

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: March 30, 2018

H. Ananthakrishnan  
Packet Design  
S. Sivabalan  
Cisco  
C. Barth  
R. Torvi  
Juniper Networks  
I. Minei  
Google, Inc  
E. Crabbe  
Individual Contributor  
D. Dhody  
Huawei Technologies  
September 26, 2017

PCEP Extensions for MPLS-TE LSP Path Protection with stateful PCE  
draft-ananthakrishnan-pce-stateful-path-protection-04

## Abstract

A stateful Path Computation Element (PCE) is capable of computing as well as controlling via Path Computation Element Protocol (PCEP) Multiprotocol Label Switching Traffic Engineering Label Switched Paths (MPLS LSP). Furthermore, it is also possible for a stateful PCE to create, maintain, and delete LSPs. This document describes PCEP extension to associate two or more LSPs to provide end-to-end path protection.

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 30, 2018.

## Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Requirements Language . . . . .	4
2. Terminology . . . . .	4
3. PCEP Extensions . . . . .	4
3.1. Path Protection Association Type . . . . .	5
3.2. Path Protection Association TLV . . . . .	6
4. Operation . . . . .	7
4.1. PCC Initiated LSPs . . . . .	7
4.2. PCE Initiated LSPs . . . . .	7
4.3. State Synchronization . . . . .	8
4.4. Session Termination . . . . .	8
4.5. Error Handling . . . . .	8
5. Other considerations . . . . .	8
6. IANA considerations . . . . .	9
6.1. Association Type . . . . .	9
6.2. PPAG TLV . . . . .	9
6.3. PCEP Errors . . . . .	10
7. Security Considerations . . . . .	10
8. Manageability Considerations . . . . .	11
8.1. Control of Function and Policy . . . . .	11
8.2. Information and Data Models . . . . .	11
8.3. Liveness Detection and Monitoring . . . . .	11
8.4. Verify Correct Operations . . . . .	11
8.5. Requirements On Other Protocols . . . . .	11
8.6. Impact On Network Operations . . . . .	11
9. Acknowledgments . . . . .	11
10. References . . . . .	11
10.1. Normative References . . . . .	12
10.2. Information References . . . . .	13
Authors' Addresses . . . . .	14

## 1. Introduction

[RFC5440] describes PCEP for communication between a Path Computation Client (PCC) and a PCE or between one a pair of PCEs as per [RFC4655]. A PCE computes paths for MPLS-TE LSPs based on various constraints and optimization criteria.

Stateful pce [RFC8231] specifies a set of extensions to PCEP to enable stateful control of paths such as MPLS TE LSPs between and across PCEP sessions in compliance with [RFC4657]. It includes mechanisms to effect LSP state synchronization between PCCs and PCEs, delegation of control of LSPs to PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions and focuses on a model where LSPs are configured on the PCC and control over them is delegated to the PCE. Furthermore, a mechanism to dynamically instantiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE, is specified in [I-D.ietf-pce-pce-initiated-lsp].

Path protection refers to a paradigm in which the working LSP is protected by one or more protection LSP(s). When the working LSP fails, protection LSP(s) is/are activated. When the working LSPs are computed and controlled by the PCE, there is benefit in a mode of operation where protection LSPs are as well.

This document specifies a stateful PCEP extension to associate two or more LSPs for the purpose of setting up path protection. The proposed extension covers the following scenarios:

- o A PCC initiates a protection LSP and retains the control of the LSP. The PCC computes the path itself or makes a request for path computation to a PCE. After the path setup, it reports the information and state of the path to the PCE. This is the passive stateful mode [RFC8051].
- o A PCC initiates a protection LSP and delegates the control of the LSP to a stateful PCE. The PCE may compute the path for the LSP and update the PCC with the information about the path as long as it controls the LSP. This is the active stateful mode [RFC8051].
- o A protection LSP could be initiated by a stateful PCE, which retains the control of the LSP. The PCE is responsible for computing the path of the LSP and updating to the PCC with the information about the path. This is the PCE Initiated mode [I-D.ietf-pce-pce-initiated-lsp].

Note that protection LSP can be established prior to the failure (in which case the LSP is said to be in standby mode) or post failure of

the corresponding working LSP according to the operator choice/policy.

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

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

This document describes a PCEP extension to associate protection LSPs by creating Path Protection Association Group (PPAG) and encoding this association in PCEP messages for stateful PCEP sessions.

### 1.1. Requirements Language

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

## 2. Terminology

The following terminologies are used in this document:

ERO: Explicit Route Object.

LSP: Label Switched Path.

PCC: Path Computation Client.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

PPAG: Path Protection Association Group.

TLV: Type, Length, and Value.

## 3. PCEP Extensions

### 3.1. Path Protection Association Type

LSPs are not associated by listing the other LSPs with which they interact, but rather by making them belong to an association group referred to as "Path Protection Association Group" (PPAG) in this document. All LSPs join a PPAG individually. PPAG is based on the generic Association object used to associate two or more LSPs specified in [I-D.ietf-pce-association-group]. A member of a PPAG can take the role of working or protection LSP. This document defines a new association type called "Path Protection Association Type" of value TBD1. A PPAG can have one working LSP and/or one or more protection LSPs. The source, destination and Tunnel ID (as carried in LSP-IDENTIFIERS TLV [RFC8231], with description as per [RFC3209]) of all LSPs within a PPAG MUST be the same. As per [RFC3209], TE tunnel is used to associate a set of LSPs during reroute or to spread a traffic trunk over multiple paths.

The format of the Association object used for PPAG is specified in [I-D.ietf-pce-association-group] and reproduced in this document for easy reference in Figure 1 and Figure 2.

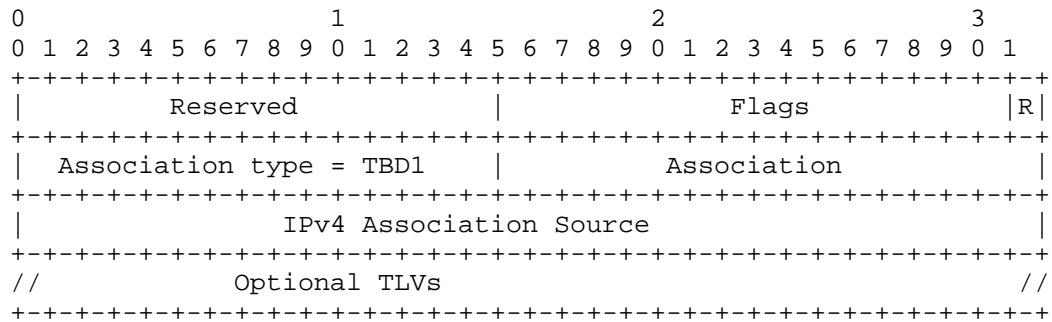


Figure 1: PPAG IPv4 ASSOCIATION Object format

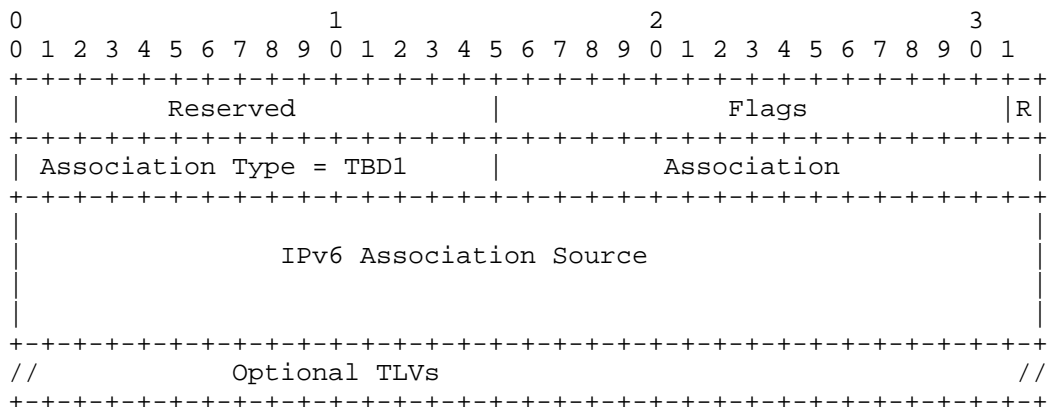


Figure 2: PPAG IPv6 ASSOCIATION Object format

This document defines a new Association type, the Path Protection Association type, value will be assigned by IANA (TBD1).

This Association-Type is dynamic in nature and created by the PCC or PCE for the LSPs belonging to the same TE tunnel (as described in [RFC3209]) originating at the same head node and terminating at the same destination. These associations are conveyed via PCEP messages to the PCEP peer. Operator-configured Association Range SHOULD NOT be set for this association-type and MUST be ignored.

### 3.2. Path Protection Association TLV

The Path Protection Association TLV is an optional TLV for use with the Path Protection Association Object Type. The Path Protection Association TLV MUST NOT be present more than once. If it appears more than once, only the first occurrence is processed and any others MUST be ignored.

The Path Protection Association TLV follows the PCEP TLV format of [RFC5440].

The type (16 bits) of the TLV is to be assigned by IANA. The length field is 16 bit-long and has a fixed value of 4.

The value comprises a single field, the Path Protection Association Flags (32 bits), where each bit represents a flag option.

The format of the Path Protection Association TLV (Figure 3) is as follows:

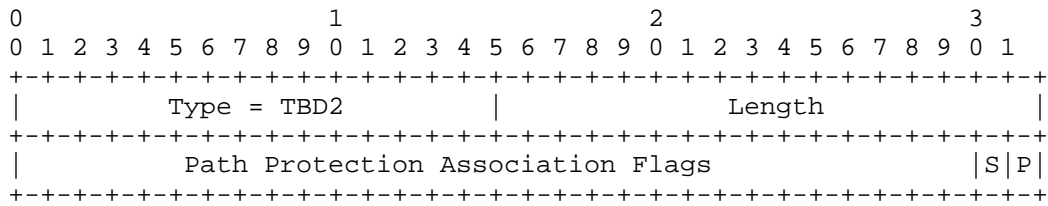


Figure 3: Path Protection Association TLV format

P (PROTECTION-LSP 1 bit) - Indicates whether the LSP associated with the PPAG is working or protection LSP. If this flag is set, the LSP is a protection LSP.

S (STANDBY 1 bit)- When the P flag is set, the S flag indicates whether the protection LSP associated with the PPAG is in standby mode. The S flag is ignored if the P flag is not set.

Unassigned bits are considered reserved. They MUST be set to 0 on transmission and MUST be ignored on receipt. If the Path Protection Association TLV is missing, it means the LSP is the working LSP.

#### 4. Operation

LSPs are associated with other LSPs with which they interact by adding them to a common association group via ASSOCIATION object. All procedures and error-handling for the ASSOCIATION object is as per [I-D.ietf-pce-association-group].

##### 4.1. PCC Initiated LSPs

A PCC can associate a set of LSPs under its control for path protection purpose. Similarly, the PCC can remove one or more LSPs under its control from the corresponding PPAG. In both cases, the PCC must report the change in association to PCE(s) via PCRpt message. A PCC can also delegate the working and protection LSPs to a stateful PCE, where PCE would control the LSPs. The stateful PCE could update the paths and attributes of the LSPs in the association group via PCUpd message. A PCE could also update the association to PCC via PCUpd message. The procedures are described in [I-D.ietf-pce-association-group].

##### 4.2. PCE Initiated LSPs

A PCE can create/update working and protection LSPs independently. As specified in [I-D.ietf-pce-association-group], Association Groups can be created by both PCE and PCC.

Further, a PCE can remove a protection LSP from a PPAG as specified in [I-D.ietf-pce-association-group].

#### 4.3. State Synchronization

During state synchronization, a PCC MUST report all the existing path protection association groups as well as any path protection flags to PCE(s). Following the state synchronization, the PCE would remove all stale information as per [I-D.ietf-pce-association-group].

#### 4.4. Session Termination

As per [I-D.ietf-pce-association-group] the association information is cleared along with the LSP state information. When a PCEP session is terminated, after expiry of State Timeout Interval at PCC, the LSP state associated with that PCEP session is reverted to operator-defined default parameters or behaviors as per [RFC8231]. Same procedure is also followed for the association information. On session termination at the PCE, when the LSP state reported by PCC is cleared, the association information is also cleared. Where there are no LSPs in a association group, the association is considered to be deleted..

#### 4.5. Error Handling

All LSPs (working or protection) within a PPAG MUST belong to the same TE Tunnel (as described in [RFC3209]) and have the same source and destination. If a PCE attempts to add an LSP to a PPAG and the Tunnel ID (as carried in LSP-IDENTIFIERS TLV [RFC8231], with description as per [RFC3209]) or source or destination of the LSP is different from the LSP(s) in the PPAG, the PCC MUST send PCErr with Error-Type= TBD (Association Error) [I-D.ietf-pce-association-group] and Error-Value = TBD3 (Tunnel ID or End points mismatch for Path Protection Association).

There MUST be only one working LSP within a PPAG. If a PCEP Speaker attempts to add another working LSP, the PCEP peer MUST send PCErr with Error-Type=TBD (Association Error) [I-D.ietf-pce-association-group] and Error-Value = TBD4 (Attempt to add another working LSP for Path Protection Association).

#### 5. Other considerations

The diversity requirement for a group of LSPs is handled via another association type called "Disjointness Association", as described in [I-D.ietf-pce-association-diversity]. The diversity requirements for the the protection LSP are also handled by including both ASSOCIATION object for the group of LSPs.



## 6. IANA considerations

### 6.1. Association Type

This document defines a new association type, originally defined in [I-D.ietf-pce-association-group], for path protection. IANA is requested to make the assignment of a new value for the sub-registry "ASSOCIATION Type Field" (request to be created in [I-D.ietf-pce-association-group]), as follows:

Association Type Value	Association Name	Reference
TBD1	Path Protection Association	This document

### 6.2. PPAG TLV

This document defines a new TLV for carrying additional information of LSPs within a path protection association group. IANA is requested to make the assignment of a new value for the existing "PCEP TLV Type Indicators" registry as follows:

TLV Type Value	TLV Name	Reference
TBD2	Path Protection Association Group TLV	This document

This document requests that a new sub-registry, named "Path protection Association Group TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the Path Protection Association Group TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

Each bit should be tracked with the following qualities:

- o Bit number (count from 0 as the most significant bit)
- o Name flag
- o Reference

Bit Number	Name	Reference
31	P - PROTECTION-LSP	This document
30	S - STANDBY	This document

Table 1: PPAG TLV

### 6.3. PCEP Errors

This document defines new Error-Type and Error-Value related to path protection association. IANA is requested to allocate new error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

Error-Type	Meaning	Reference
TBD	Association error Error-value=TBD3: Tunnel ID or End points mismatch for Path Protection Association	[I-D.ietf-pce-association-group] This document
	Error-value=TBD4: Attempt to add another working LSP for Path Protection Association	This document

## 7. Security Considerations

The security considerations described in [RFC8231], [I-D.ietf-pce-pce-initiated-lsp], and [RFC5440] apply to the extensions described in this document as well. Additional considerations related to associations where a malicious PCEP speaker could be spoofed and could be used as an attack vector by creating associations is described in [I-D.ietf-pce-association-group]. Thus securing the PCEP session using Transport Layer Security (TLS) [I-D.ietf-pce-pceps], as per the recommendations and best current practices in [RFC7525], is RECOMMENDED.

## 8. Manageability Considerations

### 8.1. Control of Function and Policy

Mechanisms defined in this document do not imply any control or policy requirements in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

### 8.2. Information and Data Models

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

The PCEP YANG module [I-D.ietf-pce-pcep-yang] supports associations.

### 8.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

### 8.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

### 8.5. Requirements On Other Protocols

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

### 8.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

## 9. Acknowledgments

We would like to thank Jeff Tantsura and Xian Zhang for their contributions to this document.

## 10. References

## 10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [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>>.
- [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>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [I-D.ietf-pce-pce-initiated-lsp] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-10 (work in progress), June 2017.
- [I-D.ietf-pce-association-group] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "PCEP Extensions for Establishing Relationships Between Sets of LSPs", draft-ietf-pce-association-group-04 (work in progress), August 2017.

## 10.2. Information References

- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <<https://www.rfc-editor.org/info/rfc4657>>.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <<https://www.rfc-editor.org/info/rfc7420>>.
- [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>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [I-D.ietf-pce-pceps]  
Lopez, D., Dios, O., Wu, Q., and D. Dhody, "Secure Transport for PCEP", draft-ietf-pce-pceps-18 (work in progress), September 2017.
- [I-D.ietf-pce-pcep-yang]  
Dhody, D., Hardwick, J., Beeram, V., and j. jeffrant@gmail.com, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-05 (work in progress), June 2017.
- [I-D.ietf-pce-association-diversity]  
Litkowski, S., Sivabalan, S., Barth, C., and D. Dhody, "Path Computation Element communication Protocol extension for signaling LSP diversity constraint", draft-ietf-pce-association-diversity-02 (work in progress), September 2017.

## Authors' Addresses

Hariharan Ananthakrishnan  
Packet Design  
1 South Almaden Blvd, #1150,  
San Jose, CA, 95113  
USA

EMail: hari@packetdesign.com

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

EMail: msiva@cisco.com

Colby Barth  
Juniper Networks  
1194 N Mathilda Ave,  
Sunnyvale, CA, 94086  
USA

EMail: cbarth@juniper.net

Raveendra Torvi  
Juniper Networks  
1194 N Mathilda Ave,  
Sunnyvale, CA, 94086  
USA

EMail: rtorvi@juniper.net

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

EMail: inaminei@google.com

Edward Crabbe  
Individual Contributor

EMail: [edward.crabbe@gmail.com](mailto:edward.crabbe@gmail.com)

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

EMail: [dhruv.ietf@gmail.com](mailto:dhruv.ietf@gmail.com)

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: August 31, 2018

H. Ananthakrishnan  
Packet Design  
S. Sivabalan  
Cisco  
C. Barth  
R. Torvi  
Juniper Networks  
I. Minei  
Google, Inc  
E. Crabbe  
Individual Contributor  
D. Dhody  
Huawei Technologies  
February 27, 2018

PCEP Extensions for MPLS-TE LSP Path Protection with stateful PCE  
draft-ananthakrishnan-pce-stateful-path-protection-05

## Abstract

A stateful Path Computation Element (PCE) is capable of computing as well as controlling via Path Computation Element Protocol (PCEP) Multiprotocol Label Switching Traffic Engineering Label Switched Paths (MPLS LSP). Furthermore, it is also possible for a stateful PCE to create, maintain, and delete LSPs. This document describes PCEP extension to associate two or more LSPs to provide end-to-end path protection.

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 31, 2018.



## Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Requirements Language . . . . .	4
2. Terminology . . . . .	4
3. PCEP Extensions . . . . .	5
3.1. Path Protection Association Type . . . . .	5
3.2. Path Protection Association TLV . . . . .	5
4. Operation . . . . .	6
4.1. State Synchronization . . . . .	6
4.2. PCC Initiated LSPs . . . . .	6
4.3. PCE Initiated LSPs . . . . .	7
4.4. Session Termination . . . . .	7
4.5. Error Handling . . . . .	7
5. Other considerations . . . . .	8
6. IANA considerations . . . . .	8
6.1. Association Type . . . . .	8
6.2. PPAG TLV . . . . .	8
6.3. PCEP Errors . . . . .	9
7. Security Considerations . . . . .	10
8. Manageability Considerations . . . . .	10
8.1. Control of Function and Policy . . . . .	10
8.2. Information and Data Models . . . . .	10
8.3. Liveness Detection and Monitoring . . . . .	10
8.4. Verify Correct Operations . . . . .	10
8.5. Requirements On Other Protocols . . . . .	10
8.6. Impact On Network Operations . . . . .	11
9. Acknowledgments . . . . .	11
10. References . . . . .	11
10.1. Normative References . . . . .	11
10.2. Information References . . . . .	12
Authors' Addresses . . . . .	13

## 1. Introduction

[RFC5440] describes PCEP for communication between a Path Computation Client (PCC) and a PCE or between one a pair of PCEs as per [RFC4655]. A PCE computes paths for MPLS-TE LSPs based on various constraints and optimization criteria.

Stateful pce [RFC8231] specifies a set of extensions to PCEP to enable stateful control of paths such as MPLS TE LSPs between and across PCEP sessions in compliance with [RFC4657]. It includes mechanisms to effect LSP state synchronization between PCCs and PCEs, delegation of control of LSPs to PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions and focuses on a model where LSPs are configured on the PCC and control over them is delegated to the PCE. Furthermore, a mechanism to dynamically instantiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE, is specified in [RFC8281].

Path protection [RFC4427] refers to a paradigm in which the working LSP is protected by one or more protection LSP(s). When the working LSP fails, protection LSP(s) is/are activated. When the working LSPs are computed and controlled by the PCE, there is benefit in a mode of operation where protection LSPs are as well.

This document specifies a stateful PCEP extension to associate two or more LSPs for the purpose of setting up path protection. The proposed extension covers the following scenarios:

- o A PCC initiates a protection LSP and retains the control of the LSP. The PCC computes the path itself or makes a request for path computation to a PCE. After the path setup, it reports the information and state of the path to the PCE. This includes the association group identifying the working and protection LSPs. This is the passive stateful mode [RFC8051].
- o A PCC initiates a protection LSP and delegates the control of the LSP to a stateful PCE. During delegation the association group identifying the working and protection LSPs is included. The PCE computes the path for the protection LSP and update the PCC with the information about the path as long as it controls the LSP. This is the active stateful mode [RFC8051].
- o A protection LSP could be initiated by a stateful PCE, which retains the control of the LSP. The PCE is responsible for computing the path of the LSP and updating to the PCC with the information about the path. This is the PCE Initiated mode [RFC8281].

Note that protection LSP can be established (signaled) prior to the failure (in which case the LSP is said to be in standby mode [RFC4427]) or post failure of the corresponding working LSP according to the operator choice/policy.

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

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

This document describes a PCEP extension to associate protection LSPs by creating Path Protection Association Group (PPAG) and encoding this association in PCEP messages for stateful PCEP sessions.

### 1.1. Requirements Language

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

## 2. Terminology

The following terminologies are used in this document:

ERO: Explicit Route Object.

LSP: Label Switched Path.

PCC: Path Computation Client.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

PPAG: Path Protection Association Group.

TLV: Type, Length, and Value.

### 3. PCEP Extensions

#### 3.1. Path Protection Association Type

LSPs are not associated by listing the other LSPs with which they interact, but rather by making them belong to an association group referred to as "Path Protection Association Group" (PPAG) in this document. All LSPs join a PPAG individually. PPAG is based on the generic Association object used to associate two or more LSPs specified in [I-D.ietf-pce-association-group]. A member of a PPAG can take the role of working or protection LSP. This document defines a new association type called "Path Protection Association Type" of value TBD1. A PPAG can have one working LSP and/or one or more protection LSPs. The source, destination and Tunnel ID (as carried in LSP-IDENTIFIERS TLV [RFC8231], with description as per [RFC3209]) of all LSPs within a PPAG MUST be the same. As per [RFC3209], TE tunnel is used to associate a set of LSPs during reroute or to spread a traffic trunk over multiple paths.

The format of the Association object used for PPAG is specified in [I-D.ietf-pce-association-group].

This document defines a new Association type, the Path Protection Association type, value will be assigned by IANA (TBD1).

This Association-Type is dynamic in nature and created by the PCC or PCE for the LSPs belonging to the same TE tunnel (as described in [RFC3209]) originating at the same head node and terminating at the same destination. These associations are conveyed via PCEP messages to the PCEP peer. Operator-configured Association Range MUST NOT be set for this association-type and MUST be ignored.

#### 3.2. Path Protection Association TLV

The Path Protection Association TLV is an optional TLV for use with the Path Protection Association Object Type. The Path Protection Association TLV MUST NOT be present more than once. If it appears more than once, only the first occurrence is processed and any others MUST be ignored.

The Path Protection Association TLV follows the PCEP TLV format of [RFC5440].

The type (16 bits) of the TLV is to be assigned by IANA. The length field is 16 bit-long and has a fixed value of 4.

The value comprises a single field, the Path Protection Association Flags (32 bits), where each bit represents a flag option.

The format of the Path Protection Association TLV (Figure 1) is as follows:

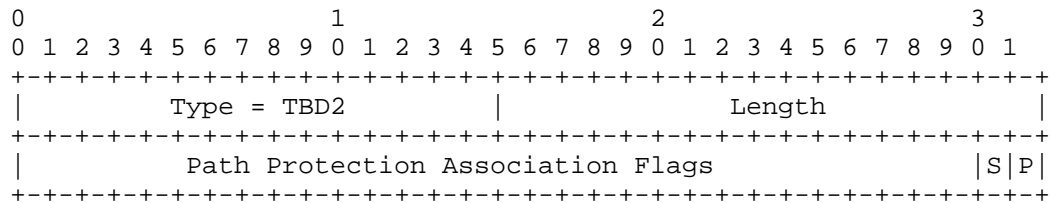


Figure 1: Path Protection Association TLV format

P (PROTECTION-LSP 1 bit) - Indicates whether the LSP associated with the PPAG is working or protection LSP. If this flag is set, the LSP is a protection LSP.

S (STANDBY 1 bit)- When the P flag is set, the S flag indicates whether the protection LSP associated with the PPAG is in standby mode. The S flag is ignored if the P flag is not set.

Unassigned bits are considered reserved. They MUST be set to 0 on transmission and MUST be ignored on receipt.

If the TLV is missing, it is considered that the LSP is the working LSP (i.e. P bit is unset).

#### 4. Operation

LSPs are associated with other LSPs with which they interact by adding them to a common association group via ASSOCIATION object. All procedures and error-handling for the ASSOCIATION object is as per [I-D.ietf-pce-association-group].

##### 4.1. State Synchronization

During state synchronization, a PCC MUST report all the existing path protection association groups as well as any path protection flags to PCE(s) as per [I-D.ietf-pce-association-group].

##### 4.2. PCC Initiated LSPs

A PCC can associate a set of LSPs under its control for path protection purpose. Similarly, the PCC can remove one or more LSPs under its control from the corresponding PPAG. In both cases, the PCC must report the change in association to PCE(s) via PCRpt message. A PCC can also delegate the working and protection LSPs to

a stateful PCE, where PCE would control the LSPs. The stateful PCE could update the paths and attributes of the LSPs in the association group via PCUpd message. A PCE could also update the association to PCC via PCUpd message. These procedures are described in [I-D.ietf-pce-association-group].

#### 4.3. PCE Initiated LSPs

A PCE can create/update working and protection LSPs independently. As specified in [I-D.ietf-pce-association-group], Association Groups can be created by both PCE and PCC. Further, a PCE can remove a protection LSP from a PPAG as specified in [I-D.ietf-pce-association-group]. The PCE uses PCUpd or PCInitiate message to communicate the association information to the PCC.

#### 4.4. Session Termination

As per [I-D.ietf-pce-association-group] the association information is cleared along with the LSP state information. When a PCEP session is terminated, after expiry of State Timeout Interval at PCC, the LSP state associated with that PCEP session is reverted to operator-defined default parameters or behaviors as per [RFC8231]. Same procedure is also followed for the association information. On session termination at the PCE, when the LSP state reported by PCC is cleared, the association information is also cleared as per [I-D.ietf-pce-association-group]. Where there are no LSPs in a association group, the association is considered to be deleted..

#### 4.5. Error Handling

All LSPs (working or protection) within a PPAG MUST belong to the same TE Tunnel (as described in [RFC3209]) and have the same source and destination. If a PCEP speaker attempts to add an LSP to a PPAG and the Tunnel ID (as carried in LSP-IDENTIFIERS TLV [RFC8231], with description as per [RFC3209]) or source or destination of the LSP is different from the LSP(s) in the PPAG, the PCC MUST send PCErr with Error-Type= 29 (Early allocation by IANA) (Association Error) [I-D.ietf-pce-association-group] and Error-Value = TBD3 (Tunnel ID or End points mismatch for Path Protection Association).

There MUST be only one working LSP within a PPAG. If a PCEP Speaker attempts to add another working LSP, the PCEP peer MUST send PCErr with Error-Type=29 (Early allocation by IANA) (Association Error) [I-D.ietf-pce-association-group] and Error-Value = TBD4 (Attempt to add another working LSP for Path Protection Association).

## 5. Other considerations

The working and protection LSPs are typically resource disjoint (e.g., node, srlg disjoint). This ensures that a single failure will not affect both the working and protection LSPs. The disjoint requirement for a group of LSPs is handled via another association type called "Disjointness Association", as described in [I-D.ietf-pce-association-diversity]. The diversity requirements for the the protection LSP are also handled by including both ASSOCIATION object identifying both the protection association group and disjoint association group for the group of LSPs.

## 6. IANA considerations

### 6.1. Association Type

This document defines a new association type, originally defined in [I-D.ietf-pce-association-group], for path protection. IANA is requested to make the assignment of a new value for the sub-registry "ASSOCIATION Type Field" (request to be created in [I-D.ietf-pce-association-group]), as follows:

Association Type Value	Association Name	Reference
TBD1	Path Protection Association	This document

### 6.2. PPAG TLV

This document defines a new TLV for carrying additional information of LSPs within a path protection association group. IANA is requested to make the assignment of a new value for the existing "PCEP TLV Type Indicators" registry as follows:

TLV Type Value	TLV Name	Reference
TBD2	Path Protection Association Group TLV	This document

This document requests that a new sub-registry, named "Path protection Association Group TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage

the Flag field in the Path Protection Association Group TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

Each bit should be tracked with the following qualities:

- o Bit number (count from 0 as the most significant bit)
- o Name flag
- o Reference

Bit Number	Name	Reference
31	P - PROTECTION-LSP	This document
30	S - STANDBY	This document

Table 1: PPAG TLV

### 6.3. PCEP Errors

This document defines new Error-Type and Error-Value related to path protection association. IANA is requested to allocate new error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

Error-Type	Meaning	Reference
29	Association error Error-value=TBD3: Tunnel ID or End points mismatch for Path Protection Association	[I-D.ietf-pce-association-group] This document
	Error-value=TBD4: Attempt to add another working LSP for Path Protection Association	This document



## 7. Security Considerations

The security considerations described in [RFC8231], [RFC8281], and [RFC5440] apply to the extensions described in this document as well. Additional considerations related to associations where a malicious PCEP speaker could be spoofed and could be used as an attack vector by creating associations is described in [I-D.ietf-pce-association-group]. Thus securing the PCEP session using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in [RFC7525], is RECOMMENDED.

## 8. Manageability Considerations

### 8.1. Control of Function and Policy

Mechanisms defined in this document do not imply any control or policy requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

### 8.2. Information and Data Models

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

The PCEP YANG module [I-D.ietf-pce-pcep-yang] supports associations.

### 8.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

### 8.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

### 8.5. Requirements On Other Protocols

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

## 8.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

## 9. Acknowledgments

We would like to thank Jeff Tantsura and Xian Zhang for their contributions to this document.

## 10. References

### 10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [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>>.
- [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>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.

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

[I-D.ietf-pce-association-group]

Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "PCEP Extensions for Establishing Relationships Between Sets of LSPs", draft-ietf-pce-association-group-04 (work in progress), August 2017.

## 10.2. Information References

[RFC4427] Mannie, E., Ed. and D. Papadimitriou, Ed., "Recovery (Protection and Restoration) Terminology for Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4427, DOI 10.17487/RFC4427, March 2006, <<https://www.rfc-editor.org/info/rfc4427>>.

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

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

[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>>.

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

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

[I-D.ietf-pce-pcep-yang]  
Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A  
YANG Data Model for Path Computation Element  
Communications Protocol (PCEP)", draft-ietf-pce-pcep-  
yang-06 (work in progress), January 2018.

[I-D.ietf-pce-association-diversity]  
Litkowski, S., Sivabalan, S., Barth, C., and D. Dhody,  
"Path Computation Element communication Protocol extension  
for signaling LSP diversity constraint", draft-ietf-pce-  
association-diversity-03 (work in progress), February  
2018.

#### Authors' Addresses

Hariharan Ananthakrishnan  
Packet Design  
1 South Almaden Blvd, #1150,  
San Jose, CA, 95113  
USA

EMail: [hari@packetdesign.com](mailto:hari@packetdesign.com)

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

EMail: [msiva@cisco.com](mailto:msiva@cisco.com)

Colby Barth  
Juniper Networks  
1194 N Mathilda Ave,  
Sunnyvale, CA, 94086  
USA

EMail: [cbarth@juniper.net](mailto:cbarth@juniper.net)

Raveendra Torvi  
Juniper Networks  
1194 N Mathilda Ave,  
Sunnyvale, CA, 94086  
USA

EMail: rtorvi@juniper.net

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

EMail: inaminei@google.com

Edward Crabbe  
Individual Contributor

EMail: edward.crabbe@gmail.com

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

EMail: dhruv.ietf@gmail.com

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 6, 2018

C. Barth  
Juniper Networks  
R. Gandhi  
Cisco Systems, Inc.  
B. Wen  
Comcast  
October 3, 2017

PCEP Extensions for  
Associated Bidirectional Label Switched Paths (LSPs)  
draft-barth-pce-association-bidir-03

## Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests. The Stateful PCE extensions allow stateful control of Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs) using PCEP.

This document defines PCEP extensions for grouping two reverse unidirectional MPLS TE LSPs into an Associated Bidirectional LSP when using a Stateful PCE for both PCE-Initiated and PCC-Initiated LSPs as well as when using a Stateless PCE.

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

## Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Conventions Used in This Document . . . . .	4
2.1. Key Word Definitions . . . . .	4
2.2. Terminology . . . . .	4
3. Overview . . . . .	4
3.1. Single-sided Initiation . . . . .	5
3.2. Double-sided Initiation . . . . .	5
3.3. Co-routed Associated Bidirectional LSP . . . . .	6
4. Protocol Extensions . . . . .	6
4.1. Association Object . . . . .	6
4.2. Bidirectional LSP Association Group TLV . . . . .	7
5. PCEP Procedure . . . . .	8
5.1. PCE Initiated LSPs . . . . .	8
5.2. PCC Initiated LSPs . . . . .	8
5.3. Stateless PCE . . . . .	8
5.4. State Synchronization . . . . .	9
5.5. Error Handling . . . . .	9
6. Security Considerations . . . . .	9
7. Manageability Considerations . . . . .	9
7.1. Control of Function and Policy . . . . .	9
7.2. Information and Data Models . . . . .	10
7.3. Liveness Detection and Monitoring . . . . .	10
7.4. Verify Correct Operations . . . . .	10
7.5. Requirements On Other Protocols . . . . .	10
7.6. Impact On Network Operations . . . . .	10
8. IANA Considerations . . . . .	10
8.1. Association Types . . . . .	10
8.2. Bidirectional LSP Association Group TLV . . . . .	10
8.2.1. Flag Fields in Bidirectional LSP Association Group TLV . . . . .	11
8.3. PCEP Errors . . . . .	11
9. References . . . . .	12
9.1. Normative References . . . . .	12
9.2. Informative References . . . . .	13
Acknowledgments . . . . .	14

Authors' Addresses . . . . . 14

## 1. Introduction

[RFC5440] describes the Path Computation Element Protocol (PCEP) as a communication mechanism between a Path Computation Client (PCC) and a Path Control Element (PCE), or between PCE and PCC, that enables computation of Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs).

