MP LSP. The local P2P LSP is from a backup ingress node to the next hop node of the ingress of the P2P LSP. It is desirable to let PCE compute these backup ingress nodes.

This document defines extensions to the Path Computation Element Communication Protocol (PCEP) for a PCC to send a request for computing a backup ingress node for an MPLS TE P2MP LSP or an MPLS TE P2P LSP to a PCE and for a PCE to compute the backup ingress node and reply to the PCC with a computation result for the backup ingress node.

2. Terminology

This document uses terminologies defined in RFC5440, RFC4090, and RFC4875.

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. Extensions to PCEP

This section describes the extensions to PCEP for computing a backup ingress of an MPLS TE P2MP LSP and an MPLS TE P2P LSP.

4.1. Backup Ingress Capability Advertisement

4.1.1. Capability TLV in Existing PCE Discovery Protocol

There are a couple of options for advertising a PCE capability for computing a backup ingress for an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

The first option is to define a new flag in the OSPF and ISIS PCE Capability Flags to indicate the capability that a PCE is capable to compute both a backup ingress for an MPLS TE P2MP LSP and a backup ingress for an MPLS TE P2P LSP.

The second option is to define two new flags. One new flag in the OSPF and ISIS PCE Capability Flags indicates the capability that a PCE is capable to compute a backup ingress for an MPLS TE P2MP LSP; and another new flag in the OSPF and ISIS PCE Capability Flags indicates the capability that a PCE is capable to compute a backup ingress for an MPLS TE P2P LSP.

This second option is preferred now.

The format of the PCE-CAP-FLAGS sub-TLV is as follows:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |             Type = 5          |              Length           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
   //                 PCE Capability Flags                         //
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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

The following capability bits have been assigned by IANA:

Bit	Capabilities
0	Path computation with GMPLS link constraints
1	Bidirectional path computation
2	Diverse path computation
3	Load-balanced path computation
4	Synchronized path computation
5	Support for multiple objective functions
6	Support for additive path constraints (max hop count, etc.)
7	Support for request prioritization
8	Support for multiple requests per message
9	Global Concurrent Optimization (GCO)
10	P2MP path computation
11-31	Reserved for future assignments by IANA.

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

For the second option, one bit such as bit 11 may be assigned to indicate that a PCE is capable to compute a backup ingress for an MPLS TE P2MP LSP and another bit such as bit 12 may be assigned to indicate that a PCE is capable to compute a backup ingress for an MPLS TE P2P LSP.

Bit	Capabilities
11	Backup ingress computation for P2MP LSP
12	Backup ingress computation for P2P LSP
13-31	Reserved for future assignments by IANA.

4.1.2. Open Message Extension

If a PCE does not advertise its backup ingress compution capability during discovery, PCEP should be used to allow a PCC to discover, during the Open Message Exchange, which PCEs are capable of supporting backup ingress computation.

To achieve this, we extend the PCEP OPEN object by defining a new optional TLV to indicate the PCE's capability to perform backup ingress compution for an MPLS TE P2MP LSP and an MPLS TE P2P LSP.

We request IANA to allocate a value such as 8 from the "PCEP TLV Type Indicators" subregistry, as documented in Section below ("Backup Ingress Capability TLV"). The description is "backup ingress capable", and the length value is 2 bytes. The value field is set to indicate the capability of a PCE for backup ingress compution for an MPLS TE LSP in details.

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

The inclusion of this TLV in an OPEN object indicates that the sender can perform backup ingress compution for an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

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

4.2. Request and Reply Message Extension

This section describes extensions to the existing RP (Request Parameters) object to allow a PCC to request a PCE for computing a backup ingress of an MPLS TE P2MP LSP or an MPLS TE P2P LSP when the PCE receives the PCEP request.

4.2.1. RP Object Extension

The following flags are added into the RP Object:

The I bit is added in the flag bits field of the RP object to tell the receiver of the message that the request/reply is for computing a bcakup ingress of an MPLS TE P2MP LSP and an MPLS TE P2P LSP.

- o I (Backup Ingress bit - 1 bit):

  0: This indicates that this is not PCReq/PCRep for backup ingress.

  1: This indicates that this is PCReq or PCRep message for backup ingress.

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

This I bit with the N bit defined in RFC6006 can indicate whether the request/reply is for a bcakup ingress of an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

- o I = 1 and N = 1: This indicates that this is a PCReq/PCRep message for backup ingress of an MPLS TE P2MP LSP.

- o I = 1 and N = 0: This indicates that this is a PCReq/PCRep message for backup ingress of an MPLS TE P2P LSP.

4.2.2. External Source Node Object

In addition to the information about the path that an MPLS TE P2MP LSP or an MPLS TE P2P LSP traverses, a request message may comprise other information that may be used for computing the backup ingress for the P2MP LSP or P2P LSP. For example, the information about an external source node, from which data traffic is delivered to the ingress node of the P2MP LSP or P2P LSP and transported to the egress node(s) via the P2MP LSP or P2P LSP, is useful for computing a backup ingress node.

The PCC can specify an external source node (ESN) Object. The ESN Object has the same format as the IRO object defined in [RFC5440] except that it only supports IPv4 and IPv6 prefix sub-objects.

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

The Object-Class and Object-types will need to be allocated by IANA. The IANA request is documented in Section below. (PCEP Objects).

4.2.3. Constraints between Ingress and Backup Ingress

A request message sent to a PCE from a PCC for computing a backup ingress of an MPLS TE P2MP LSP or an MPLS TE P2P LSP may comprise a constraint indicating that there must be a path from the backup ingress node to be computed to the ingress node of the P2MP LSP or P2P LSP and that the length of the path is within a given hop limit such as one hop.

This constraint can be considered as default by a PCE or explicitly sent to the PCE by a PCC [TBD].

4.2.4. Constraints for Backup Path

A request message sent to a PCE from a PCC for computing a backup ingress of a P2MP LSP or P2P LSP may comprise a constraint indicating that the backup ingress node to be computed may not be a node on the P2MP LSP or P2P LSP. In addition, the request message may comprise a list of nodes, each of which is a candidate for the backup ingress node.

A request message sent to a PCE from a PCC for computing a backup ingress of a P2MP LSP or P2P LSP may comprise a constraint indicating that there must be a path from the backup ingress node to be computed to the next-hop nodes of the ingress node of the P2MP LSP or P2P LSP and that there is not an internal node of the path from the backup ingress to the next-hop nodes on the P2MP LSP or P2P LSP .

Most of these constraints for the backup path can be considered as default by a PCE. The constraints for the backup path may be explicitly sent to the PCE by a PCC [TBD].

4.2.5. Backup Ingress Node

The PCE may send a reply message to the PCC in return to the request message for computing a new backup ingress node. The reply message may comprise information about the computed backup ingress node, which is contained in the path from the backup ingress computed to the next-hop node(s) of the ingress node of the P2MP LSP or P2P LSP.

The backup ingress node is the root or head node of the backup path computed.

4.2.6. Backup Ingress PCEP Error Objects and Types

In some cases, the PCE may not complete the backup ingress computation as requested, for example based on a set of constraints. As such, the PCE may send a reply message to the PCC that indicates an unsuccessful backup ingress computation attempt. The reply message may comprise a PCEP-error object, which may comprise an error-value, error-type and some detail information.

4.2.7. Request Message Format

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

Below is the message format for a request message:

```
            <PCReq Message>::= <Common Header>
                               [<svec-list>]
                               <request>
            <request>::= <RP>
                         <end-point-rro-pair-list>
                         [<OF>]
                         [<LSPA>]
                         [<BANDWIDTH>]
                         [<metric-list>]
                         [<ESNO>]
                         [<IRO>]
                         [<LOAD-BALANCING>]
       where:
             <ESNO> is an external source node object.
```

Figure 1: The Format for a Request Message

The definitions for svec-list, RP, end-point-rro-pair-list, OF, LSPA, BANDWIDTH, metric-list, IRO, and LOAD-BALANCING are described in RFC5440 and RFC6006.

4.2.8. Reply Message Format

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

Below is the message format for a reply message:

```
               <PCRep Message>::= <Common Header>
                                  <response>
               <response>::= <RP>
                             <end-point-path-pair-list>
                             [<NO-PATH>]
                             [<attribute-list>]
       where:

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


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

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

Figure 2: The Format for a Reply Message

The definitions for RP, NO-PATH, END-POINTS, OF, LSPA, BANDWIDTH, metric-list, IRO, and SERO are described in RFC5440, RFC6006 and RFC4875.

5. Security Considerations

The mechanism described in this document does not raise any new security issues for the PCEP, OSPF and IS-IS protocols.

6. IANA Considerations

This section specifies requests for IANA allocation.

6.1. Backup Ingress Capability Flag

Two new OSPF Capability Flags are defined in this document to indicate the capabilities for computing a backup ingress for an MPLS TE P2MP LSP and an MPLS TE P2P LSP. IANA is requested to make the assignment from the "OSPF Parameters Path Computation Element (PCE) Capability Flags" registry:

Bit	Description	Reference
11	Backup ingress for P2MP LSP	This I-D
12	Backup ingress for P2P LSP	This I-D

6.2. Backup Ingress Capability TLV

A new backup ingress capability TLV is defined in this document to allow a PCE to advertize its backup ingress computation capability. IANA is requested to make the following allocation from the "PCEP TLV Type Indicators" sub-registry.

Value	Description	Reference
8	Backup ingress capable	This I-D

6.3. Request Parameter Bit Flags

A new RP Object Flag has been defined in this document. IANA is requested to make the following allocation from the "PCEP RP Object Flag Field" Sub-Registry:

Bit	Description	Reference
16	Backup ingress (I-bit)	This I-D

6.4. PCEP Objects

An External Source Node Object-Type is defined in this document. IANA is requested to make the following Object-Type allocation from the "PCEP Objects" sub-registry:

Object-Class Value	33
Name	External Source Node
Object-Type	1: IPv4
	2: IPv6
	3-15: Unassigned
Reference	This I-D

7. Acknowledgement

The author would like to thank Quintin Zhao and others for their valuable comments on this draft.

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.

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

[RFC4090] Pan, P., Swallow, G., and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, May 2005.

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

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

8.2. Informative References

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

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

Author's Address

Huaimo Chen
Huawei Technologies
Boston, MA
US

Email: Huaimochen@huawei.com

Internet Engineering Task Force H. Chen
Internet-Draft Huawei Technologies
Intended status: Standards Track March 7, 2011
Expires: September 8, 2011

A Fordward-Search P2MP TE LSP Inter-Domain Path Computation
draft-chen-pce-forward-search-p2mp-path-00.txt

Abstract

This document presents a forward search procedure for computing Point-to-MultiPoint (P2MP) Traffic Engineering (TE) Label Switched Paths (LSPs) crossing a number of domains through using multiple Path Computation Elements (PCEs). In addition, extensions to the Path Computation Element Communication Protocol (PCEP) for supporting the forward search procedure are described.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

This document describes a new method called "Forward Search Shortest P2MP LSP Path Crossing Domains". The major characteristics of this method for computing an MPLS TE P2MP LSP path from a source node to a number of destination nodes crossing multiple domains include the following three ones.

(1) It guarantees that the path computed from the source node to the destination nodes is shortest.

(2) It does not depend on any domain path tree or domain sequences from the source node to the destination nodes.

(3) Navigating a mesh of domains is simple and efficient.

2. Terminology

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

ASBR: Autonomous System Border Router. Routers used to connect together ASes of the same or different service providers via one or more inter-AS links.

Boundary Node (BN): a boundary node is either an ABR in the context of inter-area Traffic Engineering or an ASBR in the context of inter-AS Traffic Engineering.

Entry BN of domain(n): a BN connecting domain(n-1) to domain(n) along a determined sequence of domains.

Exit BN of domain(n): a BN connecting domain(n) to domain(n+1) along a determined sequence of domains.

Inter-area TE LSP: A TE LSP that crosses an IGP area boundary.

Inter-AS TE LSP: A TE LSP that crosses an AS boundary.

LSP: Label Switched Path.

LSR: Label Switching Router.

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.

PCE(i) is a PCE with the scope of domain(i).

TED: Traffic Engineering Database.

This document uses terminologies defined in RFC5440.

3. Conventions Used in This Document

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

4. Forward Search P2MP Path Computation

This section gives an overview of the forward search P2MP path computation procedure and describes the procedure in details.

4.1. Overview of Procedure

Simply speaking, the idea of the Forward Search P2MP inter-domain path computation method for computing a path for an MPLS TE P2MP LSP crossing multiple domains from a source node to a number of destination nodes includes:

Start from the source node and the source domain.

Consider the optimal path segment from the source node to every exit boundary node of the source domain as a special link;

Consider the optimal path segment from an entry boundary node to every exit boundary node of a domain as a special link; and the optimal path segment is computed as needed.

The whole topology consisting of many domains can be considered as a special topology, which contains those special links, the normal links in the destination domains and the inter-domain links.

Compute a shortest path in this special topology from the source node to the multiple destination nodes using CSPF.

Forward Search P2MP inter-domain path computation method running at any PCE just grows the result path list/tree in the same way as normal CSPF on the special topology. When the result path list/tree reaches all the destination nodes, the shortest path from the source node to the destination nodes is found and a PCRep message with the shortest path is sent to the PCE/PCC that sends the PCReq message eventually.

4.2. Description of Procedure

Suppose that we have the following variables:

A current PCE named as CurrentPCE which is currently computing the path.

A number of rest destination nodes named as RestDestinationNodes, which is the number of destination nodes to which shortest paths are to be found. RestDestinationNodes is initially the number of all the destination nodes of an MPLS TE P2MP LSP.

A candidate node list named as CandidateNodeList, which contains the nodes through which the shortest path from the source node to a destination node may be. Each node C in CandidateNodeList has the following information:

the cost of the path from the source node to node C,

the previous hop node P and the link between P and C,

the PCE responsible for C, and

the flags for C. The flags include:

one bit D indicating that node C is a Destination node if it is set;

one bit S indicating that C is the Source node if it is set;

one bit E indicating that C is an Exit boundary node if it is set;

one bit I indicating that C is an entry boundary node if it is set; and

one bit N indicating that C is a Node in the destination domains if it is set.

The nodes in CandidateNodeList are ordered by path cost. Initially, CandidateNodeList contains only a Source Node, with path cost 0, PCE responsible for the source domain, and flags with S bit set.

A result path list or tree named as ResultPathTree, which contains the shortest paths from the source node to the boundary nodes or the nodes in the destination domains. Initially, ResultPathTree is empty.

The Forward Search path computation method for computing a path for an MPLS TE P2MP LSP crossing a number of domains from a source node to a number of destination nodes can be described as follows:

Initially, a PCC sets RestDestinationNodes to the number of all the destination nodes of the MPLS TE P2MP LSP, ResultPathTree to empty and CandidateNodeList to contain the source node and sends a PCE responsible for the source domain a request with the source node, RestDestinationNodes, CandidateNodeList and ResultPathTree.

When the PCE responsible for a domain (called current domain) receives a request for computing the path for the MPLS TE P2MP LSP, it checks whether the current PCE is the PCE responsible for the node C with the minimum cost in the CandidateNodeList. If it is, then remove C from CandidateNodeList and graft it into ResultPathTree; otherwise, a PCReq message is sent to the PCE for node C.

Suppose that node C has Flags. The ResultPathTree is built from C in the following steps.

If node C is a destination node (i.e., the Destination Node (D) bit in the Flags is set), then RestDestinationNodes is decreased by one. If RestDestinationNodes is zero (i.e., all the destinations are on the result path tree), then the shortest path is found and a PCRep message with the path is sent to the PCE/PCC which sends the request to the current PCE.

If node C is in the destination domain (i.e., the Node in Destination domain (N) bit in the Flags is set), then for every node N connected to node C and not on ResultPathTree, it is merged into CandidateNodeList. The cost to node N is the sum of the cost to node C and the cost of the link between C and N. The PCE for node N is the current PCE.

If node C is an Entry Boundary Node or Source Node (i.e., the Entry/ Incoming Boundary Node (I) bit or the Source Node (S) bit is set), then path segments from node C to every exit boundary node of the current domain that is not on the result path tree are computed through using CSPF and as special links. For every node N connected to node C through a special link (i.e., a path segment), it is merged into CandidateNodeList. The cost to node N is the sum of the cost to node C and the cost of the special link (i.e., path segment) between C and N. The PCE for node N is the current PCE.

If node C is an Exit Boundary Node (i.e., the Exit Boundary Node (E) bit is set) and there exist inter-domain links connected to it, then for every node N connected to C and not on the result path tree, it is merged into the candidate node list. The cost to node N is the sum of the cost to node C and the cost of the link between C and N. The PCE for node N is the PCE responsible for node N.

If the CurrentPCE is the same as the PCE of the node with the minimum cost in CandidateNodeList, then the node is removed from CandidateNodeList, grafted to ResultPathTree, and the above steps are repeated; otherwise, the CurrentPCE sends the PCE a request with the source node, RestDestinationNodes, CandidateNodeList and ResultPathTree.

5. Extensions to PCEP

The extensions to PCEP for Forward Search P2MP Inter-domain Path Computation include the definition of a new flag in the RP object, a result path list/tree and a candidate node list in a request message.

5.1. RP Object Extension

The following flag is added into the RP Object:

The F bit is added in the flag bits field of the RP object to tell the receiver of the message that the request/reply is for Forward Search Path Computation.

- o F (Forward search Path Computation bit - 1 bit):

  0: This indicates that this is not PCReq/PCRep for Forward Search Path Computation.

  1: This indicates that this is PCReq or PCRep message for Forward Search Path Computation.

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

This F bit with the N bit defined in RFC6006 can indicate whether the request/reply is for Forward Search Path Computation of an MPLS TE P2MP LSP or an MPLS TE P2P LSP.

- o F = 1 and N = 1: This indicates that this is a PCReq/PCRep message for Forward Search Path Computation of an MPLS TE P2MP LSP.

- o F = 1 and N = 0: This indicates that this is a PCReq/PCRep message for Forward Search Path Computation of an MPLS TE P2P LSP.

5.2. PCE Object

The figure below illustrates a PCE IPv4 object body (Object-Type=2), which comprises a PCE IPv4 address. The PCE IPv4 address object indicates the IPv4 address of a PCE , with which a PCE session may be established and to which a request message may be sent.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       PCE  IPv4 address                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format of the PCE 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                 PCE  IPv6 address (16 bytes)                  |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

5.3. Candidate Node List Object

The candidate-node-list-obj object contains a list of candidate nodes. A new PCEP object class and type are requested for it. The format of the candidate-node-list-obj object body is as follows:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    //                    <candidate-node-list>                    //
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The following is the definition of the candidate node list.

```
      <candidate-node-list>::= <candidate-node>
                               [<candidate-node-list>]
      <candidate-node>::= <ERO>
                          <candidate-attribute-list>

      <candidate-attribute-list>::= [<attribute-list>]
                                    [<PCE>]
                                    [<Node-Flags>]
```

The ERO in a candidate node contain just the path segment of the last link of the path, which is from the previous hop node of the tail end node of the path to the tail end node. With this information, we can graft the candidate node into the existing result path list or tree.

Simply speaking, a candidate node has the same or similar format of a path defined in RFC 5440, but the ERO in the candidate node just contain the tail end node of the path and its previous hop, and the candidate node may contain two new objects PCE and node flags.

5.4. Node Flags Object

The Node Flags object is used to indicate the characteristics of the node in a candidate node list in a request or reply message for Forward Search Inter-domain Path Computation. The Node Flags object comprises a Reserved field, and a number of Flags.

The format of the Node Flags object body is as follows:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |D|S|I|E|N|          Flags     |         Reserved               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where

- o D = 1: The node is a destination node.
- o S = 1: The node is a source node.
- o I = 1: The node is an entry boundary node.
- o E = 1: The node is an exit boundary node.
- o N = 1: The node is a node in a destination domain.

5.5. Rest Destination Nodes Object

The figure below is an illustration of an object called a number of destinations not in path tree/list , which comprises an Object Length field, a Class-num field, a C-type filed, and a number of destinations not in path. As shown, the value of Object Length field in the object may be 8, which is a length of the object in bytes; the value of Class-num field and the value of C-type field will be assigned by Internet Assigned Numbers Authority (IANA); and the value of the number of destinations not in path tree field comprises a number, which is the number of destinations that are not in the final path computed yet.

