Internet Engineering Task Force (IETF)
Request for Comments: 6006
Category: Standards Track
ISSN: 2070-1721

Q. Zhao, Ed.
Huawei Technology
D. King, Ed.
Old Dog Consulting
F. Verhaeghe
Thales Communication France
T. Takeda
NTT Corporation
Z. Ali
Cisco Systems, Inc.
J. Meuric
France Telecom

September 2010

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

Abstract

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

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

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6006.

Table of Contents

1. Introduction 3
   1.1. Terminology 4
   1.2. Requirements Language 5
2. PCC-PCE Communication Requirements 5
3. Protocol Procedures and Extensions 6
   3.1. P2MP Capability Advertisement 6
        3.1.1. P2MP Computation TLV in the Existing PCE Discovery Protocol 6
        3.1.2. Open Message Extension 7
   3.2. Efficient Presentation of P2MP LSPs 7
   3.3. P2MP Path Computation Request/Reply Message Extensions 8
        3.3.1. The Extension of the RP Object 8
        3.3.2. The New P2MP END-POINTS Object 9
   3.4. Request Message Format 12
   3.5. Reply Message Format 12
   3.6. P2MP Objective Functions and Metric Types 13
        3.6.1. New Objective Functions 13
        3.6.2. New Metric Object Types 14
   3.7. Non-Support of P2MP Path Computation 14

3.8. Non-Support by Back-Level PCE Implementations 15
3.9. P2MP TE Path Reoptimization Request 15
3.10. Adding and Pruning Leaves to/from the P2MP Tree 16
3.11. Discovering Branch Nodes 19
3.11.1. Branch Node Object 19
3.12. Synchronization of P2MP TE Path Computation Requests 19
3.13. Request and Response Fragmentation 20
3.13.1. Request Fragmentation Procedure 21
3.13.2. Response Fragmentation Procedure 21
3.13.3. Fragmentation Examples 21
3.14. UNREACH-DESTINATION Object 22
3.15. P2MP PCEP-ERROR Objects and Types 23
3.16. PCEP NO-PATH Indicator 24
4. Manageability Considerations 25
4.1. Control of Function and Policy 25
4.2. Information and Data Models 25
4.3. Liveness Detection and Monitoring 25
4.4. Verifying Correct Operation 25
4.5. Requirements for Other Protocols and Functional Components 26
4.6. Impact on Network Operation 26
5. Security Considerations 26
6. IANA Considerations 27
6.1. PCEP TLV Type Indicators 27
6.2. Request Parameter Bit Flags 27
6.3. Objective Functions 27
6.4. Metric Object Types 27
6.5. PCEP Objects 28
6.6. PCEP-ERROR Objects and Types 29
6.7. PCEP NO-PATH Indicator 30
6.8. SVEC Object Flag 30
6.9. OSPF PCE Capability Flag 30
7. Acknowledgements 30
8. References 30
8.1. Normative References 30
8.2. Informative References 32

1. Introduction

The Path Computation Element (PCE) defined in [RFC4655] is an entity that is capable of computing a network path or route based on a network graph, and applying computational constraints. A Path Computation Client (PCC) may make requests to a PCE for paths to be computed.

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

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

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

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

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

1.1. Terminology

Terminology used in this document:

TE LSP: Traffic Engineering Label Switched Path.

LSR: Label Switching Router.

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

P2MP: Point-to-Multipoint.

P2P: Point-to-Point.

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

1.2. Requirements Language

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

2. PCC-PCE Communication Requirements

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

3. Protocol Procedures and Extensions

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

3.1. P2MP Capability Advertisement

3.1.1. P2MP Computation TLV in the Existing PCE Discovery Protocol

[RFC5088] defines a PCE Discovery (PCED) TLV carried in an OSPF Router Information Link State Advertisement (LSA) defined in [RFC4970] to facilitate PCE discovery using OSPF. [RFC5088] specifies that no new sub-TLVs may be added to the PCED TLV. This document defines a new flag in the OSPF PCE Capability Flags to indicate the capability of P2MP computation.