[RFC8231] specifies extensions to PCEP to enable stateful control of MPLS TE LSPs. It describes two modes of operation - Passive Stateful PCE and Active Stateful PCE. In [RFC8231], the focus is on Active Stateful PCE where LSPs are provisioned on the PCC and control over them is delegated to a PCE. Further, [I-D.ietf-pce-pce-initiated-lsp] describes the setup, maintenance and teardown of PCE-Initiated LSPs for the Stateful PCE model.

[I-D.ietf-pce-association] introduces a generic mechanism to create a grouping of LSPs which can then be used to define associations between a set of LSPs and/or a set of attributes, for example primary and secondary LSP associations, and is equally applicable to the active and passive modes of a Stateful PCE [RFC8231] or a stateless PCE [RFC5440].

The MPLS Transport Profile (MPLS-TP) requirements document [RFC5654] specifies that MPLS-TP MUST support associated bidirectional point-to-point LSPs. [RFC7551] specifies RSVP signaling extensions for binding two reverse unidirectional LSPs [RFC3209] into an associated bidirectional LSP. The fast reroute (FRR) procedures for associated bidirectional LSPs are described in [I-D.ietf-teas-assoc-corouted-bidir-frr].

This document specifies PCEP extensions for grouping two reverse unidirectional MPLS-TE LSPs into an Associated Bidirectional LSP for both single-sided and double-sided initiation cases when using a Stateful (both active and passive modes) or Stateless PCE. The PCEP extensions cover the following cases:

- o A PCE initiates the forward and/ or reverse LSP of a single-sided or double-sided bidirectional LSP on a PCC and retains the control of the LSP. The PCE computes the path of the LSP and updates the PCC with the information about the path.
- o A PCC initiates the forward and/ or reverse LSP of a single-sided or double-sided bidirectional LSP and retains the control of the LSP. The PCC computes the path of the LSP and reports the PCE



with the information about the path (as long as it controls the LSP, as in passive Stateful PCE mode).

- o A PCC initiates the forward and/ or reverse LSP of a single-sided or double-sided bidirectional LSP and delegates the control of the LSP to a Stateful PCE. The PCE may compute the path of the LSP and update the PCC with the information about the path (as long as it controls the LSP, as in active Stateful PCE mode).
- o A PCC requests co-routed or non co-routed paths for forward and reverse LSPs of a bidirectional LSP from a Stateless PCE.

## 2. Conventions Used in This Document

### 2.1. Key Word Definitions

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

### 2.2. Terminology

The reader is assumed to be familiar with the terminology defined in [RFC5440], [RFC7551], [RFC8231], and [I-D.ietf-pce-association].

## 3. Overview

As shown in Figure 1, two reverse unidirectional LSPs can be grouped to form an associated bidirectional LSP. There are two methods of initiating the bidirectional LSP association, single-sided and double-sided as described in the following sections.

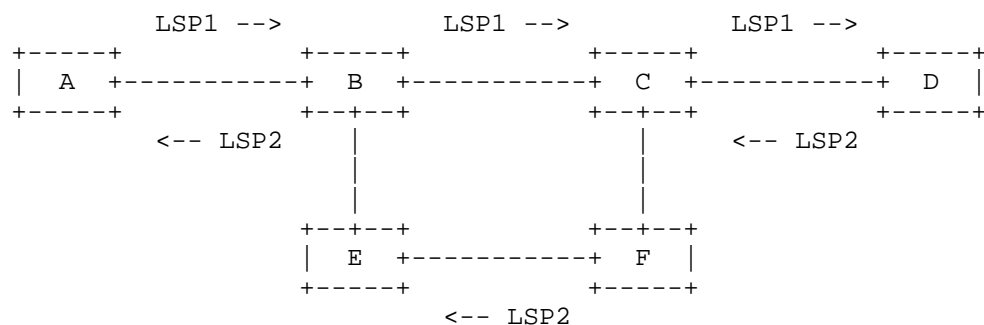


Figure 1: Example of Associated Bidirectional LSP

### 3.1. Single-sided Initiation

As specified in [RFC7551], in the single-sided case, the bidirectional tunnel is provisioned only on one endpoint node (PCC) of the tunnel. Both forward and reverse LSPs of this tunnel are initiated with the Association Type set to "Single-sided Bidirectional LSP Association" on the originating endpoint node. The forward and reverse LSPs are identified in the Bidirectional LSP Association Group TLV of their PCEP Association Objects.

The originating endpoint node signals the properties for the reverse LSP in the RSVP REVERSE\_LSP Object [RFC7551] of the forward LSP Path message. The remote endpoint then creates the corresponding reverse tunnel and signals the reverse LSP in response to the received RSVP Path message.

The two unidirectional reverse LSPs on the originating endpoint node are grouped together using the PCEP Association Object and on the remote endpoint node by the RSVP signaled Association Object.

As shown in Figure 1, the forward tunnel and both the forward LSP LSP1 and the reverse LSP LSP2 are initiated on the originating endpoint node A, either by the PCE or the PCC. The creation of reverse tunnel and reverse LSP2 on the remote endpoint node D is triggered by the RSVP signaled LSP1.

As specified in [I-D.ietf-teas-assoc-corouted-bidir-frr], for fast-reroute bypass tunnel assignment, the LSP starting from the originating node is identified as the forward LSP of the single-sided initiated bidirectional LSP.

### 3.2. Double-sided Initiation

As specified in [RFC7551], in the double-sided case, the bidirectional tunnel is provisioned on both endpoint nodes (PCCs) of the tunnel. The forward and reverse LSPs of this tunnel are initiated with the Association Type set to "Double-sided Bidirectional LSP Association" on both endpoint nodes. The forward and reverse LSPs are identified in the Bidirectional LSP Association Group TLV of their Association Objects.

The two reverse unidirectional LSPs on both the endpoint nodes are grouped together by using the PCEP Association Object.

As shown in Figure 1, the forward tunnel and LSP1 are initiated on the endpoint node A and the reverse tunnel and LSP2 are initiated on the endpoint node D, either by the PCE or the PCCs.

As specified in [I-D.ietf-teas-assoc-corouted-bidir-frr], for fast-reroute bypass tunnel assignment, the LSP with the higher Source Address [RFC3209] is identified as the forward LSP of the double-sided initiated bidirectional LSP.

### 3.3. Co-routed Associated Bidirectional LSP

In both single-sided and double-sided initiation cases, forward and reverse LSPs may be co-routed as shown in Figure 2, where both forward and reverse LSPs follow the same congruent path in the forward and reverse directions, respectively.

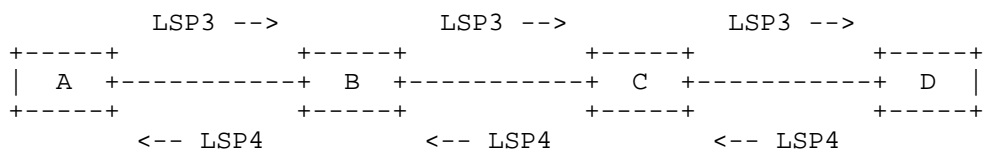


Figure 2: Example of Co-routed Associated Bidirectional LSP

## 4. Protocol Extensions

### 4.1. Association Object

As per [I-D.ietf-pce-association], LSPs are associated by adding them to a common association group. This document defines two new Bidirectional LSP Association Groups to be used by the associated bidirectional LSPs. A member of the Bidirectional LSP Association Group can take the role of a forward or reverse LSP and follows the following rules:

- o An LSP can not be part of more than one Bidirectional LSP Association Group.
- o The Tunnel (as defined in [RFC3209]) of forward and reverse LSPs of the single-sided bidirectional association MUST be the same.

This document defines two new Association Types for the Association Object as follows:

- o Association Type (TBD1) = Single-sided Bidirectional LSP Association Group
- o Association Type (TBD2) = Double-sided Bidirectional LSP Association Group

These Association Types are operator-configured associations in nature and statically created by the operator on the PCEP peers. The LSP belonging to these associations is conveyed via PCEP messages to the PCEP peer. Operator-configured Association Range TLV [I-D.ietf-pce-association] SHOULD NOT be sent for these Association Types, and MUST be ignored, so that the entire range of association ID can be used for them.

The Association ID, Association Source, optional Global Association Source and optional Extended Association ID in the Bidirectional LSP Association Group Object are also operator-configured and populated using the procedures defined in [RFC7551].

#### 4.2. Bidirectional LSP Association Group TLV

The Bidirectional LSP Association Group TLV is an optional TLV for use with the Single-sided and Double-sided Bidirectional LSP Association Group Object Types.

- o The Bidirectional LSP Association Group TLV follows the PCEP TLV format from [RFC5440].
- o The Type (16 bits) of the TLV is TBD3, to be assigned by IANA.
- o The Length is 4 Bytes.
- o The value comprises of a single field, the Bidirectional LSP Association Flags (32 bits), where each bit represents a flag option.
- o If the Bidirectional LSP Association Group TLV is missing, it means the LSP is the forward LSP.
- o The Bidirectional LSP Association Group TLV MUST NOT be present more than once. If it appears more than once, only the first occurrence is processed and any others MUST be ignored.

The format of the Bidirectional LSP Association Group TLV is shown in Figure 3:

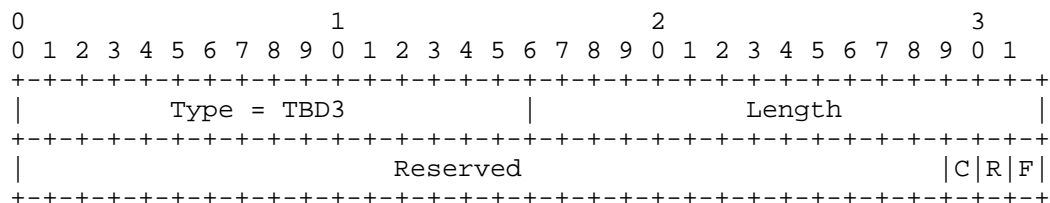


Figure 3: Bidirectional LSP Association Group TLV format

Bidirectional LSP Association Flags are defined as following.

F (Forward LSP, 1 bit) - Indicates whether the LSP associated is the forward LSP of the bidirectional LSP. If this flag is set, the LSP is a forward LSP.

R (Reverse LSP, 1 bit) - Indicates whether the LSP associated is the reverse LSP of the bidirectional LSP. If this flag is set, the LSP is a reverse LSP.

C (Co-routed LSP, 1 bit) - Indicates whether the bidirectional LSP is co-routed. This flag MUST be set for both the forward and reverse LSPs of a co-routed bidirectional LSP.

The C flag is used by the PCE (for both Stateful and Stateless) to compute bidirectional paths of the forward and reverse LSPs.

The Reserved flags MUST be set to 0 when sent and MUST be ignored when received.

## 5. PCEP Procedure

### 5.1. PCE Initiated LSPs

As specified in [I-D.ietf-pce-association], Association Groups can be created by both Stateful PCE and PCC.

A Stateful PCE can create and update the forward and reverse LSPs independently for both Single-sided and Double-sided bidirectional LSP association groups. The establishment and removal of the association relationship can be done on a per LSP basis. A PCE can create and update the association of the LSP on a PCC via PCInitiate and PCUpd messages, respectively, using the procedures described in [I-D.ietf-pce-association].

### 5.2. PCC Initiated LSPs

A PCC can associate or remove an LSP under its control from the bidirectional LSP association group. The PCC MUST report the change in LSP association to Stateful PCE via PCRpt message.

### 5.3. Stateless PCE

For a stateless PCE, it might be useful to associate a path computation request to an association group, thus enabling it to

associate a common set of configuration parameters or behaviors with the request. A PCC can request co-routed or non co-routed forward and reverse direction paths from a stateless PCE for the bidirectional LSP association group.

#### 5.4. State Synchronization

During state synchronization, a PCC MUST report all the existing bidirectional LSP association groups to the Stateful PCE. After the state synchronization, the PCE MUST remove all stale bidirectional associations.

#### 5.5. Error Handling

The LSPs (forward or reverse) in a single-sided bidirectional LSP association group MUST belong to the same TE Tunnel (as defined in [RFC3209]). If a PCE attempts to add an LSP in a single-sided bidirectional LSP association group for a different Tunnel, the PCC MUST send PCErr with Error-Type = TBD4 (Bidirectional LSP Association Error) and Error-Value = 1 (Tunnel mismatch). Similarly, if a PCC attempts to add an LSP to a single-sided bidirectional LSP association group at PCE not complying to this rule, the PCE MUST send this PCErr.

### 6. Security Considerations

The security considerations described in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp] apply to the extensions defined in this document as well.

Two new Association Types for the Association Object, Double-sided Bidirectional LSP Association Group and Single-sided Associated Bidirectional LSP Group are introduced in this document. Additional security considerations related to LSP associations due to a malicious PCEP speaker is described in [I-D.ietf-pce-association] and apply to these Association Types. Thus, securing the PCEP session using Transport Layer Security (TLS) [I-D.ietf-pce-pceps] is recommended.

### 7. Manageability Considerations

#### 7.1. Control of Function and Policy

The mechanisms defined in this document do not imply any control or policy requirements in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

## 7.2. Information and Data Models

[RFC7420] describes the PCEP MIB, there are no new MIB Objects defined for LSP associations.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] supports LSP associations.

## 7.3. Liveness Detection and Monitoring

The mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

## 7.4. Verify Correct Operations

The mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

## 7.5. Requirements On Other Protocols

The mechanisms defined in this document do not add any new requirements on other protocols.

## 7.6. Impact On Network Operations

The mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

## 8. IANA Considerations

### 8.1. Association Types

This document adds new Association Types for the Association Object defined [I-D.ietf-pce-association]. IANA is requested to make the assignment of values for the sub-registry "ASSOCIATION Type Field" (to be created in [I-D.ietf-pce-association]), as follows:

Value Name	Reference
-----	
TBD1 Single-sided Bidirectional LSP Association Group	[This document]
TBD2 Double-sided Bidirectional LSP Association Group	[This document]

### 8.2. Bidirectional LSP Association Group TLV

This document defines a new TLV for carrying additional information of LSPs within a Bidirectional LSP Association Group. IANA is requested to add the assignment of a new value in the existing "PCEP TLV Type Indicators" registry as follows:

TLV-Type	Name	Reference
TBD3	Bidirectional LSP Association Group TLV	[This document]

#### 8.2.1. Flag Fields in Bidirectional LSP Association Group TLV

This document requests that a new sub-registry, named "Bidirectional LSP Association Group TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the Bidirectional LSP Association Group TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (count from 0 as the most significant bit)
- o Description
- o Reference

The following values are defined in this document for the Flag field.

Bit No.	Description	Reference
31	F - Forward LSP	[This document]
30	R - Reverse LSP	[This document]
29	C - Co-routed LSP	[This document]

#### 8.3. PCEP Errors

IANA is requested to allocate new Error-Type and Error-Value related to bidirectional LSP association within the " PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

Error-Type	Description	Reference
TBD4	Bidirectional LSP Association Error	[This document]
	Error-value=1: Tunnel mismatch	[This document]



## 9. References

### 9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [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>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.
- [RFC7551] Zhang, F., Ed., Jing, R., and R. Gandhi, Ed., "RSVP-TE Extensions for Associated Bidirectional LSPs", RFC 7551, May 2015.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [I-D.ietf-pce-association] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "PCEP Extensions for Establishing Relationships Between Sets of LSPs", draft-ietf-pce-association-group (work in progress).
- [I-D.ietf-pce-pce-initiated-lsp] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp (work in progress).
- [I-D.ietf-teas-assoc-corouted-bidir-frr] Gandhi, R., Ed., Shah, H., and J. Whittaker, "Fast Reroute Procedures for Associated Bidirectional Label Switched Paths", draft-ietf-teas-assoc-corouted-bidir-frr (work-in-progress).

## 9.2. Informative References

- [RFC5654] Niven-Jenkins, B., Ed., Brungard, D., Ed., Betts, M., Ed., Sprecher, N., and S. Ueno, "Requirements of an MPLS Transport Profile", RFC 5654, September 2009.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, December 2014.
- [I-D.ietf-pce-pceps] Lopez, D., Dios, O., Wu, Q., and D. Dhody, "Secure Transport for PCEP", draft-ietf-pce-pceps (work in progress).
- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang (work in progress).

#### Acknowledgments

TBA.

#### Authors' Addresses

Colby Barth  
Juniper Networks

Email: cbarth@juniper.net

Rakesh Gandhi  
Cisco Systems, Inc.

Email: rgandhi@cisco.com

Bin Wen  
Comcast

Email: Bin\_Wen@cable.comcast.com

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: September 28, 2018

C. Barth  
Juniper Networks  
R. Gandhi, Ed.  
Cisco Systems, Inc.  
B. Wen  
Comcast  
March 27, 2018

PCEP Extensions for  
Associated Bidirectional Label Switched Paths (LSPs)  
draft-barth-pce-association-bidir-04

## Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests. The Stateful PCE extensions allow stateful control of Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs) using PCEP.

This document defines PCEP extensions for grouping two reverse unidirectional MPLS TE LSPs into an Associated Bidirectional LSP when using a Stateful PCE for both PCE-Initiated and PCC-Initiated LSPs as well as when using a Stateless PCE.

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

## Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Conventions Used in This Document . . . . .	4
2.1. Key Word Definitions . . . . .	4
2.2. Terminology . . . . .	4
3. Overview . . . . .	4
3.1. Single-sided Initiation . . . . .	5
3.2. Double-sided Initiation . . . . .	5
3.3. Co-routed Associated Bidirectional LSP . . . . .	6
4. Protocol Extensions . . . . .	6
4.1. Association Object . . . . .	6
4.2. Bidirectional LSP Association Group TLV . . . . .	7
5. PCEP Procedure . . . . .	8
5.1. PCE Initiated LSPs . . . . .	8
5.2. PCC Initiated LSPs . . . . .	8
5.3. Stateless PCE . . . . .	9
5.4. State Synchronization . . . . .	9
5.5. Error Handling . . . . .	9
6. Security Considerations . . . . .	9
7. Manageability Considerations . . . . .	10
7.1. Control of Function and Policy . . . . .	10
7.2. Information and Data Models . . . . .	10
7.3. Liveness Detection and Monitoring . . . . .	10
7.4. Verify Correct Operations . . . . .	10
7.5. Requirements On Other Protocols . . . . .	10
7.6. Impact On Network Operations . . . . .	10
8. IANA Considerations . . . . .	10
8.1. Association Types . . . . .	10
8.2. Bidirectional LSP Association Group TLV . . . . .	11
8.2.1. Flag Fields in Bidirectional LSP Association Group TLV . . . . .	11
8.3. PCEP Errors . . . . .	11
9. References . . . . .	13
9.1. Normative References . . . . .	13
9.2. Informative References . . . . .	13
Acknowledgments . . . . .	15
Authors' Addresses . . . . .	15

## 1. Introduction

[RFC5440] describes the Path Computation Element Protocol (PCEP) as a communication mechanism between a Path Computation Client (PCC) and a Path Control Element (PCE), or between PCE and PCC, that enables computation of Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs).

[RFC8231] specifies extensions to PCEP to enable stateful control of MPLS TE LSPs. It describes two modes of operation - Passive Stateful PCE and Active Stateful PCE. In [RFC8231], the focus is on Active Stateful PCE where LSPs are provisioned on the PCC and control over them is delegated to a PCE. Further, [RFC8281] describes the setup, maintenance and teardown of PCE-Initiated LSPs for the Stateful PCE model.

[I-D.ietf-pce-association] introduces a generic mechanism to create a grouping of LSPs which can then be used to define associations between a set of LSPs and/or a set of attributes, for example primary and secondary LSP associations, and is equally applicable to the active and passive modes of a Stateful PCE [RFC8231] or a stateless PCE [RFC5440].

The MPLS Transport Profile (MPLS-TP) requirements document [RFC5654] specifies that MPLS-TP MUST support associated bidirectional point-to-point LSPs. [RFC7551] specifies RSVP signaling extensions for binding two reverse unidirectional LSPs [RFC3209] into an associated bidirectional LSP. The fast reroute (FRR) procedures for associated bidirectional LSPs are described in [I-D.ietf-teas-assoc-corouted-bidir-frr].

This document specifies PCEP extensions for grouping two reverse unidirectional MPLS-TE LSPs into an Associated Bidirectional LSP for both single-sided and double-sided initiation cases when using a Stateful (both active and passive modes) or Stateless PCE. The PCEP extensions cover the following cases:

- o A PCC initiates the forward and/ or reverse LSP of a single-sided or double-sided bidirectional LSP and retains the control of the LSP. The PCC computes the path itself or makes a request for path computation to a PCE. After the path setup, it reports the information and state of the path to the PCE. This includes the association group identifying the bidirectional LSP. This is the Passive Stateful mode defined in [RFC8051].
- o A PCC initiates the forward and/ or reverse LSP of a single-sided or double-sided bidirectional LSP and delegates the control of the LSP to a Stateful PCE. During delegation the association group

identifying the bidirectional LSP is included. The PCE computes the path of the LSP and updates the PCC with the information about the path as long as it controls the LSP. This is the Active Stateful mode defined in [RFC8051].

- o A PCE initiates the forward and/ or reverse LSP of a single-sided or double-sided bidirectional LSP on a PCC and retains the control of the LSP. The PCE is responsible for computing the path of the LSP and updating the PCC with the information about the path as well as the association group identifying the bidirectional LSP. This is the PCE-Initiated mode defined in [RFC8281].
- o A PCC requests co-routed or non co-routed paths for forward and reverse LSPs of a bidirectional LSP from a Stateless PCE [RFC5440].

## 2. Conventions Used in This Document

### 2.1. Key Word Definitions

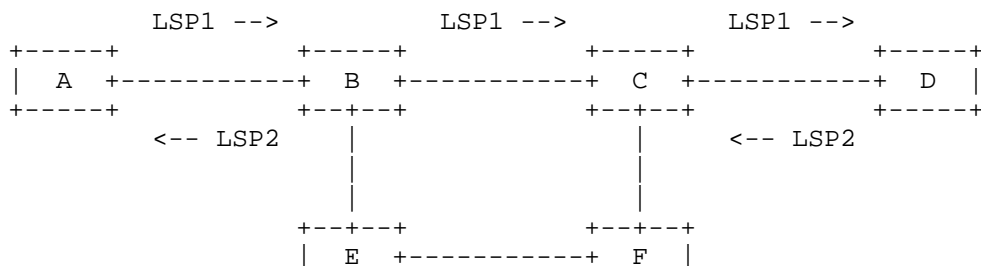
The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

### 2.2. Terminology

The reader is assumed to be familiar with the terminology defined in [RFC5440], [RFC7551], [RFC8231], and [I-D.ietf-pce-association].

## 3. Overview

As shown in Figure 1, two reverse unidirectional LSPs can be grouped to form an associated bidirectional LSP. There are two methods of initiating the bidirectional LSP association, single-sided and double-sided as described in the following sections.



```

+-----+ +-----+
<-- LSP2

```

Figure 1: Example of Associated Bidirectional LSP

### 3.1. Single-sided Initiation

As specified in [RFC7551], in the single-sided case, the bidirectional tunnel is provisioned only on one endpoint node (PCC) of the tunnel. Both forward and reverse LSPs of this tunnel are initiated with the Association Type set to "Single-sided Bidirectional LSP Association" on the originating endpoint node. The forward and reverse LSPs are identified in the Bidirectional LSP Association Group TLV of their PCEP Association Objects.

The originating endpoint node signals the properties for the reverse LSP in the RSVP REVERSE\_LSP Object [RFC7551] of the forward LSP Path message. The remote endpoint then creates the corresponding reverse tunnel and signals the reverse LSP in response to the received RSVP Path message.

The two unidirectional reverse LSPs on the originating endpoint node are grouped together using the PCEP Association Object and on the remote endpoint node by the RSVP signaled Association Object.

As shown in Figure 1, the forward tunnel and both the forward LSP LSP1 and the reverse LSP LSP2 are initiated on the originating endpoint node A, either by the PCE or the PCC. The creation of reverse tunnel and reverse LSP2 on the remote endpoint node D is triggered by the RSVP signaled LSP1.

As specified in [I-D.ietf-teas-assoc-corouted-bidir-frr], for fast-reroute bypass tunnel assignment, the LSP starting from the originating node is identified as the forward LSP of the single-sided initiated bidirectional LSP.

### 3.2. Double-sided Initiation

As specified in [RFC7551], in the double-sided case, the bidirectional tunnel is provisioned on both endpoint nodes (PCCs) of the tunnel. The forward and reverse LSPs of this tunnel are initiated with the Association Type set to "Double-sided Bidirectional LSP Association" on both endpoint nodes. The forward and reverse LSPs are identified in the Bidirectional LSP Association Group TLV of their Association Objects.

The two reverse unidirectional LSPs on both the endpoint nodes are grouped together by using the PCEP Association Object.



As shown in Figure 1, the forward tunnel and LSP1 are initiated on the endpoint node A and the reverse tunnel and LSP2 are initiated on the endpoint node D, either by the PCE or the PCCs.

As specified in [I-D.ietf-teas-assoc-corouted-bidir-frr], for fast-reroute bypass tunnel assignment, the LSP with the higher Source Address [RFC3209] is identified as the forward LSP of the double-sided initiated bidirectional LSP.

### 3.3. Co-routed Associated Bidirectional LSP

In both single-sided and double-sided initiation cases, forward and reverse LSPs may be co-routed as shown in Figure 2, where both forward and reverse LSPs follow the same congruent path in the forward and reverse directions, respectively.

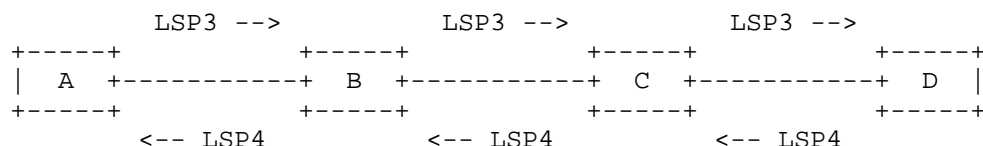


Figure 2: Example of Co-routed Associated Bidirectional LSP

## 4. Protocol Extensions

### 4.1. Association Object

As per [I-D.ietf-pce-association], LSPs are associated by adding them to a common association group. This document defines two new Bidirectional LSP Association Groups to be used by the associated bidirectional LSPs. A member of the Bidirectional LSP Association Group can take the role of a forward or reverse LSP and follows the following rules:

- o An LSP can not be part of more than one Bidirectional LSP Association Group.
- o The Tunnel (as defined in [RFC3209]) of forward and reverse LSPs of the single-sided bidirectional association MUST be the same.

This document defines two new Association Types for the Association Object as follows:

- o Association Type (TBD1) = Single-sided Bidirectional LSP Association Group

- o Association Type (TBD2) = Double-sided Bidirectional LSP Association Group

These Association Types are operator-configured associations in nature and statically created by the operator on the PCEP peers. The LSP belonging to these associations is conveyed via PCEP messages to the PCEP peer. Operator-configured Association Range TLV [I-D.ietf-pce-association] MUST NOT be sent for these Association Types, and MUST be ignored, so that the entire range of association ID can be used for them.

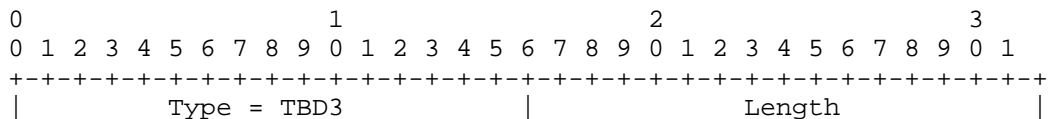
The Association ID, Association Source, optional Global Association Source and optional Extended Association ID in the Bidirectional LSP Association Group Object are initialized using the procedures defined in [RFC7551] and [I-D.ietf-pce-association].

#### 4.2. Bidirectional LSP Association Group TLV

The Bidirectional LSP Association Group TLV is an optional TLV for use with the Single-sided and Double-sided Bidirectional LSP Association Group Object Types.

- o The Bidirectional LSP Association Group TLV follows the PCEP TLV format from [RFC5440].
- o The Type (16 bits) of the TLV is TBD3, to be assigned by IANA.
- o The Length is 4 Bytes.
- o The value comprises of a single field, the Bidirectional LSP Association Flags (32 bits), where each bit represents a flag option.
- o If the Bidirectional LSP Association Group TLV is missing, it means the LSP is the forward LSP.
- o The Bidirectional LSP Association Group TLV MUST NOT be present more than once. If it appears more than once, only the first occurrence is processed and any others MUST be ignored.

The format of the Bidirectional LSP Association Group TLV is shown in Figure 3:



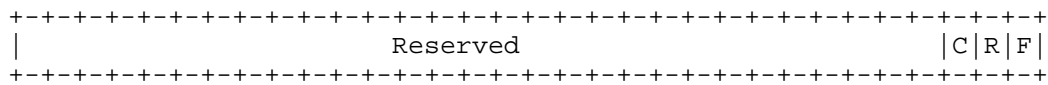


Figure 3: Bidirectional LSP Association Group TLV format

Bidirectional LSP Association Flags are defined as following.

F (Forward LSP, 1 bit) - Indicates whether the LSP associated is the forward LSP of the bidirectional LSP. If this flag is set, the LSP is a forward LSP.

R (Reverse LSP, 1 bit) - Indicates whether the LSP associated is the reverse LSP of the bidirectional LSP. If this flag is set, the LSP is a reverse LSP.

C (Co-routed LSP, 1 bit) - Indicates whether the bidirectional LSP is co-routed. This flag MUST be set for both the forward and reverse LSPs of a co-routed bidirectional LSP.

The C flag is used by the PCE (for both Stateful and Stateless) to compute bidirectional paths of the forward and reverse LSPs.

The Reserved flags MUST be set to 0 when sent and MUST be ignored when received.

## 5. PCEP Procedure

### 5.1. PCE Initiated LSPs

As specified in [I-D.ietf-pce-association], Association Groups can be created by both Stateful PCE and PCC.

A Stateful PCE can create and update the forward and reverse LSPs independently for both Single-sided and Double-sided bidirectional LSP association groups. The establishment and removal of the association relationship can be done on a per LSP basis. A PCE can create and update the association of the LSP on a PCC via PCInitiate and PCUpd messages, respectively, using the procedures described in [I-D.ietf-pce-association].

### 5.2. PCC Initiated LSPs

A PCC can create and update the forward and reverse LSPs independently for both Single-sided and Double-sided bidirectional LSP association groups. The establishment and removal of the association relationship can be done on a per LSP basis. In both

cases, the PCC must report the change in association to PCE(s) via PCRpt message. A PCC can also delegate the forward and reverse LSPs to a Stateful PCE, where PCE would control the LSPs. The Stateful PCE could update the LSPs in the association group via PCUpd message, using the procedures described in [I-D.ietf-pce-association].

### 5.3. Stateless PCE

For a stateless PCE, it might be useful to associate a path computation request to an association group, thus enabling it to associate a common set of configuration parameters or behaviors with the request. A PCC can request co-routed or non co-routed forward and reverse direction paths from a stateless PCE for the bidirectional LSP association group.

### 5.4. State Synchronization

During state synchronization, a PCC MUST report all the existing bidirectional LSP association groups to the Stateful PCE as per [I-D.ietf-pce-association]. After the statesynchronization, the PCE MUST remove all stale bidirectional associations.

### 5.5. Error Handling

The LSPs (forward or reverse) in a single-sided bidirectional LSP association group MUST belong to the same TE Tunnel (as defined in [RFC3209]). If a PCE attempts to add an LSP in a single-sided bidirectional LSP association group for a different Tunnel, the PCC MUST send PCErr with Error-Type = 29 (Early allocation by IANA) (Association Error) and Error-Value = TBD4 (Bidirectional Association Tunnel Mismatch). Similarly, if a PCC attempts to add an LSP to a single-sided bidirectional LSP association group at PCE not complying to this rule, the PCE MUST send this PCErr.

## 6. Security Considerations

The security considerations described in [RFC5440], [RFC8231], and [RFC8281] apply to the extensions defined in this document as well.

Two new Association Types for the Association Object, Double-sided Bidirectional LSP Association Group and Single-sided Associated Bidirectional LSP Group are introduced in this document. Additional security considerations related to LSP associations due to a malicious PCEP speaker is described in [I-D.ietf-pce-association] and apply to these Association Types. Hence, securing the PCEP session using Transport Layer Security (TLS) [RFC8253] is recommended.

## 7. Manageability Considerations

### 7.1. Control of Function and Policy

The mechanisms defined in this document do not imply any control or policy requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

### 7.2. Information and Data Models

[RFC7420] describes the PCEP MIB, there are no new MIB Objects defined for LSP associations.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] supports LSP associations.

### 7.3. Liveness Detection and Monitoring

The mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

### 7.4. Verify Correct Operations

The mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

### 7.5. Requirements On Other Protocols

The mechanisms defined in this document do not add any new requirements on other protocols.

### 7.6. Impact On Network Operations

The mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440], [RFC8231], and [RFC8281].

## 8. IANA Considerations

### 8.1. Association Types

This document adds new Association Types for the Association Object defined [I-D.ietf-pce-association]. IANA is requested to make the assignment of values for the sub-registry "ASSOCIATION Type Field" (to be created in [I-D.ietf-pce-association]), as follows:

Value Name	Reference
TBD1 Single-sided Bidirectional LSP Association Group	[This document]
TBD2 Double-sided Bidirectional LSP Association Group	[This document]

## 8.2. Bidirectional LSP Association Group TLV

This document defines a new TLV for carrying additional information of LSPs within a Bidirectional LSP Association Group. IANA is requested to add the assignment of a new value in the existing "PCEP TLV Type Indicators" registry as follows:

TLV-Type	Name	Reference
TBD3	Bidirectional LSP Association Group TLV	[This document]

### 8.2.1. Flag Fields in Bidirectional LSP Association Group TLV

This document requests that a new sub-registry, named "Bidirectional LSP Association Group TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the Bidirectional LSP Association Group TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (count from 0 as the most significant bit)
- o Description
- o Reference

The following values are defined in this document for the Flag field.