```
 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 Length (8)       |   Class-num   |    C-type     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Numbe of Destinations not in path tree            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

5.6. Request Message Extension

Below is the message format for a request message with the extension of a result path list and a candidate node list:

```
           <PCReq Message>::= <Common Header>
                              [<svec-list>]
                              <request-list>
           <request-list>::=<request>[<request-list>]
           <request>::= <RP>
                        <END-POINTS>
                        [<OF>]
                        [<LSPA>]
                        [<BANDWIDTH>]
                        [<metric-list>]
                        [<RRO>[<BANDWIDTH>]]
                        [<IRO>]
                        [<LOAD-BALANCING>]
                        [<result-path-list>]
                        [<candidate-node-list-obj>]
                        [<rest-destination-nodes>]

        where:
           <result-path-list>::=<path>[<result-path-list>]
           <path>::= <ERO><attribute-list>
           <attribute-list>::=[<LSPA>]
                              [<BANDWIDTH>]
                              [<metric-list>]
                              [<IRO>]

           <candidate-node-list-obj> contains a <candidate-node-list>

           <candidate-node-list>::= <candidate-node>
                                    [<candidate-node-list>]
           <candidate-node>::= <ERO>
                               <candidate-attribute-list>

           <candidate-attribute-list>::= [<attribute-list>]
                                         [<PCE>]
                                         [<Node-Flags>]
```

Figure 1: The Format for a Request Message

The definition for the result path list that may be added into a request message is the same as that for the path list in a reply message that is described in RFC5440.

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.

7.1. Request Parameter Bit Flags

A new RP Object Flag has been defined in this document. IANA is requested to make the following allocation from the "PCEP RP Object Flag Field" Sub-Registry:

Bit	Description	Reference
18	Forward Path Computation (F-bit)	This I-D

8. Acknowledgement

The author would like to thank people 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, 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.

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

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

9.2. Informative References

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

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

Author's Address

Huaimo Chen
Huawei Technologies
Boston, MA
USA

Email: Huaimochen@huawei.com

Internet Engineering Task Force   H. Chen
Internet-Draft   Huawei Technologies
Intended status: Standards Track   March 7, 2011
Expires: September 8, 2011

A Fordward-Search P2P TE LSP Inter-Domain Path Computation
draft-chen-pce-forward-search-p2p-path-computation-00.txt

Abstract

This document presents a forward search procedure for computing Point-to-Point (P2P) Traffic Engineering (TE) Label Switched Paths (LSPs) crossing a number of domains through using multiple Path Computation Elements (PCEs). In addition, extensions to the Path Computation Element Communication Protocol (PCEP) for supporting the forward search procedure are described.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Methods or procedures exist for computing MPLS TE P2P LSP inter-domain paths. They include RFC 5152 "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)" and RFC 5441 "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths". These methods or procedures have some issues.

There are a few of issues with the Backward Recursive Path Computation (BRPC) algorithm or procedure for computing an MPLS TE P2P LSP path from a source node to a destination node crossing multiple domains. These issues include:

The sequence of domains from the source node to the destination node must be known in advance.

Navigating a mesh of domains may be complex.

More importantly, the BRPC procedure can not find the optimal path if the optimal path is not in the sequence of domains from the source node to the destination node. Thus the BRPC procedure can not guarantee that the path crossing multiple domains computed by the BRPC procedure is optimal.

This document presents a forward search procedure for computing Point-to-Point (P2P) Traffic Engineering (TE) Label Switched Paths (LSPs) crossing a number of domains through using multiple Path Computation Elements (PCEs). This procedure resolves the issues mentioned above. It guarantees that the path found from the source to the destination is optimal. It does not depend on any sequence of domains from the source node to the destination node. Navigating a mesh of domains is simple and efficient.

2. Terminology

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

ASBR: Autonomous System Border Router. Routers used to connect together ASes of the same or different service providers via one or more inter-AS links.

Boundary Node (BN): a boundary node is either an ABR in the context of inter-area Traffic Engineering or an ASBR in the context of inter-AS Traffic Engineering.

Entry BN of domain(n): a BN connecting domain(n-1) to domain(n) along a determined sequence of domains.

Exit BN of domain(n): a BN connecting domain(n) to domain(n+1) along a determined sequence of domains.

Inter-area TE LSP: A TE LSP that crosses an IGP area boundary.

Inter-AS TE LSP: A TE LSP that crosses an AS boundary.

LSP: Label Switched Path.

LSR: Label Switching Router.

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.

PCE(i) is a PCE with the scope of domain(i).

TED: Traffic Engineering Database.

This document uses terminologies defined in RFC5440.

3. Conventions Used in This Document

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

4. Forward Search Path Computation

This section gives an overview of the forward search path computation procedure and describes the procedure in details.

4.1. Overview of Procedure

Simply speaking, the idea of the forward search path computation procedure for computing a path for an MPLS TE P2P LSP crossing multiple domains from a source node to a destination node includes:

Start from the source node and the source domain.

Consider the optimal path segment from the source node to every exit boundary node of the source domain as a special link;

Consider the optimal path segment from an entry boundary node to every exit boundary node of a domain as a special link; and the optimal path segment is computed as needed.

The whole topology consisting of many domains can be considered as a special topology, which contains those special links, the normal links in the destination domain and the inter-domain links.

Compute an optimal path in this special topology from the source node to the destination node using CSPF.

The forward search path computation procedure for computing a path for an MPLS TE P2P LSP starts at the source domain, in which the source (or ingress) node of the MPLS TE LSP locates. When a PCE in the source domain receives a PCReq for the path for the MPLS TE LSP, it computes the optimal path from the source node to every exit boundary node of the domain towards the destination node.

There are two lists involved in the path computation. One list is called candidate node list, which contains the nodes with brief information about the temporary optimal paths from the source node to each of these nodes currently found. The nodes in the candidate list are ordered by the cost of the path. Initially, the candidate node list contains only source node with cost 0.

The other is called result path list or tree, which contains the final optimal paths from the source node to the boundary nodes or the nodes in the destination domain. Initially, the result path list is empty.

When a PCE responsible for a domain (called current domain) receives a PCReq for computing the path for the MPLS TE LSP, it removes the node with the minimum cost from the candidate node list and put or graft the node to the result path list or tree.

If the destination node is in the current domain, the PCE computes the optimal path from the source node to the destination node and sends a PCRep with the optimal path to the PCE or PCC from which the PCReq is received.

Otherwise (i.e., if the destination is not in the domain), the PCE computes the optimal path from the source node to every exit boundary node of the current domain towards the destination node and further to the entry boundary nodes of the domain connected to the current domain, puts the new node into the candidate list in order by path cost, updates the existing node in the candidate node list with the new node with lower cost, and then sends a PCReq with the new candidate node list to the PCE that is responsible for the domain with the first node in the candidate node list.

4.2. Description of Procedure

Suppose that we have the following variables:

A current PCE named as CurrentPCE which is currently computing the path.

A candidate node list named as CandidateNodeList, which contains the nodes to each of which the temporary optimal path from the source node is currently found. The information about each node C in CandidateNodeList consists of

the cost of the path from the source node to node C,

the previous hop node P and the link between P and C,

the PCE responsible for C, and

the flags for C. The flags include

one bit D indicating that node C is a Destination node if it is set;

one bit S indicating that C is the Source node if it is set;

one bit E indicating that C is an Exit boundary node if it is set;

one bit I indicating that C is an entry boundary node if it is set; and

one bit N indicating that C is a Node in the destination domain if it is set.

The nodes in CandidateNodeList are ordered by path cost. Initially, CandidateNodeList contains only a Source Node, with path cost 0, PCE responsible for the source domain, and flags with S bit set.

A result path list or tree named as ResultPathTree, which contains the final optimal paths from the source node to the boundary nodes or the nodes in the destination domain. Initially, ResultPathTree is empty.

The Forward Search Path Computation procedure for computing the path for the MPLS TE P2P LSP is described as follows:

Initially, a PCC sets ResultPathTree to empty and CandidateNodeList to contain the source node and sends PCE responsible for the source domain a PCReq with the source node, the destination node, CandidateNodeList and ResultPathTree.

When the PCE responsible for a domain (called current domain) receives a request for computing the path for the MPLS TE P2MP LSP, it checks whether the current PCE is the PCE responsible for the node C with the minimum cost in the CandidateNodeList. If it is, then remove C from CandidateNodeList and graft it into ResultPathTree; otherwise, a PCReq message is sent to the PCE for node C.

Suppose that node C has Flags. The ResultPathTree is built from C in the following steps.

If the D (Destination Node) bit in the Flags is set, then the optimal path from the source node to the destination node is found, and a PCRep message with the path is sent to the PCE/PCC which sends the request to the current PCE.

If the N (Node in Destination domain) bit in the Flags is set, then for every node N connected to node C and not on ResultPathTree, it is merged into CandidateNodeList. The cost to node N is the sum of the cost to node C and the cost of the link between C and N. The PCE for node N is the current PCE.

If the Entry/Incoming Boundary Node (I) bit or the Source Node (S) bit is set), then path segments from node C to every exit boundary node of the current domain that is not on the result path tree are computed through using CSPF and as special links. For every node N connected to node C through a special link (i.e., a path segment), it is merged into CandidateNodeList. The cost to node N is the sum of the cost to node C and the cost of the special link (i.e., path segment) between C and N. The PCE for node N is the current PCE.

If the Exit Boundary Node (E) bit is set and there exist inter-domain links connected to it, then for every node N connected to C and not on the result path tree, it is merged into the candidate node list. The cost to node N is the sum of the cost to node C and the cost of the link between C and N. The PCE for node N is the PCE responsible for node N.

If the CurrentPCE is the same as the PCE of the node with the minimum cost in CandidateNodeList, then the node is removed from CandidateNodeList, grafted to ResultPathTree, and the above steps are repeated; otherwise, the CurrentPCE sends the PCE a request with the source node, CandidateNodeList and ResultPathTree.

5. Extensions to PCEP

This section describes the extensions to PCEP for Forward Search Path Computation. The extensions include the definition of a new flag in the RP object, a result path list and a candidate node list in the PCReq message.

5.1. RP Object Extension

The following flag is added into the RP Object:

The F bit is added in the flag bits field of the RP object to tell the receiver of the message that the request/reply is for Forward Search Path Computation.

- o F (Forward search Path Computation bit - 1 bit):

  0: This indicates that this is not PCReq/PCRep for Forward Search Path Computation.

  1: This indicates that this is PCReq or PCRep message for Forward Search Path Computation.

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

This F bit with the N bit defined in RFC6006 can indicate whether the request/reply is for Forward Search Path Computation of an MPLS TE P2P LSP or an MPLS TE P2MP LSP.

- o F = 1 and N = 0: This indicates that this is a PCReq/PCRep message for Forward Search Path Computation of an MPLS TE P2P LSP.

- o F = 1 and N = 1: This indicates that this is a PCReq/PCRep message for Forward Search Path Computation of an MPLS TE P2MP LSP.

5.2. PCE Object

The figure below illustrates a PCE IPv4 object body (Object-Type=2), which comprises a PCE IPv4 address. The PCE IPv4 address object indicates the IPv4 address of a PCE , with which a PCE session may be established and to which a request message may be sent.

```
        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
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                        PCE  IPv4 address                      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format of the PCE 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
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                               |
       |                  PCE  IPv6 address (16 bytes)                 |
       |                                                               |
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

5.3. Node Flags Object

The Node Flags object is used to indicate the characteristics of the node in a candidate node list in a request or reply message for Forward Search Inter-domain Path Computation. The Node Flags object comprises a Reserved field, and a number of Flags.

The format of the Node Flags object body is as follows:

```
       0                   1                   2                   3
       0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |D|S|I|E|N|          Flags          |         Reserved          |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where

- o D = 1: The node is a destination node.
- o S = 1: The node is a source node.
- o I = 1: The node is an entry boundary node.
- o E = 1: The node is an exit boundary node.
- o N = 1: The node is a node in a destination domain.

5.4. Candidate Node List Object

The candidate-node-list-obj object contains the nodes in the candidate node list. A new PCEP object class and type are requested for it. The format of the candidate-node-list-obj object body is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//               (a list of <candidate-node>s)                 //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The following is the definition of candidate node list, which may contain Node Flags.

```
<candidate-node-list>::= <candidate-node>
                         [<candidate-node-list>]
<candidate-node>::= <ERO>
                    <candidate-attribute-list>

<candidate-attribute-list>::= [<attribute-list>]
                              [<PCE>]
                              [<Node-Flags>]
```

The ERO in a candidate node contain just the path segment of the last link of the path, which is from the previous hop node of the tail end node of the path to the tail end node. With this information, we can graft the candidate node into the existing result path list or tree.

Simply speaking, a candidate node has the same or similar format of a path defined in RFC 5440, but the ERO in the candidate node just contain the tail end node of the path and its previous hop, and the candidate path may contain two new objects PCE and node flags.

5.5. Request Message Extension

Below is the message format for a request message with the extension of a result path list and a candidate node list:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>
<request-list>::=<request>[<request-list>]
<request>::= <RP>
             <END-POINTS>
             [<OF>]
             [<LSPA>]
             [<BANDWIDTH>]
             [<metric-list>]
             [<RRO>[<BANDWIDTH>]]
             [<IRO>]
             [<LOAD-BALANCING>]
             [<result-path-list>]
             [<candidate-node-list-obj>]

where:
   <result-path-list>::=<path>[<result-path-list>]
   <path>::= <ERO><attribute-list>
   <attribute-list>::=[<LSPA>]
                      [<BANDWIDTH>]
                      [<metric-list>]
                      [<IRO>]

   <candidate-node-list-obj> contains a <candidate-node-list>
```

Figure 1: The Format for a Request Message

The definition for the result path list that may be added into a request message is the same as that for the path list in a reply message that is described in RFC5440.

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.

7.1. Request Parameter Bit Flags

A new RP Object Flag has been defined in this document. IANA is requested to make the following allocation from the "PCEP RP Object

Flag Field" Sub-Registry:

Bit	Description	Reference
18	Forward Path Computation (F-bit)	This I-D

8. Acknowledgement

The author would like to thank people 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, 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.

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

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

9.2. Informative References

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

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

Author's Address

Huaimo Chen
Huawei Technologies
Boston, MA
US

Email: Huaimochen@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standard
Expires: September 4, 2011

D. Dhody
U. Palle
Huawei Technologies India Pvt Ltd

March 3, 2011

Standard Representation Of Domain Sequence
draft-dhody-pce-pcep-domain-sequence-00

Abstract

The ability to compute shortest constrained Traffic Engineering Label Switched Paths (TE LSPs) in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks across multiple domains has been identified as a key requirement for P2P and P2MP scenarios. In this context, a domain is a collection of network elements within a common sphere of address management or path computational responsibility such as an IGP area or an Autonomous Systems. This document specifies a standard representation of domain sequence that can be utilized in all PCE deployment scenarios.

Status of this Memo

Table of Contents

1. Introduction

RFC 5441 [A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths] mentions -

"The sequence of domains to be traversed is either administratively predetermined or discovered by some means that is outside of the scope of this document. The PCC MAY indicate the sequence of domains to be traversed using the Include Route Object (IRO) defined in [RFC5440] so that it is available to all PCEs."

This document proposes a standard way to represent domain sequence using IRO in various deployment scenarios.

It further gives examples of various deployment scenario including P2P, P2MP and HPCE.

The domain sequence (the set of domains traversed to reach the destination domain) is either administratively predetermined or discovered by some means (H-PCE) that is outside of the scope of this document. Here the focus is only on a standard representation of the domain sequence in all possible scenarios.

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.

ABR: OSPF Area Border Router. Routers used to connect two IGP areas.

AS: Autonomous System.

ASBR: Autonomous System Boundary Router.

BN: Boundary Node, Can be an ABR or ASBR.

BRPC: Backward Recursive Path Computation

Domain: Any collection of network elements within a common sphere of address management or path computational responsibility. Examples of domains include Interior Gateway Protocol (IGP) areas and Autonomous Systems (ASs).

Domain-Seq: The sequence of domains for a path.

ERO: Explicit Route Object

H-PCE: Hierarchical PCE

IGP: Interior Gateway Protocol. Either of the two routing protocols, Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS).

IRO: Include Route Object

IS-IS: Intermediate System to Intermediate System.

OSPF: Open Shortest Path First.

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.

P2MP: Point-to-Multipoint

P2P: Point-to-Point

TE LSP: Traffic Engineering Label Switched Path.

3. Detail Description

3.1. Domains

A domain is any collection of network elements within a common sphere of address management or path computation responsibility. Examples of domains include IGP areas or Autonomous Systems (ASes). To uniquely identify a domain in the domain sequence both AS and Area-id is important.

3.2. Standard Representation

The IRO (Include Route Object) is used to specify the domain sequence that the computed inter-domain path MUST traverse.

IRO Object-Class is 10.
IRO Object-Type is 1.

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

Sub-objects: The IRO is made of sub-objects.
The following sub-object types are used.

```
          Type   Sub-object
           32    Autonomous system number
           TBD  OSPF Area id
           TBD  ISIS Area id
```

Since the length of Area-id is different for OSPF and ISIS, we propose different sub-objects.

For OSPF, the area-id is a 32 bit number. The Subobject looks

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |      Area Id (4 bytes)        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Area Id (continued)           |             Resvd             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The length if fixed.

For ISIS, the area-id is of variable length and thus the length of the Subobject is variable. The Area-id is as described in ISIS by ISO standard. The Length MUST be at least 4, and MUST be a multiple of 4.

```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |L|    Type     |     Length      |                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+                               |
       //                        ISIS Area ID                           //
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

3.3. Deployment Scenarios

3.3.1. Only AS

Considering each AS to be made of a single area, in this scenario the area MAY be skipped in the domain sequence. The domain sequence could be represented with just AS numbers.

```
                                     +---------------------------------+
                                     |AS 200                           |
                                     |            +------+             |
                                     |            |      |             |
      +------------------------+     |            |      |   +------+  |
      | AS 100                 |     |            +------+   |      |  |
      |               +------+ |     | +------+              |      |  |
      |               |      +-+-----+-+      |              +------+  |
      |               |      | |     | |      |                        |
      |               +------+ |     | +------+                        |
      | +------+               |     |              +------+           |
      | |      |               |     |              |      |           |
      | |      |               |     |              |      |           |
      | +------+               |     |              +------+           |
      |                        |     |                                 |
      |               +------+ |     | +------+                        |
      |               |      +-+-----+-+      |               +------+ |
      |               |      | |     | |      |               |      | |
      |               +------+ |     | +------+               |      | |
      |                        |     |                        +------+ |
      |                        |     |                                 |
      |                        |     |                                 |
      |       +------+         |     |              +------+           |
      |       |      |         |     |              |      |           |
      |       |PCE   |         |     |              |PCE   |           |
      |       +------+         |     |              +------+           |
      |                        |     |                                 |
      +------------------------+     |                                 |
                                     +---------------------------------+
```

Both AS are made of Area 0.

This could be represented as <IRO> as:

```
    +---------+  +---------+  +---------+
    |IRO      |  |Sub      |  |Sub      |
    |Object   |  |Object As|  |Object As|
    |Header   |  |100      |  |200      |
    |         |  |         |  |         |
    |         |  |         |  |         |
    +---------+  +---------+  +---------+


    +---------+  +---------+  +---------+  +---------+  +---------+
    |IRO      |  |Sub      |  |Sub      |  |Sub      |  |Sub      |
    |Object   |  |Object As|  |Object   |  |Object As|  |Object   |
    |Header   |  |100      |  |Area 0   |  |200      |  |Area 0   |
    |         |  |         |  |         |  |         |  |         |
    |         |  |         |  |         |  |         |  |         |
    +---------+  +---------+  +---------+  +---------+  +---------+
```

Area is optional and it MAY be skipped. PCE should be able to understand both notations.

3.3.2. Only Area

Consider a case where both end of LSP belong to different area but within the same AS, this could be represented in domain sequence using the AREA sub-object. AS number MAYBE skipped.

```
 +-------------------+                            +-------------------+
 |                   |                            |                   |
 |           +--+    |                            |     +--+          |
 |  +--+     |  |    |                            |     |  |          |
 |  |  |     +--+    |                            |     +--+   +--+   |
 |  +--*             +                            +            |  |   |
 |                   |                            |            +--+   |
 |         *--+      +                            +                   |
 |         |  |      |                            |     +--+          |
 |         +--+      |                            |     |  |          |
 |                   |+--------------------------+|     +--+          |
 |                  ++++                       +-++                   |
 |                  ||||         +--+          | ||                   |
 | Area 2           ++++         |  |          +-++  Area 4           |
 +-------------------+|          +--+            |+-------------------+
                      |                          |
                      |                +--+      |
                      |    +--+        |  |      |
                      |    |  |        +--+      |
                      |    +--+                  |
                      |                          |
                      |                          |
                      |                          |
                      |                          |
                      |           +--+           |
                      |           |  |           |
                      |           +--+           |
  +------------------+|                          |+--------------------+
  |                  ++-+                      +-++                    |
  |                  || |                      | ||                    |
  |                  ++-+    Area 0            +-++                    |
  |                  |+--------------------------+|     +--+           |
  |          +--+    |                            |     |  |           |
  |          |  |    |                            |     +--+           |
  | +--+     +--+    |                            |                    |
  | |  |             +                            +            +--+    |
  | +--+             |                            |            |  |    |
  |                  +                            +            +--+    |
  |       +--+       |                            |                    |
  |       |  |       |                            |     +--+           |
  |       +--+       |                            |     |  |           |
  |                  |                            |     +--+           |
  |                  |                            |                    |
  | Area 1           |                            |  Area 5            |
  +------------------+                            +--------------------+
```

AS Number is 100.