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

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

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

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

3.1.2. Open Message Extension

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

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

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

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

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

3.2. Efficient Presentation of P2MP LSPs

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

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

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

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

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

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

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

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

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

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

3.3. P2MP Path Computation Request/Reply Message Extensions

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

3.3.1. The Extension of the RP Object

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

This document adds the following flags to the RP Object:

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

o F (RP fragmentation bit - 1 bit):

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

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

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

o N (P2MP bit - 1 bit):

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

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

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

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

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

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

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

3.3.2. The New P2MP END-POINTS Object

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

- Old leaves whose path can be modified/reoptimized;
- Old leaves whose path must be left unchanged.

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

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

- New leaves to add (leaf type = 1)
- Old leaves to remove (leaf type = 2)
- Old leaves whose path can be modified/reoptimized (leaf type = 3)
- Old leaves whose path must be left unchanged (leaf type = 4)

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

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

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

The format of the new END-POINTS object body for IPv4 (Object-Type 3) is as follows:

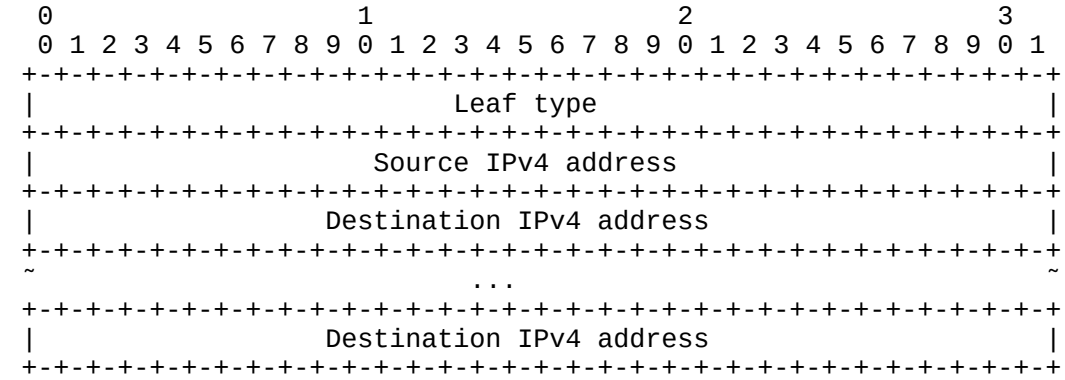

Figure 1. The New P2MP END-POINTS Object Body Format for IPv4

The format of the END-POINTS object body for IPv6 (Object-Type 4) is as follows:

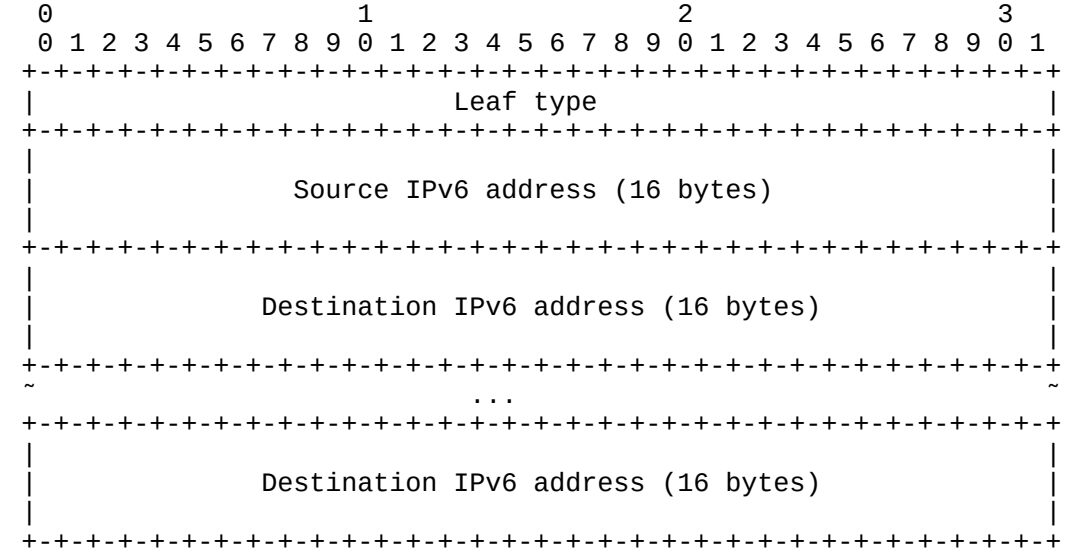

Figure 2. The New P2MP END-POINTS Object Body Format for IPv6

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

3.4. Request Message Format

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

Below is the message format for the request message:

```
        <PCReq Message>::= <Common Header>
                           <request>
     where:
            <request>::= <RP>
                         <end-point-rro-pair-list>
                         [<OF>]
                         [<LSPA>]
                         [<BANDWIDTH>]
                         [<metric-list>]
                         [<IRO>]
                         [<LOAD-BALANCING>]

     where:

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

            <RRO-List>::=<RRO>[<BANDWIDTH>][<RRO-List>]
            <metric-list>::=<METRIC>[<metric-list>]
```