Bit No.	Description	Reference
31	F - Forward LSP	[This document]
30	R - Reverse LSP	[This document]
29	C - Co-routed LSP	[This document]

## 8.3. PCEP Errors

This document defines new Error value for Error Type 29 (Association Error). IANA is requested to allocate new Error value within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

Error Type	Description	Reference
------------	-------------	-----------

```

29      Association Error
      Error value: TBD4 [This document]
      Bidirectional Association Tunnel Mismatch

```

## 9. References

### 9.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997.
- [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.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.
- [RFC7551] Zhang, F., Ed., Jing, R., and R. Gandhi, Ed., "RSVP-TE Extensions for Associated Bidirectional LSPs", RFC 7551, May 2015.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", RFC 8281, December 2017.
- [I-D.ietf-pce-association] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "PCEP Extensions for Establishing Relationships Between Sets of LSPs", draft-ietf-pce-association-group (work in progress).
- [I-D.ietf-teas-assoc-corouted-bidir-frr] Gandhi, R., Ed., Shah, H., and J. Whittaker, "Fast Reroute Procedures for Co-routed Associated Bidirectional Label Switched Paths (LSPs)", draft-ietf-teas-assoc-corouted-bidir-frr (work in progress).

### 9.2. Informative References

- [RFC5654] Niven-Jenkins, B., Ed., Brungard, D., Ed., Betts, M., Ed.,



Sprecher, N., and S. Ueno, "Requirements of an MPLS Transport Profile", RFC 5654, September 2009.

[RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, December 2014.

[RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, January 2017.

[RFC8253] Lopez, D., 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, October 2017.

[I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang (work in progress).

#### Acknowledgments

Authors would like to thank Dhruv Dhody for various discussions on association groups.

#### Authors' Addresses

Colby Barth  
Juniper Networks

Email: cbarth@juniper.net

Rakesh Gandhi (editor)  
Cisco Systems, Inc.  
Canada

Email: rgandhi@cisco.com

Bin Wen  
Comcast

Email: Bin\_Wen@cable.comcast.com

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: March 5, 2018

I. Minei  
Google, Inc.  
E. Crabbe  
Individual Contributor  
S. Sivabalan  
Cisco Systems, Inc.  
H. Ananthakrishnan  
Packet Design  
D. Dhody  
Huawei  
Y. Tanaka  
NTT Communications Corporation  
September 1, 2017

PCEP Extensions for Establishing Relationships Between Sets of LSPs  
draft-ietf-pce-association-group-04

Abstract

This document introduces a generic mechanism to create a grouping of LSPs in the context of a PCE. This grouping can then be used to define associations between sets of LSPs or between a set of LSPs and a set of attributes (such as configuration parameters or behaviors), and is equally applicable to stateful PCE (active and passive modes) and stateless PCE.

Requirements Language

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

Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

#### Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Architectural Overview . . . . .	4
3.1. Motivation . . . . .	4
3.2. Operation Overview . . . . .	4
3.3. Operator-configured Association Range . . . . .	5
4. Operator-configured Association Range TLV . . . . .	5
4.1. Procedure . . . . .	7
5. ASSOCIATION Object . . . . .	7
5.1. Object Definition . . . . .	7
5.1.1. Global Association Source TLV . . . . .	9
5.1.2. Extended Association ID TLV . . . . .	9
5.2. Object Encoding in PCEP messages . . . . .	10
5.2.1. Stateful PCEP messages . . . . .	10
5.2.2. Request Message . . . . .	12
5.3. Processing Rules . . . . .	13
6. IANA Considerations . . . . .	15
6.1. PCEP Object . . . . .	15
6.2. PCEP TLV . . . . .	15
6.3. Association Flags . . . . .	15
6.4. Association Type . . . . .	16
6.5. PCEP-Error Object . . . . .	16
7. Security Considerations . . . . .	17
8. Manageability Considerations . . . . .	17
8.1. Control of Function and Policy . . . . .	17
8.2. Information and Data Models . . . . .	17

8.3. Liveness Detection and Monitoring . . . . .	17
8.4. Verify Correct Operations . . . . .	18
8.5. Requirements On Other Protocols . . . . .	18
8.6. Impact On Network Operations . . . . .	18
9. Acknowledgements . . . . .	18
10. Contributors . . . . .	18
11. References . . . . .	19
11.1. Normative References . . . . .	19
11.2. Informative References . . . . .	19
Authors' Addresses . . . . .	20

## 1. Introduction

[RFC5440] describes the Path Computation Element Protocol PCEP. PCEP enables the communication between a Path Computation Client (PCC) and a Path Control Element (PCE), or between PCE and PCE, for the purpose of computation of Multiprotocol Label Switching (MPLS) as well as Generalized MPLS (GMPLS) for Traffic Engineering Label Switched Path (TE LSP) characteristics.

Stateful pce [I-D.ietf-pce-stateful-pce] specifies a set of extensions to PCEP to enable stateful control of TE LSPs between and across PCEP sessions in compliance with [RFC4657] and focuses on a model where LSPs are configured on the PCC and control over them is delegated to the PCE. The model of operation where LSPs are initiated from the PCE is described in [I-D.ietf-pce-pce-initiated-lsp].

This document introduces a generic mechanism to create a grouping of LSPs. This grouping can then be used to define associations between sets of LSPs or between a set of LSPs and a set of attributes (such as configuration parameters or behaviors), and is equally applicable to stateful PCE (active and passive modes) and stateless PCE. The associations could be created dynamically and conveyed to a PCEP peer within PCEP, or it could be configured by an operator on the PCEP peers. Refer Section 3.2 for more details.

## 2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [RFC8051]: Stateful PCE, Delegation.

This document uses the following terms defined in [I-D.ietf-pce-stateful-pce]: Redelegation Timeout Interval, LSP State Report, LSP Update Request.

This document uses the following terms defined in [I-D.ietf-pce-pce-initiated-lsp]: PCE-initiated LSP, LSP Initiate.

The following term is defined in this document:

Association Timeout Interval: when a PCEP session is terminated, a PCC waits for this time period before deleting associations created by the PCEP peer.

### 3. Architectural Overview

#### 3.1. Motivation

Stateful PCE provides the ability to update existing LSPs and to instantiate new ones. To enable support for PCE-controlled make-before-break and for protection, there is a need to define associations between LSPs. For example, the association between the original and the re-optimized path in the make-before break scenario, or between the working and protection path in end-to-end protection. Another use for LSP grouping is for applying a common set of configuration parameters or behaviors to a set of LSPs.

For a stateless PCE, it might be useful to associate a path computation request to an association group, thus enabling it to associate a common set of policy, configuration parameters or behaviors with the request.

Some associations could be created dynamically, such as association between the working and protection LSPs of a tunnel. Whereas some association could be created by the operator manually, such as policy based association, where the LSP could join an operator-configured existing association.

Rather than creating separate mechanisms for each use case, this draft defines a generic mechanism that can be reused as needed.

#### 3.2. Operation Overview

LSPs are associated with other LSPs with which they interact by adding them to a common association group. Association groups as defined in this document can be applied to LSPs originating at the same head end or different head ends.

Some associations could be created dynamically by a PCEP speaker and the associations (along with the set of LSPs) are conveyed to a PCEP peer. Whereas, some associations are configured by the operator on the PCEP peers involved before hand, a PCEP speaker then could ask for a LSP to join the operator-configured association. Usage of

dynamic and configured is usually dependent on the type of the association.

For the operator-configured association, the association identifier, type, as well as the association source IP address is manually configured by the operator. In case of dynamic association, the association identifier is allocated dynamically by the PCEP speaker.

The dynamically created association can be reported to the PCEP peer via the PCEP messages as per the stateful extensions. While the operator-configured association are known to the PCEP peer before hand, a PCEP peer could ask for a LSP to join the operator-configured association via the stateful PCEP messages.

The association are properties of the LSP and thus could be stored in the LSP state database. The dynamic association exist as long as the LSP state. In case of PCEP session termination, the LSP state clean up SHOULD also take care of associations.

Multiple types of associations can exist, each with their own association identifier space. The definition of the different association types and their behaviors is outside the scope of this document. The establishment and removal of the association relationship can be done on a per LSP basis. An LSP may join multiple association groups, of different or of the same association type.

### 3.3. Operator-configured Association Range

Some association types are dynamic, some are operator-configured and some could be both. For the association types that could be dynamic and operator-configured, it is necessary to configure a range of association identifiers that are marked for operator-configured associations to avoid any association identifier clash.

A range of association identifier for each association-type are kept for the operator-configured associations. Dynamic associations MUST NOT use the association identifier from this range.

This range needs to be communicated to a PCEP peer in the Open Message. A new TLV is defined in this specification for this purpose (Section 4).

### 4. Operator-configured Association Range TLV

This section defines PCEP extension to support the advertisement of the Operator-configured Association Range used for an association-type.

A new PCEP OP-CONF-ASSOC-RANGE (Operator-configured Association Range) TLV is defined. The PCEP OP-CONF-ASSOC-RANGE TLV is carried within an OPEN object. This way, during PCEP session-setup phase, a PCEP speaker can advertise to a PCEP peer the Operator-configured Association Range for an association type.

The PCEP OP-CONF-ASSOC-RANGE TLV is optional. It MAY be carried within an OPEN object sent by a PCEP speaker in an Open message to a PCEP peer. The OP-CONF-ASSOC-RANGE TLV format is compliant with the PCEP TLV format defined in [RFC5440]. That is, the TLV is composed of 2 octets for the type, 2 octets specifying the TLV length, and a Value field. The Length field defines the length of the value portion in octets.

The PCEP OP-CONF-ASSOC-RANGE TLV has the following format:

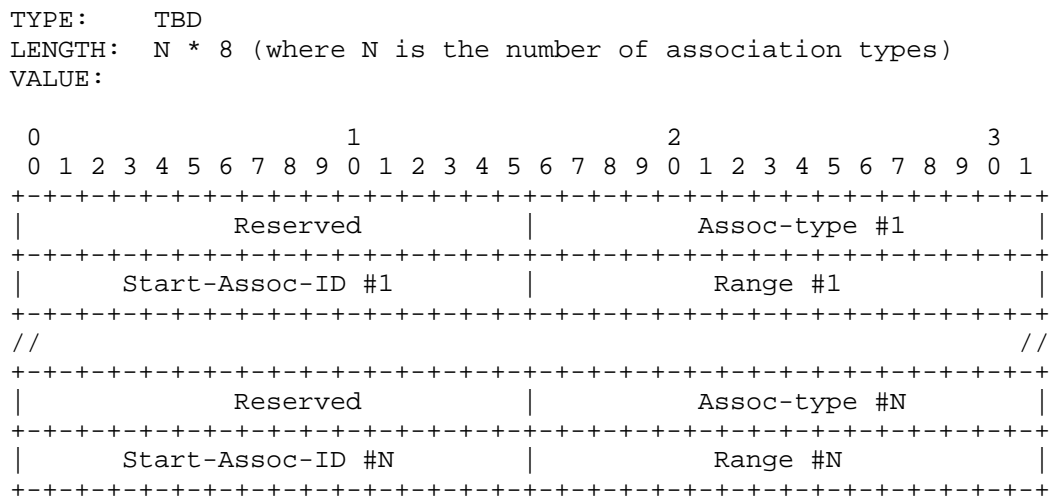


Figure 1: The OP-CONF-ASSOC-RANGE TLV format

The Value portion includes the following fields, repeated for each association type:

Reserved (2 bytes): This field MUST be set to 0 on transmission and MUST be ignored on receipt.

Assoc-type (2 bytes): The association type.



Start-Assoc-ID (2 bytes): The start association identifier for the Operator-configured Association Range for the particular association type.

Range (2 bytes): The number of associations marked for the Operator-configured Associations.

#### 4.1. Procedure

A PCEP speaker MAY include an OP-CONF-ASSOC-RANGE TLV within an OPEN object in an Open message sent to a PCEP peer in order to advertise the Operator-configured Association Range for an association type. The OP-CONF-ASSOC-RANGE TLV MUST NOT appear more than once in an OPEN object. If it appears more than once, the PCEP session MUST be rejected with error type 1 and error value 1 (PCEP session establishment failure / Reception of an invalid Open message).

As specified in [RFC5440], a PCEP peer that does not recognize the OP-CONF-ASSOC-RANGE TLV will silently ignore it.

The Operator-configured Association Range SHOULD be included for each association type that could be both dynamic and operator-configured. For association types that are only dynamic or only operator-configured, this TLV can be skipped, in which case the full range of association identifier is considered dynamic or operator-configured respectively. Each association type (that are defined in separate documents) can specify the default value for the operator-configured association range for their respective association type.

The absence of the OP-CONF-ASSOC-RANGE TLV in an OPEN object MUST be interpreted as an absence of explicit Operator-configured Association Range at the PCEP peer. In which case, the default behavior as per each association type would be applied.

### 5. ASSOCIATION Object

#### 5.1. Object Definition

Association groups and their memberships are defined using a new ASSOCIATION object.

ASSOCIATION Object-Class is to be assigned by IANA (TBD).

ASSOCIATION Object-Type is 1 for IPv4 and its format is shown in Figure 2:

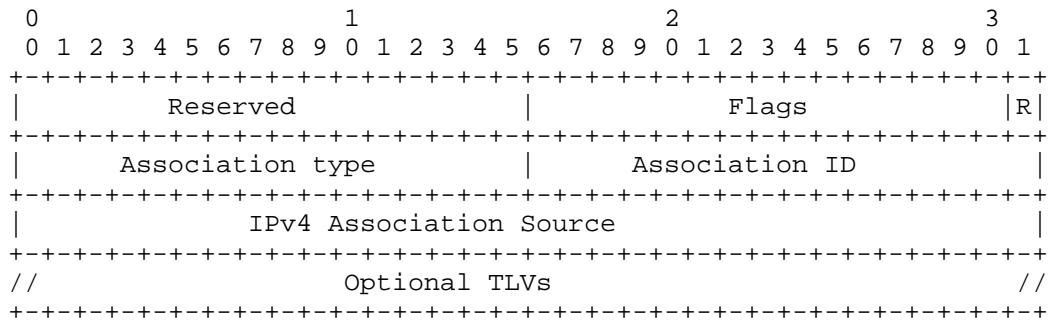


Figure 2: The IPv4 ASSOCIATION Object format

ASSOCIATION Object-Type is 2 for IPv6 and its format is shown in Figure 3:

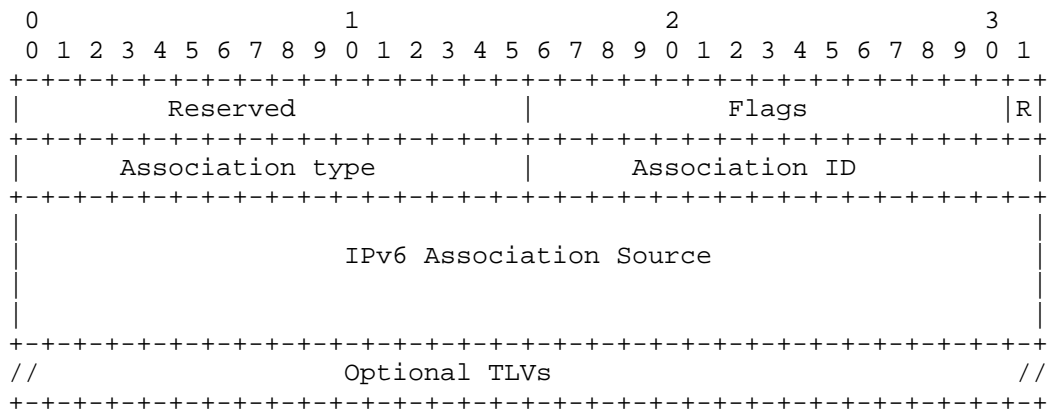


Figure 3: The IPv6 ASSOCIATION Object format

Reserved (2-byte): MUST be set to 0 and ignored upon receipt.

Flags (2-byte): The following flags are currently defined:

R (Removal - 1 bit): when set, the requesting PCE peer requires the removal of an LSP from the association group. This flag is used for ASSOCIATION object in PCRpt and PCUpd message, the flag is ignored in other PCEP messages.

Association type (2-byte): the association type (for example protection). The association type are defined in separate documents.

Association ID (2-byte): the identifier of the association group. When combined with Type and Association Source, this value uniquely identifies an association group. The value 0xffff and 0x0 are reserved. The value 0xffff is used to indicate all association groups.

Association Source: 4 or 16 bytes - An IPv4 or IPv6 address. This could be the IP address of the PCEP speaker that created a dynamic association, an operator configured IP address, or an IP address selected as per the local policy. The value such as 0.0.0.0 or ::/128 are acceptable.

Optional TLVs: The optional TLVs follow the PCEP TLV format of [RFC5440]. This document defines two optional TLVs. Other documents can define more TLVs.

#### 5.1.1. Global Association Source TLV

The Global Association Source TLV is an optional TLV for use in the Association Object.

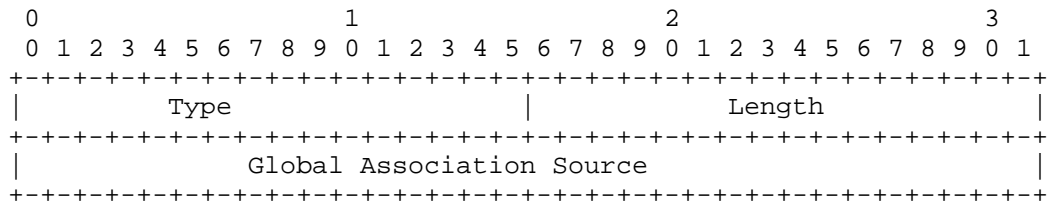


Figure 4: The Global Association Source TLV format

Type: To be allocated by IANA.

Length: Fixed value of 4 bytes.

Global Association Source: as defined in [RFC6780].

#### 5.1.2. Extended Association ID TLV

The Extended Association ID TLV is an optional TLV for use in the Association Object.

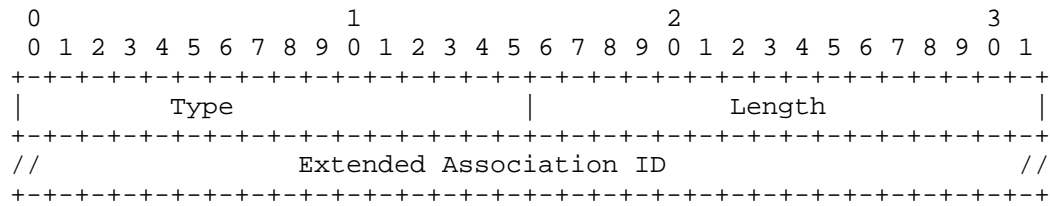


Figure 5: The Extended Association ID TLV format

Type: To be allocated by IANA.

Length: variable.

Extended Association ID: as defined in [RFC6780].

## 5.2. Object Encoding in PCEP messages

### 5.2.1. Stateful PCEP messages

The ASSOCIATION Object is OPTIONAL and MAY be carried in the Path Computation Update (PCUpd), Path Computation Report (PCRpt) and Path Computation Initiate (PCInitiate) messages.

When carried in PCRpt message, it is used to report the association group membership information pertaining to a LSP to a stateful PCE. It can also be used to remove an LSP from one or more association groups by setting the R flag to 1 in the ASSOCIATION object. Unless, a PCE wants to delete an association from an LSP, it does not need to carry the ASSOCIATION object in future messages.

The PCRpt message is defined in [I-D.ietf-pce-stateful-pce] and updated as below:

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

Where:

```
<state-report-list> ::= <state-report> [<state-report-list>]
```

```
<state-report> ::= [<SRP>]
                  <LSP>
                  [<association-list>]
                  <path>
```

Where:

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

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

When an LSP is delegated to a stateful PCE, the stateful PCE can initiate a new association group for this LSP, or associate it with one or more existing association groups. This is done by including the ASSOCIATION Object in a PCUpd message. A stateful PCE can also remove a delegated LSP from one or more association groups by setting the R flag to 1 in the ASSOCIATION object.

The PCUpd message is defined in [I-D.ietf-pce-stateful-pce] and updated as below:

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

Where:

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

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

Where:

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

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

A PCE initiating a new LSP, can include the association group information. This is done by including the ASSOCIATION Object in a PCInitiate message. The PCInitiate message is defined in [I-D.ietf-pce-pce-initiated-lsp] and updated as below:

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

Where:

```
<PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                              [<PCE-initiated-lsp-list>]
```

```
<PCE-initiated-lsp-request> ::= (<PCE-initiated-lsp-instantiation>|
                                <PCE-initiated-lsp-deletion>)
```

```
<PCE-initiated-lsp-instantiation> ::= <SRP>
                                       <LSP>
                                       [<END-POINTS>]
                                       <ERO>
                                       [<association-list>]
                                       [<attribute-list>]
```

Where:

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

#### 5.2.2. Request Message

In case of passive stateful or stateless PCE, the ASSOCIATION Object is OPTIONAL and MAY be carried in the Path Computation Request (PCReq) message.

When carried in a PCReq message, the ASSOCIATION Object is used to associate the path computation request to an association group. The association (and the other LSPs) should be known to the PCE before hand. These could be operator-configured or dynamically learned before. The R flag in ASSOCIATION object within PCReq message MUST be set to 0 while sending and ignored on receipt.

The PCReq message is defined in [RFC5440] and updated in [I-D.ietf-pce-stateful-pce], it is further updated below for association:

```

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

```

Where:

```

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

```

```

<request> ::= <RP>
              <END-POINTS>
              [<LSP>]
              [<LSPA>]
              [<BANDWIDTH>]
              [<metric-list>]
              [<association-list>]
              [<RRO> [<BANDWIDTH>]]
              [<IRO>]
              [<LOAD-BALANCING>]

```

Where:

```

<association-list> ::= <ASSOCIATION> [<association-list>]

```

Note that LSP object MAY be present for the passive stateful PCE mode.

### 5.3. Processing Rules

Association groups can be operator-configured on the necessary PCC and PCE. In addition, a PCC or a PCE can create association groups dynamically. The PCEP speaker can reports the association to its peer via PCEP messages. If a PCEP speaker does not recognize the ASSOCIATION object, it will return a PCErr message with Error-Type "Unknown Object" as described in [RFC5440]. If a PCEP speaker understand the ASSOCIATION object but does not support the association-type, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 1 "Association-type is not supported". On receiving a PCEP message with ASSOCIATION, if a PCEP speaker finds that too many LSPs belong to the association group, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 2 "Too many LSPs in the association group". If a PCEP speaker cannot handle a new associations, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 3 "Too many association groups". These number MAY be set by operator or decided based on a local policy.

If a PCE speaker receives ASSOCIATION in PCReq message, and the association information is not known (association is not configured, or created dynamically, or learned from a PCEP peer), it MUST return

a PCErr message with Error-Type TBD "Association Error" and Error-Value 4 "Association unknown". If the association information received from the peer does not match with the local operator configured information, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 5 "Operator-configured association information mismatch". On receiving association information that does not match with the association information previously received about the same association from a peer, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 6 "Association information mismatch". If a PCE peer is unwilling or unable to process the ASSOCIATION object, it MUST return a PCErr message with the Error-Type "Not supported object" and follow the relevant procedures described in [RFC5440]. On receiving a PCEP message with ASSOCIATION, if a PCEP speaker could not add the LSP to the association group for any reason, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 7 "Cannot join the association group".

The association information is cleared along with the LSP state information as per the [I-D.ietf-pce-stateful-pce]. When a PCEP session is terminated, after expiry of State Timeout Interval at PCC, the LSP state associated with that PCEP session is reverted to operator-defined default parameters or behaviors. Same procedure is also followed for the association information. On session termination at the PCE, when the LSP state reported by PCC is cleared, the association information is also cleared. Where there are no LSPs in a association group, the association is considered to be deleted.

In case the LSP is delegated to another PCE on session failure, the association information set by the PCE remains intact, unless updated by the new PCE.

Upon LSP delegation revocation, the PCC MAY clear the association created by the PCE, but in order to avoid traffic loss, it can perform this in a make-before-break fashion, which is the same as what is defined in [I-D.ietf-pce-stateful-pce] for handling LSP state cleanup.

If a PCE speaker receives ASSOCIATION object with R bit set for removal, and the association information is not known, it MUST return a PCErr message with Error-Type TBD "Association Error" and Error-Value 4 "Association unknown".



## 6. IANA Considerations

IANA maintains the "Path Computation Element Protocol (PCEP) Numbers" registry at <http://www.iana.org/assignments/pcep>.

### 6.1. PCEP Object

The "PCEP Numbers" registry contains a subregistry "PCEP Objects". This document request IANA to allocate code points from this registry.

Object-Class	Value	Name	Reference
	TBD	Association Object-Type 0: Reserved 1: IPv4 2: IPv6	[This I-D]

### 6.2. PCEP TLV

IANA is requested to make the assignment of the new code points for the existing "PCEP TLV Type Indicators" registry as follows:

Value	Meaning	Reference
TBD	Operator-configured Association Range	[This I-D]
TBD	Global Association Source	[This I-D]
TBD	Extended Association Id	[This I-D]

### 6.3. Association Flags

This document requests IANA to create a subregistry of the "PCEP Numbers" for the bits carried in the Flags field of the ASSOCIATION object. The subregistry is called "ASSOCIATION Flags Field". New values are assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

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

Bit	Description	Reference
15	R (Removal)	[This I-D]

#### 6.4. Association Type

This document requests IANA to create a subregistry of the "PCEP Numbers" for the Association Type field of the the ASSOCIATION object. The subregistry is called "ASSOCIATION Type Field". New values are to be assigned by Standards Action [RFC8126]. Each value should be tracked with the following qualities:

- o Type
- o Name
- o Reference

There are no association type specified in this document, future document should request the assignment of association types from this subregistry.

#### 6.5. PCEP-Error Object

IANA is requested to allocate new error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the "PCEP Numbers" registry, as follows:

Error-Type	Meaning
TBD	Association Error [This I-D]
	Error-value=1: Association-type is not supported
	Error-value=2: Too many LSPs in the association group
	Error-value=3: Too many association groups
	Error-value=4: Association unknown
	Error-value=5: Operator-configured association information mismatch
	Error-value=6: Association information mismatch
	Error-value=7: Cannot join the association group

## 7. Security Considerations

The security considerations described in [I-D.ietf-pce-stateful-pce] and [RFC5440] apply to the extensions described in this document as well. Additional considerations related to a malicious PCEP speaker are introduced, as associations could be spoofed and could be used as an attack vector. An attacker could report too many associations in an attempt to load the PCEP peer. The PCEP peer responds with PCErr as described in Section 5.3. An attacker could impact LSP operations by creating bogus associations. Further, association information could provides an adversary with the opportunity to eavesdrop on the relationship between the LSPs. Thus securing the PCEP session using Transport Layer Security (TLS) [I-D.ietf-pce-pceps], as per the recommendations and best current practices in [RFC7525], is RECOMMENDED.

## 8. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440] and [I-D.ietf-pce-stateful-pce] apply to PCEP protocol extensions defined in this document. In addition, requirements and considerations listed in this section apply.

### 8.1. Control of Function and Policy

A PCE or PCC implementation MUST allow operator-configured associations as described in this document. The identifier MUST be from the operator-configured identifier range Section 3.3.

### 8.2. Information and Data Models

An implementation SHOULD allow the operator to view the associations configured or created dynamically. Further implementation SHOULD allow to view associations reported by each peer, and the current set of LSPs in the association . To serve this purpose, the PCEP YANG module [I-D.ietf-pce-pcep-yang] can be extended to include association information.

### 8.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

#### 8.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [I-D.ietf-pce-stateful-pce].

#### 8.5. Requirements On Other Protocols

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

#### 8.6. Impact On Network Operations

Mechanisms defined in [RFC5440] and [I-D.ietf-pce-stateful-pce] also apply to PCEP extensions defined in this document.

### 9. Acknowledgements

We would like to thank Yuji Kamite and Joshua George for their contributions to this document. Also thanks to Venugopal Reddy, Cyril Margaria and Rakesh Gandhi for their useful comments.

### 10. Contributors

Stephane Litkowski  
Orange

Email: stephane.litkowski@orange.com

Xian Zhang  
Huawei Technologies  
F3-1-B RnD Center, Huawei Base  
Bantian, Longgang District  
Shenzhen 518129  
China

Email: zhang.xian@huawei.com

Mustapha Aissaoui  
Nokia

Email: mustapha.aissaoui@nokia.com

## 11. References

### 11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [RFC6780] Berger, L., Le Faucheur, F., and A. Narayanan, "RSVP ASSOCIATION Object Extensions", RFC 6780, DOI 10.17487/RFC6780, October 2012, <<https://www.rfc-editor.org/info/rfc6780>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [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>>.
- [I-D.ietf-pce-stateful-pce]  
Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-21 (work in progress), June 2017.
- [I-D.ietf-pce-pce-initiated-lsp]  
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-10 (work in progress), June 2017.

### 11.2. Informative References

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

- [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>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a  
Stateful Path Computation Element (PCE)", RFC 8051,  
DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [I-D.ietf-pce-pceps]  
Lopez, D., Dios, O., Wu, Q., and D. Dhody, "Secure  
Transport for PCEP", draft-ietf-pce-pceps-16 (work in  
progress), August 2017.
- [I-D.ietf-pce-pcep-yang]  
Dhody, D., Hardwick, J., Beeram, V., and j.  
jeffrant@gmail.com, "A YANG Data Model for Path  
Computation Element Communications Protocol (PCEP)",  
draft-ietf-pce-pcep-yang-05 (work in progress), June 2017.

## Authors' Addresses

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

Email: [inaminei@google.com](mailto:inaminei@google.com)

Edward Crabbe  
Individual Contributor

Email: [edward.crabbe@gmail.com](mailto:edward.crabbe@gmail.com)

Siva Sivabalan  
Cisco Systems, Inc.  
170 West Tasman Dr.  
San Jose, CA 95134  
US

Email: [msiva@cisco.com](mailto:msiva@cisco.com)

Hariharan Ananthakrishnan  
Packet Design

Email: hari@packetdesign.com

Dhruv Dhody  
Huawei  
Divyashree Techno Park, Whitefield  
Bangalore, KA 560066  
India

Email: dhruv.ietf@gmail.com

Yosuke Tanaka  
NTT Communications Corporation  
Granpark Tower 3-4-1 Shibaura, Minato-ku  
Tokyo 108-8118  
Japan

Email: yosuke.tanaka@ntt.com

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: February 2, 2020

I. Minei  
Google, Inc.  
E. Crabbe  
Individual Contributor  
S. Sivabalan  
Cisco Systems, Inc.  
H. Ananthakrishnan  
Netflix  
D. Dhody  
Huawei  
Y. Tanaka  
NTT Communications Corporation  
August 1, 2019

Path Computation Element Communication Protocol (PCEP) Extensions for  
Establishing Relationships Between Sets of Label Switched Paths (LSPs)  
draft-ietf-pce-association-group-10

Abstract

This document introduces a generic mechanism to create a grouping of Label Switched Paths (LSPs) in the context of a Path Computation Element (PCE). This grouping can then be used to define associations between sets of LSPs or between a set of LSPs and a set of attributes (such as configuration parameters or behaviours), and is equally applicable to the stateful PCE (active and passive modes) as well as the stateless PCE.

Requirements Language

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

Status of This Memo

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

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



Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 2, 2020.

#### Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Architectural Overview . . . . .	4
3.1. Motivation . . . . .	4
3.2. Relationship with the SVEC List . . . . .	5
3.3. Operation Overview . . . . .	5
3.4. Operator-configured Association Range . . . . .	6
4. Discovery of Supported Association Types . . . . .	7
4.1. ASSOC-Type-List TLV . . . . .	7
4.1.1. Procedure . . . . .	8
5. Operator-configured Association Range TLV . . . . .	8
5.1. Procedure . . . . .	10
6. ASSOCIATION Object . . . . .	11
6.1. Object Definition . . . . .	11
6.1.1. Global Association Source TLV . . . . .	13
6.1.2. Extended Association ID TLV . . . . .	13
6.1.3. Association Source . . . . .	14
6.1.4. Unique Identification for an Association Group . . . . .	14
6.2. Relationship with the RSVP ASSOCIATION . . . . .	15
6.3. Object Encoding in PCEP messages . . . . .	15
6.3.1. Stateful PCEP messages . . . . .	15
6.3.2. Request Message . . . . .	17
6.3.3. Reply Message . . . . .	18
6.4. Processing Rules . . . . .	19

7.	IANA Considerations . . . . .	21
7.1.	PCEP Object . . . . .	21
7.2.	PCEP TLV . . . . .	22
7.3.	Association Flags . . . . .	22
7.4.	Association Type . . . . .	23
7.5.	PCEP-Error Object . . . . .	23
8.	Security Considerations . . . . .	24
9.	Manageability Considerations . . . . .	24
9.1.	Control of Function and Policy . . . . .	24
9.2.	Information and Data Models . . . . .	24
9.3.	Liveness Detection and Monitoring . . . . .	25
9.4.	Verify Correct Operations . . . . .	25
9.5.	Requirements on Other Protocols . . . . .	25
9.6.	Impact on Network Operations . . . . .	25
10.	Acknowledgments . . . . .	25
11.	Contributors . . . . .	25
12.	References . . . . .	26
12.1.	Normative References . . . . .	26
12.2.	Informative References . . . . .	27
	Appendix A. Example for Operator-configured Association Range .	28
	Authors' Addresses . . . . .	29

## 1. Introduction

[RFC5440] describes the Path Computation Element (PCE) Communication Protocol (PCEP). PCEP enables the communication between a Path Computation Client (PCC) and a PCE, or between PCE and PCE, for the purpose of computation of Multiprotocol Label Switching (MPLS) as well as Generalized MPLS (GMPLS) Traffic Engineering Label Switched Path (TE LSP) characteristics.

[RFC8231] specifies a set of extensions to PCEP to enable stateful control of TE LSPs within and across PCEP sessions in compliance with [RFC4657]. It includes mechanisms to effect LSP State Synchronization between PCCs and PCEs, delegation of control over LSPs to PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions. The model of operation where LSPs are initiated from the PCE is described in [RFC8281].

[RFC4872] defines the RSVP ASSOCIATION object, which was defined in the context of GMPLS-controlled Label Switched Paths (LSPs) to be used to associate recovery LSPs with the LSP they are protecting. This object also has broader applicability as a mechanism to associate RSVP state and [RFC6780] described how the ASSOCIATION object can be more generally applied by defining the Extended ASSOCIATION Object.

This document introduces a generic mechanism to create a grouping of LSPs. This grouping can then be used to define associations between sets of LSPs or between a set of LSPs and a set of attributes (such as configuration parameters or behaviours), and is equally applicable to stateful PCE (active and passive modes) and stateless PCE. The associations could be created dynamically and conveyed to a PCEP peer within PCEP, or it could be configured manually by an operator on the PCEP peers. Refer Section 3.3 for more details.