This could be represented as <IRO> as:

```
+---------+ +---------+ +---------+ +---------+
|IRO      | |Sub      | |Sub      | |Sub      |
|Object   | |Object   | |Object   | |Object   |
|Header   | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         |
|         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+


+---------+ +---------+ +---------+ +---------+ +---------+
|IRO      | |Sub      | |Sub      | |Sub      | |Sub      |
|Object   | |Object As| |Object   | |Object   | |Object   |
|Header   | |100      | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         | |         |
|         | |         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+ +---------+
```

AS is optional and it MAY be skipped. PCE should be able to understand both notations.

3.3.3. Mix of AS and Area

In inter-AS case where an AS is further made up of multiple areas, both AS number and area should be a part of domain sequence.

```
                  |
                  |  +-------------+                +----------------+
                  |  |Area 2       |                |Area 4          |
                  |  |         +--+|                |          +--+  |
                  |  |         |  ||                |          |  |  |
                  |  |  +--+   +--+|                |   +--+   +--+  |
                  |  |  |  |       |                |   |  |         |
                  |  |  *--+       |                |   +--+         |
                  |  | /      +--+ |                |          +--+  |
                  |  |/       |  | |                |          |  |  |
                  |  /        +--+ |                |   +--+   +--+  |
                  | /|  +--+       |+--------------+|   |  |         |
                  |/ |  |  |       ++-+          +-++   +--+         |
   +-------------+/  |  +--+       || |          | ||                |
   |             /|  |             ++-+          +-++                |
   |         +--*||  +-------------+|              |+----------------+
   |         |  |||                 |     +--+     |
   |         +--+||                 |     |  |     |
   |    +--+     ||                 |     +--+     |
   |    |  |     ||                 |              |
   |    +--+     ||                 |              |
   |             ||                 |     +--+     |
   |+--+         ||                 |     |  |     |
   ||  |         ||                 |     +--+     |
   |+--+         ||                 |              |
   |             ||                 |     +--+     |
   |    +--+     ||  +------------+ |     |  |     |+----------------+
   |    |  |     ||  |Area 3      +-++    +--+   +-++ Area 5         |
   |    +--+     ||  |            | ||           | ||                |
   |             ||  |            +-++           +-++                |
   |         +--+||  |       +--+ | |  Area 0      ||   +--+         |
   |         |  |||  |       |  | | +--------------+|   |  |         |
   |         +--*||  |       +--+ |                 |   +--+         |
   |             \|  |            |                 |          +--+  |
   |Area 1       |\  |   +--+     |                 |   +--+   |  |  |
   +-------------+|\ |   |  |     |                 |   |  |   +--+  |
                  | \|   +--+  +--+                 |   +--+         |
                  |  \         |  |                 |                |
                  |  |\        +--+                 |          +--+  |
                  |  | \ +--+     |                 |          |  |  |
                  |  |  \|  |     |                 |          +--+  |
                  |  |   *--+     |                 |                |
                  |  |            |                 |                |
                  |  +------------+                 +----------------+
                  |
                  |
       As 100     |  AS 200
                  |
```

The domain sequence can be carried in IRO as shown below:

```
 +--------+ +--------+ +--------+ +--------+ +--------+ +--------+ +--------+
 |IRO     | |Sub     | |Sub     | |Sub     | |Sub     | |Sub     | |Sub     |
 |Object  | |Object  | |Object  | |Object  | |Object  | |Object  | |Object  |
 |Header  | |As 100  | |Area 1  | |AS 200  | |Area 3  | |Area 0  | |Area 4  |
 |        | |        | |        | |        | |        | |        | |        |
 +--------+ +--------+ +--------+ +--------+ +--------+ +--------+ +--------+
```

Combination of both AS and Area uniquely identify a domain in the domain sequence.

3.3.4. PCE serving multiple domains

A single PCE maybe responsible for multiple domains; for example PCE function deployed on an ABR. Domain sequence should have no impact on this. PCE which can support 2 adjacent domains can internally handle this situation without any impact on the neighboring domains.

3.3.5. P2MP

In case of P2MP the path domain tree is nothing but a series of Domain-Seq, as shown in the below figure:

```
   D1-D3-D6, D1-D3-D5 and D1-D2-D4.
               D1
              /  \
             D2  D3
            /   /  \
           D4  D5  D6
```

The same domain sequence are carried in IRO as explained above.

3.3.6. HPCE

Consider a case as shown below consisting of child and parent PCE.

```
                                +--------+
                                | Parent |
                                | PCE    |
                                +--------+

 +-------------------+                            +-------------------+
 |           +--+    |                            |     +--+          |
 |  +--+     |  |    |                            |     |  |          |
 |  |  |     +--+    |                            |     +--+   +--+   |
 |  +--*             +                            +            |  |   |
 |                   |                            |            +--+   |
 |         *--+      +                            +                   |
 |         |  |      |                            |     +--+          |
 |         +--+      |                            |     |  |          |
 |                   |+--------------------------+|     +--+          |
 |                  ++++                       +-++                   |
 |                  ||||         +--+          | ||                   |
 | Area 2           ++++         |  |          +-++  Area 4           |
 +-------------------+|          +--+            |+-------------------+
                      |                +--+      |
                      |    +--+        |  |      |
                      |    |  |        +--+      |
                      |    +--+                  |
                      |                          |
                      |           +--+           |
                      |           |  |           |
                      |           +--+           |
  +------------------+|                          |+--------------------+
  |                  ++-+                      +-++                    |
  |                  || |                      | ||                    |
  |                  ++-+    Area 0            +-++                    |
  |                  |+--------------------------+|     +--+           |
  |          +--+    |                            |     |  |           |
  |          |  |    |                            |     +--+           |
  | +--+     +--+    |                            |                    |
  | |  |             +                            +            +--+    |
  | +--+             |                            |            |  |    |
  |                  +                            +            +--+    |
  |       +--+       |                            |                    |
  |       |  |       |                            |     +--+           |
  |       +--+       |                            |     |  |           |
  |                  |                            |     +--+           |
  | Area 1           |                            |  Area 5            |
  +------------------+                            +--------------------+
```

In HPCE implementation PCE(1) can request the parent PCE to determine the domain path and return in the PCRep in form of ERO. The Subobject would be AS and Area (OSPF/ISIS). So in this case, the reply would carry the result as

```
+---------+ +---------+ +---------+ +---------+
|ERO      | |Sub      | |Sub      | |Sub      |
|Object   | |Object   | |Object   | |Object   |
|Header   | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         |
|         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+


+---------+ +---------+ +---------+ +---------+ +---------+
|ERO      | |Sub      | |Sub      | |Sub      | |Sub      |
|Object   | |Object As| |Object   | |Object   | |Object   |
|Header   | |100      | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         | |         |
|         | |         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+ +---------+
```

3.3.7. Domain Seq V/s PCE Seq

[PCE-P2MP-PROCEDURES] introduces the concept of PCE-Sequence, where a sequence of PCE based on the domain sequence should be decided and attached in the PCReq at the very beginning of path computation. It is much simpler and advantageous to carry only domain-sequence rather than PCE-Sequence.

Advantages

- o All PCE must be aware of all other PCEs in all domain for PCE-Sequence. There is no clear method for this. In domain-sequence PCE should be aware of the domains and not all the PCEs serving the domain. PCE needs to be aware of the neighboring PCEs as done by discovery protocols.

- o There maybe multiple PCE in a domain, the selection of PCE shouldn't be made at the PCC/PCE(1). This decision is made only at the neighboring PCE which is completely aware of states of PCE via notification messages.

- o Domain sequence would be compatible to P2P inter-domain BRPC method as described in RFC 5441.

There is no need for PCE-Sequence and it doesn't give any benefits over Domain Seq.

4. IANA Considerations

IANA has defined a registry for OSPF and ISIS Area sub-object.

Type	Sub-object
TBD	OSPF Area id
TBD	ISIS Area id

5. Security Considerations

This document specifies a standard representation of domain sequence, which is used in all inter-domain PCE scenarios as explained in other RFC and drafts. It does not introduce any new security considerations.

6. Manageability Considerations

TBD

7. Acknowledgments

We would like to thank Pradeep Shastry, Suresh babu, Quintin Zhao and Chen Huaimo for their useful comments and suggestions.

8. References

8.1. Normative References

[ISO] "Intermediate System to Intermediate System Intra-Domain Routeing Exchange Protocol for use in Conjunction with the Protocol for Providing the Connectionless-mode Network Service" ISO/IEC 10589:2002 Second Edition

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

8.2. Informative References

[PCE-HIERARCHY-FWK] King, D. and A. Farrel, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", September 2010.

[PCE-P2MP-PROCEDURES] Zhao, Q., Ali, Z., Saad,, T., and D. King, "PCE-based Computation Procedure To Compute Shortest Constrained P2MP Inter-domain Traffic

Engineering Label Switched Paths", January 2011.

[RFC5440] Ayyangar, A ., Farrel, A ., Oki, E., Atlas, A., Dolganow, A., Ikejiri, Y., Kumaki, K., Vasseur, J., and J. Roux, "Path Computation Element (PCE) communication Protocol (PCEP)", March 2009.

[RFC5441] Vasseur, JP., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", April 2009.

Authors' Addresses

Dhruv Dhody
Huawei Technologies India Pvt Ltd
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruvd@huawei.com

Udayasree Palle
Huawei Technologies India Pvt Ltd
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: udayasreepalle@huawei.com

PCE Working Group D. Dhody
Internet-Draft U. Palle
Intended status: Standards Track Q. Zhao
Expires: September 12, 2011 Huawei Technology
D. King
Old Dog Consulting
March 11, 2011

Management Information Base for the PCE Communications Protocol (PCEP) for Path-Key-Based Inter-Domain Path Computation
draft-dhody-pce-pcep-pathkey-mib-01

Abstract

This memo defines an experimental portion of the Management Information Base for use with network management protocols in the Internet community. In particular, it describes managed objects for modeling of the Path Computation Element communication Protocol (PCEP)for communications between a Path Computation Client (PCC)and a Path Computation Element (PCE), or between two PCEs when path-key-based inter-domain path computation is requested.

Status of this Memo

Copyright Notice

This Internet-Draft will expire on September 12, 2011.

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.

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].

If confidentiality is required between domains, Path-Key-Based mechanism is described in [RFC 5520]. For preserving the confidentiality of the "Confidential Path Segment (CPS)";the PCE returns a path containing a loose hop in place of the segment that must be kept confidential.

[PCE-PCEP-DRAFT-MIB] defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community for P2P path computations.

This memo defines an experimental portion of the Management Information Base for use with network management protocols in the Internet community. In particular, it describes managed objects for modeling of Path Computation Element communication Protocol (PCEP)[RFC5440] for communications between a Path Computation Client (PCC)and a Path Computation Element (PCE), or between two PCEs in path-key-based inter-domain path computations.

Some objects maybe moved to [PCE-PCEP-DRAFT-MIB] after consensus with the authors and working group, these are defined in section 6.2.

2. Terminology

The following terminology is used in this document.

CPS: Confidential Path Segment. A segment of a path that contains nodes and links that the AS policy requires to not be disclosed outside the AS.

Domain: Any collection of network elements within a common sphere of address management or path computational responsibility. Examples of domains include Interior Gateway Protocol (IGP) areas and Autonomous Systems (ASs).

IGP: Interior Gateway Protocol. Either of the two routing protocols, Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS).

PCC: Path Computation Client: any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.

P2P: Point-to-Point

3. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIv2, which is described in STD 58,RFC 2578 [RFC2578] and STD 58, RFC 2580 [RFC2580].

4. PCEP Pathkey MIB Module Architecture

The PCEP Pathkey MIB will contain the following information:

- o PCEP Pathkey counters, timers and configurations

- o PCEP Pathkey table of CPS related information.

5. Example of the PCEP PathKey MIB module usage

In this section we provide an example (pcePcepPathKeyTable 1) of using the MIB objects described in Section 6 (Object definitions) to monitor. While this example is not meant to illustrate every permutation of the MIB, it is intended as an aid to understanding some of the key concepts. It is meant to be read after going through the MIB itself.

```
   pcePcepPathKeyTable 1 of the PCE-PCEP-PATHKEY-DRAFT-MIB module :
   {
               pcePcepPathKey                    (4512),
               pcePcepPathKeyFirstHopIndex       (1),
               pcePcepPathKeyHopNum              (2),
               pcePcepPathKeyRequestSource       (x.x.x.x),
               pcePcepPathKeyRequestId           (10),
               pcePcepPathKeyRetrieved           (1),
               pcePcepPathKeyRetrieveSource      (y.y.y.y),
               pcePcepPathKeyDiscardTime         (10),
               pcePcepPathKeyReuseTime           (30)
   }

   pcePcepPathKeyHopTable 1 of the PCE-PCEP-PATHKEY-DRAFT-MIB module :
   {
               pcePcepPathKeyHopIndex            1,
               pcePcepPathKeyHopAddrType         ipv4 (1),
               pcePcepPathKeyHopIpAddr           "192.168.100.1",
               pcePcepPathKeyHopIpPrefixLen      32,
               pcePcepPathKeyHopType             strict (2)
   }
   {
               pcePcepPathKeyHopIndex            2,
               pcePcepPathKeyHopAddrType         ipv4 (1),
               pcePcepPathKeyHopIpAddr           "192.168.100.2",
               pcePcepPathKeyHopIpPrefixLen      32,
               pcePcepPathKeyHopType             strict (2)
   }
```

6. Object definitions

6.1. PCE-PCEP-PATHKEY-DRAFT-MIB

This MIB module makes references to the following documents.

