Internet Engineering Task Force (IETF)
Request for Comments: 8306
Obsoletes: 6006
Category: Standards Track
ISSN: 2070-1721

Q. Zhao
D. Dhody, Ed.
R. Palleti
Huawei Technologies
D. King
Old Dog Consulting

November 2017

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.

This document obsoletes RFC 6006.

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 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8306.

Table of Contents

1. Introduction

The Path Computation Element (PCE) as 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 subset of the S2L sub-LSPs. In the extreme case, the PCC may request the S2L sub-LSPs to be computed individually; the PCC is responsible for deciding whether to signal individual S2L sub-LSPs or combine the computation results to signal the entire P2MP TE LSP. Hence, the PCC may use one Path Computation Request message or may split the request across multiple path computation messages.

This document obsoletes [RFC6006] and incorporates the following errata:

- Erratum IDs 3819, 3830, 3836, 4867, and 4868 for [RFC6006]
- Erratum ID 4956 for [RFC5440]

All changes from [RFC6006] are listed in Appendix A.

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

2. PCC-PCE Communication Requirements

This section summarizes the PCC-PCE communication requirements as met by the protocol extension specified in this document for P2MP MPLS-TE LSPs. The numbering system in the list below corresponds to the requirement numbers (e.g., R1, R2) 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 computation 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. IGP Extensions for P2MP Capability Advertisement

[RFC5088] defines a PCE Discovery (PCED) TLV carried in an OSPF Router Information Link State Advertisement (LSA) as defined in [RFC7770] to facilitate PCE discovery using OSPF. [RFC5088] specifies that no new sub-TLVs may be added to the PCED TLV. This document defines a 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 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 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 an optional TLV to indicate the PCE’s capability to perform P2MP path computations.

IANA has allocated value 6 from the "PCEP TLV Type Indicators" subregistry, 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 the 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 when 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 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 are defined in [RFC3209] and [RFC3473].

The Secondary Explicit Route Object (SERO) is used to specify the explicit route of an 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 the format of an ERO as 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 class of the SRRO as 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].

PCEP Object-Class and Object-Type values for the SERO and SRRO have been assigned:

Object-Class Value	29
Name	SERO
Object-Type	0: Reserved 1: SERO 2-15: Unassigned
Reference	RFC 8306

Object-Class Value	30
Name	SRRO
Object-Type	0: Reserved 1: SRRO 2-15: Unassigned
Reference	RFC 8306

The IANA assignments are 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 the 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 (1) that the request and reply messages have been fragmented across multiple messages, (2) that they have been requested for a P2MP path, and (3) 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 Path Computation Request (PCReq) or Path Computation Reply (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 assignments are documented in Section 6.2 ("Request Parameter Bit Flags") of this document.

3.3.2. The 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 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 P2MP 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 four 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 P2MP 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 P2MP END-POINTS object body for IPv4 (Object-Type 3) is as follows:

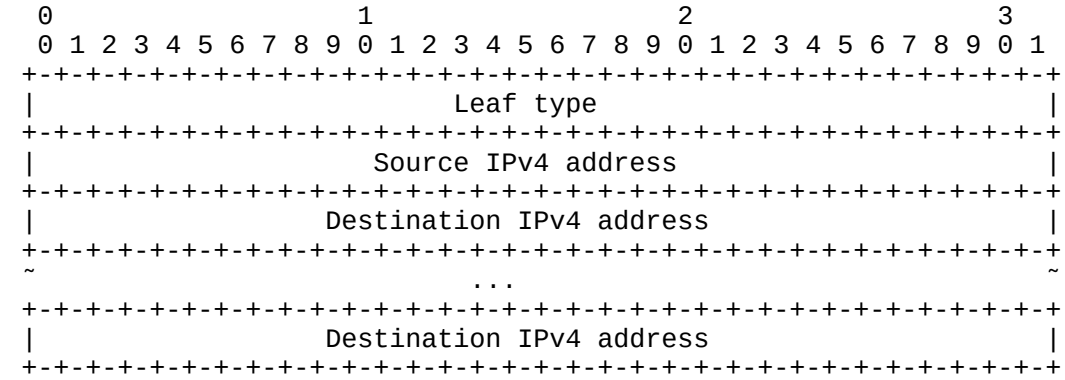

Figure 1: The P2MP END-POINTS Object Body Format for IPv4

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

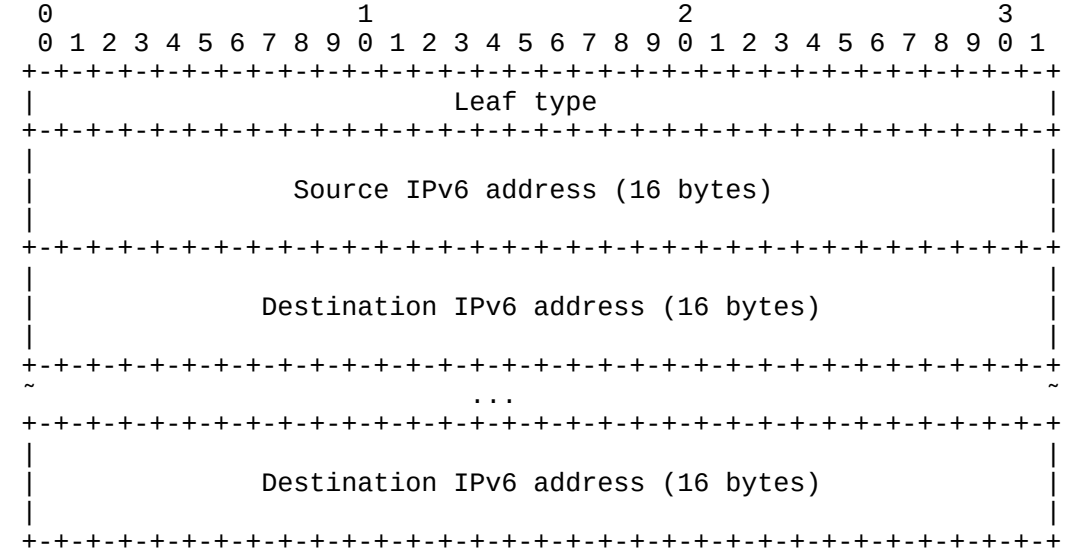

Figure 2: The P2MP END-POINTS Object Body Format for IPv6

The END-POINTS object body has a variable length. These are

- multiples of 4 bytes for IPv4
- multiples of 16 bytes, plus 4 bytes, for IPv6

3.4. Request Message Format

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

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

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

Below is the message format for the request message:

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

where:

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

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

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

where:

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

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

Figure 3: The Message Format for the Request Message

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

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

3.5. Reply Message Format

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

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

Below is the message format for the reply message:

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

where:

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

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

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

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

where:

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

Figure 4: The Message Format for the Reply Message

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

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

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

3.6. P2MP Objective Functions and Metric Types

3.6.1. 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 objective function codes are defined 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 (i.e., 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 objective functions is subject to the rules defined in [RFC5541].

3.6.2. METRIC Object-Type Values

There are three types defined for the METRIC object in [RFC5440] -- namely, the IGP metric, the TE metric, and Hop Counts. 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. The following values for these metric types have been assigned; see Section 6.4.

- 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 computation 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. The Error-Types and Error-values have been assigned; see Section 6 ("IANA Considerations") of this document.
- If the PCE does not understand the P2MP flag in the RP object, then the PCE would send a PCErr message with Error-Type=2 (Capability not supported) as per [RFC5440].

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 PCC MUST insert the list of Record Route Objects (RROs) and SRROs after each instance of the END-POINTS object in the PCReq message, as described in Section 3.4 ("Request Message Format") 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-POINTS object with leaf 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 is 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 PCC MUST build a P2MP request using END-POINTS with leaf type 1.