## 2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer, Path Computation Request (PCReq), Path Computation Reply (PCRep), and PCEP Error (PCErr).

This document uses the following terms defined in [RFC8051]: Stateful PCE, Active Stateful PCE, Passive Stateful PCE, and Delegation.

This document uses the following terms defined in [RFC8231]: LSP State Report (PCRpt), LSP Update Request (PCUpd), and State Timeout Interval.

This document uses the following terms defined in [RFC8281]: PCE-initiated LSP, and LSP Initiate Request (PCInitiate).

## 3. Architectural Overview

### 3.1. Motivation

Stateful PCE provides the ability to update existing LSPs and to instantiate new ones. There are various situations where several LSPs need to share common information. E.g., to support for PCE-controlled make-before-break, an association between the original and the re-optimized path is desired. Similarly, for end-to-end protection, the association between working and protection LSPs is required (see [I-D.ietf-pce-stateful-path-protection]). For diverse paths, an association between a group of LSPs could be used (see [I-D.ietf-pce-association-diversity]). Another use for the LSP grouping is for applying a common set of configuration parameters or behaviours to a set of LSPs.

For a stateless PCE, it might be useful to associate a path computation request to an association group, thus enabling it to associate a common set of policy, configuration parameters or behaviours with the request.

Some associations could be created dynamically, such as association between the working and protection LSPs of a tunnel, whereas some

associations could be created by the operator manually, such as policy-based association, where the LSP could join an operator-configured existing association.

Rather than creating separate mechanisms for each use case, this document defines a generic mechanism that can be reused as needed.

### 3.2. Relationship with the SVEC List

Note that, [RFC5440] defines a mechanism for the synchronization of a set of path computation requests by using the SVEC (Synchronization VECTOR) object, that specifies the list of synchronized requests that can either be dependent or independent. The SVEC object identifies the relationship between the set of path computation requests, identified by 'Request-ID-number' in RP (Request Parameters) object. [RFC6007] further clarifies the use of the SVEC list for synchronized path computations when computing dependent requests as well as describes a number of usage scenarios for SVEC lists within single-domain and multi-domain environments.

The motivation behind the association group defined in this document and the SVEC object are quite different, though some use cases may overlap. PCEP extensions that define a new association type should clarify the relationship between the SVEC object and the association type, if any.

### 3.3. Operation Overview

LSPs are associated with other LSPs with which they interact by adding them to a common association group. Association groups as defined in this document can be applied to LSPs originating at the same head end or different head ends.

Some associations could be created dynamically by a PCEP speaker and the associations (along with the set of LSPs) are conveyed to a PCEP peer. Whereas some associations are configured by the operator on the PCEP peers involved beforehand, a PCEP speaker then could ask for a LSP to join the operator-configured association. Usage of dynamic and configured association is usually dependent on the type of the association.

For the operator-configured association, the association parameters such as the association identifier, association type, as well as the association source IP address, are manually configured by the operator. In case of dynamic association, the association parameters such as the association identifier, are allocated dynamically by the PCEP speaker, the association source is set as local PCEP speaker

address, unless local policy dictates otherwise, in which case association source is set based on the local policy.

The dynamically created association can be reported to the PCEP peer via the PCEP messages as per the stateful extensions. When the operator-configured association is known to the PCEP peer beforehand, a PCEP peer could ask for a LSP to join the operator-configured association via the stateful PCEP messages.

The associations are properties of the LSP and thus could be stored in the LSP state database. The dynamic association exists as long as the LSP state exists. In case of PCEP session termination, the LSP state clean-up MUST also take care of associations.

Multiple types of associations can exist, each with their own association identifier space. The definition of the different association types and their behaviours is outside the scope of this document. The establishment and removal of the association relationship can be done on a per LSP basis. An LSP may join multiple association groups, of different or of the same association type.

#### 3.4. Operator-configured Association Range

Some association types are dynamic, some are operator-configured and some could be both. For the association types that could be both dynamic and operator-configured and use the same association source, it is necessary to distinguish a range of association identifiers that are marked for operator-configured associations to avoid any association identifier clash within the scope of the association source. This document assumes that these two ranges are configured.

A range of association identifiers for each Association type (and Association Source) are kept for the operator-configured associations. Dynamic associations MUST NOT use the association identifier from this range.

This range as set at the PCEP speaker (PCC or PCE, as an association source) needs to be communicated to a PCEP peer in the Open Message. A new TLV is defined in this specification for this purpose (Section 5). See Appendix A for an example.

Association identifier range for sources other than the PCEP speaker (for example an NMS system) is not communicated in PCEP and the procedure for operator-configured association range setting is outside the scope of this document.

#### 4. Discovery of Supported Association Types

This section defines PCEP extensions so as to support the capability advertisement of the association types supported by a PCEP speaker.

A new PCEP ASSOC-Type-List (Association Types list) TLV is defined. The PCEP ASSOC-Type-List TLV is carried within an OPEN object. This way, during PCEP session-setup phase, a PCEP speaker can advertise to a PCEP peer the list of supported Association types.

##### 4.1. ASSOC-Type-List TLV

The PCEP ASSOC-Type-List TLV is OPTIONAL. It MAY be carried within an OPEN object sent by a PCEP speaker in an Open message to a PCEP peer so as to indicate the list of supported Association types.

The PCEP ASSOC-Type-List TLV format is compliant with the PCEP TLV format defined in [RFC5440]. That is, the TLV is composed of 2 octets for the type, 2 octets specifying the TLV length, and a Value field. The Length field defines the length of the value portion in octets. The TLV is padded to 4-octet alignment, and padding is not included in the Length field (e.g., a 3-octet value would have a length of three, but the total size of the TLV would be eight octets).

The PCEP ASSOC-Type-List TLV has the following format:

TYPE: TBD  
 LENGTH:  $N * 2$  (where N is the number of association types)  
 VALUE: list of 2-byte association type code points, identifying the association types supported by the sender of the Open message.

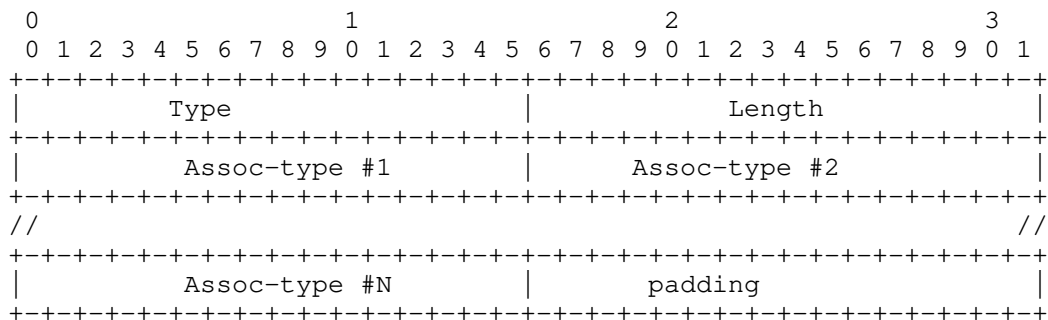


Figure 1: The ASSOC-Type-List TLV format

Assoc-type (2 bytes): Association type code point identifier. IANA manages the "ASSOCIATION Type Field" code point registry (see Section 7.4).

#### 4.1.1. Procedure

An ASSOC-Type-List TLV within an OPEN object in an Open message is included by a PCEP speaker in order to advertise a set of one or more supported association types. The ASSOC-Type-List TLV MUST NOT appear more than once in an OPEN object. If it appears more than once, the PCEP session MUST be rejected with error type 1 and error value 1 (PCEP session establishment failure / Reception of an invalid Open message). As specified in [RFC5440], a PCEP peer that does not recognize the ASSOC-Type-List TLV will silently ignore it.

Association type (to be defined in other documents) can specify if the association type advertisement is mandatory for it. Thus, the ASSOC-Type-List TLV MUST be included if at least one mandatory association type needs to be advertised and the ASSOC-Type-List TLV MAY be included otherwise. For an association type that specifies that the advertisement is mandatory, a missing Assoc-type in the ASSOC-Type-List TLV (or missing ASSOC-Type-List TLV) is to be interpreted as the association type is not supported by the PCEP speaker.

The absence of the ASSOC-Type-List TLV in an OPEN object MUST be interpreted as an absence of information on the list of supported Association types (rather than the Association type is not supported). In this case, the PCEP speaker could still use the ASSOCIATION object: if the peer does not support the association, it will react as per the procedure described in Section 6.4.

In case the use of the ASSOC-Type-List TLV is triggered by support for a mandatory association type, then it is RECOMMENDED that the PCEP implementation include all supported Association types (including optional) to ease the operations of the PCEP peer.

#### 5. Operator-configured Association Range TLV

This section defines PCEP extension to support the advertisement of the Operator-configured Association Range used for an Association type by the PCEP speaker (as an Association source).

A new PCEP OP-CONF-ASSOC-RANGE (Operator-configured Association Range) TLV is defined. The PCEP OP-CONF-ASSOC-RANGE TLV is carried within an OPEN object. This way, during PCEP session-setup phase, a PCEP speaker can advertise to a PCEP peer the Operator-configured Association Range for an association type.

The PCEP OP-CONF-ASSOC-RANGE TLV is OPTIONAL. It MAY be carried within an OPEN object sent by a PCEP speaker in an Open message to a PCEP peer. The OP-CONF-ASSOC-RANGE TLV format is compliant with the PCEP TLV format defined in [RFC5440]. That is, the TLV is composed of 2 bytes for the type, 2 bytes specifying the TLV length, and a Value field. The Length field defines the length of the value portion in bytes.

The PCEP OP-CONF-ASSOC-RANGE TLV has the following format:

TYPE: 29 (Early allocation by IANA)

LENGTH:  $N * 8$  (where N is the number of association types)

VALUE:

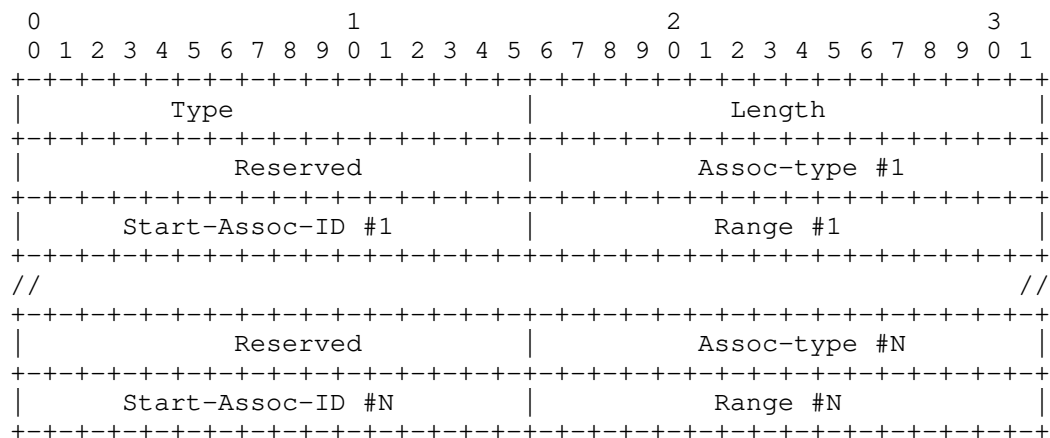


Figure 2: The OP-CONF-ASSOC-RANGE TLV format

The Value portion includes the following fields, repeated for each association type:

Reserved (2 bytes): This field MUST be set to 0 on transmission and MUST be ignored on receipt.

Assoc-type (2 bytes): The association type (Section 7.4). The association types are defined in separate documents.

Start-Assoc-ID (2 bytes): The start association identifier for the Operator-configured Association Range for the particular association type. The values 0 and 0xffff MUST NOT be used and on receipt of these values in the TLV, the session is rejected with error message sent (see Section 5.1).



Range (2 bytes): The number of associations marked for the Operator-configured Associations. The Range MUST be greater than 0, and it MUST be such that (Start-Assoc-ID + Range) do not cross the association identifier range of 0xffff. In case this condition is not satisfied, the session is rejected with error message sent (see Section 5.1).

### 5.1. Procedure

A PCEP speaker MAY include an OP-CONF-ASSOC-RANGE TLV within an OPEN object in an Open message sent to a PCEP peer in order to advertise the Operator-configured Association Range for an association type. The OP-CONF-ASSOC-RANGE TLV MUST NOT appear more than once in an OPEN object. If it appears more than once, the PCEP session MUST be rejected with error type 1 and error value 1 (PCEP session establishment failure / Reception of an invalid Open message).

As specified in [RFC5440], a PCEP peer that does not recognize the OP-CONF-ASSOC-RANGE TLV will silently ignore it.

The Operator-configured Association Range SHOULD be included for each association type that could be both dynamic and operator-configured. For association types that are only dynamic or only operator-configured, this TLV MAY be skipped, in which case the full range of association identifier is considered dynamic or operator-configured respectively. Each association type (that are defined in separate documents) can specify the default value for the operator-configured association range for their respective association type.

The absence of the OP-CONF-ASSOC-RANGE TLV in an OPEN object MUST be interpreted as an absence of explicit Operator-configured Association Range at the PCEP peer. In this case, the default behavior as per each association type applies. If the association source is not a PCEP speaker, the default value for the operator-configured association range is used for the association source.

If the Assoc-type is not recognized or supported by the PCEP speaker, it MUST ignore that respective Start-Assoc-ID and Range. If the Assoc-type is recognized/supported but the Start-Assoc-ID or Range are set incorrectly, the PCEP session MUST be rejected with error type 1 and error value 1 (PCEP session establishment failure / Reception of an invalid Open message). The incorrect range include the case when the (Start-Assoc-ID + Range) crosses the association identifier range of 0xffff.

A given Assoc-type MAY appear more than once in the OP-CONF-ASSOC-RANGE TLV in the case of a non-contiguous Operator-configured Association Range. The PCEP speaker originating this TLV MUST NOT

carry overlapping ranges for an association type. If a PCEP peer receives overlapping ranges for an association type, it MUST consider the Open message malformed and MUST reject the PCEP session with error type 1 and error value 1 (PCEP session establishment failure / Reception of an invalid Open message).

There may be cases where an operator-configured association was configured with association parameters (such as association identifier, association type and association source) at the local PCEP speaker, and later the PCEP session gets established with the association source and a new operator-configured range is learned during session establishment. At this time, the local PCEP speaker MUST remove any associations that are not in the new operator-configured range (by disassociating any LSPs that are part of it (and notifying this change to the PCEP peer)). If a PCEP speaker receives an association for an operator-configured association and the association identifier is not in the operator-configured association range for the association type and association source, it MUST generate an error (as described in Section 6.4).

## 6. ASSOCIATION Object

### 6.1. Object Definition

Association groups and their memberships are defined using a new ASSOCIATION object.

ASSOCIATION Object-Class is 40 (Early allocation by IANA).

ASSOCIATION Object-Type is 1 for IPv4 and its format is shown in Figure 3:

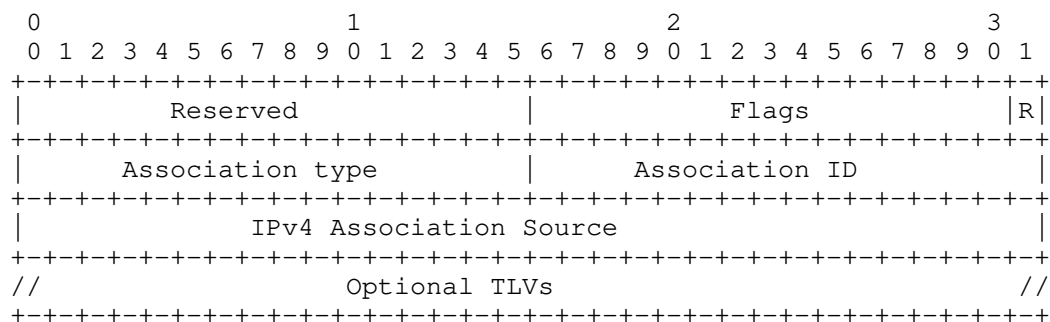


Figure 3: The IPv4 ASSOCIATION Object format

ASSOCIATION Object-Type is 2 for IPv6 and its format is shown in Figure 4:

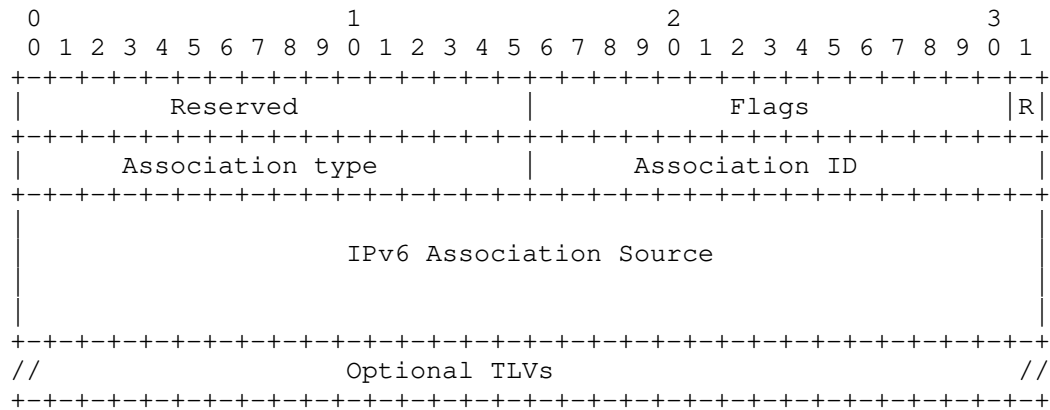


Figure 4: The IPv6 ASSOCIATION Object format

Reserved (2-byte): MUST be set to 0 and ignored upon receipt.

Flags (2-byte): The following flags are currently defined:

R (Removal - 1 bit): when set, the requesting PCEP peer requires the removal of an LSP from the association group. When unset, the PCEP peer indicates that the LSP is added or retained as part of the association group. This flag is used for the ASSOCIATION object in the PCRpt and the PCUpd message, the flag is ignored in other PCEP messages.

The unassigned flags MUST be set to zero on transmission and MUST be ignored on receipt.

Association type (2-byte): the association type (Section 7.4). The association types are defined in separate documents.

Association ID (2-byte): the identifier of the association group. When combined with other association parameters, such as Association Type and Association Source, this value uniquely identifies an association group. The values 0xffff and 0x0 are reserved. The value 0xffff is used to indicate all association groups and could be used with R flag to indicate removal for all associations for the LSP within the scope of association type and association source.

**Association Source:** Contains a valid IPv4 address (4 bytes) if the ASSOCIATION Object-Type is 1 or a valid IPv6 address (16 bytes) if the ASSOCIATION Object-Type is 2. The address provides scoping for the Association ID. See Section 6.1.3 for details.

**Optional TLVs:** The optional TLVs follow the PCEP TLV format of [RFC5440]. This document defines two optional TLVs. Other documents can define more TLVs in future.

#### 6.1.1. Global Association Source TLV

The Global Association Source TLV is an optional TLV for use in the Association Object. The meaning and the usage of Global Association Source is as per Section 4 of [RFC6780].

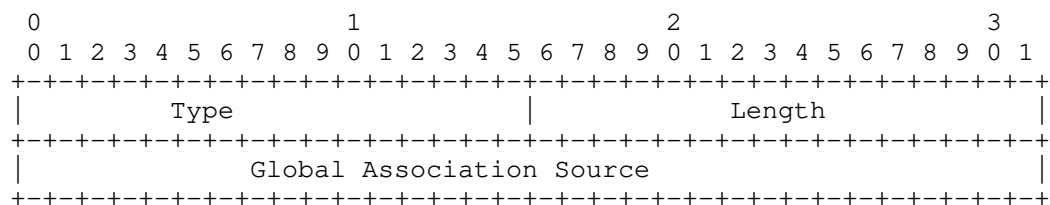


Figure 5: The Global Association Source TLV format

**Type:** 30 (Early allocation by IANA).

**Length:** Fixed value of 4 bytes.

**Global Association Source:** as defined in Section 4 of [RFC6780].

#### 6.1.2. Extended Association ID TLV

The Extended Association ID TLV is an optional TLV for use in the Association Object. The meaning and the usage of Extended Association ID is as per Section 4 of [RFC6780].

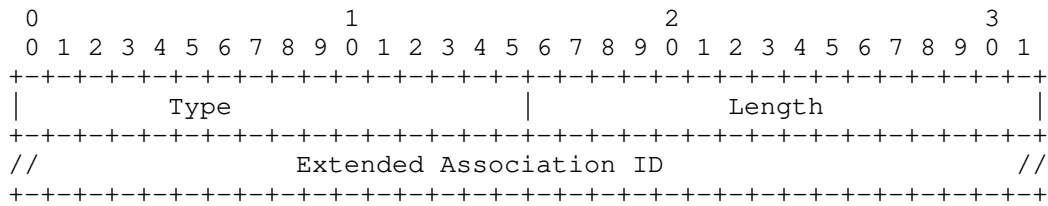


Figure 6: The Extended Association ID TLV format

Type: 31 (Early allocation by IANA).

Length: variable.

Extended Association ID: as defined in Section 4 of [RFC6780].

#### 6.1.3. Association Source

The Association Source field in the ASSOCIATION object is set to a valid IP address to identify the node that originates the association. In case of dynamic associations, the association source is usually set as the local PCEP speaker address, unless local policy dictates otherwise, in which case association source is set based on the local policy. In case of PCE redundancy, local policy could set the source as a virtual IP address which identifies all instances of the PCE. In case of operator-configured association, the association source is manually configured and it could be set as one of the PCEP speakers, Network Management System (NMS), or any other valid IP address that scopes the association identifier for the association type.

#### 6.1.4. Unique Identification for an Association Group

The combination of the mandatory fields Association type, Association ID and Association Source in the ASSOCIATION object uniquely identify the association group. If the optional TLVs - Global Association Source or Extended Association ID are included, then they MUST be included in combination with mandatory fields to uniquely identify the association group. In this document, all these fields are collectively called 'association parameters'. Note that the ASSOCIATION object MAY include other optional TLVs (not defined in this document) based on the association types, that provides 'information' related to the association type, this document uses the term 'association information' for it.

## 6.2. Relationship with the RSVP ASSOCIATION

The format of PCEP ASSOCIATION Object defined in this document is aligned with the RSVP ASSOCIATION object ([RFC6780]). Various Association types related to RSVP association are defined in [RFC4872], [RFC4873], and [RFC7551]. The PCEP extensions that define new association types, should clarify how the PCEP associations would work with RSVP associations and vice-versa.

## 6.3. Object Encoding in PCEP messages

Message formats in this document are expressed using Reduced BNF (RBNF) as used in [RFC5440] and defined in [RFC5511].

### 6.3.1. Stateful PCEP messages

The ASSOCIATION Object MAY be carried in the Path Computation Update (PCUpd), Path Computation Report (PCRpt) and Path Computation Initiate (PCInitiate) messages.

When carried in PCRpt message, it is used to report the association group membership pertaining to a LSP to a stateful PCE. The PCRpt message are used for both initial state synchronization operations (Section 5.6 of [RFC8231]) as well as whenever the state of the LSP changes. If the LSP belongs to an association group, then the associations MUST be included during the state synchronization operations.

The PCRpt message can also be used to remove an LSP from one or more association groups by setting the R flag to 1 in the ASSOCIATION object.

When an LSP is first reported to the PCE, the PCRpt message MUST include all the association groups that it belongs to. Any subsequent report message SHOULD include only the associations that are being modified or removed.

The PCRpt message is defined in [RFC8231] and updated as below:

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

Where:

```
<state-report-list> ::= <state-report> [<state-report-list>]
```

```
<state-report> ::= [<SRP>]
                    <LSP>
                    [<association-list>]
                    <path>
```

Where:

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

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

When an LSP is delegated to a stateful PCE, the stateful PCE can create a new association group for this LSP, or associate it with one or more existing association groups. This is done by including the ASSOCIATION Object in a PCUpd message. A stateful PCE can also remove a delegated LSP from one or more association groups by setting the R flag to 1 in the ASSOCIATION object.

The PCUpd message SHOULD include the association groups that are being modified or removed, there is no need to include associations that remains unchanged.

The PCUpd message is defined in [RFC8231] and updated as below:

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

Where:

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

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

Where:

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

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

Unless a PCEP speaker wants to delete an association from an LSP or make changes to the association, it does not need to carry the ASSOCIATION object in future stateful messages.

A PCE initiating a new LSP can also include the association groups that this LSP belongs to. This is done by including the ASSOCIATION Object in a PCInitiate message. The PCInitiate message MUST include all the association groups that it belongs to. The PCInitiate message is defined in [RFC8281] and updated as below:

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

Where:

```
<PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                              [<PCE-initiated-lsp-list>]
```

```
<PCE-initiated-lsp-request> ::= (<PCE-initiated-lsp-instantiation> |
                                <PCE-initiated-lsp-deletion>)
```

```
<PCE-initiated-lsp-instantiation> ::= <SRP>
                                       <LSP>
                                       [<END-POINTS>]
                                       <ERO>
                                       [<association-list>]
                                       [<attribute-list>]
```

Where:

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

### 6.3.2. Request Message

In case of passive (stateful or stateless) PCE, the ASSOCIATION Object is OPTIONAL and MAY be carried in the Path Computation Request (PCReq) message.

When carried in a PCReq message, the ASSOCIATION Object is used to associate the path computation request to an association group. The association (and the other LSPs) should be known to the PCE beforehand. These could be operator-configured or dynamically learned before via stateful PCEP messages. The R flag in ASSOCIATION object within PCReq message MUST be set to 0 while sending and ignored on receipt.

The PCReq message is defined in [RFC5440] and updated in [RFC8231], it is further updated below for association:



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

Where:

```
<svec-list> ::= <SVEC> [<svec-list>]
<request-list> ::= <request> [<request-list>]
```

```
<request> ::= <RP>
               <END-POINTS>
               [<LSP>]
               [<LSPA>]
               [<BANDWIDTH>]
               [<metric-list>]
               [<association-list>]
               [<RRO> [<BANDWIDTH>]]
               [<IRO>]
               [<LOAD-BALANCING>]
```

Where:

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

Note that the LSP object MAY be present for the passive stateful PCE mode.

#### 6.3.3. Reply Message

In case of passive (stateful or stateless) PCE, the ASSOCIATION Object is OPTIONAL and MAY be carried in the Path Computation Reply (PCRep) message with the NO-PATH object. The ASSOCIATION object in PCRep message indicates the association group that cause the PCE to fail to find a path.

The PCRep message is defined in [RFC5440] and updated in [RFC8231] , it is further updated below for association:

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

Where:

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

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

Where:

```
<association-list> ::= <ASSOCIATION> [<association-list>]
```

Note that the LSP object MAY be present for the passive stateful PCE mode.

#### 6.4. Processing Rules

Association groups can be operator-configured on the necessary PCEP speakers and the PCEP speakers can join the existing association groups. In addition, a PCC or a PCE can create association groups dynamically and the PCEP speaker can also report the associations to its peer via PCEP messages. The operator-configured associations are created via configurations (where all association parameters are manually set) and exist until explicitly removed via configurations. The PCEP speaker can add LSPs to these configured associations and carry this via stateful PCEP messages. The dynamic associations are created dynamically by the PCEP speaker (where all association parameters are populated dynamically). The association group is attached to the LSP state, and the association group exists till there is at least one LSP as part of the association. As described in Section 6.1.4, the association parameters are the combination of Association type, Association ID and Association Source as well as optional global source and extended association identifier, that uniquely identifies an association group. The information related to the association types encoded via the TLVs of a particular association type (not described in this document) are the association information (Section 6.1.4).

If a PCEP speaker does not recognize the ASSOCIATION object in the stateful message, it will return a PCErr message with Error-Type "Unknown Object" as described in [RFC5440]. In case of PCReq message, the PCE would react based on the P flag as per [RFC5440]. If a PCEP speaker understands the ASSOCIATION object but does not support the Association type, it MUST return a PCErr message with

Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 1 "Association type is not supported". If any association parameters are invalid in the ASSOCIATION object, the PCEP speaker would consider this as malformed object and handle it as malformed message [RFC5440]. On receiving a PCEP message with ASSOCIATION, if a PCEP speaker finds that too many LSPs belong to the association group, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 2 "Too many LSPs in the association group". If a PCEP speaker cannot handle a new association, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 3 "Too many association groups". These numbers MAY be set by operator or decided based on a local policy.

If a PCE peer is unwilling or unable to process the ASSOCIATION object in the stateful message, it MUST return a PCErr message with the Error-Type "Not supported object" and follow the relevant procedures described in [RFC5440]. In case of PCReq message, the PCE would react based on the P flag as per [RFC5440]. On receiving a PCEP message with ASSOCIATION, if a PCEP speaker could not add the LSP to the association group for any reason, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 7 "Cannot join the association group".

If a PCEP speaker receives an ASSOCIATION object for an operator-configured association and the association identifier is not in the operator-configured association range for the Association type and Association Source, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 8 "Association identifier not in range".

If a PCEP speaker receives ASSOCIATION in PCReq message, and the association is not known (association is not configured, or created dynamically, or learned from a PCEP peer), it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 4 "Association unknown".

If the association information (related to the association group as a whole) received from the peer does not match with the local operator-configured information, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 5 "Operator-configured association information mismatch". On receiving association information (related to the association group as a whole) that does not match with the association information previously received about the same association from a peer, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 6 "Association information mismatch". Note that information related to each LSP within the

association as part of the association information TLVs could be different.

If a PCEP speaker receives an ASSOCIATION object with the R bit set for removal, and the association group (identified by association parameters) is not known, it MUST return a PCErr message with Error-Type 26 (Early allocation by IANA) "Association Error" and Error-Value 4 "Association unknown".

The dynamic associations are cleared along with the LSP state information as per the [RFC8231]. When a PCEP session is terminated, after expiry of State Timeout Interval at PCC, the LSP state associated with that PCEP session is reverted to operator-defined default parameters or behaviours. Same procedure is also followed for the association groups. On session termination at the PCE, when the LSP state reported by PCC is cleared, the association groups are also cleared. When there are no LSPs in an association group, the association is considered to be empty and thus deleted.

In case the LSP is delegated to another PCE on session failure, the associations (and association information) set by the PCE remains intact, unless updated by the new PCE that takes over.

Upon LSP delegation revocation, the PCC MAY clear the association created by the PCE, but in order to avoid traffic loss, it SHOULD perform this in a make-before-break fashion (same as [RFC8231]).

## 7. IANA Considerations

IANA maintains the "Path Computation Element Protocol (PCEP) Numbers" registry at <<http://www.iana.org/assignments/pcep>>.

### 7.1. PCEP Object

The "PCEP Numbers" registry contains a subregistry "PCEP Objects". IANA is requested to confirm the early allocation of the following code point in the PCEP Objects registry.

Object-Class Value	Name	Reference
40 (Early allocation by IANA)	Association	[This.I-D]
	Object-Type	
	0: Reserved	
	1: IPv4	
	2: IPv6	

## 7.2. PCEP TLV

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

Value	Meaning	Reference
29 (Early allocation by IANA)	Operator-configured Association Range	[This.I-D]
30 (Early allocation by IANA)	Global Association Source	[This.I-D]
31 (Early allocation by IANA)	Extended Association ID	[This.I-D]

IANA is requested to fix the meaning for value 31 in the above registry to 'Extended Association ID', it is currently mentioned as 'Extended Association Id'.

IANA is also requested to make a new assignment for the existing "PCEP TLV Type Indicators" registry as follows:

Value	Meaning	Reference
TBD	ASSOC-Type-List	[This.I-D]

## 7.3. Association Flags

This document requests IANA to create a subregistry of the "PCEP Numbers" for the bits carried in the Flags field of the ASSOCIATION object. The subregistry is called "ASSOCIATION Flags Field". New values are assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

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

Bit	Description	Reference
15	R (Removal)	[This.I-D]

#### 7.4. Association Type

This document requests IANA to create a subregistry of the "PCEP Numbers" for the Association Type field of the the ASSOCIATION object. The subregistry is called "ASSOCIATION Type Field". New values are to be assigned by Standards Action [RFC8126]. Each value should be tracked with the following qualities:

- o Type
- o Name
- o Reference

There are no association types specified in this document, future documents should request the assignment of association types from this subregistry.

#### 7.5. PCEP-Error Object

IANA is requested to confirm the early allocation of the following code points within the "PCEP-ERROR Object Error Types and Values" sub-registry of the "PCEP Numbers" registry, as follows:

Error-Type	Meaning
26	Association Error [This.I-D]
(early	Error-value=0:
alloc by	Unassigned
IANA)	Error-value=1:
	Association type is not supported
	Error-value=2:
	Too many LSPs in the association group
	Error-value=3:
	Too many association groups
	Error-value=4:
	Association unknown
	Error-value=5:
	Operator-configured association
	information mismatch
	Error-value=6:
	Association information mismatch
	Error-value=7:
	Cannot join the association group
	Error-value=8:
	Association identifier not in range

## 8. Security Considerations

The security considerations described in [RFC8231] and [RFC5440] apply to the extensions described in this document as well. Additional considerations related to a malicious PCEP speaker are introduced, as associations could be spoofed and could be used as an attack vector. An attacker could attempt to create too many associations in an attempt to load the PCEP peer. The PCEP peer responds with PCErr as described in Section 6.4. An attacker could impact LSP operations by creating bogus associations. Further, association groups could provides an adversary with the opportunity to eavesdrop on the relationship between the LSPs. Thus securing the PCEP session using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in [RFC7525], is RECOMMENDED.

Much of the information carried in the ASSOCIATION object, as per this document is not extra sensitive. It often reflects information that can also be derived from the LSP Database, but association provides a much easier grouping of related LSPs and messages. Implementations and operator can and should use indirect values in ASSOCIATION as a way to hide any sensitive business relationships.

## 9. Manageability Considerations

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

### 9.1. Control of Function and Policy

A PCE or PCC implementation MUST allow operator-configured associations and SHOULD allow setting of the operator-configured association range (Section 3.4) as described in this document.

### 9.2. Information and Data Models

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

An implementation SHOULD allow the operator to view the associations configured or created dynamically. Further implementation SHOULD allow to view associations reported by each peer, and the current set of LSPs in the association.

It might also be useful to find out how many associations for each association type currently exist and to know how many free association identifiers are available for a particular association type and source.

### 9.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

### 9.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

### 9.5. Requirements on Other Protocols

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

### 9.6. Impact on Network Operations

Mechanisms defined in [RFC5440] and [RFC8231] also apply to PCEP extensions defined in this document.

## 10. Acknowledgments

We would like to thank Yuji Kamite and Joshua George for their contributions to this document. Also thanks to Venugopal Reddy, Cyril Margaria, Rakesh Gandhi, Mike Koldychev, and Adrian Farrel for their useful comments.