[RFC2578], [RFC2580], [RFC3411], [RFC2863], [RFC3813].

```
   PCE-PCEP-PATHKEY-DRAFT-MIB DEFINITIONS ::= BEGIN

   IMPORTS
          MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
          Unsigned32,
          Counter32,
          OCTET STRING,
          experimental
                FROM SNMPv2-SMI             --  [RFC2578]

          TimeStamp
                FROM SNMPv2-TC              --  [RFC2579]

          PcePcepIdentifier,
                FROM PCE-TC-STD-MIB

          MplsLSPID, MplsPathIndex, TeHopAddressType,
          TeHopAddress, TeHopAddressUnnum
                FROM MPLS-TC-STD-MIB        -- [RFC3811]

          MODULE-COMPLIANCE,
          OBJECT-GROUP,
          NOTIFICATION-GROUP
               FROM SNMPv2-CONF;            --  [RFC2580]

 pcePcepPathkeyDraftMIB MODULE-IDENTITY
        LAST-UPDATED "201103081200Z" --Mar 8, 2011
        ORGANIZATION "Path Computation Element (PCE) Working Group"
        CONTACT-INFO "


           Dhruv Dhody
           Udayasree Palle
           Quintin Zhao
           Huawei Technology
           Daniel King
           OldDog Consulting

      EMail: dhruvd@huawei.com
      EMail: udayasreepalle@huawei.com
      EMail: qzhao@huawei.com
      EMail: daniel@oldog.co.uk
      EMail comments directly to the PCE WG Mailing List at pce@ietf.org
      WG-URL: http://www.ietf.org/html.charters/pce-charter.html
      "

   DESCRIPTION
```

```
   "This MIB module defines a collection of objects for managing PCE
   communication protocol(PCEP) for Path-Key-Based Inter-Domain Path
   Computation"

      -- Revision history

         REVISION
              "201103081200Z"  -- 08 Mar 2011 12:00:00 EST
              DESCRIPTION
              "
              Main Changes from -00 draft :
              1. Added HopTable to store the CPS hops.
              2. Added Path Key Creation Time.

         REVISION
              "201009171200Z"  -- 17 Sep 2010 12:00:00 EST
          DESCRIPTION

              "draft-00 version"
         ::= { experimental 9999 } --

   -- Notifications --

   pcePcepPathKeyNotifications OBJECT IDENTIFIER ::=
                            { pcePcepPathKeyDraftMIB 0 }

   pcePcepPathKeyMIBObjects OBJECT IDENTIFIER ::=
                            { pcePcepPathKeyDraftMIB 1 }
   pcePcepPathKeyConformance   OBJECT IDENTIFIER ::=
                            { pcePcepPathKeyDraftMIB 2 }

   pcePcepPathKeyObjects OBJECT IDENTIFIER ::=
                            { pcePcepPathKeyMIBObjects 1 }

   --

   -- PCE Pathkey Objects

   --
```

```
   pcePcepPathKeyDiscardTimer OBJECT-TYPE
               SYNTAX  Unsigned32
               UNITS   "minutes"
               MAX-ACCESS read-write
               STATUS mandatory
               DESCRIPTION
               "The value which indicates a period of time after the
               expiration of which a PCE discard unwanted path-keys."
               ::= {  pcePcepPathKeyObjects 1 }

   pcePcepPathKeyReUseTimer OBJECT-TYPE
               SYNTAX  Unsigned32
               UNITS   "minutes"
               MAX-ACCESS read-write
               STATUS mandatory
               DESCRIPTION
                   "The value which indicates a period of time which
                    should expire before an old path-key could be
                    reused for a new CPS."
               ::= {  pcePcepPathKeyObjects 2 }

   pcePcepPathKeyRetainStatus OBJECT-TYPE
            SYNTAX      INTEGER {
                          enabled(1),
                          disabled(2)
                        }
            MAX-ACCESS  read-write
            STATUS      optional
            DESCRIPTION
                "The path-key retain status of this PCE to retain the
                path-key and CPS for debugging purposes."
               ::= {  pcePcepPathKeyObjects 3 }

   pcePcepPathKeysGenerated OBJECT-TYPE
                  SYNTAX  Counter32
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The number of path-keys generated by this PCE."
                  ::= {  pcePcepPathKeyObjects 4 }
```

```
   pcePcepPathKeyExpandUnknown OBJECT-TYPE
                SYNTAX  Counter32
                MAX-ACCESS read-only
                STATUS mandatory
                DESCRIPTION
                    "The number of attempts to expand an unknown
                     path-key."
                ::= {  pcePcepPathKeyObjects 5 }

   pcePcepPathKeyExpandExpired OBJECT-TYPE
                SYNTAX  Counter32
                MAX-ACCESS read-only
                STATUS mandatory
                DESCRIPTION
                    "The number of attempts to expand an expired
                     path-key."
                ::= {  pcePcepPathKeyObjects 6 }

   pcePcepPathKeyExpandSame OBJECT-TYPE
                  SYNTAX  Counter32
                  MAX-ACCESS read-only
                  STATUS optional
                  DESCRIPTION
                      "The number of attempts to expand the same
                       path-key."
                  ::= {  pcePcepPathKeyObjects 7 }

 pcePcepPathKeyExpiredNoExpansion OBJECT-TYPE
                SYNTAX  Counter32
                MAX-ACCESS read-only
                STATUS optional
                DESCRIPTION
                    "The number of path-keys expired without any attempt
                     to expand it."
                ::= {  pcePcepPathKeyObjects 8 }

   pcePcepPathKeyExpansionSuccess OBJECT-TYPE
                  SYNTAX  Counter32
                  MAX-ACCESS read-only
                  STATUS optional
                  DESCRIPTION
                      "The number of path-key expansion requests (PCReq)
                       which had successful retrieval."
                  ::= {  pcePcepPathKeyObjects 9 }
```

```
   pcePcepPathKeyExpansionFailures OBJECT-TYPE
                  SYNTAX  Counter32
                  MAX-ACCESS read-only
                  STATUS optional
                  DESCRIPTION
                      "The number of path-key expansion requests (PCReq)
                       which had failed retrieval."
                  ::= {  pcePcepPathKeyObjects 10 }

   pcePcepPathKeyConfig OBJECT-TYPE
                  SYNTAX      INTEGER {
                          enabled(1),
                          disabled(2)
                        }
            MAX-ACCESS  read-write
            STATUS      mandatory
                  DESCRIPTION
                      "The path-key based inter domain computation
                       configuration."
                  ::= {  pcePcepPathKeyObjects 11 }

   pcePcepPathKeyTable  OBJECT-TYPE
               SYNTAX      SEQUENCE OF pcePcepPathKeyEntry
               MAX-ACCESS  not-accessible
               STATUS      current
               DESCRIPTION
                   "This table contains information about the
                     Pathkey CPS of PCE."
                 ::= { pcePcepPathKeyObjects 12 }

 pcePcepPathKeyEntry OBJECT-TYPE
              SYNTAX      pcePcepPathKeyEntry
              MAX-ACCESS  not-accessible
              STATUS      current
              DESCRIPTION
                  "An entry in this table represents a path-key and CPS.
                   An entry is only created when a path-key generated by
                   PCE during inter-domain computation."

              INDEX       { pcePcepPathKey }

              ::= { pcePcepPathKeyTable 1 }
```

```
   pcePcepPathKeyEntry ::= SEQUENCE {
            pcePcepPathKey                   Unsigned32,
            pcePcepPathKeyFirstHopIndex      MplsPathIndex,
            pcePcepPathKeyHopNum             Unsigned32,
            pcePcepPathKeyRequestSource      PcePcepIdentifier,
            pcePcepPathKeyRequestId          Unsigned32,
            pcePcepPathKeyRetrieved          INTEGER,
            pcePcepPathKeyRetrieveSource     PcePcepIdentifier,
            pcePcepPathKeyCreationTime       TimeStamp,
            pcePcepPathKeyDiscardTime        Unsigned32,
            pcePcepPathKeyReuseTime          Unsigned32,
   }

   pcePcepPathKey OBJECT-TYPE
                  SYNTAX  Unsigned32
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The path-key value to identify a CPS."
                  ::= {  pcePcepPathKeyEntry 1 }

   pcePcepPathKeyFirstHopIndex    OBJECT-TYPE
                  SYNTAX  MplsPathIndex
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The Hop index of the first Hop in the CPS. This
                      index along with pcePcepPathKeyHopNum is used to
                      traverse the Hops in the CPS."
                  ::= {  pcePcepPathKeyEntry 2 }

   pcePcepPathKeyHopNum    OBJECT-TYPE
                  SYNTAX  Unsigned32
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The number of Hops in this CPS. This number along
                       with pcePcepPathKeyFirstHopIndex is used to
                       traverse the Hops in the CPS."
                  ::= {  pcePcepPathKeyEntry 3 }
```

```
   pcePcepPathKeyRequestSource OBJECT-TYPE
                  SYNTAX  PcePcepIdentifier
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "Source that issued the original request that led
                       to the creation of the path-key."
                  ::= {  pcePcepPathKeyEntry 4 }

   pcePcepPathKeyRequestId OBJECT-TYPE
                  SYNTAX  Unsigned32
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The request ID of the original PCReq that led
                       to the creation of the path-key."
                  ::= {  pcePcepPathKeyEntry 5 }

   pcePcepPathKeyRetrieved OBJECT-TYPE
               SYNTAX      INTEGER {
                             TRUE(1),
                             FALSE(2)
                           }
               MAX-ACCESS read-only
               STATUS mandatory
               DESCRIPTION
                  "It specifies whether the path-key is retrieved
                   or not."
               ::= {  pcePcepPathKeyEntry 6 }

   pcePcepPathKeyRetrieveSource OBJECT-TYPE
                  SYNTAX  PcePcepIdentifier
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "If the path-key is retrieved then by which
                       PCC."
                  ::= {  pcePcepPathKeyEntry 7 }

   pcePcepPathKeyCreationTime OBJECT-TYPE
                  SYNTAX  TimeStamp
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The value of sysUpTime at which Path Key
                      was generated by PCE."
                  ::= {  pcePcepPathKeyEntry 8 }
```

```
   pcePcepPathKeyDiscardTime OBJECT-TYPE
                  SYNTAX  Unsigned32
                  MAX-ACCESS read-only
                  STATUS mandatory
                  DESCRIPTION
                      "The time after which the path segment associated
                       with the path-key will be discarded."
                  ::= {  pcePcepPathKeyEntry 9 }

 pcePcepPathKeyReuseTime OBJECT-TYPE
                SYNTAX  Unsigned32
                MAX-ACCESS read-only
                STATUS mandatory
                DESCRIPTION
                    "The time after which the path-key will be available
                     for re-use."
                ::= {  pcePcepPathKeyEntry 10 }

   pcePcepPathKeyHopTable  OBJECT-TYPE
                  SYNTAX      SEQUENCE OF pcePcepPathKeyHopEntry
                  MAX-ACCESS  not-accessible
                  STATUS      current
                  DESCRIPTION
                      "This table contains information about the
                        Pathkey Hop in the CPS of PCE."
                   ::= { pcePcepPathKeyObjects 13 }

pcePcepPathKeyHopEntry OBJECT-TYPE
              SYNTAX      pcePcepPathKeyHopEntry
              MAX-ACCESS  not-accessible
              STATUS      current
              DESCRIPTION
                  "An entry in this table represents a Hop in the CPS.
                   An entry is only created when a path-key generated by
                   PCE during inter-domain computation."

              INDEX       { pcePcepPathKeyHopIndex }

              ::= { pcePcepPathKeyHopTable 1 }
```

```
   pcePcepPathKeyHopEntry ::= SEQUENCE {
            pcePcepPathKeyHopIndex         MplsPathIndex,
            pcePcepPathKeyHopAddrType      TeHopAddressType,
            pcePcepPathKeyHopIpAddr        TeHopAddress,
            pcePcepPathKeyHopIpPrefixLen   InetAddressPrefixLength,
            pcePcepPathKeyHopAddrUnnum     TeHopAddressUnnum,
            pcePcepPathKeyHopLspId         MplsLSPID,
            pcePcepPathKeyHopType          INTEGER,
   }

   pcePcepPathKeyHopIndex OBJECT-TYPE
               SYNTAX  MplsPathIndex
               MAX-ACCESS read-only
               STATUS mandatory
               DESCRIPTION
                   "The index into this table identifying a particular
                   Hop. All Hops in the CPS are added sequentially into
                   the table. The first hop index and number of Hops are
                   used for traversing the table."

               ::= {  pcePcepPathKeyHopEntry 1 }

   pcePcepPathKeyHopAddrType OBJECT-TYPE
               SYNTAX TeHopAddressType
               MAX-ACCESS read-only
               STATUS mandatory
               DESCRIPTION
                   "The Hop Address Type of this CPS hop.
                    Note that lspid(5) is a valid option only
                    for tunnels signaled via CRLDP."
               DEFVAL { ipv4 }
               ::= { pcePcepPathKeyHopEntry 2 }

 pcePcepPathKeyHopIpAddr OBJECT-TYPE
             SYNTAX TeHopAddress
             MAX-ACCESS read-only
             STATUS mandatory
             DESCRIPTION
                 "The Hop Address for this CPS hop.
                  The type of this address is determined by the
                  value of the corresponding pcePcepPathKeyHopAddrType."
             DEFVAL { '00000000'h } -- IPv4 address 0.0.0.0
             ::= { pcePcepPathKeyHopEntry 3 }
```

```
   pcePcepPathKeyHopIpPrefixLen OBJECT-TYPE
               SYNTAX InetAddressPrefixLength
               MAX-ACCESS read-only
               STATUS current
               DESCRIPTION
                   "If pcePcepPathKeyHopAddrType is set to ipv4(1) or
                    ipv6(2), then this value will contain an
                    appropriate prefix length for the IP address in
                    object pcePcepPathKeyHopIpAddr. Otherwise this value
                    is irrelevant and should be ignored."
               DEFVAL { 32 }
               ::= { pcePcepPathKeyHopEntry 4 }

   pcePcepPathKeyHopAddrUnnum OBJECT-TYPE
               SYNTAX TeHopAddressUnnum
               MAX-ACCESS read-only
               STATUS current
               DESCRIPTION
                   "If pcePcepPathKeyHopAddrType is set to unnum(4),
                    then this value will contain the interface
                    identifier of the unnumbered interface for this
                    hop. This object should be used in conjunction
                    with pcePcepPathKeyHopIpAddr which would contain
                    the LSR Router ID in this case."
               ::= { pcePcepPathKeyHopEntry 5 }

   pcePcepPathKeyHopLspId OBJECT-TYPE
               SYNTAX MplsLSPID
               MAX-ACCESS read-only
               STATUS current
               DESCRIPTION
                   "If pcePcepPathKeyHopAddrType is set to lspid(5),
                    then this value will contain the LSPID of a tunnel
                    of this hop. The present tunnel being configured is
                    tunneled through this hop (using label stacking).
                    This object is otherwise insignificant and should
                    contain a value of 0 to indicate this fact."
               ::= { pcePcepPathKeyHopEntry 6 }
```

```
   pcePcepPathKeyHopType OBJECT-TYPE
               SYNTAX INTEGER {
                           strict(1),
                           loose(2)
                           }
               MAX-ACCESS read-only
               STATUS mandatory
               DESCRIPTION
                   "Denotes whether this hop is routed in a
                    strict or loose fashion. "
               DEFVAL { strict }
               ::= { pcePcepPathKeyHopEntry 7 }

   ---

   --- Notifications

   ---

   pcePcepPathKeyExpandUnknownNtf NOTIFICATION-TYPE
             OBJECTS      {
                             pcePcepPathKeyExpandUnknown
                          }
             STATUS       mandatory
             DESCRIPTION
                "This notification is sent when an attempt to expand an
                 unknown path-key is made. The value of the counter
                 pcePcepPathKeyExpandUnknown is also increased at this
                 time."
             ::= { pcePcepPathKeyNotifications 1 }

   pcePcepPathKeyExpandExpiredNtf NOTIFICATION-TYPE
             OBJECTS      {
                             pcePcepPathKeyExpandExpired
                          }
             STATUS       mandatory
             DESCRIPTION
                "This notification is sent when an attempt to expand an
                 expired path-key is made. The value of the counter
                 pcePcepPathKeyExpandExpired is also increased at this
                 time."
             ::= { pcePcepPathKeyNotifications 2 }
```

```
   pcePcepPathKeyExpandSameNtf NOTIFICATION-TYPE
            OBJECTS     {
                           pcePcepPathKeyExpandSame
                        }
            STATUS      optional
            DESCRIPTION
               "This notification is sent when a duplicate attempt to
                expand the same path-key is made. The value of the
                counter pcePcepPathKeyExpandSame is also increased at
                this time."
            ::= { pcePcepPathKeyNotifications 3 }

   pcePcepPathKeyExpiredNoExpansionNtf NOTIFICATION-TYPE
            OBJECTS     {
                           pcePcepPathKeyExpiredNoExpansion
                        }
            STATUS      optional
            DESCRIPTION
               "This notification is sent when path-key expires without
                any attempt to expand it. The value of the counter
                pcePcepPathKeyExpiredNoExpansion is also increased at
                this time."
            ::= { pcePcepPathKeyNotifications 4 }

   --*****************************************************************
   -- Module Conformance Statement
   --*****************************************************************

   pcePcepPathKeyGroups
          OBJECT IDENTIFIER ::= { pcePcepPathKeyConformance 1 }

   pcePcepPathKeyCompliances
          OBJECT IDENTIFIER ::= { pcePcepPathKeyConformance 2 }
```

```
   --
   -- Full Compliance
   --

   pcePcepPathKeyModuleFullCompliance MODULE-COMPLIANCE
           STATUS current
           DESCRIPTION
                "The Module is implemented with support
                for read-create and read-write.  In other
                words, both monitoring and configuration
                are available when using this MODULE-COMPLIANCE."

           MODULE -- this module
              MANDATORY-GROUPS    { pcePcepPathKeyGeneralGroup,
                                    pcePcepPathKeyNotificationsGroup
                                  }

           ::= { pcePcepPathKeyCompliances 1 }

  --
  -- Read-Only Compliance
  --

  pcePcepPathKeyModuleReadOnlyCompliance MODULE-COMPLIANCE
              STATUS current
              DESCRIPTION
                  "The Module is implemented with support
                  for read-only.  In other words, only monitoring
                  is available by implementing this MODULE-COMPLIANCE."

              MODULE -- this module
                  MANDATORY-GROUPS    { pcePcepPathKeyGeneralGroup,
                                        pcePcepPathKeyNotificationsGroup
                                      }
              ::= { pcePcepPathKeyCompliances 2 }

   -- units of conformance
```

```
           pcePcepPathKeyGeneralGroup OBJECT-GROUP
               OBJECTS {
                   pcePcepPathKeyDiscardTimer,
                   pcePcepPathKeyReUseTimer,
                   pcePcepPathKeysGenerated,
                   pcePcepPathKeyExpandUnknown,
                   pcePcepPathKeyExpandExpired,
                   pcePcepPathKeyConfig,
                   pcePcepPathKey,
                   pcePcepPathKeyFirstHopIndex,
                   pcePcepPathKeyHopNum,
                   pcePcepPathKeyRequestSource,
                   pcePcepPathKeyRequestId,
                   pcePcepPathKeyRetrieved,
                   pcePcepPathKeyRetrieveSource,
                   pcePcepPathKeyCreationTime,
                   pcePcepPathKeyDiscardTime,
                   pcePcepPathKeyReuseTime,
                   pcePcepPathKeyHopIndex,
                   pcePcepPathKeyHopAddrType,
                   pcePcepPathKeyHopIpAddr,
                   pcePcepPathKeyHopIpPrefixLen,
                   pcePcepPathKeyHopType
              }
               STATUS    current
               DESCRIPTION
                   "Objects that apply to all PCEP Pathkey MIB
                    implementations."

               ::= { pcePcepPathKeyGroups 1 }

   pcePcepPathKeyNotificationsGroup NOTIFICATION-GROUP
       NOTIFICATIONS { pcePcepPathKeyExpandUnknownNtf,
                       pcePcepPathKeyExpandExpiredNtf
                          }
       STATUS   current


       DESCRIPTION
           "The notifications for a PCEP Pathkey MIB implementation."
       ::= { pcePcepPathKeyGroups 2 }

       END
```

6.2. Objects for inclusion in module PCE-PCEP-DRAFT-MIB

Following object maybe moved to [PCE-PCEP-DRAFT-MIB] after consensus with the authors and working group.

pcePcepPathKeyConfig

7. IANA Considerations

TBD

8. Security Considerations

This MIB module can be used for configuration of certain objects, and anything that can be configured can be incorrectly configured, with potentially disastrous results.

There are a number of management objects defined in this MIB module with a MAX-ACCESS clause of read-create. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negatie effect on network operations. These are the tables and objects and their sensitivity/vulnerability:

- pcePcepPathKeyDiscardTimer: Setting this value incorrectly may cause the expiration of Pathkey before attempt to retrieve the CPS.

- pcePcepPathKeyReUseTimer: Setting this value incorrectly may cause the re-use of pathkey which may not guarantee the uniqueness of path-key values.

The user of the PCE-PCEP-PATHKEY-DRAFT-MIB module must therefore be aware that support for SET operations in a non-secure environment without proper protection can have a negative effect on network operations.

The readable objects in the PCE-PCEP-PATHKEY-DRAFT-MIB module (i.e., those with MAX-ACCESS other than not-accessible) may be considered sensitive in some environments since, collectively, they provide information about the amount and frequency of path computation requests and responses within the network and can reveal some aspects of their configuration.

In such environments it is important to control also GET and NOTIFY access to these objects and possibly even to encrypt their values when sending them over the network via SNMP.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPsec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB module.

It is RECOMMENDED that implementers consider the security features as provided by the SNMPv3 framework (see [RFC3410], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

9. References

9.1. Normative References

[RFC2578] McCloghrie, k., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", April 1999.

[RFC2579] McCloghrie, k., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIv2", April 1999.

[RFC2580] McCloghrie, k., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIv2", April 1999.

[RFC2863] McCloghrie, k. and F. Kastenholz, "The Interfaces Group MIB", June 2000.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", December 2002.

[RFC3811] Nadeau, T. and J. Cucchiara, "Definition of Textual Conventions and for Multiprotocol Label Switching (MPLS) Management", June 2004.

[RFC3813] Srinivasan, C., Viswanathan, A., and T. Nadeau, "MPLS Multiprotocol Label Switching (MPLS) Label Switch Router Management Information Base", June 2004.

[RFC5440] Ayyangar, A ., Farrel, A ., Oki, E., Atlas, A., Dolganow, A., Ikejiri, Y., Kumaki, K., Vasseur, J., and J. Roux, "Path Computation Element (PCE) communication Protocol (PCEP)", March 2009.

9.2. Informative References

[PCE-PCEP-DRAFT-MIB] Kiran Koushik, A S., Stephan, E., Zhao, Q., and D. King, "PCE communication protocol(PCEP) Management Information Base", July 2010.

[RFC3410] Case, J ., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", December 2002.

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

[RFC5520] Bradford, R., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", April 2009.

Authors' Addresses

Dhruv Dhody
Huawei Technology
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruvd@huawei.com

Udayasree Palle
Huawei Technology
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: Udayasreepalle@huawei.com

Quintin Zhao
Huawei Technology
125 Nagog Technology Park
Acton, MA 01719
US

EMail: qzhao@huawei.com

Daniel King
Old Dog Consulting
UK

EMail: daniel@olddog.co.uk

Network Working Group C. Margaria, Ed.
Internet-Draft Nokia Siemens Networks
Intended status: Standards Track O. Gonzalez de Dios, Ed.
Expires: September 12, 2011 Telefonica Investigacion y Desarrollo
F. Zhang, Ed.
Huawei Technologies
March 11, 2011

PCEP extensions for GMPLS
draft-ietf-pce-gmpls-pcep-extensions-02

Abstract

This memo provides extensions for the Path Computation Element communication Protocol (PCEP) for the support of GMPLS control plane.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

PCEP RFCs [RFC5440], [RFC5521], [RFC5541], [RFC5520] are focused on path computation requests in MPLS networks. [RFC4655] defines the PCE framework also for GMPLS networks. This document complements these RFCs by providing some consideration of GMPLS applications and routing requests, for example for OTN and WSON networks.

The requirements on PCE extensions to support those characteristics are described in [I-D.ietf-pce-gmpls-aps-req] and [I-D.ietf-pce-wson-routing-wavelength].

1.1. Contributing Authors

Elie Sfeir, Franz Rambach (Nokia Siemens Networks) Francisco Javier Jimenez Chico (Telefonica Investigacion y Desarrollo) Suresh BR, Young Lee, SenthilKumar S, Jun Sun (Huawei Technologies), Ramon Casellas (CTTC)

1.2. PCEP requirements for GMPLS

This section provides a set of PCEP requirements to support GMPLS LSPs and assure signal compatibility in the path. When requesting a path computation (PCReq) to PCE, the PCC should be able to indicate, according to [I-D.ietf-pce-gmpls-aps-req] and to RSVP procedures like explicit label control (ELC), the following additional attributes:

(1) Switching capability: for instance PSC1-4, L2SC, TDM, LSC, FSC

(2) Encoding type: as defined in [RFC4202], [RFC4203], e.g., Ethernet, SONET/SDH, Lambda, etc.

(3) Signal Type: Indicates the type of elementary signal that constitutes the requested LSP. A lot of signal types with different granularity have been defined in SONET/SDH and G.709 ODUk, such as VC11, VC12, VC2, VC3 and VC4 in SDH, and ODU1, ODU2 and ODU3 in G.709 ODUk [RFC4606], [RFC4328] and other signal types like the one defined in [I-D.ceccarelli-ccamp-gmpls-ospf-g709] or [I-D.zhang-ccamp-gmpls-evolving-g709] .

(4) Concatenation Type: In SDH/SONET and G.709 OTN networks, two kinds of concatenation modes are defined: contiguous concatenation which requires co-route for each member signal and requires all the interfaces along the path to support this capability, and virtual concatenation which allows diverse routes for the member signals and only requires the ingress and egress interfaces to support this capability. Note that for the virtual concatenation, it also may specify co-routed or separated-routed. See [RFC4606]

and [RFC4328] about concatenation information.

(5) Concatenation Number: Indicates the number of signals that are requested to be contiguously or virtually concatenated. See also [RFC4606] and [RFC4328].

(6) Technology specific label(s) such as wavelength label as defined in [I-D.ietf-ccamp-gmpls-g-694-lambda-labels]

(7) e2e Path protection type: as defined in [RFC4872], e.g., 1+1 protection, 1:1 protection, (pre-planned) rerouting, etc.