Figure 3. The Message Format for the Request Message

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

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

3.5. Reply Message Format

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

Below is the message format for the reply message:

```
<PCRep Message>::= <Common Header>
                        <response>
<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 4. The Message Format for the Reply Message

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

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

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

3.6. P2MP Objective Functions and Metric Types

3.6.1. New Objective Functions

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

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

The description of the two new objective functions is as follows.

Objective Function Code: 7

Name: Shortest Path Tree (SPT)

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

Objective Function Code: 8

Name: Minimum Cost Tree (MCT)

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

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

3.6.2. New Metric Object Types

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

- P2MP IGP metric: T=8
- P2MP TE metric: T=9
- P2MP hop count metric: T=10

3.7. Non-Support of P2MP Path Computation

- If a PCE receives a P2MP path request and it understands the P2MP flag in the RP object, but the PCE is not capable of P2MP computation, the PCE MUST send a PCErr message with a PCEP-ERROR object and corresponding Error-Value. The request MUST then be cancelled at the PCC. New Error-Types and Error-Values are requested in Section 6 ("IANA Considerations") of this document.

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

3.8. Non-Support by Back-Level PCE Implementations

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

3.9. P2MP TE Path Reoptimization Request

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

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

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

Figure 5. PCReq Message Example 1 for Optimization

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

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

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

Figure 6. PCReq Message Example 2 for Optimization

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

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

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

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

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

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

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

Case 1: Adding leaves with full reoptimization of existing paths

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

Case 2: Adding leaves with partial reoptimization of existing paths

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

Case 3: Adding leaves without reoptimization of existing paths

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

Case 4: Pruning Leaves with full reoptimization of existing paths

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

Case 5: Pruning leaves with partial reoptimization of existing paths

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

Case 6: Pruning leaves without reoptimization of existing paths

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

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

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

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

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

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

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

3.11. Discovering Branch Nodes

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

3.11.1. Branch Node Object

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

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

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

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

3.12. Synchronization of P2MP TE Path Computation Requests

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

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

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