We would like to thank Julien Meuric for shepherding this document and providing comments with text suggestions.

Thanks to Stig Venaas for the RTGDIR review comments.

Thanks to Alvaro Retana, Mirja Kuhlewind, Martin Vigoureux, Barry Leiba, Eric Vyncke, Suresh Krishnan, and Benjamin Kaduk for the IESG comments.

## 11. Contributors



Stephane Litkowski  
Orange

Email: [stephane.litkowski@orange.com](mailto:stephane.litkowski@orange.com)

Xian Zhang  
Huawei Technologies  
F3-1-B RnD Center, Huawei Base  
Bantian, Longgang District  
Shenzhen 518129  
China

Email: [zhang.xian@huawei.com](mailto:zhang.xian@huawei.com)

Mustapha Aissaoui  
Nokia

Email: [mustapha.aissaoui@nokia.com](mailto:mustapha.aissaoui@nokia.com)

## 12. References

### 12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [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>>.
- [RFC6780] Berger, L., Le Faucheur, F., and A. Narayanan, "RSVP ASSOCIATION Object Extensions", RFC 6780, DOI 10.17487/RFC6780, October 2012, <<https://www.rfc-editor.org/info/rfc6780>>.

- [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>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.

## 12.2. Informative References

- [RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <<https://www.rfc-editor.org/info/rfc4657>>.
- [RFC4872] Lang, J., Ed., Rekhter, Y., Ed., and D. Papadimitriou, Ed., "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, DOI 10.17487/RFC4872, May 2007, <<https://www.rfc-editor.org/info/rfc4872>>.
- [RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel, "GMPLS Segment Recovery", RFC 4873, DOI 10.17487/RFC4873, May 2007, <<https://www.rfc-editor.org/info/rfc4873>>.

- [RFC6007] Nishioka, I. and D. King, "Use of the Synchronization VECtor (SVEC) List for Synchronized Dependent Path Computations", RFC 6007, DOI 10.17487/RFC6007, September 2010, <<https://www.rfc-editor.org/info/rfc6007>>.
- [RFC7551] Zhang, F., Ed., Jing, R., and R. Gandhi, Ed., "RSVP-TE Extensions for Associated Bidirectional Label Switched Paths (LSPs)", RFC 7551, DOI 10.17487/RFC7551, May 2015, <<https://www.rfc-editor.org/info/rfc7551>>.
- [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>>.
- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-12 (work in progress), July 2019.
- [I-D.ietf-pce-stateful-path-protection] Ananthakrishnan, H., Sivabalan, S., Barth, C., Minei, I., and M. Negi, "PCEP Extensions for Associating Working and Protection LSPs with Stateful PCE", draft-ietf-pce-stateful-path-protection-07 (work in progress), June 2019.
- [I-D.ietf-pce-association-diversity] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for LSP Diversity Constraint Signaling", draft-ietf-pce-association-diversity-08 (work in progress), July 2019.

#### Appendix A. Example for Operator-configured Association Range

Consider an association type T1 (which allows both dynamic and operator-configured association with a default range of <0x1000, 0xffff>). Consider that, because of need of the network, the PCE needs to create more dynamic associations and would like to change the association range to <0xbffe, 0xffff> instead. During PCEP session establishment the PCE would advertise the new range, the PCC could skip advertising as the default values are used. If a PCC is creating a dynamic association (with PCC as association source) it needs to pick a free association identifier for type T1 in the range of <0x1, 0x0fff> whereas if a PCE is creating a dynamic association (with PCE as association source) it needs to pick a free association

identifier from the range <0x1, 0xbffd>. Similarly if an operator-configured association is manually configured with the PCC as association source, it should be from the range <0x1000, 0xffff> whereas if the PCE is association source, it should be from <0xbffe, 0xffff>. In case the association source is not a PCEP peer (for example an NMS system), then the default range of <0x1000, 0xffff> is considered.

#### Authors' Addresses

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

Email: inaminei@google.com

Edward Crabbe  
Individual Contributor

Email: edward.crabbe@gmail.com

Siva Sivabalan  
Cisco Systems, Inc.  
170 West Tasman Dr.  
San Jose, CA 95134  
US

Email: msiva@cisco.com

Hariharan Ananthakrishnan  
Netflix

Email: hari@netflix.com

Dhruv Dhody  
Huawei  
Divyashree Techno Park, Whitefield  
Bangalore, KA 560066  
India

Email: dhruv.ietf@gmail.com

Yosuke Tanaka  
NTT Communications Corporation  
Granpark Tower 3-4-1 Shibaura, Minato-ku  
Tokyo 108-8118  
Japan

Email: [yosuke.tanaka@ntt.com](mailto:yosuke.tanaka@ntt.com)

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: May 3, 2018

U. Palle  
D. Dhody  
Huawei Technologies  
Y. Tanaka  
NTT Communications  
V. Beeram  
Juniper Networks  
October 30, 2017

Path Computation Element (PCE) Protocol Extensions for Stateful PCE  
usage for Point-to-Multipoint Traffic Engineering Label Switched Paths  
draft-ietf-pce-stateful-pce-p2mp-05

## Abstract

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 provides extensions required for Path Computation Element communication Protocol (PCEP) so as to enable the usage of a stateful PCE capability in supporting P2MP TE LSPs.

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 3, 2018.

## Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Requirements Language . . . . .	4
2. Terminology . . . . .	4
3. Supporting P2MP TE LSP for Stateful PCE . . . . .	4
3.1. Motivation . . . . .	4
3.2. Objectives . . . . .	5
4. Functions to Support P2MP TE LSPs for Stateful PCEs . . . . .	5
5. Architectural Overview of Protocol Extensions . . . . .	6
5.1. Extension of PCEP Messages . . . . .	6
5.2. Capability Advertisement . . . . .	6
5.3. IGP Extensions for Stateful PCE P2MP Capabilities Advertisement . . . . .	7
5.4. State Synchronization . . . . .	8
5.5. LSP Delegation . . . . .	8
5.6. LSP Operations . . . . .	8
5.6.1. Passive Stateful PCE . . . . .	8
5.6.2. Active Stateful PCE . . . . .	9
5.6.3. PCE-Initiated LSP . . . . .	9
5.6.3.1. P2MP TE LSP Instantiation . . . . .	9
5.6.3.2. P2MP TE LSP Deletion . . . . .	9
5.6.3.3. Adding and Pruning Leaves for the P2MP TE LSP . .	10
5.6.3.4. P2MP TE LSP Delegation and Cleanup . . . . .	10
6. PCEP Message Extensions . . . . .	10
6.1. The PCRpt Message . . . . .	10
6.2. The PCUpd Message . . . . .	12
6.3. The PCReq Message . . . . .	13
6.4. The PCRep Message . . . . .	14
6.5. The PCInitiate message . . . . .	15
6.6. Example . . . . .	16
6.6.1. P2MP TE LSP Update Request . . . . .	17
6.6.2. P2MP TE LSP Report . . . . .	17
7. PCEP Object Extensions . . . . .	18
7.1. Extension of LSP Object . . . . .	18
7.2. P2MP-LSP-IDENTIFIER TLV . . . . .	19
7.3. S2LS Object . . . . .	21
8. Message Fragmentation . . . . .	22
8.1. Report Fragmentation Procedure . . . . .	22
8.2. Update Fragmentation Procedure . . . . .	23
8.3. PCInitiate Fragmentation Procedure . . . . .	23

9. Non-Support of P2MP TE LSPs for Stateful PCE . . . . .	23
10. Manageability Considerations . . . . .	24
10.1. Control of Function and Policy . . . . .	24
10.2. Information and Data Models . . . . .	24
10.3. Liveness Detection and Monitoring . . . . .	25
10.4. Verify Correct Operations . . . . .	25
10.5. Requirements On Other Protocols . . . . .	25
10.6. Impact On Network Operations . . . . .	25
11. IANA Considerations . . . . .	25
11.1. PCE Capabilities in IGP Advertisements . . . . .	25
11.2. STATEFUL-PCE-CAPABILITY TLV . . . . .	26
11.3. LSP Object . . . . .	26
11.4. PCEP-Error Object . . . . .	26
11.5. PCEP TLV Type Indicators . . . . .	27
11.6. PCEP object . . . . .	27
11.7. S2LS object . . . . .	28
12. Security Considerations . . . . .	28
13. Acknowledgments . . . . .	29
14. References . . . . .	29
14.1. Normative References . . . . .	29
14.2. Informative References . . . . .	30
Appendix A. Contributor Addresses . . . . .	32
Authors' Addresses . . . . .	32

## 1. Introduction

As per [RFC4655], the Path Computation Element (PCE) 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.

[RFC4857] 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 PCEP is designed as a communication protocol between PCCs and PCEs for point-to-point (P2P) path computations and is defined in [RFC5440]. The extensions of PCEP to request path computation for P2MP TE LSPs are described in [I-D.ietf-pce-rfc6006bis].

Stateful PCEs are shown to be helpful in many application scenarios, in both MPLS and GMPLS networks, as illustrated in [RFC8051]. These scenarios apply equally to P2P and P2MP TE LSPs. [RFC8231] provides the fundamental extensions needed for stateful PCE to support general functionality for P2P TE LSP. [I-D.ietf-pce-pce-initiated-lsp]



provides the an extensions needed for stateful PCE-initiated P2P TE LSP. Complementarily, this document focuses on the extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs. This document describes the setup, maintenance and teardown of PCE-initiated P2MP LSPs under the stateful PCE model.

### 1.1. Requirements Language

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

## 2. Terminology

Terminology used in this document is same as terminology used in [RFC8231], [I-D.ietf-pce-pce-initiated-lsp], and [I-D.ietf-pce-rfc6006bis].

## 3. Supporting P2MP TE LSP for Stateful PCE

### 3.1. Motivation

[RFC8051] presents several use cases, demonstrating scenarios that benefit from the deployment of a stateful PCE including optimization, recovery, etc which are equally applicable to P2MP TE LSPs. [RFC8231] defines the extensions to PCEP for P2P TE LSPs. Complementarily, this document focuses on the extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs.

In addition to that, the stateful nature of a PCE simplifies the information conveyed in PCEP messages since it is possible to refer to the LSPs via PLSP-ID ([RFC8231]). For P2MP this is an added advantage, where the size of message is much larger. In case of stateless PCE, a modification of P2MP tree requires encoding of all leaves along with the paths in PCReq message, but using a stateful PCE with P2MP capability, the PCEP message can be used to convey only the modifications (the other information can be retrieved from the P2MP LSP identifier in the LSP database (LSPDB)).

In environments where the P2MP TE LSP placement needs to change in response to application demands, it is useful to support dynamic creation and tear down of P2MP TE LSPs. The ability for a PCE to trigger the creation of P2MP TE LSPs on demand can be seamlessly integrated into a controller-based network architecture, where intelligence in the controller can determine when and where to set up

paths. Section 3 of [I-D.ietf-pce-pce-initiated-lsp] further describes the motivation behind the PCE-Initiation capability, which are equally applicable for P2MP TE LSPs.

### 3.2. Objectives

The objectives for the protocol extensions to support P2MP TE LSP for stateful PCE are same as the objectives described in section 3.2 of [RFC8231].

## 4. Functions to Support P2MP TE LSPs for Stateful PCEs

[RFC8231] specifies new functions to support a stateful PCE. It also specifies that a function can be initiated either from a PCC towards a PCE (C-E) or from a PCE towards a PCC (E-C).

This document extends these functions to support P2MP TE LSPs.

**Capability Advertisement (E-C,C-E):** both the PCC and the PCE must announce during PCEP session establishment that they support PCEP Stateful PCE extensions for P2MP using mechanisms defined in Section 5.2.

**LSP State Synchronization (C-E):** after the session between the PCC and a stateful PCE with P2MP capability is initialized, the PCE must learn the state of a PCC's P2MP TE LSPs before it can perform path computations or update LSP attributes in a PCC.

**LSP Update Request (E-C):** a stateful PCE with P2MP capability requests modification of attributes on a PCC's P2MP TE LSP.

**LSP State Report (C-E):** a PCC sends an LSP state report to a PCE whenever the state of a P2MP TE LSP changes.

**LSP Control Delegation (C-E,E-C):** a PCC grants to a PCE the right to update LSP attributes on one or more P2MP TE LSPs; the PCE becomes the authoritative source of the LSP's attributes as long as the delegation is in effect (See Section 5.7 of [RFC8231]); the PCC may withdraw the delegation or the PCE may give up the delegation at any time.

**PCE-initiated LSP instantiation (E-C):** a PCE sends an LSP Initiate Message to a PCC to instantiate or delete a P2MP TE LSP.

## 5. Architectural Overview of Protocol Extensions

### 5.1. Extension of PCEP Messages

New PCEP messages are defined in [RFC8231] to support stateful PCE for P2P TE LSPs. In this document these messages are extended to support P2MP TE LSPs.

**Path Computation State Report (PCRpt):** Each P2MP TE LSP State Report in a PCRpt message can contain actual P2MP TE LSP path attributes, LSP status, etc. An LSP State Report carried on a PCRpt message is also used in delegation or revocation of control of a P2MP TE LSP to/from a PCE. The extension of PCRpt message is described in Section 6.1.

**Path Computation Update Request (PCUpd):** Each P2MP TE LSP Update Request in a PCUpd message MUST contain all LSP parameters that a PCE wishes to set for a given P2MP TE LSP. An LSP Update Request carried on a PCUpd message is also used to return LSP delegations if at any point PCE no longer desires control of a P2MP TE LSP. The PCUpd message is described in Section 6.2.

A new PCEP message is defined in [I-D.ietf-pce-pce-initiated-lsp] to support stateful PCE instantiation of P2P TE LSPs. In this document this message is extended to support P2MP TE LSPs.

**Path Computation LSP Initiate Message (PCInitiate):** is a PCEP message sent by a PCE to a PCC to trigger P2MP TE LSP instantiation or deletion. The PCInitiate message is described in Section 6.5.

The path computation request (PCReq) and path computation reply (PCRep) messages are also extended to support stateful PCE for P2P TE LSP in [RFC8231]. In this document these messages are extended to support P2MP TE LSPs as well.

### 5.2. Capability Advertisement

During PCEP Initialization Phase, as per Section 7.1.1 of [RFC8231], PCEP speakers advertises Stateful capability via Stateful PCE Capability TLV in open message. Two new flags are defined for the STATEFUL-PCE-CAPABILITY TLV defined in [RFC8231] and updated in [I-D.ietf-pce-pce-initiated-lsp] and [RFC8232].

Three new bits N (P2MP-CAPABILITY), M (P2MP-LSP-UPDATE-CAPABILITY), and P (P2MP-LSP-INSTANTIATION-CAPABILITY) are added in this document:

N (P2MP-CAPABILITY bit - TBD4): if set to 1 by a PCC, the N Flag indicates that the PCC is willing to send P2MP LSP State Reports whenever P2MP LSP parameters or operational status changes.; if set to 1 by a PCE, the N Flag indicates that the PCE is interested in receiving LSP State Reports whenever LSP parameters or operational status changes. The P2MP-CAPABILITY Flag must be advertised by both a PCC and a PCE for PCRpt messages P2MP extension to be allowed on a PCEP session.

M (P2MP-LSP-UPDATE-CAPABILITY bit - TBD5): if set to 1 by a PCC, the M Flag indicates that the PCC allows modification of P2MP LSP parameters; if set to 1 by a PCE, the M Flag indicates that the PCE is capable of updating P2MP LSP parameters. The P2MP-LSP-UPDATE-CAPABILITY Flag must be advertised by both a PCC and a PCE for PCUpd messages P2MP extension to be allowed on a PCEP session.

P (P2MP-LSP-INSTANTIATION-CAPABILITY bit - TBD6): If set to 1 by a PCC, the P Flag indicates that the PCC allows instantiation of an P2MP LSP by a PCE. If set to 1 by a PCE, the P flag indicates that the PCE supports P2MP LSP instantiation. The P2MP-LSP-INSTANTIATION-CAPABILITY flag must be set by both PCC and PCE in order to support PCE-initiated P2MP LSP instantiation.

A PCEP speaker should continue to advertise the basic P2MP capability via mechanisms as described in [I-D.ietf-pce-rfc6006bis].

### 5.3. IGP Extensions for Stateful PCE P2MP Capabilities Advertisement

When PCCs are LSRs participating in the IGP (OSPF or IS-IS), and PCEs are either LSRs or servers also participating in the IGP, an effective mechanism for PCE discovery within an IGP routing domain consists of utilizing IGP advertisements. Extensions for the advertisement of PCE Discovery Information are defined for OSPF and for IS-IS in [RFC5088] and [RFC5089] respectively.

The PCE-CAP-FLAGS sub-TLV, defined in [RFC5089], is an optional sub-TLV used to advertise PCE capabilities. It MAY be present within the PCED sub-TLV carried by OSPF or IS-IS. [RFC5088] and [RFC5089] provide the description and processing rules for this sub-TLV when carried within OSPF and IS-IS, respectively.

The format of the PCE-CAP-FLAGS sub-TLV is included below for easy reference:

Type: 5

Length: Multiple of 4.

Value: This contains an array of units of 32 bit flags with the most significant bit as 0. Each bit represents one PCE capability.

PCE capability bits are defined in [RFC5088]. This document defines new capability bits for the stateful PCE with P2MP as follows:

Bit	Capability
TBD1	Active Stateful PCE with P2MP
TBD2	Passive Stateful PCE with P2MP
TBD3	PCE-Initiation with P2MP

Note that while active, passive or initiation stateful PCE with P2MP capabilities may be advertised during discovery, PCEP Speakers that wish to use stateful PCEP MUST advertise stateful PCEP capabilities during PCEP session setup, as specified in the current document. A PCC MAY initiate stateful PCEP P2MP capability advertisement at PCEP session setup even if it did not receive any IGP PCE capability advertisements.

#### 5.4. State Synchronization

State Synchronization operations described in Section 5.6 of [RFC8231] are applicable for P2MP TE LSPs as well. The optimizations described in [RFC8232] can also be applied for P2MP.

#### 5.5. LSP Delegation

LSP delegation operations described in Section 5.7 of [RFC8231] are applicable for P2MP TE LSPs as well.

#### 5.6. LSP Operations

##### 5.6.1. Passive Stateful PCE

LSP operations for passive stateful PCE described in Section 5.8.1 of [RFC8231] are applicable for P2MP TE LSPs as well.

The Path Computation Request and Response message format for P2MP TE LSPs is described in Section 3.4 and Section 3.5 of [I-D.ietf-pce-rfc6006bis] respectively.

The Request and Response message for P2MP TE LSPs are extended to support encoding of LSP object, so that it is possible to refer to a LSP with a unique identifier and simplify the PCEP message exchange. For example, in case of modification of one leaf in a P2MP tree, there should be no need to carry the full P2MP tree in PCReq message.

The extension for the Request and Response message for passive stateful operations on P2MP TE LSPs are described in Section 6.3 and Section 6.4. The extension for the Path Computation LSP State Report (PCRpt) message is described in Section 6.1.

#### 5.6.2. Active Stateful PCE

LSP operations for active stateful PCE described in Section 5.8.2 of [RFC8231] are applicable for P2MP TE LSPs as well.

The extension for the Path Computation LSP Update (PCUpd) message for active stateful operations on P2MP TE LSPs are described in Section 6.2.

#### 5.6.3. PCE-Initiated LSP

As per section 5.1 of [I-D.ietf-pce-pce-initiated-lsp], the PCE sends a Path Computation LSP Initiate Request (PCInitiate) message to the PCC to suggest instantiation or deletion of a P2P TE LSP. This document extends the PCInitiate message to support P2MP TE LSP (see details in Section 6.5).

P2MP TE LSP suggested instantiation and deletion operations are same as P2P LSP as described in section 5.3 and 5.4 of [I-D.ietf-pce-pce-initiated-lsp].

##### 5.6.3.1. P2MP TE LSP Instantiation

The Instantiation operation of P2MP TE LSP is same as defined in section 5.3 of [I-D.ietf-pce-pce-initiated-lsp] including handling of PLSP-ID, SYMBOLIC-PATH-NAME TLV etc. Rules of processing and error codes remains unchanged. The N bit MUST be set in LSP object in PCInitiate message by PCE to specify the instantiation is for P2MP TE LSP.

Though N bit is set in the LSP object, P2MP-LSP-IDENTIFIER TLV MUST NOT be included in the LSP object in PCInitiate message as it SHOULD be generated by PCC and carried in PCRpt message.

##### 5.6.3.2. P2MP TE LSP Deletion

The deletion operation of P2MP TE LSP is same as defined in section 5.4 of [I-D.ietf-pce-pce-initiated-lsp] by sending an LSP Initiate Message with an LSP object carrying the PLSP-ID of the LSP to be removed and an SRP object with the R flag set (LSP-REMOVE as per section 5.2 of [I-D.ietf-pce-pce-initiated-lsp]). Rules of processing and error codes remains unchanged.

#### 5.6.3.3. Adding and Pruning Leaves for the P2MP TE LSP

Adding of new leaves and Pruning of old Leaves for the PCE initiated P2MP TE LSP MUST be carried in PCUpd message as per Section 6.2 for P2MP TE LSP extensions. As defined in [I-D.ietf-pce-rfc6006bis], leaf type = 1 for adding of new leaves, leaf type = 2 for pruning of old leaves of P2MP END-POINTS Object are used in PCUpd message.

PCC MAY use the Incremental State Update mechanism as described in [RFC4875] to signal adding and pruning of leaves.

#### 5.6.3.4. P2MP TE LSP Delegation and Cleanup

P2MP TE LSP delegation and cleanup operations are same as defined in section 6 of [I-D.ietf-pce-pce-initiated-lsp]. Rules of processing and error codes remains unchanged.

### 6. PCEP Message Extensions

#### 6.1. The PCRpt Message

As per Section 6.1 of [RFC8231], PCRpt message is used to report the current state of a P2P TE LSP. This document extends the PCRpt message in reporting the status of P2MP TE LSP.

The format of PCRpt message is as follows:

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

Where:

<state-report-list> ::= <state-report>  
                            [<state-report-list>]

<state-report> ::= [<SRP>]  
                            <LSP>  
                            <end-point-intended-path-pair-list>  
                            [<actual-attribute-list>  
                            <end-point-actual-path-pair-list>]  
                            <intended-attribute-list>

Where:

<end-point-intended-path-pair-list> ::=  
                            [<END-POINTS>]  
                            [<S2LS>]  
                            <intended-path>  
                            [<end-point-intended-path-pair-list>]

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

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

<actual-path> ::= (<RRO>|<SRRO>)  
                            [<actual-path>]

<intended-attribute-list> is defined in [RFC5440] and  
extended by PCEP extensions.

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

The P2MP END-POINTS object defined in [I-D.ietf-pce-rfc6006bis] is  
mandatory for specifying address of P2MP leaves grouped based on leaf  
types.

- o New leaves to add (leaf type = 1)
- o Old leaves to remove (leaf type = 2)



- o Old leaves whose path can be modified/reoptimized (leaf type = 3)
- o Old leaves whose path must be left unchanged (leaf type = 4)

When reporting the status of a P2MP TE LSP, the destinations are grouped in END-POINTS object based on the operational status (O field in S2LS object) and leaf type (in END-POINTS). This way the leaves that share the same operational status are grouped together. For reporting the status of delegated P2MP TE LSP, leaf-type = 3, where as for non-delegated P2MP TE LSP, leaf-type = 4 is used.

For delegated P2MP TE LSP configuration changes are reported via PCRpt message. For example, adding of new leaves END-POINTS (leaf-type = 1) is used where as removing of old leaves (leaf-type = 2) is used.

Note that we preserve compatibility with the [RFC8231] definition of <state-report>. At least one instance of <END-POINTS> MUST be present in this message for P2MP LSP.

During state synchronization, the PCRpt message must report the status of the full P2MP TE LSP.

The S2LS object MUST be carried in PCRpt message along with END-POINTS object when N bit is set in LSP object for P2MP TE LSP. If the S2LS object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=TBD11 (S2LS object missing). If the END-POINTS object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=3 (END-POINTS object missing) (defined in [RFC5440]).

## 6.2. The PCUpd Message

As per Section 6.2 of [RFC8231], PCUpd message is used to update P2P TE LSP attributes. This document extends the PCUpd message in updating the attributes of P2MP TE LSP.

The format of a PCUpd message is as follows:

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

Where:

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

```
<update-request> ::= <SRP>
                     <LSP>
                     <end-point-path-pair-list>
                     <attribute-list>
```

Where:

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

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

<attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

Note that we preserve compatibility with the [RFC8231] definition of <update-request>.

The PCC MAY use the make-before-break or sub-group-based procedures described in [RFC4875] based on a local policy decision.

The END-POINTS object MUST be carried in PCUpd message when N bit is set in LSP object for P2MP TE LSP. If the END-POINTS object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=3 (END-POINTS object missing) (defined in [RFC5440]).

### 6.3. The PCReq Message

As per Section 3.4 of [I-D.ietf-pce-rfc6006bis], PCReq message is used for a P2MP path computation request. This document extends the PCReq message such that a PCC MAY include the LSP object in the PCReq message if the stateful PCE P2MP capability has been negotiated on a PCEP session between the PCC and a PCE.

The format of PCReq message is as follows:

```

<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>
             [<LSP>]
             [<OF>]
             [<LSPA>]
             [<BANDWIDTH>]
             [<metric-list>]
             [<IRO>|<BNC>]
             [<LOAD-BALANCING>]

<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>]

```

#### 6.4. The PCRep Message

As per Section 3.5 of [I-D.ietf-pce-rfc6006bis], PCRep message is used for a P2MP path computation reply. This document extends the PCRep message such that a PCE MAY include the LSP object in the PCRep message if the stateful PCE P2MP capability has been negotiated on a PCEP session between the PCC and a PCE.

The format of PCRep message is as follows:

```

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

where:

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

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

<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>]

```

#### 6.5. The PCInitiate message

As defined in section 5.1 of [I-D.ietf-pce-pce-initiated-lsp], PCE sends a PCInitiate message to a PCC to recommend instantiation of a P2P TE LSP, this document extends the format of PCInitiate message for the creation of P2MP TE LSPs but the creation and deletion operations of P2MP TE LSP are same to the P2P TE LSP.

The format of PCInitiate message is as follows:

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

Where:

```
<PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                              [<PCE-initiated-lsp-list>]
```

```
<PCE-initiated-lsp-request> ::=
(<PCE-initiated-lsp-instantiation> | <PCE-initiated-lsp-deletion>)
```

```
<PCE-initiated-lsp-instantiation> ::= <SRP>
                                       <LSP>
                                       <end-point-path-pair-list>
                                       [<attribute-list>]
```

```
<PCE-initiated-lsp-deletion> ::= <SRP>
                                   <LSP>
```

Where:

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

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

<attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

The PCInitiate message with an LSP object with N bit (P2MP) set is used to convey operation on a P2MP TE LSP. The SRP object is used to correlate between initiation requests sent by the PCE and the error reports and state reports sent by the PCC as described in [RFC8231].

The END-POINTS object MUST be carried in PCInitiate message when N bit is set in LSP object for P2MP TE LSP. If the END-POINTS object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=3 (END-POINTS object missing) (defined in [RFC5440]).

## 6.6. Example

#### 6.6.1. P2MP TE LSP Update Request

LSP Update Request message is sent by an active stateful PCE to update the P2MP TE LSP parameters or attributes. An example of a PCUpd message for P2MP TE LSP is described below:

```
Common Header
SRP
LSP with P2MP flag set
END-POINTS for leaf type 3
ERO list
```

In this example, a stateful PCE request updation of path taken by some of the leaves in a P2MP tree. The update request uses the END-POINT type 3 (modified/reoptimized). The ERO list represents the S2LS path after modification. The update message does not need to encode the full P2MP tree in this case.

#### 6.6.2. P2MP TE LSP Report

LSP State Report message is sent by a PCC to report or delegate the P2MP TE LSP. An example of a PCRpt message for a delegated P2MP TE LSP is described below to add new leaves to an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 1
S2LS (O=DOWN)
ERO list (empty)
```

An example of a PCRpt message for P2MP TE LSP is described below to prune leaves from an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 2
S2LS (O=UP)
ERO list
```

An example of a PCRpt message for a delegated P2MP TE LSP is described below to report status of leaves in an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 3
  S2LS (O=UP)
  ERO list
END-POINTS for leaf type 3
  S2LS (O=DOWN)
  ERO list
```

An example of a PCRpt message for a non-delegated P2MP TE LSP is described below to report status of leaves:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 4
  S2LS (O=ACTIVE)
  ERO list
END-POINTS for leaf type 4
  S2LS (O=DOWN)
  ERO list
```

## 7. PCEP Object Extensions

The PCEP TLV defined in this document is compliant with the PCEP TLV format defined in [RFC5440].

### 7.1. Extension of LSP Object

LSP Object is defined in Section 7.3 of [RFC8231]. It specifies PLSP-ID to uniquely identify an LSP that is constant for the life time of a PCEP session. Similarly for P2MP tunnel, PLSP-ID identify a P2MP TE LSP uniquely. This document adds the following flags to the LSP Object:

N (P2MP bit - TBD7): If the bit is set to 1, it specifies the message is for P2MP TE LSP which MUST be set in PCRpt or PCUpd message for a P2MP TE LSP.

F (Fragmentation bit - TBD8): If the bit is set to 1, it specifies the message is fragmented.

If P2MP bit is set, the following P2MP-LSP-IDENTIFIER TLV MUST be present in LSP object.

## 7.2. P2MP-LSP-IDENTIFIER TLV

The P2MP LSP Identifier TLV MUST be included in the LSP object in PCRpt message for RSVP-TE signaled P2MP TE LSPs. If the TLV is missing, the PCE will generate an error with error-type 6 (mandatory object missing) and error-value TBD12 (P2MP-LSP-IDENTIFIER TLV missing) and close the PCEP session.

The P2MP LSP Identifier TLV MAY be included in the LSP object in PCUpd message for RSVP-TE signaled P2MP TE LSPs. The special value of all zeros for this TLV is used to refer to all paths pertaining to a particular PLSP-ID.

There are two P2MP LSP Identifier TLVs, one for IPv4 and one for IPv6.

The format of the IPV4-P2MP-LSP-IDENTIFIER TLV is shown in the following figure:

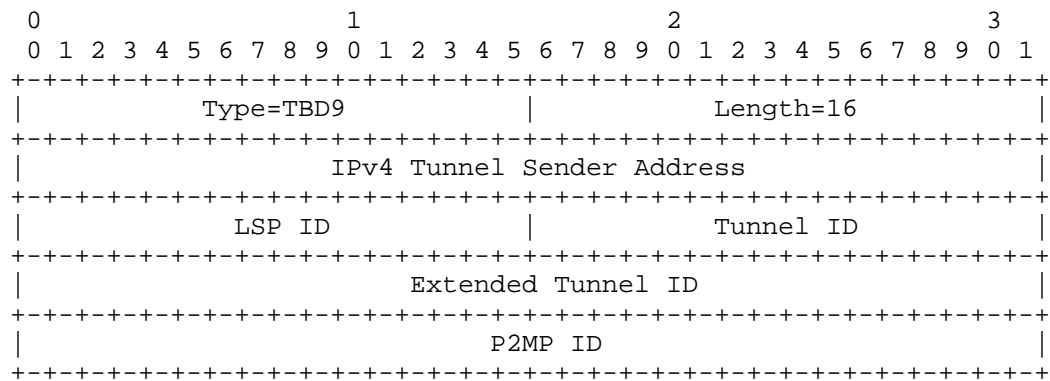


Figure 6: IPV4-P2MP-LSP-IDENTIFIER TLV format

The type (16-bit) of the TLV is TBD9 to be assigned by IANA. The length (16-bit) has a fixed value of 16 octets. The value contains the following fields:

**IPv4 Tunnel Sender Address:** contains the sender node's IPv4 address, as defined in [RFC3209], Section 4.6.2.1 for the LSP\_TUNNEL\_IPv4 Sender Template Object.

**LSP ID:** contains the 16-bit 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.1 for the LSP\_TUNNEL\_IPv4 Sender Template Object.



Tunnel ID: contains the 16-bit 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP\_TUNNEL\_IPv4 Session Object.

Extended Tunnel ID: contains the 32-bit 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP\_TUNNEL\_IPv4 Session Object.

P2MP ID: contains the 32-bit 'P2MP ID' identifier defined in Section 19.1.1 of [RFC4875] for the P2MP LSP Tunnel IPv4 SESSION Object.

The format of the IPV6-P2MP-LSP-IDENTIFIER TLV is shown in the following figure:

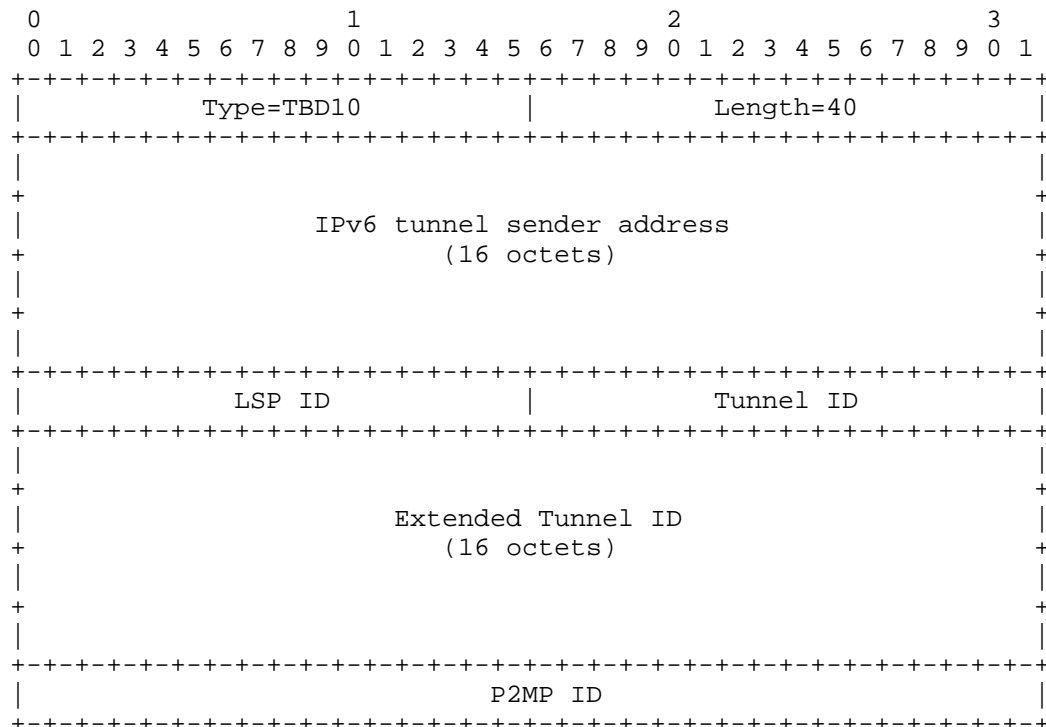


Figure 7: IPV6-P2MP-LSP-IDENTIFIER TLV format

The type of the TLV is TBD10 to be assigned by IANA. The length (16-bit) has a fixed length of 40 octets. The value contains the following fields:

IPv6 Tunnel Sender Address: contains the sender node's IPv6 address, as defined in [RFC3209], Section 4.6.2.2 for the LSP\_TUNNEL\_IPv6 Sender Template Object.