(8) Link Protection type: as defined in [RFC4203]

(9) Support for unnumbered interfaces: as defined in [RFC3477]

(10) Support for asymmetric bandwidth requests.

(11) Ability to indicate the requested granularity for the path ERO: node, link, label. This is to allow the use of the explicit label control of RSVP.

(12) In order to support the label control the Path computation response should provide label information matching signaling capabilities

(13) The PCC should be able to provide label restrictions similar to RSVP on the requests.

We describe in this document a proposal to fulfill those requirements.

1.3. PCEP existing objects related to GMPLS

PCEP as of [RFC5440], [RFC5521] and [I-D.ietf-pce-inter-layer-ext], supports the following information (in the PCReq and PCRep) related to the described requirements.

From [RFC5440]:

- o numbered endpoints
- o bandwidth (encoded as IEEE float)
- o ERO
- o LSP attributes (setup and holding priorities)

o Request attribute (include some LSP attributes)

From [RFC5521],Extensions to PCEP for Route Exclusions, definition of a XRO object and a new semantic (F bit):

o This object also allows to exclude (strict or not) resources; XRO include the diversity level (node, link, SRLG). The requested diversity is expressed in the XRO

o This Object with the F bit set indicates that the existing route is failed and resources present in the RRO can be reused.

From [I-D.ietf-pce-inter-layer-ext]:

o INTER-LAYER : indicates if inter-layer computation is allowed

o SWITCH-LAYER : indicates which layer(s) should be considered, can be used to represent the RSVP-TE generalized label request

o REQ-ADAP-CAP : indicates the adaptation capabilities requested, can also be used for the endpoints in case of mono-layer computation

The shortcomings of the existing PCEP information are:

The BANDWIDTH and LOAD-BALANCING objects do not describe the details of the traffic request (for example NVC, multiplier) in the context of GMPLS networks, for instance TDM or OTN networks.

The END-POINTS object does not allow specifying an unnumbered interface, nor the labels on the interface. Those parameters are of interest in case of switching constraints.

Current attributes do not allow to express the requested link level protection and end-to-end protection attributes.

The covered PCEP extensions are:

New objects are introduced (GENERALIZED-BANDWIDTH and GENERALIZED-LOAD-BALANCING) for flexible bandwidth encoding,

New Objects are introduced (LABEL-SET and SUGGESTED-LABEL-SET) on order to allow the PCC to restrict/influence the range of labels returned

A new object type is introduced for the END-POINTS object (generalized-endpoint), A new TLV is added to the LSPA object.

In order to indicate the mandatory routing granularity in the response, a new flag in the RP object is added.

1.4. 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.

2. PCEP objects and extensions

This section describes the required PCEP objects and extensions. The PCReq and PCRep messages are defined in [RFC5440]. The format of the request and response messages with the proposed extensions (GENERALIZED-BANDWIDTH, GENERALIZED-LOAD-BALANCING, SUGGESTED-LABEL-SET and LABEL-SET) is as follows:

```
<request>::= <RP>
             <segment-computation>|<path-key-expansion>

<segment-computation> ::=
  <END-POINTS>
  [<LSPA>]
  [<BANDWIDTH>]
  [<BANDWIDTH>]
  [<GENERALIZED-BANDWIDTH>]
  [<GENERALIZED-BANDWIDTH>]
  [<metric-list>]
  [<OF>]
  [<RRO>[<BANDWIDTH>]
  [<GENERALIZED-BANDWIDTH>]
  [<GENERALIZED-BANDWIDTH>]]
  [<IRO>]
  [<SUGGESTED-LABEL-SET>]
  [<LABEL-SET>...]
  [<LOAD-BALANCING>]
  [<GENERALIZED-LOAD-BALANCING>]
  [<GENERALIZED-LOAD-BALANCING>]
  [<XRO>]

<path-key-expansion> ::= <PATH-KEY>

<response>::=<RP>
  [<NO-PATH>]
  [<attribute-list>]
  [<path-list>]

<path-list>::=<path>[<path-list>]
<path>::= <ERO><attribute-list>
<metric-list>::=<METRIC>[<metric-list>]
```

Where:

```
      <attribute-list>::=[<LSPA>]
      [<BANDWIDTH>]
      [<LABEL-SET>...]
      [<SUGGESTED-LABEL-SET>...]
      [<GENERALIZED-BANDWIDTH>]
      [<GENERALIZED-BANDWIDTH>]
      [<GENERALIZED-LOAD-BALANCING>]
      [<GENERALIZED-LOAD-BALANCING>]
      [<metric-list>]
      [<IRO>]
```

For point-to-multipoint(P2MP) computations, the proposed grammar is:

```
     <segment-computation> ::=
         <end-point-rro-pair-list>
         [<LSPA>]
         [<BANDWIDTH>]
         [<GENERALIZED-BANDWIDTH>]
         [<GENERALIZED-BANDWIDTH>]
         [<metric-list>]
         [<IRO>]
         [<SUGGESTED-LABEL-SET>]
         [<LABEL-SET>]
         [<LOAD-BALANCING>]
         [<GENERALIZED-LOAD-BALANCING>]
         [<GENERALIZED-LOAD-BALANCING>]
         [<XRO>]

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

      <RRO-List>::=<RRO>[<BANDWIDTH>]
      [< GENERALIZED-BANDWIDTH>][<RRO-List>]
```

2.1. RP object extension

Explicit label control (ELC) is a procedure supported by RSVP-TE, where the outgoing label(s) is(are) encoded in the ERO. In consequence, the PCE may be able to provide such label(s) directly in the path ERO. The PCC, depending on policies or switching layer, may be required to use explicit label control or expect explicit link, thus it need to indicate in the PCReq which granularity it is expecting in the ERO. The possible granularities can be node, link, label. The granularities are inter-dependent, in the sense that link granularity imply the presence of node information in the ERO,similarly a label granularity imply that the ERO contain node, link and label information.

A new 2-bit routing granularity (RG) flag is defined in the RP object. The values are defined as follows

0 : node

1 : link

2 : label

3 : reserved

When the RP object appears in a request within a PCReq message the flag indicates the requested route granularity. The PCE SHOULD try to follow this granularity and MAY return a NO-PATH if the requested granularity cannot be provided. The PCE MAY return more details on the route based on its policy. The PCC can decide if the ERO is acceptable based on its content.

When the RP object appears in a response within a PCRep message the flag indicates the granularity provided in the response. The PCE MAY indicate the granularity of the returned ERO. The RG flag is backward-compatible with previous RFCs: the value sent by an implementation not supporting it will indicate a node granularity. This flag is optional for responses. A new capability flag in the PCE-CAP-FLAGS from [RFC5088] and [RFC5089] may be added.

2.2. Traffic parameters encoding, GENERALIZED-BANDWIDTH

The PCEP BANDWIDTH does not describe the details of the signal (for example NVC, multiplier), hence the bandwidth information should be extended to use the RSVP Tspec object encoding. The PCEP BANDWIDTH object defines two types: 1 and 2. C-Type 2 is representing the existing bandwidth in case of re-optimization.

The following possibilities cannot be represented in the BANDWIDTH object:

- Asymmetric bandwidth (different bandwidth in forward and reverse direction), as described in [RFC5467]

- GMPLS (SDH/SONET, G.709, ATM, MEF etc) parameters are not supported.

According to [RFC5440] the BANDWIDTH object has no TLV and has a fixed size of 4 bytes. This definition does not allow extending it with the required information. To express this information, a new object named GENERALIZED-BANDWIDTH having the following format is defined:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Traffic Spec Length         | Reserved                    |R|O|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                       Traffic Spec                            ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                       Optional TLVs                           ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The GENERALIZED-BANDWIDTH has a variable length. The Traffic spec length field indicates the length of the Traffic spec field. The bits R and O have the following meaning:

O bit : when set the value refers to the previous bandwidth in case of re-optimization

R bit : when set the value refers to the bandwidth of the reverse direction

The Object type determines which type of bandwidth is represented by the object. The following object types are defined:

1. Intserv

2. SONET/SDH

3. G.709

4. Ethernet

The encoding of the field Traffic Spec is the same as in RSVP-TE, it can be found in the following references.

Object Type	Name	Reference
0	Reserved	
1	Reserved	
2	Intserv	[RFC2210]
3	Re served	
4	SONET/SDH	[RFC4606]
5	G.709	[RFC4328]
6	Ethernet	[RFC6003]

Traffic Spec field encoding

The GENERALIZED-BANDWIDTH MAY appear more than once in a PCReq message. If more than one GENERALIZED-BANDWIDTH have the same Object Type, Reserved, R and O values, only the first one is processed, the others are ignored. On the response the GENERALIZED-BANDWIDTH object that was considered in the processing SHOULD be included.

When a PCC needs to get a bi-directional path with asymmetric bandwidth, it SHOULD specify the different bandwidth in forward and reverse directions through two separate GENERALIZED-BANDWIDTH objects. The PCE MUST compute a path that satisfies the asymmetric bandwidth constraint and return the path to PCC if the path computation is successful.

PCE MAY return several path based on the request with NVC in the GENERALIZED-BANDWIDTH if the request cannot be fulfilled on one path. The PCC should check the path applicability to its policy.

Optional TLVs may be included within the object body to specify more specific bandwidth requirements. The specification of such TLVs is outside the scope of this document.

2.3. Traffic parameters encoding, GENERALIZED-LOAD-BALANCING

The LOAD-BALANCING object is used to request a set of maximum Max-LSP TE-LSP having in total the bandwidth specified in BANDWIDTH, each TE-LSP having a minimum of min-bandwidth bandwidth. The LOAD-BALANCING follows the bandwidth encoding of the BANDWIDTH object, it does not describe enough details for the traffic specification expected by GMPLS. A PCC should be allowed to request a set of TE-LSP also in case of GMPLS traffic specification.

According to [RFC5440] the LOAD-BALANCING object has no TLV and has a fixed size of 8 bytes. This definition does not allows extending it with the required information. To express this information, a new Object named GENERALIZED-LOAD-BALANCING is defined.

The GENERALIZED-LOAD-BALANCING object, as the LOAD-BALANCING object, allows the PCC to request a set of TE-LSP having in total the GENERALIZED-BANDWIDTH traffic specification with potentially Max-Lsp, each TE-LSP having a minimum of Min Traffic spec. The GENERALIZED-LOAD-BALANCING is optional.

GENERALIZED-LOAD-BALANCING Object-Class is to be assigned by IANA. The GENERALIZED-LOAD-BALANCING Object type determines which type of minimum bandwidth is represented by the object. The following object types are defined:

1. Intserv

2. SONET/SDH

3. G.709

4. Ethernet

The GENERALIZED-LOAD-BALANCING has a variable length.

The format of the GENERALIZED-LOAD-BALANCING object body is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Traffic spec length        |      Flags    |R|    Max-LSP  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Min  Traffic Spec                                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                Optional   TLVs                                ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Traffic spec length (16 bits): the total length of the min traffic specification. It should be noted that the RSVP traffic specification may also include TLV different than the PCEP TLVs.

Flags (8 bits): The undefined Flags field MUST be set to zero on transmission and MUST be ignored on receipt. The following flag is defined:

R Flag : (1 bit) set when the value refer to the bandwidth of the reverse direction

Max-LSP (8 bits): maximum number of TE LSPs in the set.

Min-Traffic spec (variable): Specifies the minimum traffic spec of each element of the set of TE LSPs.

The encoding of the field Traffic Spec is the same as in RSVP-TE, it can be found in the following references.

Object Type	Name	Reference
2	Intserv	[RFC2210]
4	SONET/SDH	[RFC4606]
5	G.709	[RFC4328]
6	Ethernet	[RFC6003]

Traffic Spec field encoding

The GENERALIZED-LOAD-BALANCING MAY appear more than once in a PCReq message. If more than one GENERALIZED-LOAD-BALANCING have the same Object Type, and R Flag, only the first one is processed, the others are ignored. On the response the object that were considered in the processing SHOULD be included.

When a PCC needs to get a bi-directional path with asymmetric bandwidth, it SHOULD specify the different bandwidth in forward and reverse directions through two separate GENERALIZED-LOAD-BALANCING objects with different R Flag. The PCE MUST compute a path that satisfies the asymmetric bandwidth constraint and return the path to PCC if the path computation is successful.

Optional TLVs may be included within the object body to specify more specific bandwidth requirements. The specification of such TLVs is outside the scope of this document.

The GENERALIZED-LOAD-BALANCING object has the same semantic as the LOAD-BALANCING object; If a PCC requests the computation of a set of TE LSPs so that the total of their generalized bandwidth is X, the maximum number of TE LSPs is N, and each TE LSP must at least have a bandwidth of B, it inserts a GENERALIZED-BANDWIDTH object specifying X as the required bandwidth and a GENERALIZED-LOAD-BALANCING object with the Max-LSP and Min-traffic spec fields set to N and B, respectively.

For example a request for one co-signaled n x VC-4 TE-LSP will not use the GENERALIZED-LOAD-BALANCING. In case the V4 components can use different paths, the GENERALIZED-BANDWIDTH will contain a traffic specification indicating the complete n x VC4 traffic specification and the GENERALIZED-LOAD-BALANCING the minimum co-signaled VC4. For a SDH network, a request to have a TE-LSP group with 10 VC4 container, each path using at minimum 2VC4 container, can be represented with a GENERALIZED-BANDWIDTH object with OT=4, the content of the Traffic specification is ST=6,RCC=0,NCC=0,NVC=10,MT=1. The GENERALIZED-LOAD-BALANCING, OT=4,R=0,Max-LSP=5, min Traffic spec is (ST=6,RCC=0,NCC=0,NVC=2,MT=1). The PCE can respond with a response with maximum 5 path, each of then having a GENERALIZED-BANDWIDTH OT=4,R=0, and traffic spec matching the minimum traffic spec from the GENERALIZED-LOAD-BALANCING object of the corresponding request.

2.4. END-POINTS Object extensions

The END-POINTS object is used in a PCReq message to specify the source and destination of the path for which a path computation is requested. From [RFC3471] the source IP address and the destination IP address are used to identify those. A new Object Type is defined to address the following possibilities:

- Different endpoint types.
- Label restrictions on the endpoint.
- Specification of unnumbered endpoints type as seen in GMPLS networks.

The Object encoding is described in the following sections.

2.4.1. Generalized Endpoint Object Type

In GMPLS context the endpoints can:

- Be unnumbered
- Have label(s) associated to them
- May have different switching capabilities

The IPv4 and IPv6 endpoints are used to represent the source and destination IP addresses. The scope of the IP address (Node or Link) is not explicitly stated. It should also be possible to request a Path between a numbered link and an unnumbered link, or a P2MP path between different type of endpoints.

Since the PCEP END-POINTS object only support endpoints of the same type a new C-Type is proposed that support different endpoint types, including unnumbered. This new C-Type also supports the specification of constraints on the endpoint label to be use. The PCE might know the interface restrictions but this is not a requirement. On the path calculation request only the tspec and switch layer need to be coherent, the endpoint labels could be different (supporting a different tspec). Hence the label restrictions include a Generalized label request in order to interpret the labels.

The proposed object format consists of a body and a list of TLVs with the following defined TLVs (described in Section 2.4.2). TLVs are used instead of subobject because the restriction information do not only apply to the endpoints but can also be applied to the complete path. The object in which the TLV appear indicate if its a path or endpoint restriction. TLV makes the encoding more convenient.

1. IPv4 address.

2. IPv6 address.

3. Unnumbered endpoint.

4. Label request.

5. Label.

6. Upstream label.

7. Label set.

8. Suggested label set.

The labels TLV are used to restrict the label allocation in the PCE. They follow the set of restrictions provided by signaling with explicit value (label and upstream label), mandatory range restrictions (Label set) and optional range restriction (suggested label set). Single suggested value is using the suggested label set. The label range restriction are valid in GMPLS networks, either by PCC policy or depending on the switching technology used, for instance on given Ethernet or ODU equipment having limited hardware capabilities restricting the label range. Label set restriction also applies to WSON networks where the optical sender and receivers are limited in their frequency tunability ranges, restricting then in GMPLS the possible label ranges on the interface. The END-POINTS Object with Generalized Endpoint object type is encoded as follow:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |      Reserved                                  | endpoint type |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    ~                           TLVs                                ~
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Reserved bits should be set to 0 when a message is sent and ignored when the message is received

the endpoint type is defined as follow:

Value	Type	Meaning
0	Point-to-Point	
1	Point-to-Multipoint	New leaves to add
2		Old leaves to remove
3		Old leaves whose path can be modified/reoptimized
4		Old leaves whose path must be left unchanged
5-244	Reserved	
245-255	Experimental range	

The endpoint type is used to cover both point-to-point and different point-to-multipoint endpoint semantic. Endpoint type 0 MUST be accepted by the PCE, other endpoint type MAY be supported if the PCE implementation supports P2MP path calculation. The TLVs present in the object body MUST follow the following grammar:

```
<generalized-endpoint-tlvs>::=
  <p2p-endpoints> | <p2mp-endpoints>

<p2p-endpoints> ::=
  <source-endpoint>
  <destination-endpoint>

<source-endpoint> ::=
  <endpoint>
  [<endpoint-restriction-list>]

<destination-endpoint> ::=
  <endpoint>
  [<endpoint-restriction-list>]

<p2mp-endpoints> ::=
  <endpoint> [<endpoint-restriction-list>]
  [<endpoint> [<endpoint-restriction-list>] ...]
```

For endpoint type Point-to-Multipoint several endpoint objects may be present in the message and represent a leave, exact meaning depend on the endpoint type defined of the object.

An endpoint is defined as follows:

```
<endpoint>::=<IPV4-ADDRESS>|<IPV6-ADDRESS>|<UNNUMBERED-ENDPOINT>
<endpoint-restriction-list> ::=
             <endpoint-restriction>
             [<endpoint-restriction-list>]

<endpoint-restriction> ::=
                 <LABEL-REQUEST><label-restriction-list>

<label-restriction-list> ::= <label-restriction>
                             [<label-restriction-list>]
<label-restriction> ::= <LABEL>|<UPSTREAM-LABEL>|
                        <LABEL-SET>|
                        <SUGGESTED-LABEL-SET>
```

The different TLVs are described in the following sections

2.4.2. END-POINTS TLVs extensions

All endpoint TLVs have the standard PCEP TLV header as defined in [RFC5440] section 7.1

2.4.2.1. IPV4-ADDRESS

This TLV represent a numbered endpoint using IPv4 numbering, the format of the IPv4-ADDRESS TLV value (TLV-Type=TBA) 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          IPv4 address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

2.4.2.2. IPV6-ADDRESS TLV

This TLV represent a numbered endpoint using IPV6 numbering, the format of the IPv6-ADDRESS TLV value (TLV-Type=TBA) 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              IPv6 address (16 bytes)                          |
|                                                               |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

2.4.2.3. UNNUMBERED-ENDPOINT TLV

This TLV represent an unnumbered interface. This TLV has the same semantic as in [RFC3477] The TLV value is encoded as follow (TLV-Type=TBA)

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          LSR's Router ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Interface ID (32 bits)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

2.4.2.4. LABEL-REQUEST TLV

The LABEL-REQUEST TLV indicates the switching capability and encoding type of the label restriction list. Its format is the same as described in [RFC3471] Section 3.1 Generalized label request. The LABEL-REQUEST TLV use TLV-Type=TBA. The fields are encoded as in the RSVP-TE. The Encoding Type indicates the encoding type, e.g., SONET/SDH/GigE etc., that will be used with the data associated with the LSP. The Switching type indicates the type of switching that is being requested on the link. G-PID identifies the payload of the TE-LSP.

2.4.2.5. Labels TLV

Label or label range restrictions may be specified for the TE-LSP endpoints. Those are encoded in the TLVs. The label value need to be interpreted with a description on the Encoding and switching type. The REQ-ADAP-CAP object from [I-D.ietf-pce-inter-layer-ext] can be used in case of mono-layer request, however in case of multilayer it is possible to have in the future more than one object, so it is better to have a dedicated TLV for the label and label request (the scope is then more clear). TLVs are encoded as follow (following [RFC5440]) :

- o LABEL TLV, Type=TBA. The TLV Length is variable, the value is the same as [RFC3471] Section 3.2 Generalized label. This represent the downstream label

- o UPSTREAM-LABEL TLV, Type=TBA, The TLV Length is variable, the value is the same as [RFC3471] Section 3.2 Generalized label. This represent the upstream label

- o LABEL-SET TLV, Type=TBA. The TLV Length is variable, Encoding follow [RFC3471] Section 3.5 "Label set" with the addition of a U bit : the U bit is set for upstream direction in case of bidirectional 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Action     |      Reserved     |U|        Label Type       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Subchannel 1                         |
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
:                               :                               :
:                               :                               :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Subchannel N                         |
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- o SUGGESTED-LABEL-SET TLV Set, Type=TBA. The TLV length is variable, Encoding is as LABEL-SET TLV.