To remove old leaves, the PCC MUST build a P2MP request using END-POINTS with leaf type 2. If no type-2 END-POINTS exist, then the PCE MUST send Error-Type 17, Error-value 1: the PCE cannot satisfy 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. Specific PCEP-ERROR objects and types are used 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-POINTS" error will be used if a PCC receives a request that has an inconsistent END-POINTS setting (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 PCC 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) as defined in [RFC5440], except that it only supports IPv4 and IPv6 prefix sub-objects. Two Object-Type parameters 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 computation request may carry at most one Branch Node object.

The Object-Class and Object-Type values have been allocated by IANA. The IANA assignments are 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 SVEC functionality as defined in [RFC5440].

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

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

Figure 7: PCReq Message Example for Synchronization

This specification also defines two flags for the SVEC Object Flag Field for P2MP path-dependent computation requests. The first flag allows 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 allows 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 assignments are 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 is outlined in Section 3.3.1 ("The Extension of the RP Object") of this document. The F-bit is used in the RP object to signal that the initial request or response was too large to fit into a single message and will be fragmented into multiple messages. In order to identify the single request or response, each message will use the same request ID.

3.13.1. Request Fragmentation Procedure

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

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

3.13.2. Response Fragmentation Procedure

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

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

3.13.3. Fragmentation Example

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 Req-ID1 specified in the RP object, and with the F-bit set on the first message and the F-bit 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. If the timer expires and it still 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 (for IPv4 and IPv6) are defined:

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:

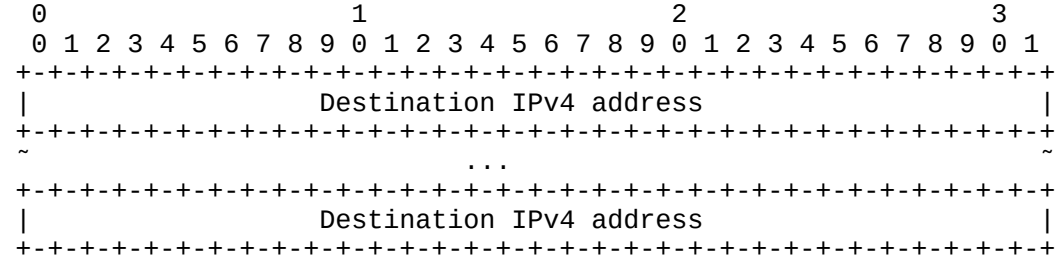

Figure 9: UNREACH-DESTINATION Object Body for IPv4

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

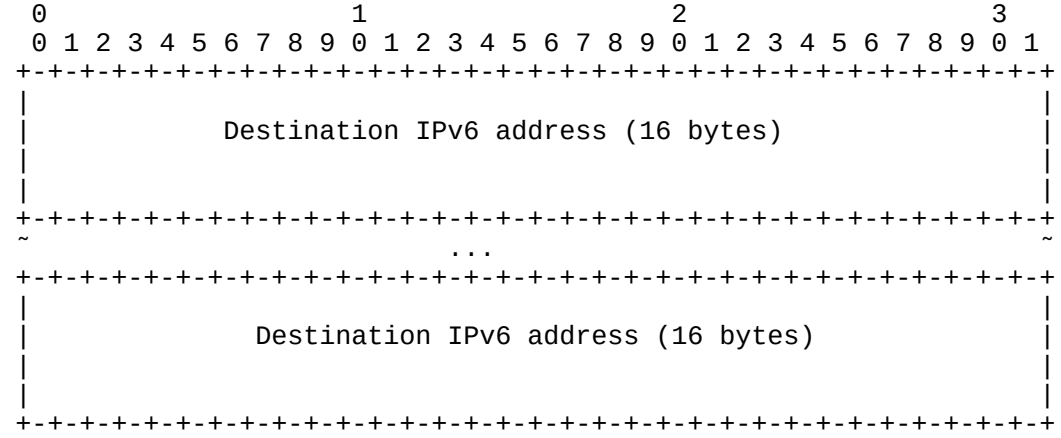

Figure 10: UNREACH-DESTINATION Object Body for IPv6

3.15. P2MP PCEP-ERROR Objects and Types

To indicate an error associated with a policy violation, the Error-value "P2MP Path computation is not allowed" has been added to the existing error code for Error-Type 5 ("Policy violation") as defined in [RFC5440] (see also Section 6.6 of this document):

> 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 of 7. The corresponding P2MP path computation request MUST also be cancelled.