LSP ID: contains the 16-bit 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.2 for the LSP\_TUNNEL\_IPv6 Sender Template Object.

Tunnel ID: contains the 16-bit 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP\_TUNNEL\_IPv6 Session Object.

Extended Tunnel ID: contains the 128-bit 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP\_TUNNEL\_IPv6 Session Object.

P2MP ID: As defined above in IPV4-P2MP-LSP-IDENTIFIERS TLV.

Tunnel ID remains constant over the life time of a tunnel.

### 7.3. S2LS Object

The S2LS (Source-to-Leaves) Object is used to report RSVP-TE state of one or more destinations (leaves) encoded within the END-POINTS object for a P2MP TE LSP. It MUST be carried in PCRpt message along with END-POINTS object when N bit is set in LSP object.

S2LS Object-Class is TBD19.

S2LS Object-Types is 1.

The format of the S2LS object is shown in the following figure:

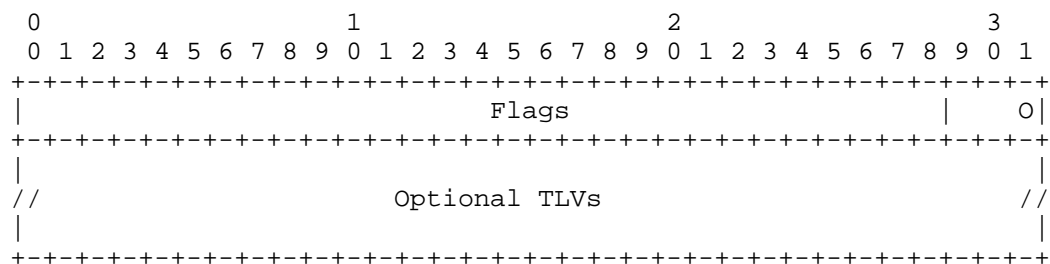


Figure 8: S2LS object format

Flags(32 bits):

O(Operational - 3 bits) the O Field represents the operational status of the group of destinations. The values are as per Operational field in LSP object defined in Section 7.3 of [RFC8231].

When N bit is set in LSP object then the O field in LSP object represents the operational status of the full P2MP TE LSP and the O field in S2LS object represents the operational status of a group of destinations encoded within the END-POINTS object.

Future documents MAY define optional TLVs that MAY be included in the S2LS Object.

## 8. Message Fragmentation

The total PCEP message length, including the common header, is 16 bytes. In certain scenarios the P2MP report and update request may not fit into a single PCEP message (e.g. initial report or update). The F-bit is used in the LSP object to signal that the initial report, update, or initiate message was too large to fit into a single message and will be fragmented into multiple messages. In order to identify the single report or update each message will use the same PLSP-ID. In order to identify that a series of PCInitiate messages represents a single Initiate, each message will use the same PLSP-ID (in this case 0) and SRP-ID-number.

Fragmentation procedure described below for report or update message is similar to [I-D.ietf-pce-rfc6006bis] which describes request and response message fragmentation.

### 8.1. Report Fragmentation Procedure

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

To indicate P2MP message fragmentation errors associated with a P2MP Report, a Error-Type (18) for "P2MP Fragmentation Error" and a new error-value TBD13 is used if a PCE has not received the last piece of the fragmented message, it should send an error message to the PCC to signal that it has received an incomplete message (i.e., "Fragmented Report failure").

## 8.2. Update Fragmentation Procedure

Once the PCE computes and updates a path for some or all leaves in a P2MP TE LSP, an update message is sent to the PCC. If the update is too large to fit into a single update message, the PCE will split the update over multiple messages. Each update message sent by the PCE, except the last one, will have the F-bit set in the LSP object to signify that the update has been fragmented into multiple messages. In order to identify that a series of update messages represents a single update, each message will use the same PLSP-ID and SRP-ID-number.

To indicate P2MP message fragmentation errors associated with a P2MP Update request, a Error-Type (18) for "P2MP Fragmentation Error" and a new error-value TBD14 is used if a PCC has not received the last piece of the fragmented message, it should send an error message to the PCE to signal that it has received an incomplete message (i.e., "Fragmented Update failure").

## 8.3. PCInitiate Fragmentation Procedure

Once the PCE initiates to set up the P2MP TE LSP, a PCInitiate message is sent to the PCC. If the PCInitiate is too large to fit into a single PCInitiate message, the PCE will split the PCInitiate over multiple messages. Each PCInitiate message sent by the PCE, except the last one, will have the F-bit set in the LSP object to signify that the PCInitiate has been fragmented into multiple messages. In order to identify that a series of PCInitiate messages represents a single Initiate, each message will use the same PLSP-ID (in this case 0) and SRP-ID-number.

To indicate P2MP message fragmentation errors associated with a P2MP PCInitiate, a Error-Type (18) for "P2MP Fragmentation Error" and a new error-value TBD15 is used if a PCC has not received the last piece of the fragmented message, it should send an error message to the PCE to signal that it has received an incomplete message (i.e., "Fragmented Instantiation failure").

## 9. Non-Support of P2MP TE LSPs for Stateful PCE

The PCEP protocol extensions described in this document for stateful PCEs with P2MP capability MUST NOT be used if PCE has not advertised its stateful capability with P2MP as per Section 5.2. If the PCEP Speaker on the PCC supports the extensions of this draft (understands the P2MP flag in the LSP object) but did not advertise this capability, then upon receipt of PCUpd message from the PCE, it SHOULD generate a PCErr with error-type 19 (Invalid Operation), error-value TBD17 (Attempted LSP Update Request for P2MP if active

stateful PCE capability for P2MP was not advertised). If the PCEP Speaker on the PCE supports the extensions of this draft (understands the P2MP flag in the LSP object) but did not advertise this capability, then upon receipt of a PCRpt message from the PCC, it SHOULD generate a PCErr with error-type 19 (Invalid Operation), error-value TBD16 (Attempted LSP State Report for P2MP if stateful PCE capability for P2MP was not advertised) and it will terminate the PCEP session.

If a Stateful PCE receives a P2MP TE LSP report message and the PCE does not understand the P2MP flag in the LSP object, and therefore the PCEP extensions described in this document, then the Stateful PCE would act as per [RFC8231].

The PCEP protocol extensions described in this document for PCC or PCE with instantiation capability for P2MP TE LSPs MUST NOT be used if PCC or PCE has not advertised its stateful capability with Instantiation and P2MP capability as per Section 5.2. If the PCEP Speaker on the PCC supports the extensions of this draft (understands the P (P2MP-LSP-INSTANTIATION-CAPABILITY) flag in the LSP object) but did not advertise this capability, then upon receipt of PCInitiate message from the PCE, it SHOULD generate a PCErr with error-type 19 (Invalid Operation), error-value TBD18 (Attempted LSP Instantiation Request for P2MP if stateful PCE instantiation capability for P2MP was not advertised).

## 10. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440], [I-D.ietf-pce-rfc6006bis], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp] apply to PCEP protocol extensions defined in this document. In addition, requirements and considerations listed in this section apply.

### 10.1. Control of Function and Policy

A PCE or PCC implementation MUST allow configuring the stateful PCEP capability, the LSP Update capability, and the LSP Initiation capability for P2MP LSPs.

### 10.2. Information and Data Models

The PCEP YANG module [I-D.ietf-pce-pcep-yang] SHOULD be extended to include advertised P2MP stateful capabilities, P2MP synchronization status, and delegation status of P2MP LSP etc. The statistics module should also count P2MP LSP related data.

### 10.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

### 10.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [I-D.ietf-pce-rfc6006bis], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

### 10.5. Requirements On Other Protocols

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

### 10.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440], [I-D.ietf-pce-rfc6006bis], [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp].

Stateful PCE feature for P2MP LSP would help with network operations.

## 11. IANA Considerations

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

### 11.1. PCE Capabilities in IGP Advertisements

IANA is requested to allocate new bits in the OSPF Parameters "PCE Capability Flags" registry, as follows:

Bit	Meaning	Reference
TBD1	Active Stateful PCE with P2MP	[This I-D]
TBD2	Passive Stateful PCE with P2MP	[This I-D]
TBD3	Stateful PCE Initiation with P2MP	[This I-D]

### 11.2. STATEFUL-PCE-CAPABILITY TLV

The STATEFUL-PCE-CAPABILITY TLV is defined in [RFC8231] and a registry is requested to be created to manage the flags in the TLV. IANA is requested to make the following allocations in the aforementioned registry.

Bit	Description	Reference
TBD4	P2MP-CAPABILITY	[This I-D]
TBD5	P2MP-LSP-UPDATE-CAPABILITY	[This I-D]
TBD6	P2MP-LSP-INSTANTIATION-CAPABILITY	[This I-D]

### 11.3. LSP Object

The LSP object is defined in [RFC8231] and a registry is created to manage the Flags field of the LSP object.

IANA is requested to make the following allocations in the aforementioned registry.

Bit	Description	Reference
TBD7	P2MP	[This I-D]
TBD8	Fragmentation	[This I-D]

### 11.4. PCEP-Error Object

IANA is requested to allocate new error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

Error-Type	Meaning
6	Mandatory Object missing [RFC5440] Error-value=TBD11: S2LS object missing Error-value=TBD12: P2MP-LSP-IDENTIFIER TLV missing
18	P2MP Fragmentation Error [I-D.ietf-pce-rfc6006bis] Error-value= TBD13. Fragmented Report failure Error-value= TBD14. Fragmented Update failure Error-value= TBD15. Fragmented Instantiation failure
19	Invalid Operation [RFC8231] Error-value= TBD16. Attempted LSP State Report for P2MP if stateful PCE capability for P2MP was not advertised Error-value= TBD17. Attempted LSP Update Request for P2MP if active stateful PCE capability for P2MP was not advertised Error-value= TBD18. Attempted LSP Instantiation Request for P2MP if stateful PCE instantiation capability for P2MP was not advertised

Reference for all new Error-Value above is [This I-D].

#### 11.5. PCEP TLV Type Indicators

IANA is requested to make the assignment of a new value for the existing "PCEP TLV Type Indicators" registry as follows:

Value	Meaning	Reference
TBD9	P2MP-IPV4-LSP-IDENTIFIERS	[This I-D]
TBD10	P2MP-IPV6-LSP-IDENTIFIERS	[This I-D]

#### 11.6. PCEP object

IANA is requested to allocate new object-class values and object types within the "PCEP Objects" sub-registry of the PCEP Numbers registry, as follows.



Object-Class	Value	Name	Reference
TBD19		S2LS	[This.I-D]
		Object-Type	
		0: Reserved	
		1: S2LS	

### 11.7. S2LS object

This document requests that a new sub-registry, named "S2LS Object Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the S2LS object. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

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

The following values are defined in this document:

Bit	Description	Reference
29-31	Operational (3-bit)	[This.I-D]

## 12. Security Considerations

The stateful operations on P2MP TE LSP are more CPU-intensive and also utilize more bandwidth on wire. In the event of an unauthorized stateful P2MP operations, or a denial of service attack, the subsequent PCEP operations may be disruptive to the network. Consequently, it is important that implementations conform to the relevant security requirements of [RFC5440], [I-D.ietf-pce-rfc6006bis] and [RFC8231], and [I-D.ietf-pce-pce-initiated-lsp]. Further [RFC8253] discusses an enhanced approach to provide secure transport for PCEP via Transport Layer Security (TLS).

### 13. Acknowledgments

Thanks to Quintin Zhao, Avantika and Venugopal Reddy for his comments.

### 14. References

#### 14.1. 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>>.
- [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>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.

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

[I-D.ietf-pce-pce-initiated-lsp]  
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-11 (work in progress), October 2017.

[I-D.ietf-pce-rfc6006bis]  
Zhao, Q., Dhody, D., Palleti, R., and D. King, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", draft-ietf-pce-rfc6006bis-04 (work in progress), September 2017.

#### 14.2. Informative References

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

[RFC4857] Fogelstroem, E., Jonsson, A., and C. Perkins, "Mobile IPv4 Regional Registration", RFC 4857, DOI 10.17487/RFC4857, June 2007, <<https://www.rfc-editor.org/info/rfc4857>>.

[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>>.

[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>>.

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

- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [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>>.
- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and j. jeffrant@gmail.com, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-05 (work in progress), June 2017.

Appendix A. Contributor Addresses

Yuji Kamite  
NTT Communications Corporation  
Granpark Tower  
3-4-1 Shibaura, Minato-ku  
Tokyo 108-8118  
Japan

EMail: y.kamite@ntt.com

Authors' Addresses

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

EMail: udayasreeredy@gmail.com

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

EMail: dhruv.ietf@gmail.com

Yosuke Tanaka  
NTT Communications Corporation  
Granpark Tower  
3-4-1 Shibaura, Minato-ku  
Tokyo 108-8118  
Japan

EMail: yosuke.tanaka@ntt.com

Vishnu Pavan Beeram  
Juniper Networks

EMail: vbeeram@juniper.net

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: October 14, 2019

U. Palle  
D. Dhody  
Huawei Technologies  
Y. Tanaka  
NTT Communications  
V. Beeram  
Juniper Networks  
April 12, 2019

Path Computation Element (PCE) Protocol Extensions for Stateful PCE  
usage for Point-to-Multipoint Traffic Engineering Label Switched Paths  
draft-ietf-pce-stateful-pce-p2mp-13

## Abstract

The Path Computation Element (PCE) has been identified as an appropriate technology for the determination of the paths of point-to-multipoint (P2MP) TE Label Switched Paths (LSPs). This document provides extensions required for Path Computation Element Communication Protocol (PCEP) so as to enable the usage of a stateful PCE capability in supporting P2MP TE LSPs.

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on October 14, 2019.

## Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Requirements Language . . . . .	4
2. Terminology . . . . .	4
3. Supporting P2MP TE LSPs for Stateful PCE . . . . .	4
3.1. Motivation . . . . .	4
3.2. Objectives . . . . .	5
4. Functions to Support P2MP TE LSPs for Stateful PCEs . . . . .	5
5. Architectural Overview of Protocol Extensions . . . . .	6
5.1. Extension of PCEP Messages . . . . .	6
5.2. Capability Advertisement . . . . .	6
5.3. IGP Extensions for Stateful PCE P2MP Capabilities Advertisement . . . . .	7
5.4. State Synchronization . . . . .	8
5.5. LSP Delegation . . . . .	8
5.6. LSP Operations . . . . .	8
5.6.1. Passive Stateful PCE . . . . .	8
5.6.2. Active Stateful PCE . . . . .	9
5.6.3. PCE-Initiated LSP . . . . .	9
5.6.3.1. P2MP TE LSPs Instantiation . . . . .	9
5.6.3.2. P2MP TE LSPs Deletion . . . . .	9
5.6.3.3. Adding and Pruning Leaves for the P2MP TE LSP . .	10
5.6.3.4. P2MP TE LSPs Delegation and Cleanup . . . . .	10
6. PCEP Message Extensions . . . . .	10
6.1. The PCRpt Message . . . . .	10
6.2. The PCUpd Message . . . . .	13
6.3. The PCReq Message . . . . .	14
6.4. The PCRep Message . . . . .	14
6.5. The PCInitiate message . . . . .	15
6.6. Example . . . . .	16
6.6.1. P2MP TE LSPs Update Request . . . . .	17
6.6.2. P2MP TE LSP Report . . . . .	17
6.6.3. P2MP TE LSPs Initiation Request . . . . .	18
7. PCEP Object Extensions . . . . .	19
7.1. Extension of LSP Object . . . . .	19
7.1.1. P2MP-LSP-IDENTIFIERS TLV . . . . .	19
7.2. S2LS Object . . . . .	22
8. Message Fragmentation . . . . .	23
8.1. Report Fragmentation Procedure . . . . .	23
8.2. Update Fragmentation Procedure . . . . .	24

8.3. PCInitiate Fragmentation Procedure . . . . .	24
9. Non-Support of P2MP TE LSPs for Stateful PCE . . . . .	24
10. Manageability Considerations . . . . .	25
10.1. Control of Function and Policy . . . . .	25
10.2. Information and Data Models . . . . .	25
10.3. Liveness Detection and Monitoring . . . . .	26
10.4. Verify Correct Operations . . . . .	26
10.5. Requirements On Other Protocols . . . . .	26
10.6. Impact On Network Operations . . . . .	26
11. IANA Considerations . . . . .	26
11.1. PCE Capabilities in IGP Advertisements . . . . .	26
11.2. STATEFUL-PCE-CAPABILITY TLV . . . . .	27
11.3. LSP Object . . . . .	27
11.4. PCEP-Error Object . . . . .	28
11.5. PCEP TLV Type Indicators . . . . .	29
11.6. PCEP object . . . . .	29
11.7. S2LS object . . . . .	29
12. Security Considerations . . . . .	30
13. Acknowledgments . . . . .	30
14. References . . . . .	31
14.1. Normative References . . . . .	31
14.2. Informative References . . . . .	33
Appendix A. Contributor Addresses . . . . .	34
Authors' Addresses . . . . .	34

## 1. Introduction

As per [RFC4655], the Path Computation Element (PCE) 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. [RFC5671] examines the applicability of PCE for the path computation for P2MP TE LSPs.

The PCEP is designed as a communication protocol between PCCs and PCEs for point-to-point (P2P) path computations and is defined in [RFC5440]. The extensions of PCEP to request path computation for P2MP TE LSPs are described in [RFC8306].

Stateful PCEs are shown to be helpful in many application scenarios, in both MPLS and GMPLS networks, as illustrated in [RFC8051]. These scenarios apply equally to P2P and P2MP TE LSPs. [RFC8231] provides the fundamental extensions to PCEP needed for stateful PCE to support



general functionality for P2P TE LSP. [RFC8281] provides extensions to PCEP needed for stateful PCE-initiated P2P TE LSP. This document complements that work by focusing on PCEP extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs. This document describes the setup, maintenance, and teardown of PCE-initiated P2MP LSPs under the stateful PCE model.

### 1.1. Requirements Language

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

## 2. Terminology

Terminology used in this document is the same as terminology used in [RFC8231], [RFC8281], and [RFC8306].

## 3. Supporting P2MP TE LSPs for Stateful PCE

### 3.1. Motivation

[RFC8051] presents several use cases, demonstrating scenarios that benefit from the deployment of a stateful PCE including optimization, recovery, etc., which are equally applicable to P2MP TE LSPs. [RFC8231] defines the extensions to PCEP needed for stateful operation of P2P TE LSPs. This document complements the previous work by focusing on extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs.

In addition to that, the stateful nature of a PCE simplifies the information conveyed in PCEP messages since it is possible to refer to the LSPs via a PCEP-specific LSP identifier (PLSP-ID) ([RFC8231]). For P2MP, where the size of message is much larger, this is an added advantage. When using a stateless PCE, a request to modify an existing P2MP tree requires that all the leaves are presented in the PCEP messages along with all the path information. But when using a stateful PCE, the PCEP messages can use a PLSP-ID to represent all information about the LSP that has previously been exchanged in PCEP messages, and it is only necessary to encode the modifications (such as new or removed leaf nodes). The PLSP-ID provides an index into the LSP-DB at the PCE, and identifies the LSP at the PCC.

In environments where the P2MP TE LSPs placement needs to change in response to application demands, it is useful to support dynamic creation and tear down of P2MP TE LSPs. The ability for a PCE to

trigger the creation of P2MP TE LSPs on demand can be seamlessly integrated into a controller-based network architecture, where intelligence in the controller can determine when and where to set up paths. Section 3 of [RFC8281] further describes the motivation behind the PCE-Initiation capability, which is equally applicable to P2MP TE LSPs.

### 3.2. Objectives

The objectives for the protocol extensions to support P2MP TE LSPs for stateful PCE are same as the objectives described in section 3.2 of [RFC8231].

### 4. Functions to Support P2MP TE LSPs for Stateful PCEs

[RFC8231] specifies new functions to support a stateful PCE. It also specifies that a function can be initiated either from a PCC towards a PCE (C-E) or from a PCE towards a PCC (E-C).

This document extends these functions to support P2MP TE LSPs.

Capability Advertisement (E-C,C-E): both the PCC and the PCE must announce during PCEP session establishment that they support Stateful PCE extensions for P2MP using mechanisms defined in Section 5.2.

LSP State Synchronization (C-E): after the session between the PCC and a stateful PCE with P2MP capability is initialized, the PCE must learn the state of a PCC's P2MP TE LSPs before it can perform path computations or update LSP attributes in a PCC.

LSP Update Request (E-C): a stateful PCE with P2MP capability requests modification of attributes on a PCC's P2MP TE LSPs.

LSP State Report (C-E): a PCC sends an LSP state report to a PCE whenever the state of a P2MP TE LSP changes.

LSP Control Delegation (C-E,E-C): a PCC grants to a PCE the right to update LSP attributes on one or more P2MP TE LSPs; the PCE becomes the authoritative source of the LSP's attributes as long as the delegation is in effect (See Section 5.7 of [RFC8231]); the PCC may withdraw the delegation or the PCE may give up the delegation at any time.

PCE-initiated LSP instantiation (E-C): a PCE sends an LSP Initiate Message to a PCC to instantiate or delete a P2MP TE LSP [RFC8281].

## 5. Architectural Overview of Protocol Extensions

### 5.1. Extension of PCEP Messages

New PCEP messages are defined in [RFC8231] to support stateful PCE for P2P TE LSPs. In this document these messages are extended to support P2MP TE LSPs.

**Path Computation State Report (PCRpt):** Each P2MP TE LSP State Report in a PCRpt message contains the actual P2MP TE LSP path attributes, the LSP status, etc. An LSP State Report carried in a PCRpt message is also used in delegation or revocation of control of a P2MP TE LSP to/from a PCE. The extension of PCRpt message is described in Section 6.1.

**Path Computation Update Request (PCUpd):** Each P2MP TE LSP Update Request in a PCUpd message MUST contain all LSP parameters that a PCE wishes to set for a given P2MP TE LSP. An LSP Update Request carried in a PCUpd message is also used to return LSP delegations if at any point PCE no longer desires control of a P2MP TE LSP. The PCUpd message is described in Section 6.2.

A PCEP message is defined in [RFC8281] to support stateful PCE instantiation of P2P TE LSPs. In this document this message is extended to support P2MP TE LSPs.

**Path Computation LSP Initiate Message (PCInitiate):** PCInitiate is a PCEP message sent by a PCE to a PCC to trigger P2MP TE LSPs instantiation or deletion. The PCInitiate message is described in Section 6.5.

The Path Computation Request (PCReq) and Path Computation Reply (PCRep) messages are also extended to support passive stateful PCE for P2P TE LSP in [RFC8231]. In this document these messages are extended to support P2MP TE LSPs as well.

### 5.2. Capability Advertisement

During PCEP Initialization Phase, as per Section 7.1.1 of [RFC8231], PCEP speakers advertise Stateful capability via the STATEFUL-PCE-CAPABILITY TLV in the OPEN object. Various flags are defined for the STATEFUL-PCE-CAPABILITY TLV defined in [RFC8231] and updated in [RFC8281] and [RFC8232].

Three new flags N (P2MP-CAPABILITY), M (P2MP-LSP-UPDATE-CAPABILITY), and P (P2MP-LSP-INSTANTIATION-CAPABILITY) are added in this document:

- N (P2MP-CAPABILITY flag - 1 bit): if set to 1 by a PCC, the N Flag indicates that the PCC is willing to send P2MP LSP State Reports whenever any parameters or operational status change of the P2MP LSP; if set to 1 by a PCE, the N Flag indicates that the PCE is interested in receiving LSP State Reports whenever there is any parameters or operational status change of the P2MP LSP. The P2MP-CAPABILITY Flag MUST be advertised by both a PCC and a PCE for PCRpt messages P2MP extension to be allowed on a PCEP session.
- M (P2MP-LSP-UPDATE-CAPABILITY flag - 1 bit): if set to 1 by a PCC, the M Flag indicates that the PCC allows modification of P2MP LSP parameters; if set to 1 by a PCE, the M Flag indicates that the PCE is capable of updating P2MP LSP parameters. The P2MP-LSP-UPDATE-CAPABILITY Flag MUST be advertised by both a PCC and a PCE for PCUpd messages P2MP extension to be allowed on a PCEP session.
- P (P2MP-LSP-INstantiation-CAPABILITY flag - 1 bit): If set to 1 by a PCC, the P Flag indicates that the PCC allows instantiation of a P2MP LSP by a PCE. If set to 1 by a PCE, the P flag indicates that the PCE supports P2MP LSP instantiation. The P2MP-LSP-INstantiation-CAPABILITY flag MUST be set by both PCC and PCE in order to support PCE-initiated P2MP LSP instantiation.

A PCEP speaker should continue to advertise the basic P2MP capability via mechanisms as described in [RFC8306].

### 5.3. IGP Extensions for Stateful PCE P2MP Capabilities Advertisement

When PCC is a Label Switching Router (LSR), participating in the IGP (OSPF or IS-IS), and PCEs are either LSRs or servers also participating in the IGP, an effective mechanism for PCE discovery within an IGP routing domain consists of utilizing IGP advertisements. Extensions for the advertisement of PCE Discovery Information are defined for OSPF and for IS-IS in [RFC5088] and [RFC5089] respectively.

The PCE-CAP-FLAGS sub-TLV, defined in [RFC5089], is an optional sub-TLV used to advertise PCE capabilities. It MAY be present within the PCE Discovery (PCED) TLV carried by OSPF or IS-IS. [RFC5088] and [RFC5089] provide the description and processing rules for this sub-TLV when carried within OSPF and IS-IS, respectively.

The format of the PCE-CAP-FLAGS sub-TLV is included below for easy reference:

Type: 5

Length: Multiple of 4.

Value: This contains an array of units of 32 bit flags with the most significant bit as 0. Each bit represents one PCE capability.

PCE capability bit flags are defined in [RFC5088]. This document defines new capability bits (early allocated by IANA) for the stateful PCE with P2MP as follows:

Bit	Capability
13	Active Stateful PCE with P2MP
14	Passive Stateful PCE with P2MP
15	PCE-Initiation with P2MP

Note that while active, passive or initiation stateful PCE with P2MP capabilities may be advertised during discovery, PCEP Speakers that wish to use stateful PCEP MUST advertise stateful PCEP capabilities during PCEP session setup, as specified in the current document. A PCC MAY initiate stateful PCEP P2MP capability advertisement at PCEP session setup even if it did not receive any IGP PCE capability advertisements.

#### 5.4. State Synchronization

State Synchronization operations (described in Section 5.6 of [RFC8231]) are applicable for the P2MP TE LSPs as well. The optimizations described in [RFC8232] can also be applied for P2MP TE LSPs.

#### 5.5. LSP Delegation

LSP delegation operations (described in Section 5.7 of [RFC8231]) are applicable for P2MP TE LSPs as well.

#### 5.6. LSP Operations

##### 5.6.1. Passive Stateful PCE

LSP operations for passive stateful PCE (described in Section 5.8.1 of [RFC8231]) are applicable for P2MP TE LSPs as well.

The PCReq and PCRep message format for P2MP TE LSPs is described in Section 3.4 and Section 3.5 of [RFC8306] respectively.

The PCReq and PCRep message for P2MP TE LSPs are extended to support encoding of LSP object, so that it is possible to refer to an LSP with a unique identifier and simplify the PCEP message exchange. For example, in case of modification of one leaf in a P2MP tree, there should be no need to carry the full P2MP tree in PCReq message.

The extension for the Request and Response message for passive stateful operations on P2MP TE LSPs are described in Section 6.3 and Section 6.4. The extension for the Path Computation LSP State Report (PCRpt) message is described in Section 6.1.

#### 5.6.2. Active Stateful PCE

LSP operations for active stateful PCE (described in Section 5.8.2 of [RFC8231]) are applicable for P2MP TE LSPs as well.

The extension for the Path Computation LSP Update (PCUpd) message for active stateful operations on P2MP TE LSPs are described in Section 6.2.

#### 5.6.3. PCE-Initiated LSP

As per section 5.1 of [RFC8281], the PCE sends a Path Computation LSP Initiate Request (PCInitiate) message to the PCC to suggest instantiation or deletion of a P2P TE LSP. This document extends the PCInitiate message to support P2MP TE LSPs (see details in Section 6.5).

The P2MP TE LSPs suggested instantiation and deletion operations are same as for P2P LSP as described in section 5.3 and 5.4 of [RFC8281].

##### 5.6.3.1. P2MP TE LSPs Instantiation

The Instantiation operation of P2MP TE LSPs is the same as defined in section 5.3 of [RFC8281], including handling of PLSP-ID, SYMBOLIC-PATH-NAME TLV, etc. The rules for processing and use of error codes remains unchanged. The N (P2MP) flag (Section 7.1) MUST be set in the LSP object in PCInitiate message by the PCE to specify that the instantiation is for P2MP TE LSPs. Like the PLSP-ID (as per [RFC8281]), the P2MP-LSP-IDENTIFIERS TLV SHOULD NOT be included in the LSP object in PCInitiate messages and MUST be ignored on receipt. These identifiers are generated by the PCC on receipt of PCInitiate message and reported via PCRpt message to the PCE.

##### 5.6.3.2. P2MP TE LSPs Deletion

The deletion operation of P2MP TE LSPs is the same as defined in section 5.4 of [RFC8281] by sending an LSP Initiate Message with an LSP object carrying the PLSP-ID of the LSP to be removed and an SRP object with the R flag set (LSP-REMOVE as per section 5.2 of [RFC8281]). Rules of processing and error codes remains unchanged.

#### 5.6.3.3. Adding and Pruning Leaves for the P2MP TE LSP

Adding of new leaves and Pruning of old Leaves for the PCE initiated P2MP TE LSP MUST be carried in PCUpd message as per Section 6.2 for P2MP TE LSP extensions. As defined in [RFC8306], leaf type = 1 for adding of new leaves, leaf type = 2 for pruning of old leaves of P2MP END-POINTS Object are used in PCUpd message.

PCC MAY use the Incremental State Update mechanism as described in [RFC4875] to signal adding and pruning of leaves.

Section 3.10 of [RFC8306] defines the error-handling procedures when adding new leaves to or removing old leaves from the existing P2MP tree for PCReq message. The same error handling and error-codes are also applicable to the stateful PCE messages as described in this document.

#### 5.6.3.4. P2MP TE LSPs Delegation and Cleanup

P2MP TE LSPs delegation and cleanup operations are same as defined in section 6 of [RFC8281]. Rules of processing and error codes remains unchanged.

### 6. PCEP Message Extensions

Message formats in this section, as those in [RFC8231], [RFC8281], and [RFC5440], are presented using Routing Backus-Naur Format (RBNF) as specified in [RFC5511].

#### 6.1. The PCRpt Message

As per Section 6.1 of [RFC8231], PCRpt message is used to report the current state of a P2P TE LSP. This document extends the PCRpt message in reporting the status of P2MP TE LSPs.

The format of PCRpt message is as follows:

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

Where:

```
<state-report-list> ::= <state-report>
                        [<state-report-list>]
```

```
<state-report> ::= [<SRP>  
                    <LSP>  
                    <path>
```

Where:

```

<path> ::= <end-point-intended-path-pair-list>
          [<actual-attribute-list>
           <end-point-actual-path-pair-list>]
          [<intended-attribute-list>]

```

```
<end-point-intended-path-pair-list>::=
    [<END-POINTS>]
    [<S2LS>]
    <intended-path>
    [<end-point-intended-path-pair-list>]
```

```
<end-point-actual-path-pair-list>::=
    [<END-POINTS>]
    [<S2LS>]
    <actual-path>
    [<end-point-actual-path-pair-list>]
```

$$\langle \text{intended-path} \rangle ::= (\langle \text{ERO} \rangle | \langle \text{SERO} \rangle) [\langle \text{intended-path} \rangle]$$
$$\langle \text{actual-path} \rangle ::= (\langle \text{RRO} \rangle | \langle \text{SRRO} \rangle) [\langle \text{actual-path} \rangle]$$

<intended-attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

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

The P2MP END-POINTS object defined in [RFC8306] is mandatory for specifying address of P2MP leaves, grouped by leaf types.

- o New leaves to add (leaf type = 1)
- o Old leaves to remove (leaf type = 2)



- o Old leaves whose path can be modified/reoptimized (leaf type = 3)
- o Old leaves whose path must be left unchanged (leaf type = 4)

When reporting the status of a P2MP TE LSP, the destinations MUST be grouped in END-POINTS object based on the operational status (O field in S2LS object) and leaf type (in END-POINTS). This way, leaves of the same type that share the same operational status can be grouped together. For reporting the status of delegated P2MP TE LSPs leaf type = 3 is used, whereas for non-delegated P2MP TE LSPs, leaf type = 4 is used.

For a delegated P2MP TE LSP configuration changes are reported via PCRpt message. For example, adding of new leaves END-POINTS (leaf type = 1) is used, and removing of old leaves (leaf type = 2) is used.

Note that the compatibility with the [RFC8231] definition of <state-report> is preserved. At least one instance of <END-POINTS> MUST be present in this message for P2MP LSP.

Note that the ordering of <end-point-intended-path-pair-list>, <actual-attribute-list>, <end-point-actual-path-pair-list>, and <intended-attribute-list> is done to retain compatibility with state reports for the P2P LSPs as per [RFC8231].

During state synchronization, the PCRpt message reports the status of the full P2MP tree.

The S2LS object MUST be carried in PCRpt message along with END-POINTS object when N (P2MP) flag is set in LSP object for P2MP TE LSPs. If the S2LS object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 ("Mandatory Object missing") and Error-value=13 (early allocated by IANA) ("S2LS object missing"). If the END-POINTS object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 ("Mandatory Object missing") and Error-value=3 ("END-POINTS object missing") (defined in [RFC5440]).

The S2LS object could be used in conjunction with the intended-path (ERO) as well as the actual-path (RRO); for the same leaf, the state encoded in the S2LS object associated with the actual-path MUST be used over the intended-path.

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

## 6.2. The PCUpd Message

As per Section 6.2 of [RFC8231], PCUpd message is used to update P2P TE LSP attributes. This document extends the PCUpd message in updating the attributes of a P2MP TE LSP.