```
Common Header
SVEC for sync of LSP1 and LSP2
OF (optional)
END-POINTS1 for P2MP
  RRO1 list
END-POINTS2 for P2MP
  RRO2 list
```

Figure 7. PCReq Message Example for Synchronization

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

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

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

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

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

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

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

3.13. Request and Response Fragmentation

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

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

3.13.1. Request Fragmentation Procedure

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

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

3.13.2. Response Fragmentation Procedure

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

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

3.13.3. Fragmentation Examples

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

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

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

Figure 8. PCReq Message Fragmentation Example

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

3.14. UNREACH-DESTINATION Object

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

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

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

The following UNREACH-DESTINATION objects will be required:

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

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

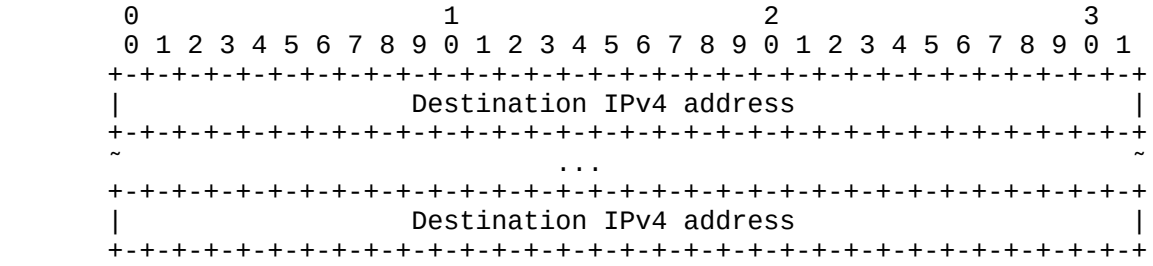

Figure 9. UNREACH-DESTINATION Object Body for IPv4

The format of the UNREACH-DESTINATION object body for IPv6 (Object-Type=2) is as follows:

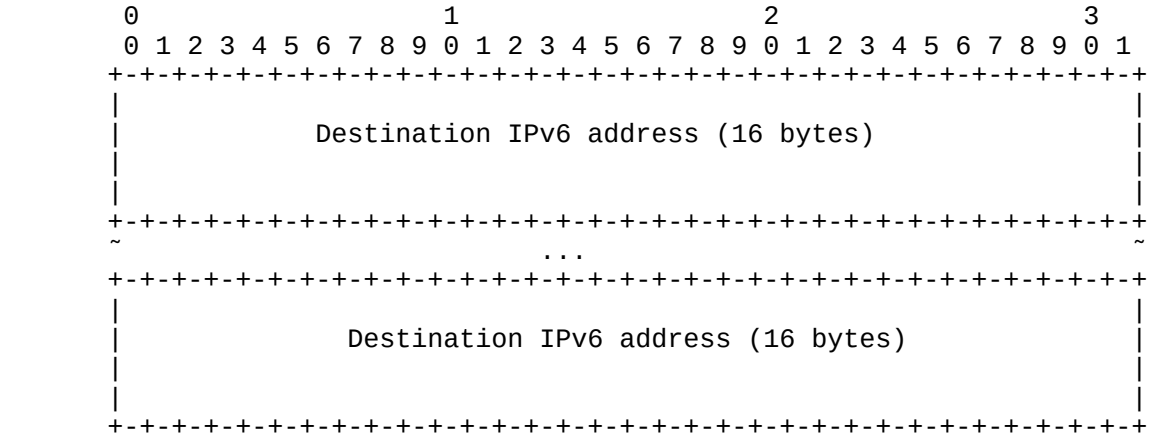

Figure 10. UNREACH-DESTINATION Object Body for IPv6

3.15. P2MP PCEP-ERROR Objects and Types

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

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

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

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

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

To indicate P2MP message fragmentation errors associated with a P2MP path request, a new Error-Type (17) and subsequent error-values are defined as follows for inclusion in the PCEP-ERROR object:

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

3.16. PCEP NO-PATH Indicator

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

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

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

4. Manageability Considerations

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

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

4.1. Control of Function and Policy

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

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

4.2. Information and Data Models

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

4.3. Liveness Detection and Monitoring

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

4.4. Verifying Correct Operation

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

4.5. Requirements for Other Protocols and Functional Components

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

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

4.6. Impact on Network Operation

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

5. Security Considerations

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

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

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

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

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

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

6. IANA Considerations

IANA maintains a registry of PCEP parameters. A number of IANA considerations have been highlighted in previous sections of this document. IANA has made the following allocations.

6.1. PCEP TLV Type Indicators

As described in Section 3.1.2., the newly defined P2MP capability TLV allows the PCE to advertise its P2MP path computation capability. IANA has made the following allocation from the "PCEP TLV Type Indicators" sub-registry.

Value	Description	Reference
6	P2MP capable	RFC 6006

6.2. Request Parameter Bit Flags

As described in Section 3.3.1, three new RP Object Flags have been defined. IANA has made the following allocations from the PCEP "RP Object Flag Field" sub-registry:

Bit	Description	Reference
18	Fragmentation (F-bit)	RFC 6006
19	P2MP (N-bit)	RFC 6006
20	ERO-compression (E-bit)	RFC 6006

6.3. Objective Functions

As described in Section 3.6.1, two new Objective Functions have been defined. IANA has made the following allocations from the PCEP "Objective Function" sub-registry:

Code Point	Name	Reference
7	SPT	RFC 6006
8	MCT	RFC 6006

6.4. Metric Object Types

As described in Section 3.6.2, three new metric object T fields have been defined. IANA has made the following allocations from the PCEP "METRIC Object T Field" sub-registry:

Value	Description	Reference
8	P2MP IGP metric	RFC 6006
9	P2MP TE metric	RFC 6006
10	P2MP hop count metric	RFC 6006

6.5. PCEP Objects

As discussed in Section 3.3.2, two new END-POINTS Object-Types are defined. IANA has made the following Object-Type allocations from the "PCEP Objects" sub-registry:

Object-Class Value	4
Name	END-POINTS
Object-Type	3: IPv4
	4: IPv6
	5-15: Unassigned
Reference	RFC 6006

As described in Section 3.2, Section 3.11.1, and Section 3.14, four PCEP Object-Classes and six PCEP Object-Types have been defined. IANA has made the following allocations from the "PCEP Objects" sub-registry:

Object-Class Value	28
Name	UNREACH-DESTINATION
Object-Type	1: IPv4
	2: IPv6
	3-15: Unassigned
Reference	RFC 6006

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

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

Object-Class Value	31
Name	Branch Node Capability Object
Object-Type	1: Branch node list 2: Non-branch node list 3-15: Unassigned
Reference	RFC 6006

6.6. PCEP-ERROR Objects and Types

As described in Section 3.15, a number of new PCEP-ERROR Object Error Types and Values have been defined. IANA has made the following allocations from the PCEP "PCEP-ERROR Object Error Types and Values" sub-registry:

Error Type	Meaning	Reference
5	Policy violation	
	Error-value=7: P2MP Path computation is not allowed	RFC 6006
16	P2MP Capability Error	
	Error-Value=0: Unassigned	RFC 6006
	Error-Value=1: The PCE is not capable to satisfy the request due to insufficient memory	RFC 6006
	Error-Value=2: The PCE is not capable of P2MP computation	RFC 6006
17	P2MP END-POINTS Error	
	Error-Value=0: Unassigned	RFC 6006
	Error-Value=1: The PCE is not capable to satisfy the request due to no END-POINTS with leaf type 2	RFC 6006
	Error-Value=2: The PCE is not capable to satisfy the request due to no END-POINTS with leaf type 3	RFC 6006
	Error-Value=3: The PCE is not capable to satisfy the request due to no END-POINTS with leaf type 4	RFC 6006
	Error-Value=4: The PCE is not capable to satisfy the request due to inconsistent END-POINTS	RFC 6006
18	P2MP Fragmentation Error	
	Error-Value=0: Unassigned	RFC 6006
	Error-Value=1: Fragmented request failure	RFC 6006

6.7. PCEP NO-PATH Indicator

As discussed in Section 3.16, a new NO-PATH-VECTOR TLV Flag Field has been defined. IANA has made the following allocation from the PCEP "NO-PATH-VECTOR TLV Flag Field" sub-registry:

Bit	Description	Reference
24	P2MP Reachability Problem	RFC 6006

6.8. SVEC Object Flag

As discussed in Section 3.12, two new SVEC Object Flags are defined. IANA has made the following allocation from the PCEP "SVEC Object Flag Field" sub-registry:

Bit	Description	Reference
19	Partial Path Diverse	RFC 6006
20	Link Direction Diverse	RFC 6006

6.9. OSPF PCE Capability Flag

As discussed in Section 3.1.1, a new OSPF Capability Flag is defined to indicate P2MP path computation capability. IANA has made the following assignment from the OSPF Parameters "Path Computation Element (PCE) Capability Flags" registry:

Bit	Description	Reference
10	P2MP path computation	RFC 6006

7. Acknowledgements

The authors would like to thank Adrian Farrel, Young Lee, Dan Tappan, Autumn Liu, Huaimo Chen, Eiji Okim, Nick Neate, Suresh Babu K, Dhruv Dhody, Udayasree Palle, Gaurav Agrawal, Vishwas Manral, Dan Romascanu, Tim Polk, Stewart Bryant, David Harrington, and Sean Turner for their valuable comments and input on this document.

8. References

8.1. Normative References

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

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

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

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

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

[RFC4970] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 4970, July 2007.

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

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

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

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

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

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

8.2. Informative References

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

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

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

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

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

[PCEP-MIB] Koushik, K., Stephan, E., Zhao, Q., and D. King, "PCE communication protocol (PCEP) Management Information Base", Work in Progress, July 2010.

Contributors

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

Mohamad Chaitou
France
EMail: mohamad.chaitou@gmail.com

Authors' Addresses

Quintin Zhao (editor)
Huawei Technology
125 Nagog Technology Park
Acton, MA 01719
US
EMail: qzhao@huawei.com

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

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

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

Zafar Ali
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada
EMail: zali@cisco.com

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