To indicate capability errors associated with the P2MP path computation request, 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 computation 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 of 1. The corresponding P2MP path computation request MUST also be cancelled.
>
> Error-Type=16; Error-value=2: if a PCE receives a P2MP path computation 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 of 2. The corresponding P2MP path computation request MUST also be cancelled.

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

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

3.16. PCEP NO-PATH Indicator

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

One 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 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 for 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 PCE may be configured to enable or disable P2MP path computations.
- 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 ("IGP Extensions for P2MP Capability Advertisement") 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 in [RFC7420] for general PCEP control and monitoring of P2P computations. [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.

The "ietf-pcep" PCEP YANG module is specified in [PCEP-YANG]. The P2MP capability of a PCEP entity or a configured peer can be set using this YANG module. Also, support for P2MP path computation can be learned using this module. The statistics are maintained in the "ietf-pcep-stats" YANG module as specified in [PCEP-YANG]. This YANG module will be required to be augmented to also include the P2MP-related statistics.

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 ("IGP Extensions for P2MP Capability Advertisement") of this document.

The relevant IANA assignment is 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 that specifically help to minimize or negate unauthorized P2MP path computation requests and denial-of-service attacks. These mechanisms include the following:

- Securing the PCEP session requests and responses is RECOMMENDED using TCP security techniques such as the TCP Authentication Option (TCP-AO) [RFC5925] or using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in [RFC7525].

- Authenticating the PCEP requests and responses to ensure that the message is intact and sent from an authorized node using the TCP-AO or TLS is RECOMMENDED.

- Policy control could be provided by explicitly defining which PCCs are allowed to send P2MP path computation requests to the PCE via IP access lists.

PCEP operates over TCP, so it is also important to secure the PCE and PCC against TCP denial-of-service attacks.

As stated in [RFC6952], PCEP implementations SHOULD support the TCP-AO [RFC5925] and not use TCP MD5 because of TCP MD5’s known vulnerabilities and weakness.

6. IANA Considerations

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

6.1. PCEP TLV Type Indicators

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

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

Value	Description	Reference
6	P2MP capable	RFC 8306

6.2. Request Parameter Bit Flags

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

IANA had previously made allocations from the PCEP "RP Object Flag Field" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Bit	Description	Reference
18	Fragmentation (F-bit)	RFC 8306
19	P2MP (N-bit)	RFC 8306
20	ERO-compression (E-bit)	RFC 8306

6.3. Objective Functions

As described in Section 3.6.1, this document defines two objective functions.

IANA had previously made allocations from the PCEP "Objective Function" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Code Point	Name	Reference
7	SPT	RFC 8306
8	MCT	RFC 8306

6.4. METRIC Object-Type Values

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

IANA had previously made allocations from the PCEP "METRIC Object T Field" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Value	Description	Reference
8	P2MP IGP metric	RFC 8306
9	P2MP TE metric	RFC 8306
10	P2MP hop count metric	RFC 8306

6.5. PCEP Objects

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

IANA had previously made the Object-Type allocations from the "PCEP Objects" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Object-Class Value	4
Name	END-POINTS
Object-Type	3: IPv4 4: IPv6 5-15: Unassigned
Reference	RFC 8306

As described in Sections 3.2, 3.11.1, and 3.14, four PCEP Object-Class values and six PCEP Object-Type values have been defined.

IANA had previously made allocations from the "PCEP Objects" subregistry, where RFC 6006 was the reference. IANA has updated the reference to point to this document.