A LABEL TLV represent the label used on the unnumbered interface, bit

U is used to indicate which exact direction is considered. The label type indicates which type of label is carried. A LABEL-SET TLV represents a set of possible labels that can be used on the unnumbered interface. the label allocated on the first link SHOULD be within the label set range. The action parameter in the Label set indicates the type of list provided. Those parameters are described by [RFC3471] section 3.5.1 A SUGGESTED-LABEL-SET TLV has the same encoding as the LABEL-SET TLV, it indicates to the PCE a set of preferred (ordered) set of labels to be used. the PCE MAY use those labels for label allocation.

The U bit has the following meaning:

U: Upstream direction: set when the label or label set is in the reverse direction

2.5. LABEL-SET object

The LABEL-SET object is carried in a request within a PCReq message to restrict the set of labels to be assigned during the path computation. Any label allocated by the PCE (and included in the ERO object on the response) must be in the range stated in the LABEL-SET. The LABEL-SET Object encoding is defined as 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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 //                  TLVs                                       //
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

where TLVs follow the following grammar

<label-set-tlvs> ::= <LABEL-REQUEST><LABEL-SET>[<LABEL-SET>]

The LABEL-REQUEST and LABEL-SET TLVs are as defined in Section 2.4.2.5, See also [RFC3471] and [RFC3473] for the definitions of the fields.

It is allowed to have more than one LABEL-SET object per request within a PCReq message (for example in case of multiple SWITCH-LAYER present).

In the case of unsuccessful path computation the LABEL-SET object MAY be used to indicate the set of constraint that could not be satisfied.

2.6. SUGGESTED-LABEL-SET object

Similar to the endpoint restriction SUGGESTED-LABEL-SET TLV, but with end-to-end scope the SUGGESTED-LABEL-SET object indicate an optional set of label that the PCE MAY use when selecting the labels. The SUGGESTED-LABEL-SET object is carried within a PCReq or PCRep message to indicate the preferred set of label to be assigned during the path computation. The encoding is the same as the LABEL-SET object. It is allowed to have more than one SUGGESTED LABEL-SET object per PCReq (for example in case of multiple SWITCH-LAYER present).

2.7. LSPA extensions

The LSPA carries the LSP attributes. In the end-to-end protection context this also includes the protection state information. The LSPA object can be extended by a protection TLV type: Type TBA: protection attribute

```
 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                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S|P|N|O|  Reserved  | LSP Flags |     Reserved      | Link Flags|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|I|R|    Reserved    | Seg.Flags |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The content is as defined in [RFC4872], [RFC4873].

LSP Flags can be considered for routing policy based on the protection type. The other attributes are only meaningful for a stateful PCE.

2.8. NO-PATH Object Extension

The NO-PATH object is used in PCRep messages in response to an unsuccessful path computation request (the PCE could not find a path satisfying the set of constraints). In this scenario, PCE MUST include a NO-PATH object in the PCRep message. The NO-PATH object may carries the NO-PATH-VECTOR TLV that specifies more information on the reasons that led to a negative reply. In case of GMPLS networks there could be some more additional constraints that led to the failure like protection mismatch, lack of resources, and so on. Few new flags have been introduced in the 32-bit flag field of the NO-PATH-VECTOR TLV and no modifications have been made in the NO-PATH object.

2.8.1. Extensions to NO-PATH-VECTOR TLV

The modified NO-PATH-VECTOR TLV carrying the additional information is as follows: New fields PM and NR are defined in the 23th and 22th bit of the Flags field respectively.

Bit number TBA - Protection Mismatch (1-bit). Specifies the mismatch of the protection type in the request.

Bit number TBA - No Resource (1-bit). Specifies that the resources are not currently sufficient to provide the path.

Bit number TBA - Granularity not supported (1-bit). Specifies that the PCE is not able to provide a route with the requested granularity.

Bit number TBA - No endpoint label resource (1-bit). Specifies that the PCE is not able to provide a route because of the endpoint label restriction.

Bit number TBA - No endpoint label resource in range (1-bit). Specifies that the PCE is not able to provide a route because of the endpoint label set restriction.

Bit number TBA - No label resource in range (1-bit). Specifies that the PCE is not able to provide a route because of the label set restriction.

3. Additional Error Type and Error Values Defined

A PCEP-ERROR object is used to report a PCEP error and is characterized by an Error-Type that specifies the type of error while Error-value that provides additional information about the error type. An additional error type and few error values are defined to represent some of the errors related to the newly identified objects related to SDH networks. For each PCEP error, an Error-Type and an Error-value are defined. Error-Type 1 to 10 are already defined in [RFC5440]. Additional Error- values are defined for Error-Type 10 and A new Error-Type is introduced (value TBA).

Error-Type	Error-value	
10	Reception of an invalid object	
	Error-value=TBA:	Bad Generalized Bandwidth Object value.
	Error-value=TBA:	Unsupported LSP Protection Type in protection attribute TLV.
	Error-value=TBA:	Unsupported LSP Protection Flags in protection attribute TLV.
	Error-value=TBA:	Unsupported Secondary LSP Protection Flags in protection attribute TLV.
	Error-value=TBA:	Unsupported Link Protection Type in protection attribute TLV.
	Error-value=TBA:	Unsupported Link Protection Type in protection attribute TLV.
TBA	Path computation failure	
	Error-value=TBA:	Unacceptable request message.
	Error-value=TBA:	Generalized bandwidth object not supported.
	Error-value=TBA:	Label Set constraint could not be met.
	Error-value=TBA:	Label constraint could not be met.
	Error-value=TBA:	Unsupported endpoint type in END-POINTS Generalized Endpoint object type

Error-value=TBA: Unsupported TLV present in END-POINTS Generalized Endpoint object type

Error-value=TBA: Unsupported granularity in the RP object flags

4. Manageability Considerations

Liveness Detection and Monitoring This document makes no change to the basic operation of PCEP and so there are no changes to the requirements for liveness detection and monitoring set out in [RFC4657] and [RFC5440].

5. IANA Considerations

IANA assigns values to the PCEP protocol objects and TLVs. IANA is requested to make some allocations for the newly defined objects and TLVs introduced in this document. Also, IANA is requested to manage the space of flags that are newly added in the TLVs.

5.1. PCEP Objects

As described in Section 2.2 and Section 2.3new Objects are defined IANA is requested to make the following Object-Type allocations from the "PCEP Objects" sub-registry.

Object Class to be assigned

Name GENERALIZED-BANDWIDTH

Object-Type 0 to 6

Reference This document (section Section 2.2)

Object Class to be assigned

Name GENERALIZED-LOAD-BALANCING

Object-Type 0 to 6

Reference This document (section Section 2.3)

Object Class to be assigned

Name LABEL-SET

Object-Type 0

Reference This document (section Section 2.5)

Object Class to be assigned

Name SUGGESTED-LABEL-SET

Object-Type 0

Reference This document (section Section 2.6)

As described in Section 2.4.1 a new Object type is defined IANA is requested to make the following Object-Type allocations from the "PCEP Objects" sub-registry. The values here are suggested for use by IANA.

Object Class 4

Name END-POINTS

Object-Type 5 : Generalized Endpoint

6-15 : unassigned

Reference This document (section Section 2.2)

5.2. END-POINTS object, Object Type Generalized Endpoint

IANA is requested to create a registry to manage the endpoint type field of the END-POINTS object, Object Type Generalized Endpoint and manage the code space.

New endpoint type in the Reserved range may be allocated by an IETF consensus action. Each endpoint type should be tracked with the following qualities:

- o endpoint type
- o Description
- o Defining RFC

New endpoint type in the Experimental range are for experimental use; these will not be registered with IANA and MUST NOT be mentioned by RFCs.

The following values have been defined by this document. (Section 2.4.1, Table 4):

Value	Type	Meaning
0	Point-to-Point	
1	Point-to-Multipoint	New leaves to add
2		Old leaves to remove
3		Old leaves whose path can be modified/reoptimized
4		Old leaves whose path must be left unchanged
5-244	Reserved	
245-255	Experimental range	

5.3. New PCEP TLVs

IANA manages the PCEP TLV code point registry (see [RFC5440]). This is maintained as the "PCEP TLV Type Indicators" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry. This document defines new PCEP TLVs, to be carried in the END-POINTS object with Generalized Endpoint object Type. IANA is requested to do the following allocation. The values here are suggested for use by IANA.

Value	Meaning	Reference
7	IPv4 endpoint	This document (section Section 2.4.2.1)
8	IPv6 endpoint	This document (section Section 2.4.2.2)
9	Unnumbered endpoint	This document (section Section 2.4.2.3)
10	Label request	This document (section Section 2.4.2.4)
11	Requested GMPLS Label	This document (section Section 2.4.2.5)
12	Requested GMPLS Upstream Label	This document (section Section 2.4.2.5)

13	Requested GMPLS Label Set	This document (section Section 2.4.2.5)
14	Suggested GMPLS Label Set	This document (section Section 2.4.2.5)
15	LSP Protection Information	This document (section Section 2.7)

5.4. RP Object Flag Field

As described in Section 2.1 new flag are defined in the RP Object Flag IANA is requested to make the following Object-Type allocations from the "RP Object Flag Field" sub-registry. The values here are suggested for use by IANA.

Bit	Description	Reference
bit 17-16	routing granularity (RG)	This document, Section 2.1

5.5. New PCEP Error Codes

As described in Section Section 3, new PCEP Error-Type and Error Values are defined. IANA is requested to make the following allocation in the "PCEP-ERROR Object Error Types and Values" registry. The values here are suggested for use by IANA.

Error	name	Reference
Type=10	Reception of an invalid object	[RFC5440]
Value=2:	Bad Generalized Bandwidth Object value.	This Document
Value=3:	Unsupported LSP Protection Type in protection attribute TLV.	This Document
Value=4:	Unsupported LSP Protection Flags in protection attribute TLV.	This Document
Value=5:	Unsupported Secondary LSP Protection Flags in protection attribute TLV.	This Document
Value=6:	Unsupported Link Protection Type in protection attribute TLV.	This Document
Value=7:	Unsupported Link Protection Type in protection attribute TLV.	This Document
Type=14	Path computation failure	This Document
Value=1:	Unacceptable request message.	This Document
Value=2:	Generalized bandwidth object not supported.	This Document
Value=3:	Label Set constraint could not be met.	This Document
Value=4:	Label constraint could not be met.	This Document
Value=5:	Unsupported endpoint type in END-POINTS Generalized Endpoint object type	This Document
Value=6:	Unsupported TLV present in END-POINTS Generalized Endpoint object type	This Document
Value=7:	Unsupported granularity in the RP object flags	This Document

5.6. New NO-PATH-VECTOR TLV Fields

As described in Section Section 2.8.1, new NO-PATH-VECTOR TLV Flag Fields have been defined. IANA is requested to do the following allocations in the "NO-PATH-VECTOR TLV Flag Field" sub-registry. The values here are suggested for use by IANA.

Bit number 23 - Protection Mismatch (1-bit). Specifies the mismatch of the protection type in the request.

Bit number 22 - No Resource (1-bit). Specifies that the resources are not currently sufficient to provide the path.

Bit number 21 - Granularity not supported (1-bit). Specifies that the PCE is not able to provide a route with the requested granularity.

Bit number 20 - No endpoint label resource (1-bit). Specifies that the PCE is not able to provide a route because of the endpoint label restriction.

Bit number 19 - No endpoint label resource in range (1-bit). Specifies that the PCE is not able to provide a route because of the endpoint label set restriction.

Bit number 18 - No label resource in range (1-bit). Specifies that the PCE is not able to provide a route because of the label set restriction.

6. Security Considerations

None.

7. Contributing Authors

Nokia Siemens Networks:

Elie Sfeir
St Martin Strasse 76
Munich, 81541
Germany

Phone: +49 89 5159 16159
Email: elie.sfeir@nsn.com

Franz Rambach
St Martin Strasse 76
Munich, 81541
Germany

Phone: +49 89 5159 31188
Email: franz.rambach@nsn.com

Francisco Javier Jimenez Chico
Telefonica Investigacion y Desarrollo
C/ Emilio Vargas 6
Madrid, 28043
Spain

Phone: +34 91 3379037
Email: fjjc@tid.es

Huawei Technologies

Suresh BR
Shenzhen
China
Email: sureshbr@huawei.com

Young Lee
1700 Alma Drive, Suite 100
Plano, TX 75075
USA

Phone: (972) 509-5599 (x2240)
Email: ylee@huawei.com

SenthilKumar S
Shenzhen
China
Email: senthilkumars@huawei.com

Jun Sun
Shenzhen
China
Email: johnsun@huawei.com

CTTC - Centre Tecnologic de Telecomunicacions de Catalunya

Ramon Casellas
PMT Ed B4 Av. Carl Friedrich Gauss 7
08860 Castelldefels (Barcelona)
Spain
Phone: (34) 936452916
Email: ramon.casellas@cttc.es

8. Acknowledgments

The research of Ramon Casellas, Francisco Javier Jimenez Chico, Oscar Gonzalez de Dios, Cyril Margaria, and Franz Rambach leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement no 247674.

9. References

9.1. Normative References

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

[RFC2210] Wroclawski, J., "The Use of RSVP with IETF Integrated Services", RFC 2210, September 1997.

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

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

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, January 2003.

[RFC4202] Kompella, K. and Y. Rekhter, "Routing Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4202, October 2005.

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

[RFC4328] Papadimitriou, D., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for G.709 Optical Transport Networks Control", RFC 4328, January 2006.

[RFC4606] Mannie, E. and D. Papadimitriou, "Generalized Multi-Protocol Label Switching (GMPLS) Extensions for Synchronous Optical Network (SONET) and Synchronous Digital Hierarchy (SDH) Control", RFC 4606, August 2006.

[RFC4872] Lang, J., Rekhter, Y., and D. Papadimitriou, "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, May 2007.

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

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

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

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

[RFC5520] Bradford, R., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", RFC 5520, April 2009.

[RFC5521] Oki, E., Takeda, T., and A. Farrel, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions", RFC 5521, April 2009.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, June 2009.

[RFC6003] Papadimitriou, D., "Ethernet Traffic Parameters", RFC 6003, October 2010.

9.2. Informative References

[I-D.ceccarelli-ccamp-gmpls-ospf-g709]
Ceccarelli, D., Caviglia, D., Zhang, F., Li, D., Xu, Y., Belotti, S., Grandi, P., and J. Drake, "Traffic Engineering Extensions to OSPF for Generalized MPLS (GMPLS) Control of Evolving G.709 OTN Networks", draft-ceccarelli-ccamp-gmpls-ospf-g709-04 (work in progress), October 2010.

[I-D.ietf-ccamp-gmpls-g-694-lambda-labels]
Otani, T., Rabbat, R., Shiba, S., Guo, H., Miyazaki, K., Caviglia, D., Li, D., and T. Tsuritani, "Generalized Labels for Lambda-Switching Capable Label Switching Routers", draft-ietf-ccamp-gmpls-g-694-lambda-labels-11 (work in progress), January 2011.

[I-D.ietf-pce-gmpls-aps-req]
Otani, T., Ogaki, K., Caviglia, D., and F. Zhang, "Document: draft-ietf-pce-gmpls-aps-req-03.txt", draft-ietf-pce-gmpls-aps-req-03 (work in progress), October 2010.

[I-D.ietf-pce-inter-layer-ext]
Oki, E., Takeda, T., Roux, J., and A. Farrel, "Extensions to the Path Computation Element communication Protocol (PCEP) for Inter-Layer MPLS and GMPLS Traffic Engineering", draft-ietf-pce-inter-layer-ext-04 (work in progress), July 2010.

[I-D.ietf-pce-wson-routing-wavelength]
Lee, Y., Bernstein, G., Martensson, J., Takeda, T., and T. Tsuritani, "PCEP Requirements for WSON Routing and Wavelength Assignment", draft-ietf-pce-wson-routing-wavelength-04 (work in progress), March 2011.

[I-D.zhang-ccamp-gmpls-evolving-g709]
Zhang, F., Zhang, G., Belotti, S., Ceccarelli, D., Lin, Y., Xu, Y., Grandi, P., and D. Caviglia, "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for the evolving G.709 Optical Transport Networks Control", draft-zhang-ccamp-gmpls-evolving-g709-07 (work in progress), March 2011.

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

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

[RFC5467] Berger, L., Takacs, A., Caviglia, D., Fedyk, D., and J. Meuric, "GMPLS Asymmetric Bandwidth Bidirectional Label Switched Paths (LSPs)", RFC 5467, March 2009.

Authors' Addresses

Cyril Margaria (editor)
Nokia Siemens Networks
St Martin Strasse 76
Munich, 81541
Germany

Phone: +49 89 5159 16934
Email: cyril.margaria@nsn.com

Oscar Gonzalez de Dios (editor)
Telefonica Investigacion y Desarrollo
C/ Emilio Vargas 6
Madrid, 28043
Spain

Phone: +34 91 3374013
Email: ogondio@tid.es

Fatai Zhang (editor)
Huawei Technologies
F3-5-B R&D Center, Huawei Base
Bantian, Longgang District
Shenzhen, 518129
P.R.China

Email: zhangfatai@huawei.com

Network Working Group Y. Lee
Internet Draft Huawei
Intended status: Standard R. Casellas
Expires: September 2011 CTTC
C. Margaria
NSN
O. G. de Dios
Telefonica

March 4, 2011

PCEP Extension for WSON Routing and Wavelength Assignment

draft-lee-pce-wson-rwa-ext-01.txt

Status of this Memo

Copyright Notice

Abstract

This draft provides the Path Computation Element communication Protocol (PCEP) extensions for the support of Routing and Wavelength Assignment (RWA) in Wavelength Switched Optical Networks (WSON). Lightpath provisioning in WSONs requires a routing and wavelength assignment (RWA) process. From a path computation perspective, wavelength assignment is the process of determining which wavelength can be used on each hop of a path and forms an additional routing constraint to optical light path computation.

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 0.

Table of Contents

1. Introduction

[RFC4655] defines the PCE based Architecture and explains how a Path Computation Element (PCE) may compute Label Switched Paths (LSP) in Multiprotocol Label Switching Traffic Engineering (MPLS-TE) and Generalized MPLS (GMPLS) networks at the request of Path Computation Clients (PCCs). A PCC is shown to be any network component that makes such a request and may be for instance an Optical Switching Element within a Wavelength Division Multiplexing (WDM) network. The PCE, itself, can be located anywhere within the network, and may be within an optical switching element, a Network Management System (NMS) or Operational Support System (OSS), or may be an independent network server.

The PCE communications Protocol (PCEP) is the communication protocol used between PCC and PCE, and may also be used between cooperating PCEs. [RFC4657] sets out the common protocol requirements for PCEP. Additional application-specific requirements for PCEP are deferred to separate documents.

This document provides the PCEP extension for the support of Routing and Wavelength Assignment (RWA) in Wavelength Switched Optical Networks (WSON) based on the requirements specified in [PCE-RWA].

WSON refers to WDM based optical networks in which switching is performed selectively based on the wavelength of an optical signal. In this document, it is assumed that wavelength converters require electrical signal regeneration. Consequently, WSONs can be transparent (A transparent optical network is made up of optical devices that can switch but not convert from one wavelength to another, all within the optical domain) or translucent (3R regenerators are sparsely placed in the network).

A LSC Label Switched Path (LSP) may span one or several transparent segments, which are delimited by 3R regenerators (typically with electronic regenerator and wavelength conversion). Each transparent segment or path in WSON is referred to as an optical path. An optical path may span multiple fiber links and the path should be assigned the same wavelength for each link. In such case, the optical path is said to satisfy the wavelength-continuity constraint. Figure 1 illustrates the relationship between a LSC LSP and transparent segments (optical paths).