Also, for the following four PCEP objects, codepoint 0 for the Object-Type field is marked "Reserved", as per Erratum ID 4956 for RFC 5440. IANA has updated the reference to point to this document.

Object-Class Value	28
Name	UNREACH-DESTINATION
Object-Type	0: Reserved 1: IPv4 2: IPv6 3-15: Unassigned
Reference	RFC 8306

Object-Class Value	29
Name	SERO
Object-Type	0: Reserved 1: SERO 2-15: Unassigned
Reference	RFC 8306

Object-Class Value	30
Name	SRRO
Object-Type	0: Reserved 1: SRRO 2-15: Unassigned
Reference	RFC 8306

Object-Class Value	31
Name	BNC
Object-Type	0: Reserved 1: Branch node list 2: Non-branch node list 3-15: Unassigned
Reference	RFC 8306

6.6. PCEP-ERROR Objects and Types

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

IANA had previously made allocations from the PCEP "PCEP-ERROR Object Error Types and Values" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Error Type	Meaning	Reference
5	Policy violation	
	Error-value=7: P2MP Path computation is not allowed	RFC 8306
16	P2MP Capability Error	
	Error-value=0: Unassigned	RFC 8306
	Error-value=1: The PCE cannot satisfy the request due to insufficient memory	RFC 8306
	Error-value=2: The PCE is not capable of P2MP computation	RFC 8306
17	P2MP END-POINTS Error	
	Error-value=0: Unassigned	RFC 8306
	Error-value=1: The PCE cannot satisfy the request due to no END-POINTS with leaf type 2	RFC 8306
	Error-value=2: The PCE cannot satisfy the request due to no END-POINTS with leaf type 3	RFC 8306
	Error-value=3: The PCE cannot satisfy the request due to no END-POINTS with leaf type 4	RFC 8306
	Error-value=4: The PCE cannot satisfy the request due to inconsistent END-POINTS	RFC 8306
18	P2MP Fragmentation Error	
	Error-value=0: Unassigned	RFC 8306
	Error-value=1: Fragmented request failure	RFC 8306

6.7. PCEP NO-PATH Indicator

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

IANA had previously made an allocation from the PCEP "NO-PATH-VECTOR TLV Flag Field" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Bit	Description	Reference
24	P2MP Reachability Problem	RFC 8306

6.8. SVEC Object Flag

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

IANA had previously made allocations from the PCEP "SVEC Object Flag Field" subregistry, where RFC 6006 was the reference. IANA has updated the reference as follows to point to this document.

Bit	Description	Reference
19	Partial Path Diverse	RFC 8306
20	Link Direction Diverse	RFC 8306

6.9. OSPF PCE Capability Flag

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

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

Bit	Description	Reference
10	P2MP path computation	RFC 8306

7. References

7.1. Normative References

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

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

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

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

[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, DOI 10.17487/RFC4875, May 2007, <https://www.rfc-editor.org/info/rfc4875>.

[RFC5073] Vasseur, J., Ed., and J. Le Roux, Ed., "IGP Routing Protocol Extensions for Discovery of Traffic Engineering Node Capabilities", RFC 5073, DOI 10.17487/RFC5073, December 2007, <https://www.rfc-editor.org/info/rfc5073>.

[RFC5088] Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R. Zhang, "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5088, DOI 10.17487/RFC5088, January 2008, <https://www.rfc-editor.org/info/rfc5088>.

[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, DOI 10.17487/RFC5089, January 2008, <https://www.rfc-editor.org/info/rfc5089>.

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

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

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, DOI 10.17487/RFC5541, June 2009, <https://www.rfc-editor.org/info/rfc5541>.

[RFC7770] Lindem, A., Ed., Shen, N., Vasseur, JP., Aggarwal, R., and S. Shaffer, "Extensions to OSPF for Advertising Optional Router Capabilities", RFC 7770, DOI 10.17487/RFC7770, February 2016, <https://www.rfc-editor.org/info/rfc7770>.