```
   +---+       +-----+       +-----+      +-----+         +-----+
   |   |I1     |     |       |     |      |     |       I2|     |
   |   |o------|     |-------[(3R) ]------|     |--------o|     |
   |   |       |     |       |     |      |     |         |     |
   +---+       +-----+       +-----+      +-----+         +-----+
       [X  LSC]     [LSC  LSC]    [LSC  LSC]     [LSC  X]       SwCap
       <------->    <------->      <----->       <------->
       <----------------------->  <----------------------->
         Transparent Segment         Transparent Segment
       <------------------------------------------------->
                              LSC LSP
```

Figure 1: Illustration of a LSC LSP and transparent segments

Note that two optical paths within a WSON LSP need not operate on the same wavelength (due to the wavelength conversion capabilities). Two optical paths that share a common fiber link cannot be assigned the same wavelength. To do otherwise would result in both signals interfering with each other. Note that advanced additional multiplexing techniques such as polarization based multiplexing are not addressed in this document since the physical layer aspects are not currently standardized. Therefore, assigning the proper wavelength on a lightpath is an essential requirement in the optical path computation process.

When a switching node has the ability to perform wavelength conversion, the wavelength-continuity constraint can be relaxed, and a LSC Label Switched Path (LSP) may use different wavelengths on different links along its route from origin to destination. It is, however, to be noted that wavelength converters may be limited due to their relatively high cost, while the number of WDM channels that can be supported in a fiber is also limited. As a WSON can be composed of network nodes that cannot perform wavelength conversion, nodes with limited wavelength conversion, and nodes with full wavelength conversion abilities, wavelength assignment is an additional routing constraint to be considered in all lightpath computation.

[Ed note: in general, WSON LSC may not be the only switching layer with switching constraints. From a GMPLS/PCEP perspective, wavelength assignment corresponds to label allocation. This document should align with GMPLS extensions for PCEP. Wavelength restrictions and constraints should be formulated in terms of labels (i.e. LABEL_SET, SUGGESTED_LABEL, UPSTREAM_LABEL, etc.)

[Ed Note] For example, within a translucent WSON, a LSC LSP may be established between interfaces I1 and I2, spanning 2 transparent segments (optical paths) where the wavelength continuity constraint applies (i.e. the same unique wavelength MUST be assigned to the LSP at each TE link of the segment). If the LSC LSP induced a Forwarding Adjacency / TE link, the switching capabilities of the TE link would be [X X] where X < LSC (PSC, TDM, ...).

In addition to those label switching constraints, each optical path is constrained by the optical signal quality. The optical signal quality depends first on the optical sender and receiver capabilities. Second contributors to optical signal constraints are the optical elements used on the path (optical fibers, amplifiers, boosters, optical components). All those elements have an impact on the optical signal quality that limits the ability of the optical path to carry traffic. In order to improve the signal quality and limit some optical effects several advanced modulation processing are used. Those modulation properties contribute not only to optical signal quality checks but also constrain the selection of sender and receiver, as they should have matching signal processing capabilities.

The optical modulation properties, also referred to as signal compatibility, are already considered in signaling in [RWA-Encode] and [WSON-OSPF].

This document includes signal compatibility constraint as part of RWA path computation. That is, the signal processing capabilities (e.g., modulation and FEC) must be compatible between the sender and the receiver of the optical path across all optical elements.

This document, however, does not address optical impairments as part of RWA path computation. See [WSON-Imp] and [PSVP-Imp] for more information on optical impairments and GMPLS.

.

Listed below are some relevant drafts addressed in the IETF CCAMP WG.

- WSON RWA Framework:
  - draft-ietf-ccamp-rwa-wson-framework
- Label switching constraints:
  - draft-ietf-ccamp-general-constraint-encode
  - draft-ietf-ccamp-rwa-wson-encode
  - draft-ietf-ccamp-rwa-info
- Signal processing capabilities:
  - draft-ietf-ccamp-rwa-wson-encode
  - draft-ietf-ccamp-wson-signal-compatibility-ospf
  - draft-ietf-ccamp-wson-signaling
- Optical Impairment:
  - draft-ietf-ccamp-wson-impairments
  - draft-agraz-ccamp-wson-impairment-rsvp
  - draft-eb-ccamp-ospf-wson-impairments

The remainder of this document uses terminology from [RFC4655].

2. WSON PCE Architectures and Requirements

Figure 2 shows one typical PCE based implementation, which is referred to as Combined Process (R&WA). With this architecture, the two processes of routing and wavelength assignment are accessed via a single PCE. This architecture is the base architecture from which the requirements have been specified in [PCE-RWA] and the PCEP extensions that are going to be specified in this document based on this architecture.

```
                    +----------------------------+
  +-----+           |     +--------+     +--+     |
  |     |           |     |Routing |     |WA|     |
  | PCC |<----->    |     +--------+     +--+     |
  |     |           |                             |
  +-----+           |             PCE             |
                    +----------------------------+
```

Figure 2: Combined Process (R&WA) architecture

2.1. Encoding of a new RWA path request

The current RP object is used to indicate routing related information in a new path request per [RFC5440]. Since a new RWA path request involves both routing and wavelength assignment, the wavelength assignment related information in the request SHOULD be coupled in the path request.

[Ed note: align with [GMPLS-PCEP] in the sense that Wavelength Assignment is a particular case of Label Allocation]

Label allocation can be performed by the PCE by different means:

a) By means of Explicit Label Control, in the sense that one (or two) allocated labels MAY appear after an interface route subobject.
b) By means of a Suggested Label (and, for bidirectional LSPs, an Upstream Label) provided by the PCE
c) By means of a Label Set, containing one or more allocated Labels, provided by the PCE.

Note that in the b) and c) cases, except when c) includes only one Label, the label allocation can be considered an optimization or suggestion, allowing to be completed with distributed label allocation (performed during signaling).

Additionally, given a range of potential labels to allocate, the request SHOULD convey the heuristic / mechanism to the allocation, including vendor-specific approaches.

The format of a PCReq message after incorporating the WA object is as follows:

```
<PCReq Message> ::= <Common Header>

                    [<svec-list>]
```

```
                          <request-list>

      Where:

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

         <request>::= <RP>

                      <ENDPOINTS>

                      <WA>

                      [other optional objects...]
```

Note: if WA object is present in the request, the WA object MUST be encoded after the ENDPOINTS object.

The format of the Wavelength Assignment (WA) object body is as follows:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Flags                              |E|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   //                      Optional TLVs                          //
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: WA Object Body Format

Flags (32 bits)

The following new flags SHOULD be set

- o E (Explicit - 1 bit): When E bit is set to 1, this indicates that the label assigned by the PCE must be explicit. That is, the selected way to convey the allocated wavelength is by means of Explicit Label Control (ELC) [RFC4003] for each hop of a computed LSP. Otherwise, the label assigned by the PCE needs not be explicit (i.e., in the form of label sets). This is to allow the distributed WA.

2.1.1. Wavelength Range Constraint

For any request that contains a wavelength assignment, the requester (PCC) MUST be able to specify a restriction on the wavelengths to be used. This restriction is to be interpreted by the PCE as a constraint on the tuning ability of the origination laser transmitter or on any other maintenance related constraints. Note that if the LSP LSC spans different segments, the PCE MUST have mechanisms to know the tunability restrictions of the involved wavelength converters / regenerators, e.g. by means of the IGP. Even if the PCE knows the tenability of the transmitter, the PCC MUST be able to apply additional constraints to the request.

[Ed Note: to align with [PCEP-GMPLS]

WA TLVs:

*) WA_PREFERENCES_TLV (TBD) - Allow FF, LF, Random, vendor-specific

*) IPv4_ADDRESS_TLV | IPv6_ADDRESS_TLV | UNNUMBERED_IF_ID_TLV

*) LABEL_SET_TLV

The Wavelength is defined in [Lambda-Label] 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Grid | C.S   |    Identifier     |              n                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

See [Lambda-Label] for a description of Grid, C.S, Identifier and n.

Note that each 32 bit Wavelength Field is designated to represent one wavelength restriction on the associated link identifier.

2.1.2. Signal processing capability restrictions

Path computation for WSON include the check of signal processing capabilities, those capability MAY be provided by the IGP, however this is not a MUST. Moreover, a PCC should be able to indicate additional restrictions for those signal compatibility, either on the endpoint or any given link.

The supported signal processing capabilities are the one described in [RWA-Info]:

- Modulation Type List
- FEC Type List
- Bit rate
- Client signal

The Bit-rate restriction is already expressed in [PCEP-GMPLS] in the GENERALIZED-BANDWIDTH object.

The client signal information can be expressed in the [PCEP-Layer] REQ-ADAP-CAP object.

In order to support the Modulation and FEC information two new TLV are introduced as endpoint-restriction in the END-POINTS type Generalized endpoint:

- Modulation restriction TLV
- FEC restriction TLV.

The END-POINTS type generalized endpoint is extended as follow:

```
<endpoint-restrictions> ::= <VENDOR-ENDPOINT-RESTRICTION>|

                            <signal-compatibility-restriction> |

                             <LABEL-REQUEST><label-restriction>

                             [<endpoint-restrictions>]
```

Where

```
      signal-compatibility-restriction ::=

            <MODULATION-FORMAT>|<FEC>
```

The MODULATION-FORMAT and FEC TLV are described in the following sections.

2.1.2.1. MODULATION-FORMAT TLV

This optional TLV represents a modulation format restriction. This TLV MAY appear more than once in the endpoint-restriction.

The TLV type is TBD, recommended value 17.

The TLV data is defined as follow:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S|I| Modulation ID               |     Reserved               |X|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Modulation ID/S bit dependent body                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format follows the definition from [WSON-Encode] section 4.2.1 with the exception that the modulation length is already represented in the TLV Length field.

The S and I bit are set as described in [WSON-Encode] section 4.2.1.

The Modulation ID is as defined in [WSON-Encode] section 4.2.1.

The X bit is set to 1 to exclude the Modulation format, the X bit is set to 0 to include the modulation format.

The reserved bits MUST be set to 0 on transmit and MUST be ignore on receive.

The rest of the TLV is encoded following [WSON-Encode] section 4.2.1.

2.1.2.2. FEC TLV

This optional TLV represents a FEC restriction. This TLV MAY appear more than once in the endpoint-restriction.

The TLV type is TBD, recommended value 18.

The TLV data is defined as follow:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S|I|          FEC ID             |     Reserved               |X|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        FEC ID/S bit dependent body                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format follows the definition from [WSON-Encode] section 4.2.2 with the exception that the FEC length is already represented in the TLV Length field.

The S and I bit are set as described in [WSON-Encode] section 4.2.2.

The FEC ID is as defined in [WSON-Encode] section 4.2.2.

The X bit is set to 1 to exclude the FEC; the X bit is set to 0 to include the FEC.

The reserved bits MUST be set to 0 on transmit and MUST be ignore on receive.

The rest of the TLV is encoded following [WSON-Encode] section 4.2.2.

2.1.3. New XRO sub-object: signal processing exclusion

The endpoint restriction only applies to the END-POINTS object.

The PCC/PCE should be able to exclude a signal processing along the path in order to handle client restriction or multi-domain path computation.

In order to support the exclusion a new XRO sub-object is defined: the signal processing exclusion:

```
    0                   1                   2                   3

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

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|  Type = X   |     Length      |   Reserved    | Attribute     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 sub-sub objects                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The Attribute field indicates how the exclusion sub-object is to be interpreted. The Attribute can only be 0 (Interface) or 1 (Node).

The sub-sub objects are encoded as in RSVP signaling definition [WSON-Sign].

2.1.4. IRO sub-object: signal processing inclusion

Similar to the XRO sub-object the PCC/PCE should be able to include a signal processing along the path in order to handle client restriction or multi-domain path computation.

This is supported by adding the sub-object "processing" defined for ERO in [WSON-Sign] to the PCEP IRO object.

2.2. Encoding of a RWA Path Reply

The ERO is used to encode the path of a TE LSP through the network. The ERO is carried within a given path of a PCEP response, which is in turn carried in a PCRep message to provide the computed TE LSP if the path computation was successful. The preferred way to convey the allocated wavelength is by means of Explicit Label Control (ELC) [RFC4003].

In order to encode wavelength assignment, the Wavelength Assignment (WA) Object needs to be employed to be able to specify wavelength assignment. Since each segment of the computed optical path is associated with wavelength assignment, the WA Object should be aligned with the ERO object.

[Ed note: to align with [PCEP-GMPLS] the response WA MAY also include

* SUGGESTED_LABEL_TLV

* UPSTREAM_LABEL_TLV

* LABEL_SET_TLV

specifying the allocated labels according to the requested policies]

2.3. Error Indicator

To indicate errors associated with the RWA request, a new Error Type (TDB) and subsequent error-values are defined as follows for inclusion in the PCEP-ERROR Object:

A new Error-Type (TDB) and subsequent error-values are defined as follows:

o Error-Type=TBD; Error-value=1: if a PCE receives a RWA request and the PCE is not capable of processing the request due to insufficient memory, the PCE MUST send a PCErr message with a PCEP-ERROR Object (Error-Type=TDB) and an Error-value(Error-value=1). The PCE stops processing the request. The corresponding RWA request MUST be cancelled at the PCC.

o Error-Type=TBD; Error-value=2: if a PCE receives a RWA request and the PCE is not capable of RWA computation, the PCE MUST send a PCErr message with a PCEP-ERROR Object (Error-Type=15) and an Error-value (Error-value=2). The PCE stops processing the request. The corresponding RWA computation MUST be cancelled at the PCC.

2.4. NO-PATH Indicator

To communicate the reason(s) for not being able to find RWA for the path request, the NO-PATH object can be used in the PCRep message. The format of the NO-PATH object body is defined in [RFC5440]. The object may contain a NO-PATH-VECTOR TLV to provide additional information about why a path computation has failed.

Two new bit flags are defined to be carried in the Flags field in the NO-PATH-VECTOR TLV carried in the NO-PATH Object.

o Bit TDB: When set, the PCE indicates no feasible route was found that meets all the constraints associated with RWA.

- o Bit TDB: When set, the PCE indicates that no wavelength was assigned to at least one hop of the route in the response.

- o Bit TDB: When set, the PCE indicate that no path was found satisfying the signal compatibility constraints.

3. Manageability Considerations

Manageability of WSON Routing and Wavelength Assignment (RWA) with PCE must address the following considerations:

3.1. Control of Function and Policy

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

- o The ability to send a WSON RWA request.

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

- o The support for WSON RWA.

- o A set of WSON RWA specific policies (authorized sender, request rate limiter, etc).

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

3.2. Information and Data Models, e.g. MIB module

Extensions to the PCEP MIB module defined in [PCEP-MIB] should be defined, so as to cover the WSON RWA information introduced in this document. A future revision of this document will list the information that should be added to the MIB module.

3.3. 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 section 8.3 of [RFC5440].

3.4. Verifying Correct Operation

Mechanisms defined in this document do not imply any new verification requirements in addition to those already listed in section 8.4 of [RFC5440]

3.5. Requirements on Other Protocols and Functional Components

The PCE Discovery mechanisms ([RFC5089] and [RFC5088]) may be used to advertise WSON RWA path computation capabilities to PCCs.

3.6. Impact on Network Operation

Mechanisms defined in this document do not imply any new network operation requirements in addition to those already listed in section 8.6 of [PCEP].

4. Security Considerations

This document has no requirement for a change to the security models within PCEP [PCEP]. However the additional information distributed in order to address the RWA problem represents a disclosure of network capabilities that an operator may wish to keep private. Consideration should be given to securing this information.

5. IANA Considerations

A future revision of this document will present requests to IANA for codepoint allocation.

6. Acknowledgments

The authors would like to thank Adrian Farrel for many helpful comments that greatly improved the contents of this draft.

This document was prepared using 2-Word-v2.0.template.dot.

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.

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

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

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, January 2003.

[RFC4003] Berger, L., "GMPLS Signaling Procedure for Egress Control", RFC 4003, February 2005.

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

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

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

7.2. Informative References

[PCEP-GMPLS] Margaria, et al., "PCEP extensions for GMPLS", draft-ietf-pce-gmpls-pcep-extensions, work in progress.

[PCEP-Layer] Oki, Takeda, Le Roux, and Farrel, "Extensions to the Path Computation Element communication Protocol (PCEP) for Inter-Layer MPLS and GMPLS Traffic Engineering", draft-ietf-pce-inter-layer-ext, work in progress.

[WSON-Frame] Lee, Y. and Bernstein, G. (Editors), and W. Imajuku, "Framework for GMPLS and PCE Control of Wavelength Switched Optical Networks", draft-ietf-ccamp-rwa-wson-framework, work in progress.

[PCE-RWA] Lee, Y., et. al., "PCEP Requirements for WSON Routing and Wavelength Assignment", draft-ietf-pce-wson-routing-wavelength, work in progress.

[Lambda-Label] Tomohiro, O. and D. Li, "Generalized Labels for Lambda-Switching Capable Label Switching Routers", draft-ietf-ccamp-gmpls-g-694-lambda-labels, work in progress.

[WSON-Sign] Bernstein et al," Signaling Extensions for Wavelength Switched Optical Networks", draft-ietf-ccamp-wson-signaling, work in progress.

[WSON-OSPF] Lee and Bernstein," OSPF Enhancement for Signal and Network Element Compatibility for Wavelength Switched Optical Networks", draft-ietf-ccamp-wson-signal-compatibility-ospf, work in progress.

[RWA-Info] Bernstein and Lee, "Routing and Wavelength Assignment Information Model for Wavelength Switched Optical Networks",draft-ietf-ccamp-rwa-info, work in progress.

[RWA-Encode] Bernstein and Lee, "Routing and Wavelength Assignment Information Encoding for Wavelength Switched Optical Networks", draft-ietf-ccamp-rwa-wson-encode, work in progress.

[WSON-Imp] Y. Lee, G. Bernstein, D. Li, G. Martinelli, "A Framework for the Control of Wavelength Switched Optical Networks (WSON) with Impairments", draft-ietf-ccamp-wson-impairments, work in progress.

[RSVP-Imp] agraz, "RSVP-TE Extensions in Support of Impairment Aware Routing and Wavelength Assignment in Wavelength Switched Optical Networks WSONs)", draft-agraz-ccamp-wson-impairment-rsvp, work in progress.

[OSPF-Imp] Bellagamba, et al., "OSPF Extensions for Wavelength Switched Optical Networks (WSON) with Impairments", draft-eb-ccamp-ospf-wson-impairments, work in progress.

8. Contributors

Authors' Addresses

Young Lee
Huawei Technologies
1700 Alma Drive, Suite 100
Plano, TX 75075, USA
Phone: (972) 509-5599 (x2240)
Email: leeyoung@huawei.com

Ramon Casellas
CTTC PMT Ed B4 Av. Carl Friedrich Gauss 7
08860 Castelldefels (Barcelona)
Spain
Phone: (34) 936452916
Email: ramon.casellas@cttc.es

Cyril Margaria
Nokia Siemens Networks
St Martin Strasse 76
Munich, 81541
Germany
Phone: +49 89 5159 16934
Email: cyril.margaria@nsn.com

Oscar Gonzalez de Dios
Telefonica Investigacion y Desarrollo
C/ Emilio Vargas 6
Madrid, 28043
Spain
Phone: +34 91 3374013
Email: ogondio@tid.es

Intellectual Property Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

Network Working Group
Internet-Draft
Intended status: Informational
Expires: July 17, 2011

Xuerong Wang
ZTE Corporation
Jan 13, 2011

Extensions of Backward-Recursive PCE-Based Computation (BRPC) to Support Inter-Autonomous System (AS) Bidirectional LSP Path Computation

draft-wang-pce-inter-as-extentions-00.txt

Abstract

This document provides extensions for the Backward-Recursive PCE-Based Computation (BRPC) to support bidirectional LSP path computation.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

Requirements for establishing Multiprotocol Label Switching Traffic Engineering (MPLS-TE) Label Switched Paths (LSPs) that cross multiple Autonomous Systems (ASes) are described in [RFC4216]. As described in [RFC4216], a method SHOULD provide the ability to compute a path spanning multiple ASes. So a path computation entity that may be the head-end Label Switching Router (LSR), an AS Border Router (ASBR), or a Path Computation Element [PCE] needs to know the TE information not only of the links within an AS, but also of the links that connect to other ASes.

As described in [RFC5392], two new LSAs are defined to advertise inter-AS TE information for OSPFv2 and OSPFv3 separately, and three new sub-TLVs are added to the existing Link TLV to carry the information about the neighboring AS and the remote ASBR. [RFC5316] defines similar extensions for [ISIS].

In order for bidirectional path computation, PCE needs to get bidirectional Inter-AS TE link information. [RFC5392] introduces a "proxy" for the ASBR at the edge of the other AS and generate a bidirectional TE link.

This document extends BRPC in order to support the bidirectional path computation within single procedure. Based on the mechanism in this document, we don't need to introduce the 'proxy'. It shows how the Backward-Recursive PCE-Based Computation (BRPC) - procedures for Inter-AS TE Links can be extended in order for deriving the optimum end-to-end bidirectional path.