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

7.2. Informative References

[PCEP-YANG] Dhody, D., Ed., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", Work in Progress, draft-ietf-pce-pcep-yang-05, July 2017.

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

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

[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, DOI 10.17487/RFC5671, October 2009, <https://www.rfc-editor.org/info/rfc5671>.

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

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

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

[RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <https://www.rfc-editor.org/info/rfc6952>.

[RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <https://www.rfc-editor.org/info/rfc7420>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <https://www.rfc-editor.org/info/rfc7525>.

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

Appendix A. Summary of Changes from RFC 6006

- Updated the text to use the term "PCC" instead of "user" while describing the encoding rules in Section 3.10.
- Updated the example in Figure 7 to explicitly include the RP object.
- Corrected the description of the F-bit in the RP object in Section 3.13, as per Erratum ID 3836.
- Corrected the description of the fragmentation procedure for the response in Section 3.13.2, as per Erratum ID 3819.
- Corrected the Error-Type for fragmentation in Section 3.15, as per Erratum ID 3830.
- Updated the references for the OSPF Router Information Link State Advertisement (LSA) [RFC7770] and the PCEP MIB [RFC7420].
- Added current information and references for PCEP YANG [PCEP-YANG] and PCEPS [RFC8253].
- Updated the Security Considerations section to include the TCP-AO and TLS.
- Updated the IANA Considerations section (Section 6.5) to mark codepoint 0 as "Reserved" for the Object-Type defined in this document, as per Erratum ID 4956 for [RFC5440]. IANA references have also been updated to point to this document.

Appendix A.1. RBNF Changes from RFC 6006

- Updates to the RBNF for the request message format, per Erratum ID 4867:
  - Updated the request message to allow for the bundling of multiple path computation requests within a single PCReq message.
  - Added <svec-list> in PCReq messages. This object was missed in [RFC6006].
  - Added the BNC object in PCReq messages. This object is required to support P2MP. The BNC object shares the same format as the IRO, but it only supports IPv4 and IPv6 prefix sub-objects.

  * Updated the <RRO-List> format to also allow the SRRO. This object was missed in [RFC6006].

  * Removed the BANDWIDTH object followed by the RRO from <RRO-List>. The BANDWIDTH object was included twice in RFC 6006 -- once as part of <end-point-path-pair-list> and also as part of <RRO-List>. The latter has been removed, and the RBNF is backward compatible with [RFC5440].

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

- Updates to the RBNF for the reply message format, per Erratum ID 4868:

  * Updated the reply message to allow for bundling of multiple path computation replies within a single PCRep message.

  * Added the UNREACH-DESTINATION object in PCRep messages. This object was missed in [RFC6006].

Acknowledgements

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

Thanks to Deborah Brungard for handling related errata for RFC 6006.

The authors would like to thank Jonathan Hardwick and Adrian Farrel for providing review comments with suggested text for this document.

Thanks to Jonathan Hardwick for being the document shepherd and for providing comments and guidance.

Thanks to Ben Niven-Jenkins for RTGDIR reviews.

Thanks to Roni Even for GENART reviews.

Thanks to Fred Baker for the OPSDIR review.

Thanks to Deborah Brungard for being the responsible AD and guiding the authors.

Thanks to Mirja Kuehlewind, Alvaro Retana, Ben Campbell, Adam Roach, Benoit Claise, Suresh Krishnan, and Eric Rescorla for their IESG review and comments.

Contributors

Fabien Verhaeghe
Thales Communication France
160 boulevard 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: tomonori.takeda@ntt.com

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

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

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

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

Udayasree Palle
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India
Email: udayasreereddy@gmail.com

Authors' Addresses

Quintin Zhao
Huawei Technologies
125 Nagog Technology Park
Acton, MA 01719
United States of America

Email: quintin.zhao@huawei.com

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

Email: dhruv.ietf@gmail.com

Ramanjaneya Reddy Palleti
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: ramanjaneya.palleti@huawei.com

Daniel King
Old Dog Consulting
United Kingdom

Email: daniel@olddog.co.uk