This document does not propose or define any mechanisms to advertise any other extra-AS TE information within IGP.

1.1. 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].

2. Terminology

- ASBR: Autonomous System Border Router. Router used to connect together ASes of the same or different service providers via one or more inter-AS links.

- Boundary Node (BN): a boundary node is either an ABR in the context of inter-area Traffic Engineering or an ASBR in the context of inter-AS Traffic Engineering.

- o Entry BN of domain(n): a BN connecting domain(n-1) to domain(n) along a determined sequence of domains.

- o Exit BN of domain(n): a BN connecting domain(n) to domain(n+1) along a determined sequence of domains.

- o PCC: Path Computation Client. Any client application requesting a path computation to be performed by a Path Computation Element.

- o 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.

- o PCE(i) is a PCE with the scope of domain(i).d

- o TED: Traffic Engineering Database.

- o VSPT: Virtual Shortest Path Tree.

- o IVSPT: Inter-AS Virtual Shortest Path Tree.

3. Problem statement

3.1. Inter-AS TE link

As mentioned in [RFC5392] and [RFC5316], Hellos MUST NOT be exchanged over the Inter-AS TE link, and consequently, an IGP adjacency MUST NOT be formed.

In the current operation of TE IGP, the LSRs at each end of a TE link emit LSAs describing the link. The databases in the LSRs then have two entries (one locally generated, the other from the peer) that describe the different 'directions' of the link. This enables Constrained Shortest Path First (CSPF) to do a two-way check on the link when performing path computation and eliminate it from consideration unless both directions of the link satisfy the required constraints.

In the case we are considering here (i.e., of a TE link to another AS), there is, by definition, no IGP peering and hence no bidirectional TE link information.

The information advertised comes from the ASBR's knowledge of the unidirectional TE capabilities of the link, the ASBR's knowledge of the unidirectional current status and usage of the link, and configuration at the ASBR of the remote AS number and remote ASBR TE Router ID.

For other properties, e.g., bandwidth and metrics, an ASBR is difficult or impossible to get the latest value of these properties about reverse directional of Inter-AS TE links timely. In order for the CSPF route computation entity to include the link as a candidate path, we have to find a way to solve this problem.

3.2. Backward Recursive Path Computation

The Backward Recursive Path Computation (BRPC) [RFC5441] procedure involves cooperation and communication between PCEs in order to compute an optimal end-to-end path across multiple domains. In particular, the PCC sends a PCReq to a PCE in its domain. The request is forwarded between PCEs, domain-by-domain, until the PCE responsible for the domain containing the LSP destination is reached. The PCE in the destination domain creates a tree of potential paths to the destination (the Virtual Shortest Path Tree - VSPT) and passes this back to the previous PCE in a PCRep. Each PCE in turn adds to the VSPT and passes it back until the PCE in the source domain uses the VSPT to select an end-to-end path that the PCE sends to the PCC.

VSPT(i) defined in [RFC5441]:

In each domain i:

- o There is a set of X-en(i) entry BNs noted BN-en(k,i) where BN-en(k,i) is the kth entry BN of domain(i).

- o There is a set of X-ex(i) exit BNs noted BN-ex(k,i) where BN-ex(k,i) is the kth exit BN of domain(i).

VSPT(i): MP2P (multipoint-to-point) tree returned by PCE(i) to PCE(i-1):

```
                     Root (TE LSP destination)
                     /        |          \
                BN-en(1,i)   BN-en(2,i) ... BN-en(j,i).

                where [X-en(i)] is the number of
                entry BNs in domain i and j<= [X-en(i)]
```

Figure 1: MP2P VSPT Tree in RFC5441

Each link of tree VSPT(i) represents the shortest constrained path between BN-en(j,i) and the TE LSP destination that satisfies the set of required constraints for the TE LSP (bandwidth, affinities, etc.).

Note that PCE(i) only considers the entry BNs of domain(i), i.e., only the BNs that provide connectivity from domain(i-1). In other words, the set BN-en(k,i) is only made of those BNs that provide connectivity from domain (i-1) to domain(i). Furthermore, some BNs may be excluded according to policy constraints (either due to local policy or policies signaled in the path computation request).

BRPC procedure defined in [RFC5441]:

Step 1: First, the PCC needs to determine the PCE capable of serving its path computation request (this can be done with local configuration or via IGP discovery (see [RFC5088] and [RFC5089])). The path computation request is then relayed until reaching a PCE(n) such that the TE LSP destination resides in the domain(n).

Step 2: PCE(n) computes VSPT(n), the tree made of the list of shortest constrained paths between every BN-en(j,n) and the TE LSP destination using a suitable path computation algorithm (e.g., CSPF) and returns the computed VSPT(n) to PCE(n-1).

Step i: For i=n-1 to 2: PCE(i) computes VSPT(i), the tree made of the shortest constrained paths between each BN-en(j,i) and the TE LSP destination. It does this by considering its own TED and the information in VSPT(i+1).

Step n: Finally, PCE(1) computes the end-to-end shortest constrained path

In the case of inter-domain LSP computation, PCE(i)(i=n-1 to 2) also requires adding the inter-AS TE links that connect the domain(i) and the domain(i+1). So the BRPC procedure requires the knowledge of the traffic engineering attributes of the bidirectional inter-domain TE links at step i.

4. Solutions of Inter-AS bidirectional path computation

When multiple PCEs cooperate each other to compute a bidirectional Inter-AS LSP by using BRPC, PCE(i+1) selects some proper Inter-AS TE links including traffic engineering capabilities, current status and usage whose direction is from AS(i+1) to AS(i). These Inter-AS TE links satisfy the required set of unidirectional TE constraints, So PCE(i+1) sends PCRep message to PCE(i) including the selected unidirectional Inter-AS TE links. The bidirectional Inter-AS TE links that satisfy the constraints can then be derived by PCE(i).

4.1. Extensions of Backward-Recursive PCE-Based Computation (BRPC)

In order to solve the problem about Inter-AS TE links described in previous section, the extension of BRPC procedure for bidirectional Inter-AS TE LSP is described as follows:

o After computing the shortest constrained paths(i.e., VSPT) between every entry BN and the TE LSP destination, PCE(i+1) selects the Inter-AS TE links from AS(i+1) to AS(i) that satisfy the constraints and passes them back to the PCE(i) in a PCRep.

o Then, the PCE(i) should choose among the Inter-AS TE links carried in received PCRep message that satisfy the constraints in the reverse direction ,and compute the shortest constrained paths between every exit BN and the TE LSP destination.

Following is the extended BRPC procedure:

o Step 1:

First, the PCC needs to determine the PCE capable of serving its path computation request (this can be done with local configuration or via IGP discovery (see [RFC5088] and [RFC5089])). The path computation request is then relayed until reaching a PCE(n) such that the TE LSP destination resides in the domain(n). This step is the same as described in [RFC5441].

o Step 2:

2.1. PCE(n) computes the list of shortest constrained paths between every BN-en(j,n) and the TE LSP destination;

2.2. PCE(n) selects the Inter-AS TE links that satisfy the constraints from all of the Inter-AS TE links that provide connectivity from domain (n) to domain(n-1);

2.3. PCE(n) returns PCRep (including result of 2.1 and 2.2) to PCE(n-1).

Note that for unidirectional LSP computation, step 2.2 may not be performed.

o Step i:

For i=n-1 to 2: {

i.1. With the Inter-AS TE links returned from PCE(i+1), PCE(i) chooses the links that satisfy the constraints in the reverse direction(i.e., the selected Inter-AS TE links satisfy the required constraints in both of the directions.)

i.2. PCE(i) computes the shortest constrained paths between each BN-ex(j,i) and the TE LSP destination;

i.3. PCE(i) computes the shortest constrained paths between each BN-en(j,i) and the TE LSP destination;

i.4. PCE(i) selects the Inter-AS TE links that unidirectionally satisfy the constraints from all of the Inter-AS TE links that provide connectivity from domain (i) to domain(i-1);

i.5. PCE(i) returns PCRep (including result of i.3 and i.4) to PCE(i-1). }

Note that for unidirectional LSP computation, step i.1, i.2 and i.4 may not be performed.

o Step n:

n.1. With the Inter-AS TE links returned from PCE(2), PCE(1) chooses the links that satisfy the constraints in the reverse direction (i.e., the selected Inter-AS TE links satisfy the required constraints in both of the directions.);

n.2. PCE(1) computes the shortest constrained paths between each BN-ex(j,1) and the TE LSP destination;

n.3. Finally, PCE(1) computes the end-to-end shortest constrained path.

Note that for unidirectional LSP computation, step n.1.and n.2. may not be performed.

Note that uni-direction represents the direction from entry BNs of local domain i to exit BNs of domain i-1, the reverse direction represents the direction from exit BNs of local domain i to entry BNs of domain i+1.

4.2. Extensions of PCEP

4.2.1. IVSPT flag

PCEP needs to be introduced a new flag in RP object carried within the PCReq message (defined in [RFC5440]). The PCE(i) set this flag in PCReq to indicates that the Inter-AS TE links from AS(i+1) to AS(i) satisfying the constraints must be return. In other words, the PCE(i) requests the computation of an inter-domain TE LSP using the new BRPC procedure defined in this document. The IVSPT Flag set in PCRep to indicates that requested PCE support the new enhanced BRPC procedure, and Inter-AS TE links from AS(i+1) to AS(i) satisfying the constraints have been included in PCRep.

The following new flag of the RP object is defined:

IVSPT Flag

Bit Number	Name Flag
TBD	IVSPT

4.2.2. BRPC Procedure Completion Failure

If PCE(i) send IVSPT flag to PCE(i+1) who doesn't recognizes the IVSPT flag of RP object, PCE(i+1) MUST generate PCErr message with an Error-Type=4 (Not supported object), Error-value=4 (Unsupported parameter). The PCE may include the parent object (RP object) up to and including (but no further than) the unknown or unsupported parameter. In this case where the unknown or unsupported parameter is a bit flag (IVSPT flag), the included RP object should contain the whole bit flag field with all bits after the parameter at issue set to zero. The corresponding path computation request is then cancelled by the PCE without further notification.

If PCE(i) send IVSPT flag to PCE(i+1) who recognizes IVSPT flag of RP object but does not support the new BRPC procedure extended in this document, it MUST return a PCErr message to the upstream PCE with an Error-Type " Enhanced BRPC procedure unsupported".

The PCErr message MUST be relayed to the requesting PCC.

PCEP-ERROR objects are used to report a PCEP protocol error and are characterized by an Error-Type that specifies the type of error and an Error-value that provides additional information about the error type. Both the Error-Type and the Error-value are managed by IANA.

A new Error-Type is defined that relates to the BRPC procedure.

Error-Type	Meaning
TBD	Enhanced BRPC procedure unsupported
	Error-value 1 Enhanced BRPC procedure not supported by one or more PCEs along the domain path

4.2.3. Inter-AS TE links carried in PCEP message

For the enhanced bidirectional Inter-AS TE LSP BRPC procedure referenced in this document, PCE(n) should select the unidirectional Inter-AS TE links that satisfy the constraints from all of the Inter-AS TE links that provide connectivity from domain (n) to domain(n-1),and then PCE(n) should return the selected Inter-AS TE links in PCRep message.

Two methods of carrying Inter-AS TE links in PCRep are introduced in this document. One is to introduce a new object (Inter-AS Virtual Shortest Path Tree -IVSPT) to carry Inter-AS TE links separately(see 4.2.3.1.). The other is to extend VSPT to include Inter-AS TE links(see 4.2.3.2.).

4.2.3.1. Inter-AS Virtual Shortest Path Tree object (IVSPT)

4.2.3.1.1. Definition of IVSPT(i)

Mode of BRPC Operation is introduced in [RFC 5441]:

Definition of VSPT(i)

In each domain i:

- o Step 1: There is a set of X-en(i) entry BNs noted BN-en(k,i) where BN-en(k,i) is the kth entry BN of domain(i).

- o There is a set of X-ex(i) exit BNs noted BN-ex(k,i) where BN-ex(k,i) is the kth exit BN of domain(i).

VSPT(i): MP2P (multipoint-to-point) tree returned by PCE(i) to PCE(i-1):

```
                   Root (TE LSP destination)
                     /        |           \
               BN-en(1,i)   BN-en(2,i) ... BN-en(j,i).

                   where [X-en(i)] is the number of
               entry BNs in domain i and j<= [X-en(i)]
```

Figure 2: MP2P VSPT

Each link of tree VSPT(i) represents the shortest constrained path between BN-en(j,i) and the TE LSP destination that satisfies the set of required constraints for the TE LSP (bandwidth, affinities, etc.).These are path segments to reach the TE LSP destination from BN-en(j,i).

Note that PCE(i) only considers the entry BNs of domain(i), i.e., only the BNs that provide connectivity from domain(i-1).

Besides VSPT, this document defines Inter-AS Virtual Shortest Path Tree (IVSPT) used for describing unidirectional Inter-AS paths whose direction is from AS(i) to AS(i-1).

Definition of IVSPT(j,i) :

IVSPT(j,i): jth MP2P (multipoint-to-point) tree returned by PCE(i) to PCE(i-1)

IVSPT(1,i):

```
                    BN-en(1,i)
              /          |         \
  BN-ex(1,i-1)  BN-ex(2,i-1) ... BN-ex(k1,i-1).
```

IVSPT(2,i):

```
                     BN-en(2,i)
              /          |         \
  BN-ex(1,i-1)  BN-ex(2,i-1) ... BN-ex(k2,i-1).
```

IVSPT(j,i):

```
                    BN-en(j,i)
              /          |         \
  BN-ex(1,i-1)  BN-ex(2,i-1) ... BN-ex(kj,i-1).
```

where
[X-en(i)] is the number of entry BNs in domain i and j<= [X-en(i)],
[Y-ex(i-1)] is the number of exit BNs in domain i-1
and k1,k2,...,kj<= [X-ex(i-1)]

Figure 3: IVSPT

IVSPT(j,i) represents the Inter-AS paths from BN-en(j,i) of domain i to exit BNs of domain i-1 that satisfies the set of required constraints for the TE LSP (bandwidth, affinities, etc.).

4.2.3.1.2. Constrain Route Object(CRO)

The CRO is used to encode the Inter-AS paths that satisfies the set of required constraints for the TE LSP.

The CRO is carried within a PCRep message to provide the selected Inter-AS links if the path computation was successful.

The contents of this object are identical to the contents of the RSVP-TE ERO defined in [RFC3209], [RFC3473], and [RFC3477]. That is, the object is constructed from a series of sub-objects. Any RSVP-TE ERO sub-object already defined or that could be defined in the future for use in the RSVP-TE ERO is acceptable in this object.

```
        0                   1                   2                   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                               |
       //                        (Sub-objects)                        //
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       Sub-objects:
              Type   Sub-object
               1     IPv4 prefix
               2     IPv6 prefix
               4     Unnumbered Interface ID
```

Figure 4: Format of CRO

The format of the PCRep message is updated as follows :

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

   where:

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

     <response>::=<RP>
                  [<NO-PATH>]
                  [<attribute-list>]
                  [<path-list>]

     <path-list>::=<path>[<path-list>][<CRO-list>]

     <path>::= <ERO><attribute-list>

     where:

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

       <metric-list>::=<METRIC>[<metric-list>]

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

4.2.3.1.3. IVSPT Encoding

The IVSPT is returned within a PCRep message. The encoding consists of a non-ordered list of Constrain Route Objects (CROs) where each CRO represents an Inter-AS link that satisfy the required constraint from domain i to domain i+1.

Example:

```
R1------R3----R5-----R7------R9-----R11---- R13
        | \   | \     |     / |             /
        |  \  |   \   |  ---- |  ----------
        |   \ |     \ | /     | /
R2------R4----R6-----R8------R10----R12
           :              :
<-- AS1 -->:<---- AS2 --->:<------- AS3 --------->
```

Figure 5: An Example of Inter-AS path computation

In the example shown in Figure 5, if we make the assumption that a constrained path exists between each ABR and the destination R13, the VSPT computed by a PCE(3) serving AS 3 consists of the following non-ordered set of EROs:

- ERO1: R9(TE Router ID)-R11(Interface IP address)-R13(TE Router ID)

- ERO2: R10(TE Router ID)-R13(TE Router ID)

If we make the assumption that Inter-AS links R9-->R7 ,R9-->R8 and R10-->R8 satisfy the required constraints, the IVSPT selected by a PCE(3) serving AS 3 consists of the following non-ordered set of CROs:

- CRO1: R9(Interface IP address),R7(TE Router ID)

- CRO2: R9(Interface IP address),R8(TE Router ID)

- CRO3: R10(Interface IP address),R8(TE Router ID)

4.2.3.2. Extend VSPT to include Inter-AS links

4.2.3.2.1. New definition of VSPT(i)

In each domain i:

- There is a set of X-en(i) entry BNs noted BN-en(k,i) where BN-en(k,i) is the kth entry BN of domain(i).

- o There is a set of X-ex(i) exit BNs noted BN-ex(k,i) where BN-ex(k,i) is the kth exit BN of domain(i).

For the bidirectional LSP computation, VSPT(i) may be extended as follows:

```
              Root(TE LSP destination)
             /          |         |       \
            /           |   ...   |        \
   BN-en(1,i)...BN-en(1,i)... BN-en(j,i)...BN-en(j,i)
       |           |         |          |
 BN-ex(1,i-1)...BN-ex(k1,i-1) BN-ex(1,i-1)...BN-ex(kj,i-1)

 where:

 [X-en(i)] is the number of entry BNs in domain i and j<= [X-en(i)];

 [Y-ex(i-1)] is the number of exit BNs in domain i-1,
 kj is the number of exit BNs in domain i-1 that connect
 BN-en(j,i)(i.e., the jth entry BN in domain i),
 and k1,k2,...,kj <= [Y-ex(i-1)]
```

Figure 6: IVSPT

IVSPT(i) includes links that represent the shortest constrained path between BN-en(j,i) and the TE LSP destination ，and Inter-AS links that satisfie the set of required constraints for the TE LSP (bandwidth, affinities, etc.) from AS(i) to AS(i+1). These are path segments to reach the TE LSP destination from BN-ex(j,i-1).

4.2.3.2.2. VSPT Encoding

In the example shown in Figure 5, if we make the assumption that a constrained path exists between each ABR and the destination R13(i.e.,R9-R11-R13 and R10-R13), and Inter-AS links R9-->R7 ,R9-->R8 and R10-->R8 satisfy the required constraints, the VSPT computed for the bidirectional LSP by a PCE(3) serving AS 3 consists of the following non-ordered set of EROs:

- o ERO1: R7(TE Router ID)-R9(Interface IP address)-R11(Interface IP address)-R13(TE Router ID)

- o ERO2: R8(TE Router ID)-R9(Interface IP address)-R11(Interface IP address)-R13(TE Router ID)

- o ERO3: R8(TE Router ID)-R10(Interface IP address)-R13(TE Router ID)

For the unidirectional LSP computation , VSPT is the same as defined in [RFC5441].

5. Security Considerations

TBD.

6. IANA considerations

TBD.

7. Acknowledgments

TBD.

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.

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

[RFC4726] Farrel, A., Vasseur, J., and A. Ayyangar, "A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering", RFC 4726, November 2006.

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

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

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

[RFC5152] Vasseur, JP., Ayyangar, A., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, February 2008.

[RFC5316] Chen, M., Zhang, R., and X. Duan, "ISIS Extensions in Support of Inter-Autonomous System (AS) MPLS and GMPLS Traffic Engineering", RFC 5316, December 2008.

[RFC5376] Bitar, N., Zhang, R., and K. Kumaki, "Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)", RFC 5376, November 2008.

[RFC5392] Chen, M., Zhang, R., and X. Duan, "OSPF Extensions in Support of Inter-Autonomous System (AS) MPLS and GMPLS Traffic Engineering", RFC 5392, January 2009.

[RFC5394] Bryskin, I., Papadimitriou, D., Berger, L., and J. Ash, "Policy-Enabled Path Computation Framework", RFC 5394, December 2008.

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

[RFC5441] Vasseur, JP., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", RFC 5441, April 2009.

8.2. Informative References

[H-PCE] D. King, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS & GMPLS", draft-king-pce-hierarchy-fwk-03.txt .

Author's Address

Xuerong Wang
ZTE Corporation
6F, R&D Building 3, ZTE Industrial Park, XiLi LiuXian Road,
Nanshan District, Shenzhen 518055
P.R.China

Phone: +86 755 26773609
Email: wang.xuerong@zte.com.cn
URI: http://www.zte.com.cn/