The format of a PCUpd message is as follows:

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

Where:

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

```
<update-request> ::= <SRP>
                     <LSP>
                     <path>
```

Where:

```
<path> ::= <end-point-path-pair-list>
           <intended-attribute-list>
```

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

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

<intended-attribute-list> is the attribute-list defined in [RFC5440] and extended by PCEP extensions.

Note that the compatibility with the [RFC8231] definition of <update-request> is preserved.

The PCC SHOULD use the make-before-break or sub-group-based procedures described in [RFC4875] based on a local policy decision.

The END-POINTS object MUST be carried in PCUpd message when N flag is set in LSP object for a P2MP TE LSP. If the END-POINTS object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 ("Mandatory Object missing") and Error-value=3 ("END-POINTS object missing") (defined in [RFC5440]).

### 6.3. The PCReq Message

As per Section 3.4 of [RFC8306], PCReq message is used for a P2MP Path Computation Request. This document extends the PCReq message such that a PCC MAY include the LSP object in the PCReq message if the stateful PCE P2MP capability has been negotiated on a PCEP session between the PCC and a PCE.

The format of PCReq message is as follows:

```
<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>
              [<LSP>]
              [<OF>]
              [<LSPA>]
              [<BANDWIDTH>]
              [<metric-list>]
              [<IRO> | <BNC>]
              [<LOAD-BALANCING>]
```

```
<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>]
```

### 6.4. The PCRep Message

As per Section 3.5 of [RFC8306], PCRep message is used for a P2MP Path Computation Reply. This document extends the PCRep message such that a PCE MAY include the LSP object in the PCRep message if the

stateful PCE P2MP capability has been negotiated on a PCEP session between the PCC and a PCE.

The format of PCRep message is as follows:

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

where:

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

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

```
<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>]
```

#### 6.5. The PCInitiate message

As defined in section 5.1 of [RFC8281], PCE sends a PCInitiate message to a PCC to recommend instantiation of a P2P TE LSP. This document extends the format of PCInitiate message for the creation of P2MP TE LSPs but the creation and deletion operations of P2MP TE LSPs are same to the P2P TE LSPs.

The format of PCInitiate message is as follows:

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

Where:

```
<PCE-initiated-lsp-list> ::= <PCE-initiated-lsp-request>
                              [<PCE-initiated-lsp-list>]
```

```
<PCE-initiated-lsp-request> ::=
(<PCE-initiated-lsp-instantiation> | <PCE-initiated-lsp-deletion>)
```

```
<PCE-initiated-lsp-instantiation> ::= <SRP>
                                       <LSP>
                                       <end-point-path-pair-list>
                                       [<attribute-list>]
```

```
<PCE-initiated-lsp-deletion> ::= <SRP>
                                   <LSP>
```

Where:

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

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

<attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

The PCInitiate message with an LSP object with N flag (P2MP) set is used to convey operation on a P2MP TE LSP. The SRP object is used to correlate between initiation requests sent by the PCE and the error reports and state reports sent by the PCC as described in [RFC8231].

The END-POINTS object MUST be carried in PCInitiate message when N flag is set in LSP object for a P2MP TE LSP. If the END-POINTS object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 ("Mandatory Object missing") and Error-value=3 ("END-POINTS object missing") (defined in [RFC5440]).

## 6.6. Example

#### 6.6.1. P2MP TE LSPs Update Request

An LSP Update Request message is sent by an active stateful PCE to update the P2MP TE LSPs parameters or attributes. An example of a PCUpd message for P2MP TE LSPs is described below:

```
Common Header
SRP
LSP with P2MP flag set
END-POINTS for leaf type 3
ERO list
```

In this example, a stateful PCE requests an update of the path taken to some of the leaves in a P2MP tree. The update request uses the END-POINT type 3 (modified/reoptimized). The ERO list represents the source to leaves path after modification. The update message does not need to encode the full P2MP tree in this case.

#### 6.6.2. P2MP TE LSP Report

The LSP State Report message is sent by a PCC to report or delegate the P2MP TE LSP. The leaves of the P2MP TE LSP are grouped in the END-POINTS object based on the operational status and the leaf type. An example of a PCRpt message for a delegated P2MP TE LSPs is described below to add new leaves to an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 1 (add)
S2LS (O=DOWN)
ERO list (empty)
```

An example of a PCRpt message for a P2MP TE LSP is described below to prune leaves from an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 2 (remove)
S2LS (O=UP)
ERO list (empty)
```

An example of a PCRpt message for a delegated P2MP TE LSP is described below to report status of leaves in an existing P2MP TE LSP:

```
Common Header
SRP
LSP with P2MP flag set
END-POINTS for leaf type 3 (modify)
  S2LS (O=UP)
  RRO list
END-POINTS for leaf type 3 (modify)
  S2LS (O=DOWN)
  ERO list (empty)
```

In this example, the PCRpt message is in response to a PCUpd message (with corresponding SRP) object indicating some leaves that are up (with the actual path) and some are down.

An example of a PCRpt message for a non-delegated P2MP TE LSP is described below to report status of leaves:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 4 (unchanged)
  S2LS (O=ACTIVE)
  RRO list
END-POINTS for leaf type 4 (unchanged)
  S2LS (O=DOWN)
  ERO list (empty)
```

#### 6.6.3. P2MP TE LSPs Initiation Request

An LSP Initiation Request message is sent by an stateful PCE to create a P2MP TE LSP. An example of a PCInitiate message for a P2MP TE LSP is described below:

```
Common Header
SRP
LSP with P2MP flag set
END-POINTS for leaf type 1 (add)
  ERO list
```

In this example, a stateful PCE request creation of a P2MP TE LSP. The initiation request uses the END-POINT type 1 (new leaves). The ERO list represents the source to leaves path. The initiate message encodes the full P2MP tree in this case.

## 7. PCEP Object Extensions

The new PCEP TLVs defined in this document are in compliance with the PCEP TLV format defined in [RFC5440].

### 7.1. Extension of LSP Object

The LSP Object is defined in Section 7.3 of [RFC8231]. It specifies the PLSP-ID to uniquely identify an LSP that is constant for the life time of a PCEP session. Similarly for a P2MP tunnel, the PLSP-ID identify a P2MP TE LSP uniquely. This document adds the following flags to the LSP Object:

N (P2MP flag - 1 bit): If the N flag is set to 1, it indicates that the message is for a P2MP TE LSP.

F (Fragmentation flag - 1 bit): If the F flag is unset (0), it indicates that the LSP is not fragmented or it is the last piece of the fragmented LSP. If the F flag is set to 1, it indicates that the LSP is fragmented and this is not the last piece of the fragmented LSP. The receiver needs to wait for additional fragments until it receives an LSP with the same PLSP-ID and with the F-bit set to 0. See Section 8 for further details.

E (ERO-compression flag - 1 bit): If the E flag is set to 1, it indicates the route is in compressed format (that is, Secondary Explicit Route Object (SERO) and Secondary Record Route Object (SRRO) objects [RFC8306] are in use).

The flags defined in this section (N, F, E flags) are used in PCRpt, PCUpd, or PCInitiate message. In case of PCReq and PCRep message, these flags have no meaning and thus MUST be ignored. The corresponding flags in the RP (Request Parameters) object are used as described in [RFC8306].

#### 7.1.1. P2MP-LSP-IDENTIFIERS TLV

[RFC8231] specify the LSP-IDENTIFIERS TLVs to be included in the LSP object. For P2MP TE LSP, this document defines P2MP-LSP-IDENTIFIERS TLVs for the LSP object. There are two P2MP-LSP-IDENTIFIERS TLVs, one for IPv4 and one for IPv6. The P2MP-LSP-IDENTIFIERS TLV MUST be included in the LSP object in PCRpt message for P2MP TE LSPs. If the N bit is set in the LSP object in the PCRpt message but the P2MP-LSP-IDENTIFIER TLV is absent, the PCE MUST respond with a PCErr message carrying error-type 6 ("mandatory object missing") and error-value 14 (early allocated by IANA) ("P2MP-LSP-IDENTIFIER TLV missing") and close the PCEP session.



The P2MP-LSP-IDENTIFIERS TLV MAY be included in the LSP object in the PCUpd message for P2MP TE LSPs. The special value of all zeros for all the fields in the value portion of the TLV is used to refer to all paths pertaining to a particular PLSP-ID. The length of the TLV remains fixed based on the IP version.

The P2MP-LSP-IDENTIFIERS TLV SHOULD NOT be used in PCInitiate (see Section 5.6.3.1) and MAY optionally be included in the LSP object in the PCReq and the PCRep message for P2MP TE LSP.

The format of the IPV4-P2MP-LSP-IDENTIFIERS TLV is shown in the Figure 6:

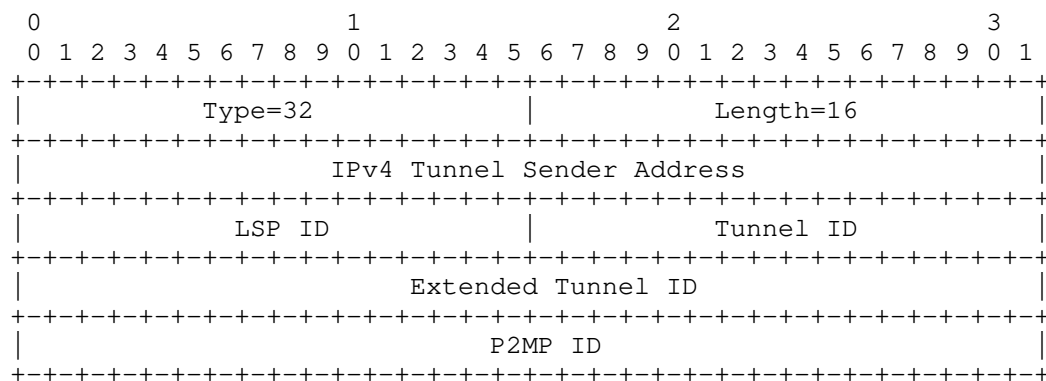


Figure 6: IPV4-P2MP-LSP-IDENTIFIERS TLV format

The type (16-bits) of the TLV is 32 (early allocated by IANA). The length (16-bits) has a fixed value of 16 octets. The value contains the following fields:

**IPv4 Tunnel Sender Address:** contains the sender node's IPv4 address, as defined in [RFC3209], Section 4.6.2.1 for the LSP\_TUNNEL\_IPv4 Sender Template Object.

**LSP ID:** contains the 16-bits 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.1 for the LSP\_TUNNEL\_IPv4 Sender Template Object.

**Tunnel ID:** contains the 16-bits 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP\_TUNNEL\_IPv4 Session Object.

Extended Tunnel ID: contains the 32-bits 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP\_TUNNEL\_IPv4 Session Object.

P2MP ID: contains the 32-bits 'P2MP ID' identifier defined in Section 19.1.1 of [RFC4875] for the P2MP LSP Tunnel IPv4 SESSION Object.

The format of the IPV6-P2MP-LSP-IDENTIFIERS TLV is shown in the Figure 7:

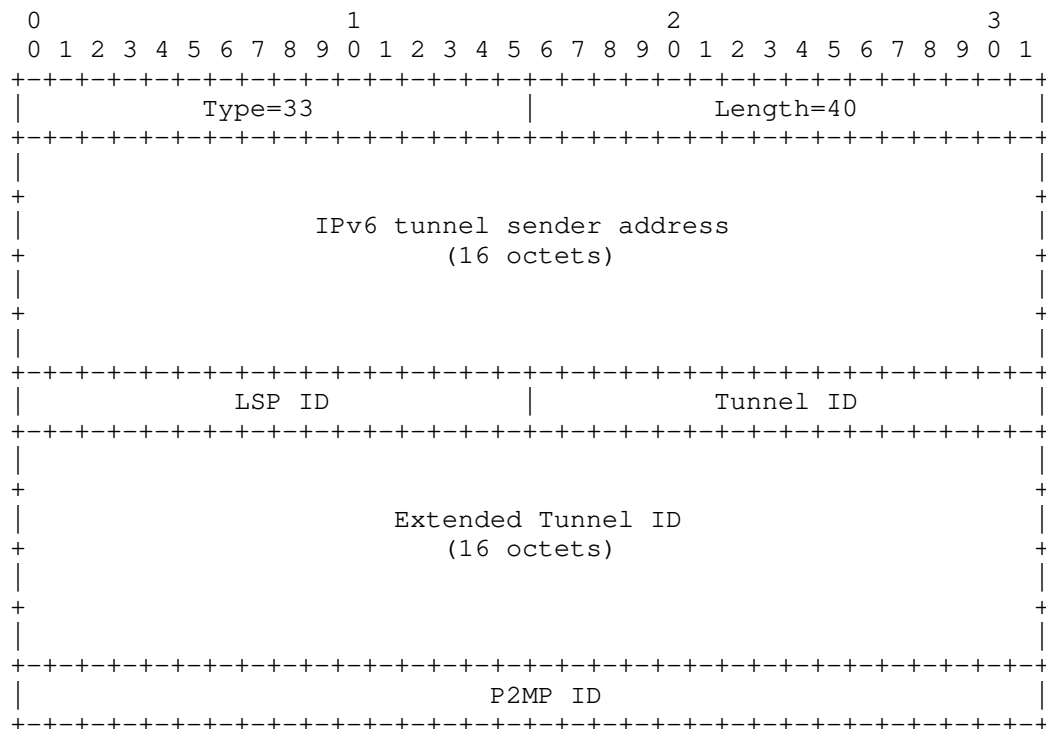


Figure 7: IPV6-P2MP-LSP-IDENTIFIERS TLV format

The type (16-bits) of the TLV is 33 (early allocated by IANA). The length (16-bits) has a fixed length of 40 octets. The value contains the following fields:

IPv6 Tunnel Sender Address: contains the sender node's IPv6 address, as defined in [RFC3209], Section 4.6.2.2 for the LSP\_TUNNEL\_IPv6 Sender Template Object.

LSP ID: contains the 16-bits 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.2 for the LSP\_TUNNEL\_IPv6 Sender Template Object.

Tunnel ID: contains the 16-bits 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP\_TUNNEL\_IPv6 Session Object.

Extended Tunnel ID: contains the 128-bits 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP\_TUNNEL\_IPv6 Session Object.

P2MP ID: As defined above in IPV4-P2MP-LSP-IDENTIFIERS TLV.

Tunnel ID remains constant over the life time of a tunnel.

## 7.2. S2LS Object

The S2LS (Source-to-Leaves) Object is used to report state of one or more destinations (leaves) encoded within the END-POINTS object for a P2MP TE LSP. It MUST be carried in PCRpt message along with END-POINTS object when N flag is set in LSP object.

S2LS Object-Class is 41 (Early allocated by IANA).

S2LS Object-Types is 1.

The format of the S2LS object is shown in the following figure:

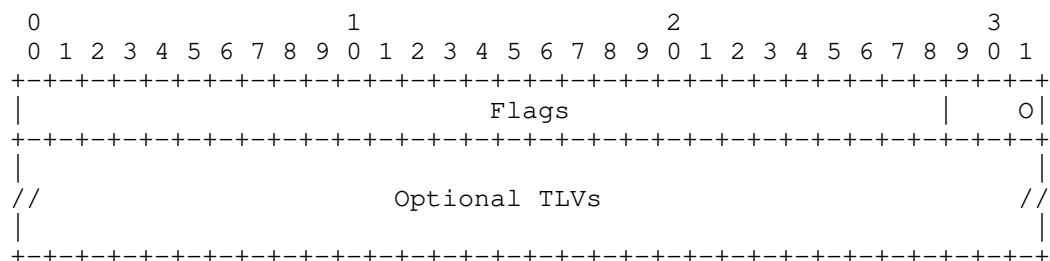


Figure 8: S2LS object format

Flags(32-bits): the following flags are currently defined -

O(Operational - 3 bits) the O Field represents the operational status of the group of destinations. The values are as per Operational field in LSP object defined in Section 7.3 of [RFC8231].

Unassigned bits are reserved for future uses. They MUST be set to 0 on transmission and MUST be ignored on receipt.

When N flag is set in LSP object then the O field in LSP object represents the operational status of the full P2MP TE LSP and the O field in S2LS object represents the operational status of a group of destinations encoded within the END-POINTS object. If there is a conflict between the O field in the LSP and the S2LS object (for example, O field in LSP corresponds to down whereas the O field in S2LS is up), the PCEP speaker MUST generate an error with error-type 10 ("Reception of an invalid object") and error-value TBD1 (to be allocated by IANA) ("Mis-match of O field in S2LS and LSP object").

Future documents might define optional TLVs that could be included in the S2LS Object.

## 8. Message Fragmentation

The total PCEP message length, including the common header, is  $(2^{16})-1$  bytes. In certain scenarios the P2MP report and update request may not fit into a single PCEP message (e.g. initial report or update). The F flag is used in the LSP object to signal that the initial report, update, or initiate message was too large to fit into a single message and will be fragmented into multiple messages. In order to identify the single report or update each message will use the same PLSP-ID. In order to identify that a series of PCInitiate messages represents a single Initiate, each message will use the same PLSP-ID (in this case 0) and SRP-ID-number.

The fragmentation procedure described below for report or update message is similar to [RFC8306] which describes request and response message fragmentation.

### 8.1. Report Fragmentation Procedure

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

The Error-Type value 18 ("P2MP Fragmentation Error") is used to report any error associated with the fragmentation of a P2MP PCEP message. A new error-value 2 (early allocated by IANA) indicates "Fragmented report failure" and is used if a PCE does not receive the last part of the fragmented message.

## 8.2. Update Fragmentation Procedure

Once the PCE computes and updates a path for some or all leaves in a P2MP TE LSP, an update message is sent to the PCC. If the update is too large to fit into a single update message, the PCE will split the update over multiple messages. Each update message sent by the PCE, except the last one, will have the F flag set in the LSP object to signify that the update has been fragmented into multiple messages. In order to identify that a series of update messages represents a single update, each message will use the same PLSP-ID and SRP-ID-number.

The Error-Type value 18 ("P2MP Fragmentation Error") is used to report any error associated with the fragmentation of a P2MP PCEP message. A new error-value 3 (early allocated by IANA) indicates "Fragmented update failure" and is used if a PCC does not receive the last part of the fragmented message.

## 8.3. PCInitiate Fragmentation Procedure

Once the PCE initiates to set up a P2MP TE LSP, a PCInitiate message is sent to the PCC. If the PCInitiate is too large to fit into a single PCInitiate message, the PCE will split the PCInitiate over multiple messages. Each PCInitiate message sent by the PCE, except the last one, will have the F flag set in the LSP object to signify that the PCInitiate has been fragmented into multiple messages. In order to identify that a series of PCInitiate messages represents a single Initiate, each message will use the same PLSP-ID (in this case 0) and SRP-ID-number.

The Error-Type value 18 ("P2MP Fragmentation Error") is used to report any error associated with the fragmentation of a P2MP PCEP message. A new error-value 4 (early allocated by IANA) indicates "Fragmented instantiation failure" and is used if a PCC does not receive the last part of the fragmented message.

## 9. Non-Support of P2MP TE LSPs for Stateful PCE

The PCEP extensions described in this document for stateful PCEs with P2MP capability MUST NOT be used if PCE has not advertised its stateful capability with P2MP as per Section 5.2. If the PCC supports the extensions as per this document (understands the N (P2MP-CAPABILITY) and M (P2MP-LSP-UPDATE-CAPABILITY) flags in the LSP object) but did not advertise this capability, then upon receipt of PCUpd message from the PCE, it SHOULD generate a PCErr with error-type 19 ("Invalid Operation"), error-value 12 (early allocated by IANA) ("Attempted LSP Update Request for P2MP if active stateful PCE capability for P2MP was not advertised") and terminate the PCEP

session. If the PCE supports the extensions as per this document (understands the N (P2MP-CAPABILITY) flag in the LSP object) but did not advertise this capability, then upon receipt of a PCRpt message from the PCC, it SHOULD generate a PCErr with error-type 19 ("Invalid Operation"), error-value 11 (early allocated by IANA) ("Attempted LSP State Report for P2MP if stateful PCE capability for P2MP was not advertised") and it SHOULD terminate the PCEP session.

If a Stateful PCE receives a P2MP TE LSP report message and the PCE does not understand the N (P2MP-CAPABILITY) flag in the LSP object, and therefore the PCEP extensions described in this document, then the Stateful PCE would act as per Section 6.1 of [RFC8231] (and consider the PCRpt message as invalid).

The PCEP extensions described in this document for PCC or PCE with instantiation capability for P2MP TE LSPs MUST NOT be used if PCC or PCE has not advertised its stateful capability with Instantiation and P2MP capability as per Section 5.2. If the PCC supports the extensions as per this document (understands the P (P2MP-LSP-INSTANTIATION-CAPABILITY) flag) but did not advertise this capability, then upon receipt of PCInitiate message from the PCE, it SHOULD generate a PCErr with error-type 19 ("Invalid Operation"), error-value 13 (early allocated by IANA) ("Attempted LSP Instantiation Request for P2MP if stateful PCE instantiation capability for P2MP was not advertised") and terminate the PCEP session..

## 10. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440], [RFC8306], [RFC8231], and [RFC8281] apply to PCEP extensions defined in this document. In addition, requirements and considerations listed in this section apply.

### 10.1. Control of Function and Policy

A PCE or PCC implementation MUST allow configuring the stateful PCEP capability, the LSP Update capability, and the LSP Initiation capability for P2MP LSPs.

### 10.2. Information and Data Models

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to include advertised P2MP stateful capabilities, P2MP synchronization status, and delegation status of P2MP LSP etc. The statistics module should also count P2MP LSP related data.

### 10.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

### 10.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8306], [RFC8231], and [RFC8281].

### 10.5. Requirements On Other Protocols

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

### 10.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440], [RFC8306], [RFC8231], and [RFC8281].

Stateful PCE feature for P2MP LSP would help with network operations.

## 11. IANA Considerations

This document requests IANA to confirm the early allocation of the code-points for the protocol elements defined in this document.

### 11.1. PCE Capabilities in IGP Advertisements

IANA is requested to confirm the early allocation for the new bits in the OSPF Parameters "PCE Capability Flags" registry, as follows:

Bit	Meaning	Reference
13	Active Stateful PCE with P2MP	[This.I-D]
14	Passive Stateful PCE with P2MP	[This.I-D]
15	Stateful PCE Initiation with P2MP	[This.I-D]

### 11.2. STATEFUL-PCE-CAPABILITY TLV

The STATEFUL-PCE-CAPABILITY TLV is defined in [RFC8231] and the 'STATEFUL-PCE-CAPABILITY TLV Flag Field' subregistry was created to manage the flags in the TLV. IANA is requested to confirm the early allocation of the following code-points in the aforementioned registry.

Bit	Description	Reference
25	P2MP-CAPABILITY	[This.I-D]
24	P2MP-LSP-UPDATE-CAPABILITY	[This.I-D]
23	P2MP-LSP-INSTANTIATION-CAPABILITY	[This.I-D]

### 11.3. LSP Object

The LSP object is defined in [RFC8231] and the 'LSP Object Flag Field' subregistry was created to manage the Flags field of the LSP object.

IANA is requested to confirm the early allocation of the following code-points in the aforementioned registry.

Bit	Description	Reference
3	P2MP	[This.I-D]
2	Fragmentation	[This.I-D]

Additionally, IANA is requested to allocate an additional code-point in this registry.

Bit	Description	Reference
TBD	ERO-compression	[This.I-D]



## 11.4. PCEP-Error Object

IANA is requested to confirm the early allocation of the new error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, as follows:

Error-Type	Meaning
6	Mandatory Object missing [RFC5440] Error-value=13: S2LS object missing Error-value=14: P2MP-LSP-IDENTIFIERS TLV missing
18	P2MP Fragmentation Error [RFC8306] Error-value= 2. Fragmented Report failure Error-value= 3. Fragmented Update failure Error-value= 4. Fragmented Instantiation failure
19	Invalid Operation [RFC8231] Error-value= 11. Attempted LSP State Report for P2MP if stateful PCE capability for P2MP was not advertised Error-value= 12. Attempted LSP Update Request for P2MP if active stateful PCE capability for P2MP was not advertised Error-value= 13. Attempted LSP Instantiation Request for P2MP if stateful PCE instantiation capability for P2MP was not advertised

Reference for all new Error-Value above is [This.I-D].

Additionally, IANA is requested to allocate an additional code-point in this registry.

Error-Type	Meaning
10	Reception of an invalid object [RFC5440] Error-value=TBD1: Mis-match of O field in S2LS and LSP object

Reference for all new Error-Value above is [This.I-D].

### 11.5. PCEP TLV Type Indicators

IANA is requested to confirm the early allocation of the following code-points in the existing "PCEP TLV Type Indicators" registry as follows:

Value	Meaning	Reference
32	P2MP-IPV4-LSP-IDENTIFIERS	[This.I-D]
33	P2MP-IPV6-LSP-IDENTIFIERS	[This.I-D]

### 11.6. PCEP object

IANA is requested to confirm the early allocation for the new object-class values and object types within the "PCEP Objects" sub-registry of the PCEP Numbers registry, as follows.

Object-Class Value	Name	Reference
41	S2LS Object-Type 0: Reserved 1: S2LS	[This.I-D]

### 11.7. S2LS object

This document requests that a new sub-registry, named "S2LS Object Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the 32-bits Flag field of the S2LS object. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

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

The following values are defined in this document:

Bit	Description	Reference
29-31	Operational (3-bits)	[This.I-D]
0-28	Unassigned	

## 12. Security Considerations

The stateful operations on P2MP TE LSPs are more CPU-intensive and also utilize more bandwidth on wire (in comparison to P2P TE LSPs). If a rogue PCC were able to request unauthorized stateful PCE operations then it may be able to mount a DoS attack against a PCE, which would disrupt the network and deny service to other PCCs. Similarly an attacker may flood the PCC with PCUpd messages at a rate that exceeds either the PCC's ability to process them or the network's ability to signal the changes, by either spoofing messages or compromising the PCE itself.

Consequently, it is important that implementations conform to the relevant security requirements as listed below -

- o As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).
- o Security considerations for path computation requests and responses are as per [RFC8306].
- o Security considerations for stateful operations (such as state report, synchronization, delegation, update, etc.) are as per [RFC8231].
- o Security considerations for LSP instantiation mechanism are as per [RFC8231].
- o Security considerations as stated in Section 10.1, Section 10.6, and Section 10.7 of [RFC5440] continue to apply.

## 13. Acknowledgments

Thanks to Quintin Zhao, Avantika and Venugopal Reddy for the review comments.

Thanks to Adrian Farrel (and Jonathan Hardwick) for the review as document shepherds.

Thanks to Andy Malis for the RTGDIR review. Thanks to Donald Eastlake for the SECDIR review. Thanks to David Schinazi for the GENART review.

Thanks to Suresh Krishnan, Mirja Kuhlewind, Roman Danyliw, and Benjamin Kaduk for the IESG reviews.

## 14. References

### 14.1. 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>>.
- [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>>.
- [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>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8306] Zhao, Q., Dhody, D., Ed., Palleti, R., and D. King, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 8306, DOI 10.17487/RFC8306, November 2017, <<https://www.rfc-editor.org/info/rfc8306>>.

## 14.2. Informative References

- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [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>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-11 (work in progress), March 2019.

Appendix A. Contributor Addresses

Yuji Kamite  
NTT Communications Corporation  
Granpark Tower  
3-4-1 Shibaura, Minato-ku  
Tokyo 108-8118  
Japan

EMail: y.kamite@ntt.com

Authors' Addresses

Udayasree Palle  
Huawei Technologies

EMail: udayasreereddy@gmail.com

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

EMail: dhruv.ietf@gmail.com

Yosuke Tanaka  
NTT Communications Corporation  
Granpark Tower  
3-4-1 Shibaura, Minato-ku  
Tokyo 108-8118  
Japan

EMail: yosuke.tanaka@ntt.com

Vishnu Pavan Beeram  
Juniper Networks

EMail: vbeeram@juniper.net

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: March 1, 2018

S. Litkowski  
Orange  
S. Sivabalan  
Cisco  
D. Dhody  
Huawei  
August 28, 2017

Inter Stateful Path Computation Element communication procedures  
draft-litkowski-pce-state-sync-02

Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests. The stateful PCE extensions allow stateful control of Multi-Protocol Label Switching (MPLS) Traffic Engineering Label Switched Paths (TE LSPs) using PCEP.

A Path Computation Client (PCC) can synchronize an LSP state information to a Stateful Path Computation Element (PCE). The stateful PCE extension allows a redundancy scenario where a PCC can have redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control on a LSP to only a single PCE, and only one PCE is responsible for path computation for this delegated LSP. The document does not state the procedures related to an inter-PCE stateful communication.

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

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



## Requirements Language

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

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 1, 2018.

## Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction and problem statement . . . . .	3
1.1. Reporting LSP changes . . . . .	3
1.2. Split-brain . . . . .	4
1.3. Applicability to H-PCE . . . . .	11
2. Proposed solution . . . . .	11
2.1. State-sync session . . . . .	11
2.2. Master/Slave relationship between PCE . . . . .	13
3. Procedures and protocol extensions . . . . .	13

3.1.	Opening a state-sync session . . . . .	13
3.1.1.	Capability advertisement . . . . .	13
3.2.	State synchronization . . . . .	14
3.3.	Incremental updates and report forwarding rules . . . . .	15
3.4.	Maintaining LSP states from different sources . . . . .	16
3.5.	Computation priority between PCEs and sub-delegation . . . . .	17
3.6.	Passive stateful procedures . . . . .	18
3.7.	PCE initiation procedures . . . . .	19
4.	Examples . . . . .	19
4.1.	Example 1 . . . . .	19
4.2.	Example 2 . . . . .	21
4.3.	Example 3 . . . . .	23
5.	Using Master/Slave computation and state-sync sessions to increase scaling . . . . .	24
6.	PCEP-PATH-VECTOR-TLV . . . . .	26
7.	Security Considerations . . . . .	27
8.	Acknowledgements . . . . .	27
9.	IANA Considerations . . . . .	27
9.1.	PCEP-Error Object . . . . .	27
9.2.	PCEP TLV Type Indicators . . . . .	27
9.3.	STATEFUL-PCE-CAPABILITY TLV . . . . .	28
10.	References . . . . .	28
10.1.	Normative References . . . . .	28
10.2.	Informative References . . . . .	28
	Authors' Addresses . . . . .	29

## 1. Introduction and problem statement

### 1.1. Reporting LSP changes

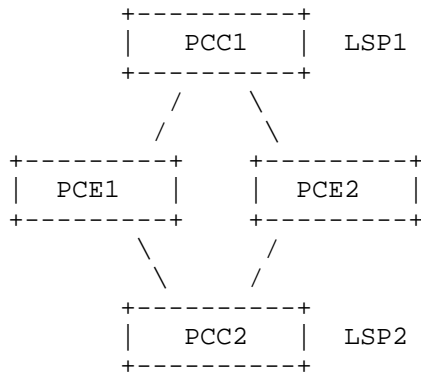
When using a stateful PCE ([I-D.ietf-pce-stateful-pce]), a Path Computation Client (PCC) can synchronize an LSP state information to the stateful Path Computation Element (PCE). If the PCC grants the control on the LSP to the PCE, the PCE can update the LSP parameters at any time.

In a multi PCE deployment (redundancy, loadbalancing...), with the current specification defined in [I-D.ietf-pce-stateful-pce], the PCC will be in charge of reporting the other PCEs of the LSP parameter change which brings additional hops and delays in notifying the overall network of the LSP parameter change.

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

Apart from the synchronization from the PCC, it is also useful if there is synchronization mechanism between the stateful PCEs. As stateful PCE make changes to its delegated LSPs, these changes

(pending LSPs and the sticky resources [RFC7399]) can be synchronized immediately to the other PCEs.



In the figure above, we consider a loadbalanced PCE architecture, so PCE1 is responsible to compute paths for PCC1 and PCE2 is responsible to compute paths for PCC2. When PCE1 triggers an LSP update for LSP1, it sends a PCUpdate message to PCC1 for LSP1 containing the new parameters. PCC1 will take the parameters into account and will send a PCReport to PCE1 and PCE2 reflecting the changes. PCE2 will so be notified of the change only after receiving the PCReport from PCC1.

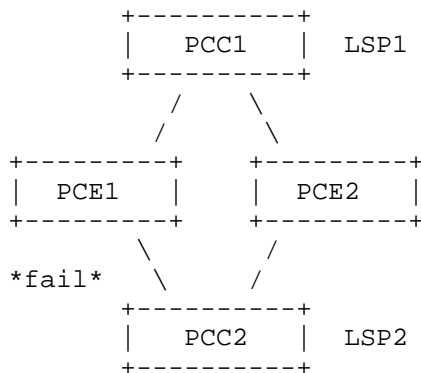
Let's consider that the LSP1 parameters changed in a such way that LSP1 will take over resources from LSP2 with an higher priority. After receiving the report from PCC1, PCE2 will so try to find a new path for LSP2. If we consider that there is a round trip delay of about 150msec between the PCEs and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

Adding a PCEP session between PCE1 and PCE2 may allow to reduce to the notification time, so PCE2 can react more quickly by taking the pending LSPs and sticky resources into account during path computation and reoptimization.

## 1.2. Split-brain

In a resiliency case, a PCC has redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control on an LSP to a single PCE only, and only this PCE is responsible for the path computation for the delegated LSP: the PCC achieves this by setting the D flag only to the active PCE. The election of the active PCE to delegate an LSP is controlled by each PCC. The PCC usually elects the active PCE by a local configured policy (by setting a priority).

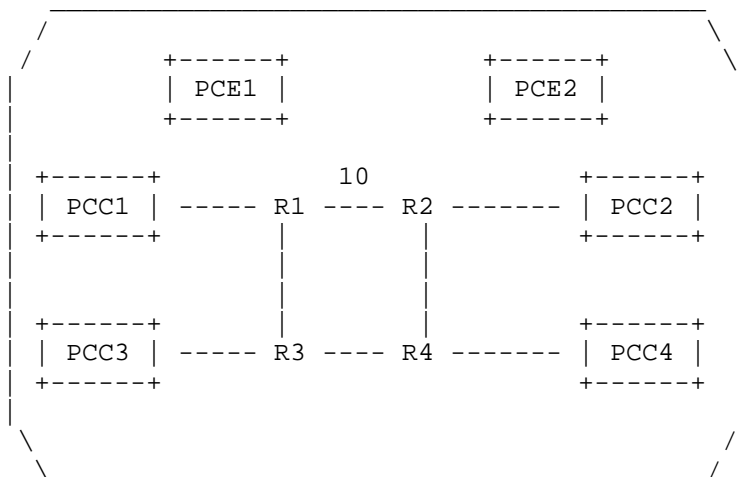
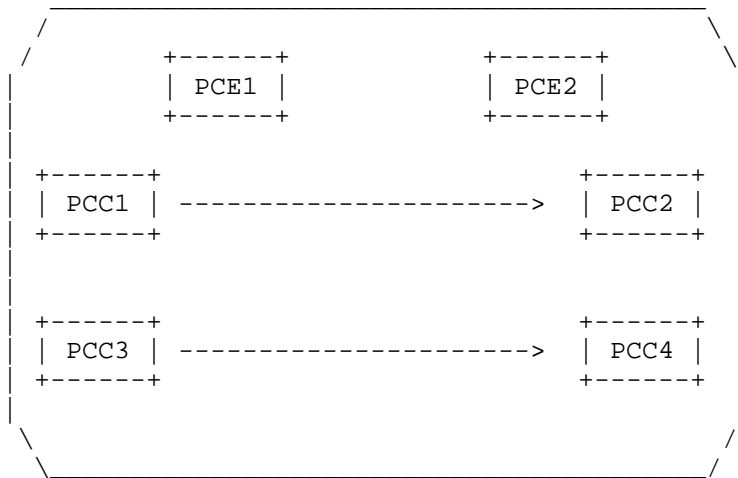
Upon PCEP session failure, or active PCE failure, PCC may decide to elect a new active PCE by sending new PCRpt message with D flag set to this new active PCE. When the failed PCE or PCEP session comes back online, it will be up to the vendor to implement preemption. Doing preemption may lead to some traffic disruption on the existing path if path results from both PCEs are not exactly the same. By considering a network with multiple PCCs and implementing multiple stateful PCEs for redundancy purpose, there is no guarantee that at any time all the PCCs delegate their LSPs to the same PCE.



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

We call this situation a split-brain scenario as there are multiple computation brains running at the same time while a central computation unit was required in some deployments.

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



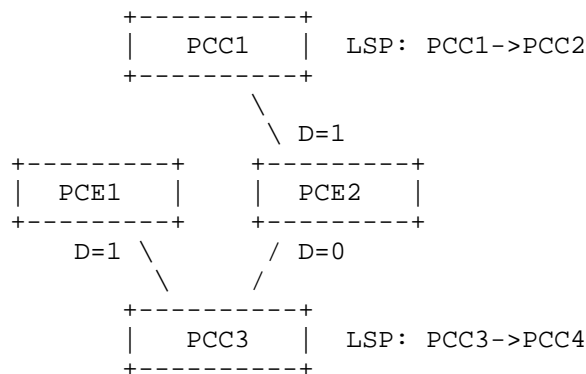
In the figure above, we want to create two link-disjoint LSPs: PCC1->PCC2 and PCC3->PCC4. In the topology, all link metrics are equal to 1 except the link R1-R2 which has a metric of 10. The PCEs are responsible for the path computation and PCE1 is the active PCE for all PCCs in the nominal case.

Scenario 1:

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

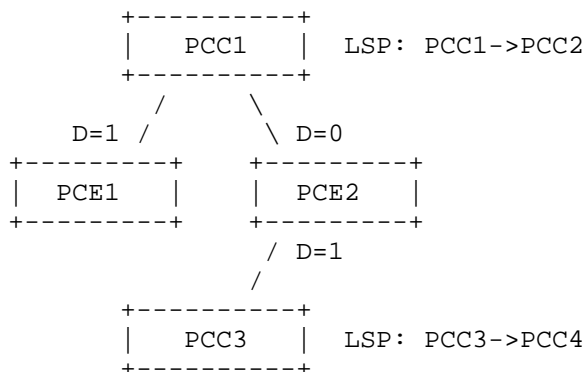
#### Scenario 2:

Now we consider that PCC1 loses its PCEP session with PCE1 (all other PCEP sessions are UP). PCC1 delegates its LSP to PCE2.



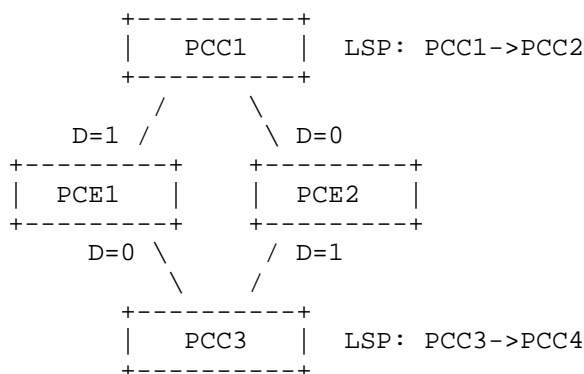
We first configure PCC1->PCC2 LSP, as the only constraint is path disjointness, PCE2 (which is the new active PCE for PCC1) sends a PCUpdate message to PCC1 with the ERO: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE1 is not aware anymore of LSPs from PCC1, so it cannot compute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

#### Scenario 3:



With this new PCEP session topology, we first configure PCC1->PCC2, PCE1 computes the shortest path as it is the only LSP in the disjoint-group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this LSP. The only solution found is to move PCC1->PCC2 LSP on another path, but PCE2 cannot do it as it does not have delegation for this LSP. In this setup, PCEs are not able to find a disjoint path.

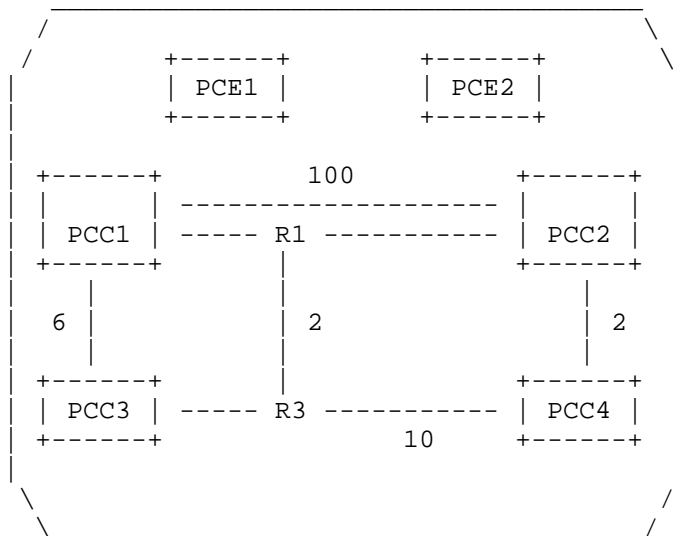
Scenario 4:



With this new PCEP session topology, we consider that PCEs are configured to fallback to shortest path if disjointness cannot be found. We first configure PCC1->PCC2, PCE1 computes shortest path as it is the only LSP in the disjoint-group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this LSP. The only solution found is to move PCC1->PCC2 LSP on another path, but PCE2 cannot do it as it does not have delegation for this LSP. PCE2 then provides

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

Scenario 5:



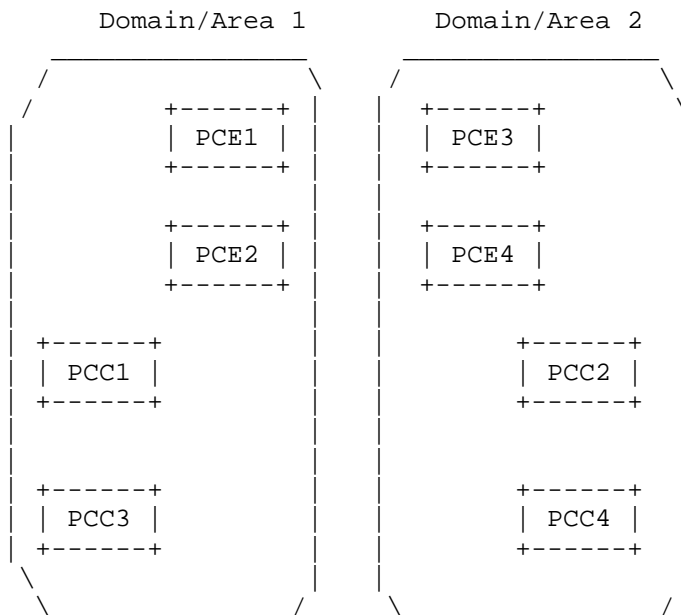
Now we consider a new network topology with the same PCEP session topology as the previous example. We configure both LSPs almost at the same time. PCE1 will compute a path for PCC1->PCC2 while PCE2 will compute a path for PCC3->PCC4. As each other is not aware of the path of the second LSP in the group (not reported yet), each PCE is computing shortest path for the LSP. PCE1 computes ERO: R1->PCC2 for PCC1->PCC2 and PCE2 computes ERO: R3->R1->PCC2->PCC4 for PCC3->PCC4. When these shortest paths will be reported to each PCE. Each PCE will recompute disjointness. PCE1 will provide a new path for PCC1->PCC2 with ERO: PCC1->PCC2. PCE2 will provide also a new path for PCC3->PCC4 with ERO: R3->PCC4. When those new paths will be reported to both PCEs, this will trigger CSPF again. PCE1 will provide a new more optimal path for PCC1->PCC2 with ERO: R1->PCC2 and PCE2 will also provide a more optimal path for PCC3->PCC4 with ERO: R3->R1->PCC2->PCC4. So we come back to the initial state. When those paths will be reported to both PCEs, this will trigger CSPF



again. An infinite loop of CSPF computation is then happening with a permanent flap of paths because of the split-brain situation.

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

Scenario 6: multi-domain



In the example above, we want to create disjoint LSPs from PCC1 to PCC2 and from PCC4 to PCC3. All the PCEs have the knowledge of both domain topologies (e.g. using BGP-LS). For operation/management reason, each domain uses its own group of redundant PCEs. PCE1/PCE2 in domain 1 have PCEP sessions with PCC1 and PCC3 while PCE3/PCE4 in domain 2 have PCEP sessions with PCC2 and PCC4. As PCE1/2 do not know about LSPs from PCC2/4 and PCE3/4 do not know about LSPs from PCC1/3, there is no possibility to compute the disjointness constraint. This scenario can also be seen as a split-brain scenario. This multi-domain architecture (with multiple groups of PCEs) can also be used in a single domain, where an operator wants to limit the failure domain by creating multiple groups of PCEs maintaining a subset of PCCs. As for the multi-domain example, there

will be no possibility to compute disjoint path starting from head-ends managed by different PCE groups.

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

### 1.3. Applicability to H-PCE

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

## 2. Proposed solution

Our solution is based on :

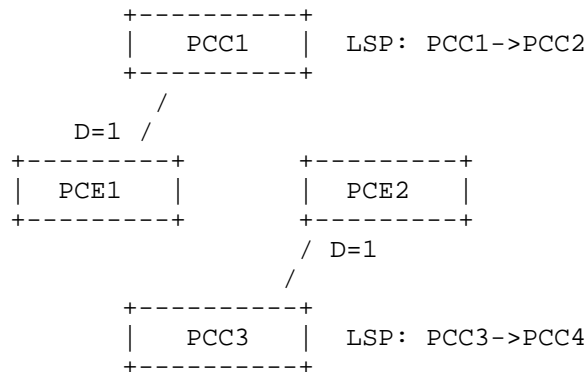
- o The creation of the inter-PCE stateful PCEP session with specific procedures.
- o A Master/Slave relationship between PCEs.

### 2.1. State-sync session

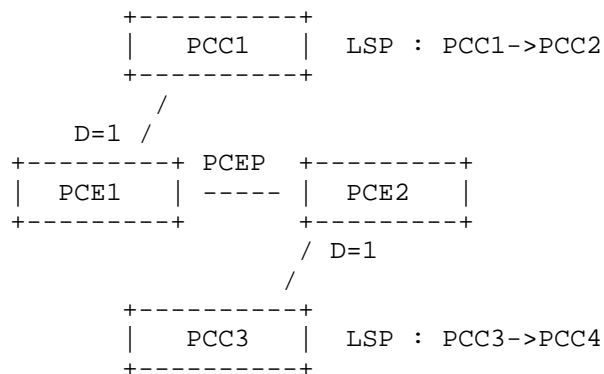
We propose to create a PCEP session between the stateful PCEs. Creating such session is already authorized by multiple scenarios like the one described in [RFC4655] (multiple PCEs that are handling part of the path computation) and [RFC6805] (hierarchical PCE) but was only focused on stateless PCEP sessions.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

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

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



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



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

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

## 2.2. Master/Slave relationship between PCE

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

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

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

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

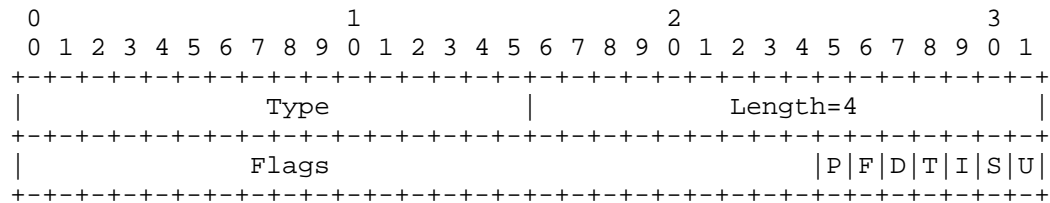
## 3. Procedures and protocol extensions

### 3.1. Opening a state-sync session

#### 3.1.1. Capability advertisement

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

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



This document only updates the Flags field with :

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

The U flag MUST be set when sending the STATEFUL-PCE-CAPABILITY TLV with the P flag set. S flag MAY be set if optimized synchronization is required as per [I-D.ietf-pce-stateful-sync-optimizations].

### 3.2. State synchronization

When the INTER-PCE-CAPABILITY has been negotiated, each PCEP speaker will behave as a PCE and as a PCC at the same time regarding the state synchronization as defined in [I-D.ietf-pce-stateful-pce]. This means that each PCEP Speaker:

- o MUST send a PCRpt message towards its neighbor with S flag set for each LSP in its LSP database learned from a PCC. (PCC role)
- o MUST send the End Of Synchronization Marker towards its neighbor when all LSPs have been reported. (PCC role)
- o MUST wait for the LSP synchronization from its neighbor to end (receiving an End Of Synchronization Marker). (PCE role)

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

Optimized synchronization MAY be used as defined in [I-D.ietf-pce-stateful-sync-optimizations].

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

### 3.3. Incremental updates and report forwarding rules

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

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

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

When a PCE receives a new PCReport from a PCC without the LSP-DB-VERSION, it SHOULD NOT forward the PCReport on any state-sync sessions.

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

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

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

The ORIGINAL-LSP-DB-VERSION TLV is encoded as follows and SHOULD always contain the LSP-DB-VERSION received from the PCC owner of the LSP:

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     |                                     |
|               Type=TBD2             |               Length=8           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     |
|               LSP State DB Version Number                             |
|                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

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

### 3.4. Maintaining LSP states from different sources

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

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

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

the current ORIGINAL-LSP-DB-VERSION of the stored LSP state, it SHOULD add the source speaker address in the source list.

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

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

### 3.5. Computation priority between PCEs and sub-delegation

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

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

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

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



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

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

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

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

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

### 3.6. Passive stateful procedures

In the passive stateful PCE architecture, the PCC is responsible of triggering a path computation request using a PCRequest message to its PCE. Similarly to PCReports which remains unchanged for passive mode, if a PCE receives a PCRequest for an LSP and if this PCE finds

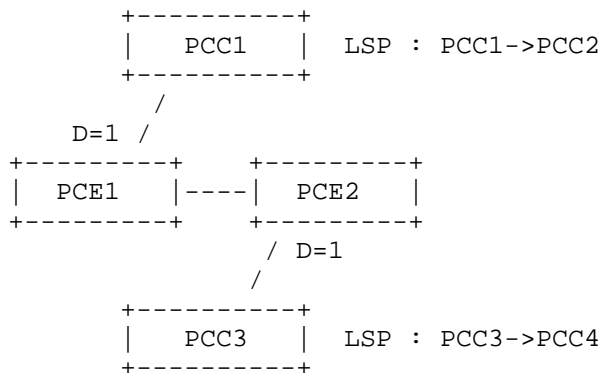
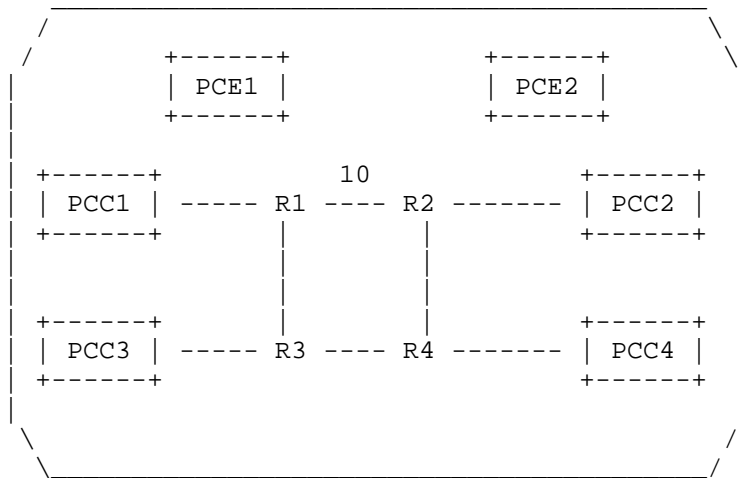
that it does not have the highest computation priority of this LSP, or groups..., it MUST forward the PCRequest to the highest priority PCE over the state-sync session. When the highest priority PCE receives the PCRequest, it computes the path and generates a PCReply only to the PCE that is received the PCRequest from. This PCE will then forward the PCReply to the requesting PCC. The handling of LSP object and the SPEAKER-IDENTITY-TLV in PCRequest and PCReply is similar to PCReport/PCUpdate.

### 3.7. PCE initiation procedures

TBD

## 4. Examples

### 4.1. Example 1



PCE1 computation priority 100  
PCE2 computation priority 200

With this PCEP session topology where computation priority is global for all LSPs, we still want to have link disjoint LSPs PCC1->PCC2 and PCC3->PCC4.

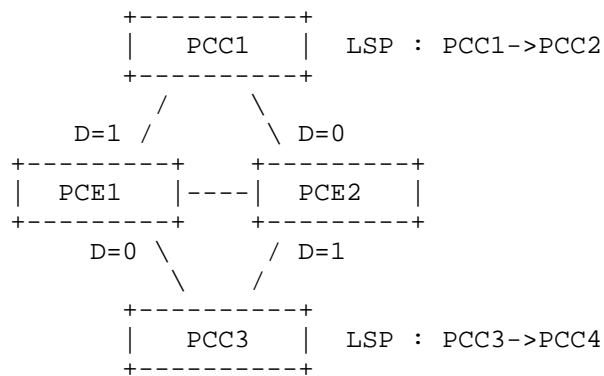
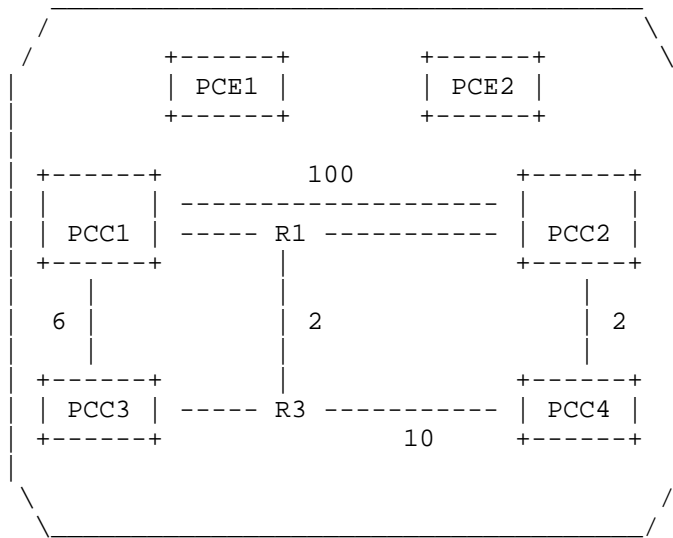
We first configure PCC1->PCC2, PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it will sub-delegate the LSP to PCE2 by sending a PCReport with D=1 and including the SPEAKER-IDENTITY-TLV over the state-sync session. PCE2 receives the PCReport and as it has delegation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

TLV). PCC1 acknowledges the PCUpdate by a PCReport to PCE1. PCE1 forwards the PCReport to PCE2.

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

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

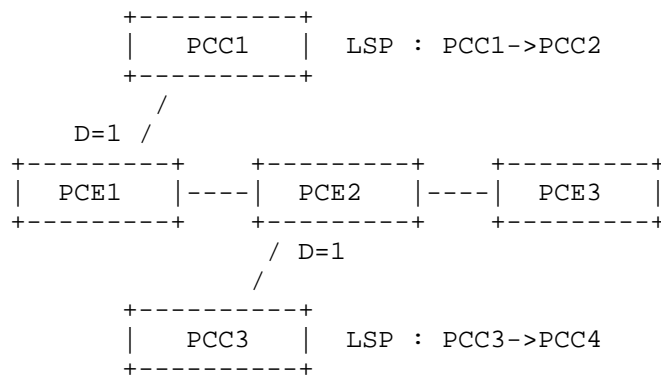
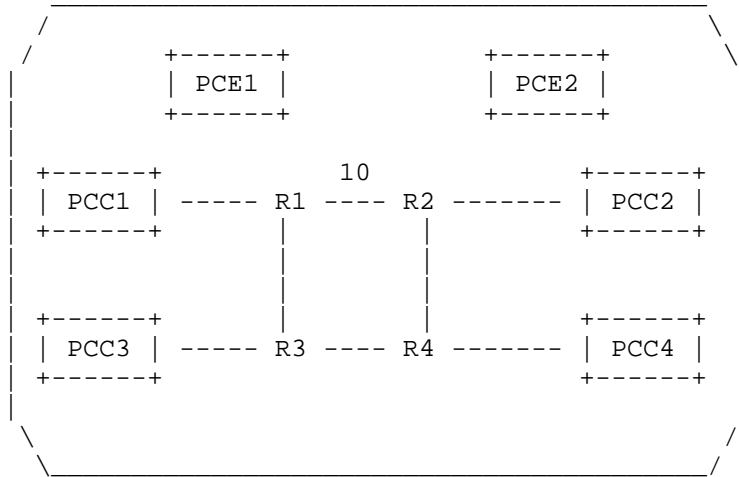
#### 4.2. Example 2



PCE1 computation priority 200  
PCE2 computation priority 100

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

## 4.3. Example 3



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

With this PCEP session topology, we still want to have link disjoint LSPs PCC1->PCC2 and PCC3->PCC4.

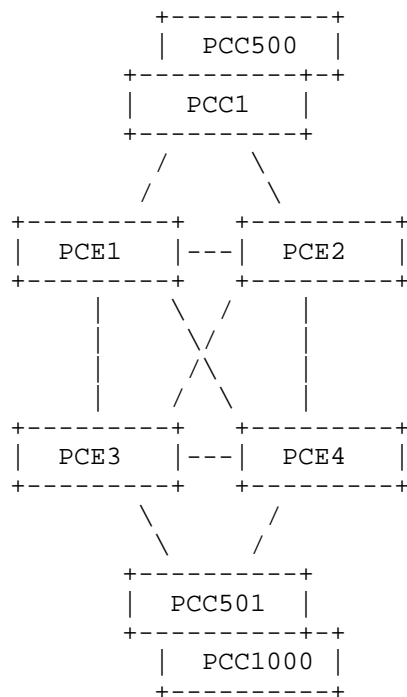
We first configure PCC1->PCC2, PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it will sub-delegate the LSP to PCE2 (as it cannot reach PCE3 through a state-sync session). PCE2 cannot compute a path for PCC1->PCC2 as it does not have the highest priority and cannot sub-delegate the LSP again towards PCE3.

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

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

5. Using Master/Slave computation and state-sync sessions to increase scaling

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



In the figure above, two groups of PCEs are created: PCE1/2 maintain PCEP sessions with PCC1 up to PCC500, while PCE3/4 maintain PCEP sessions with PCC501 up to PCC1000. A granular master/slave policy is setup as follows to loadshare computation between PCEs:

- o PCE1 has priority 200 for association ID 1 up to 300, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.
- o PCE3 has priority 200 for association ID 301 up to 500, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.

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



## 6. PCEP-PATH-VECTOR-TLV

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

This document introduces the PCEP-PATH-VECTOR-TLV (type TBD2) with the following format:

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Type=TBD3                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     PCEP-SPEAKER-INFORMATION#1                               |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     ...                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     PCEP-SPEAKER-INFORMATION#2                               |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     ...                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

The TLV format and padding rules are as per [RFC5440].

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

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Length (variable)      |      ID Length (variable)      |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Speaker Entity identity (variable)      |
+-----+-----+-----+-----+-----+-----+-----+-----+
|      SubTLVs (optional)      |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

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

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

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

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

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

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

## 7. Security Considerations

TBD.

## 8. Acknowledgements

TBD.

## 9. IANA Considerations

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

### 9.1. PCEP-Error Object

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

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

### 9.2. PCEP TLV Type Indicators

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

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

### 9.3. STATEFUL-PCE-CAPABILITY TLV

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

Bit	Description	Reference
TBD	INTER-PCE-CAPABILITY	This document

## 10. References

### 10.1. Normative References

- [I-D.ietf-pce-stateful-pce]  
Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-21 (work in progress), June 2017.
- [I-D.ietf-pce-stateful-sync-optimizations]  
Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", draft-ietf-pce-stateful-sync-optimizations-10 (work in progress), March 2017.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.

### 10.2. Informative References

- [I-D.dhodylee-pce-stateful-hpce]  
Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and O. Dios, "Hierarchical Stateful Path Computation Element (PCE).", draft-dhodylee-pce-stateful-hpce-03 (work in progress), March 2017.
- [I-D.ietf-pce-association-diversity]  
Litkowski, S., Sivabalan, S., Barth, C., and D. Dhody, "Path Computation Element communication Protocol extension for signaling LSP diversity constraint", draft-ietf-pce-association-diversity-01 (work in progress), March 2017.

- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <<https://www.rfc-editor.org/info/rfc6805>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.

## Authors' Addresses

Stephane Litkowski  
Orange

Email: [stephane.litkowski@orange.com](mailto:stephane.litkowski@orange.com)

Siva Sivabalan  
Cisco

Email: [msiva@cisco.com](mailto:msiva@cisco.com)

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

Email: [dhruv.ietf@gmail.com](mailto:dhruv.ietf@gmail.com)

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: August 26, 2021

S. Litkowski  
Cisco  
S. Sivabalan  
Ciena Corporation  
C. Li  
H. Zheng  
Huawei Technologies  
February 22, 2021

Inter Stateful Path Computation Element (PCE) Communication Procedures.  
draft-litkowski-pce-state-sync-10

## Abstract

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

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

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

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

## Requirements Language

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

## Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 26, 2021.

## Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction and Problem Statement . . . . .	3
1.1. Reporting LSP Changes . . . . .	4
1.2. Split-Brain . . . . .	5
1.3. Applicability to H-PCE . . . . .	12
2. Proposed solution . . . . .	12
2.1. State-sync session . . . . .	12

2.2. Primary/Secondary relationship between PCE . . . . .	14
3. Procedures and Protocol Extensions . . . . .	14
3.1. Opening a state-sync session . . . . .	14
3.1.1. Capability Advertisement . . . . .	14
3.2. State synchronization . . . . .	15
3.3. Incremental updates and report forwarding rules . . . . .	16
3.4. Maintaining LSP states from different sources . . . . .	17
3.5. Computation priority between PCEs and sub-delegation . . . . .	18
3.6. Passive stateful procedures . . . . .	19
3.7. PCE initiation procedures . . . . .	20
4. Examples . . . . .	20
4.1. Example 1 . . . . .	20
4.2. Example 2 . . . . .	22
4.3. Example 3 . . . . .	24
5. Using Primary/Secondary Computation and State-sync Sessions to increase Scaling . . . . .	25
6. PCEP-PATH-VECTOR TLV . . . . .	27
7. Security Considerations . . . . .	28
8. Acknowledgements . . . . .	28
9. IANA Considerations . . . . .	28
9.1. PCEP-Error Object . . . . .	28
9.2. PCEP TLV Type Indicators . . . . .	29
9.3. STATEFUL-PCE-CAPABILITY TLV . . . . .	29
10. References . . . . .	29
10.1. Normative References . . . . .	29
10.2. Informative References . . . . .	30
Appendix A. Contributors . . . . .	31
Authors' Addresses . . . . .	31

## 1. Introduction and Problem Statement

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

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

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

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

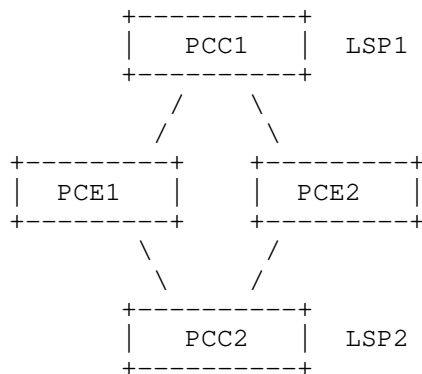
### 1.1. Reporting LSP Changes

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

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

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

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



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



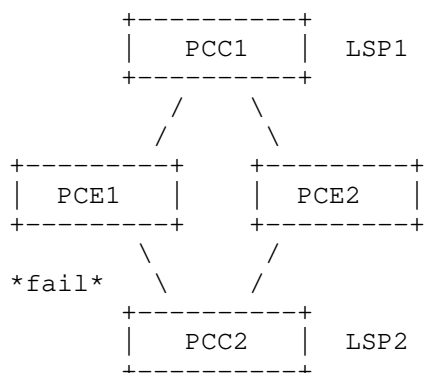
be notified of the change only after receiving the PCRpt message from PCC1.

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

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

### 1.2. Split-Brain

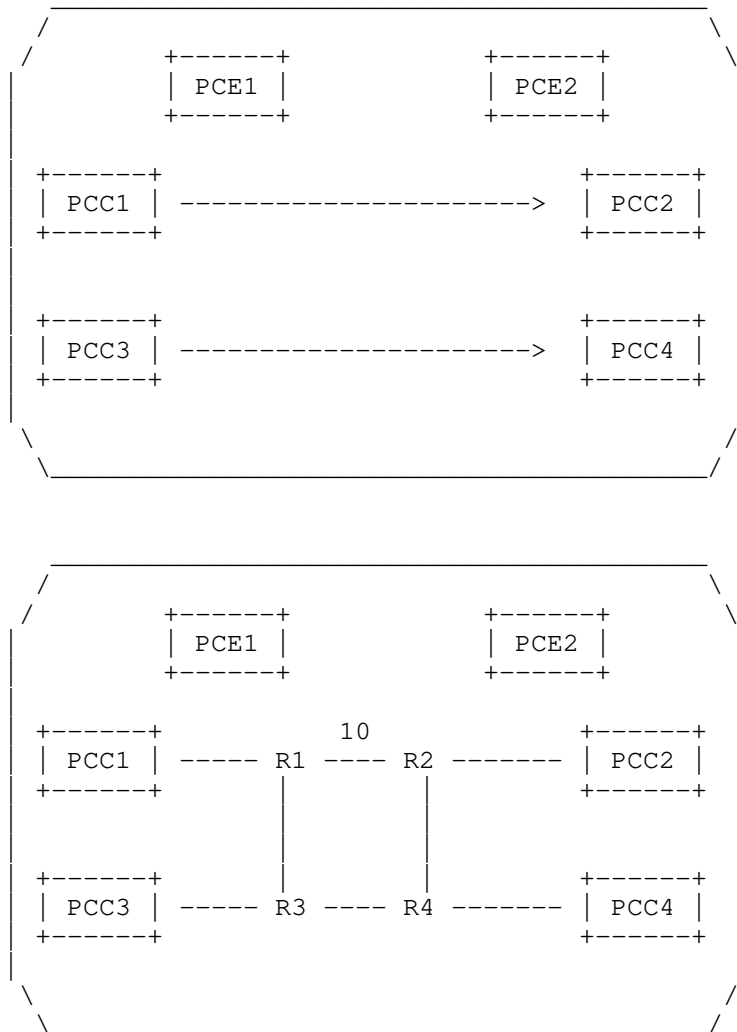
In a resiliency case, a PCC has redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control on an LSP to a single PCE only, and only this PCE is responsible for the path computation for the delegated LSP: the PCC achieves this by setting the D flag only towards the active PCE [RFC8231]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



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

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

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



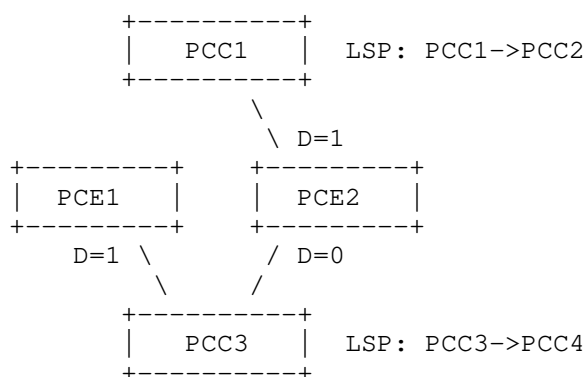
In the figure above, the requirement is to create two link-disjoint LSPs: PCC1->PCC2 and PCC3->PCC4. In the topology, all links cost metric is set to 1 except for the link 'R1-R2' which has a metric of 10. The PCEs are responsible for the path computation and PCE1 is the active primary PCE for all PCCs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

Scenario 1:

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

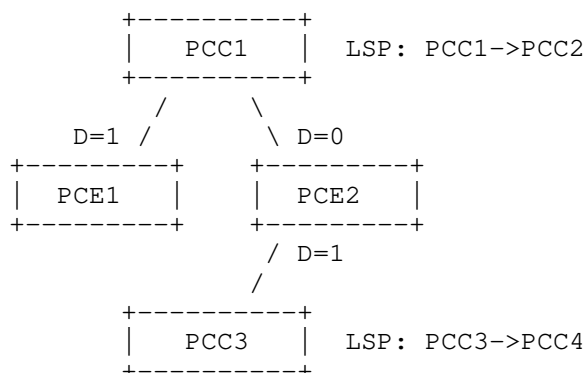
#### Scenario 2:

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



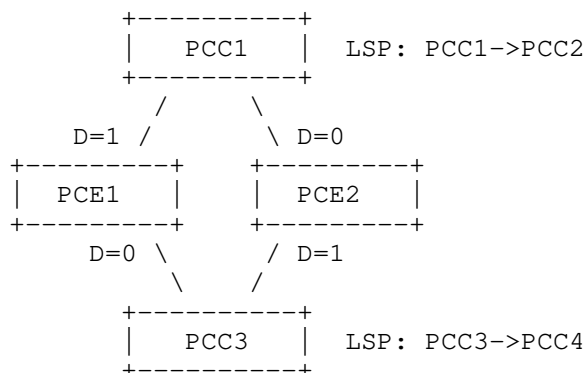
Consider that the PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE2 (which is the new active primary PCE for PCC1) sends a PCUpd message to PCC1 with the ERO: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE1 is not aware of LSPs from PCC1 any more,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

#### Scenario 3:



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

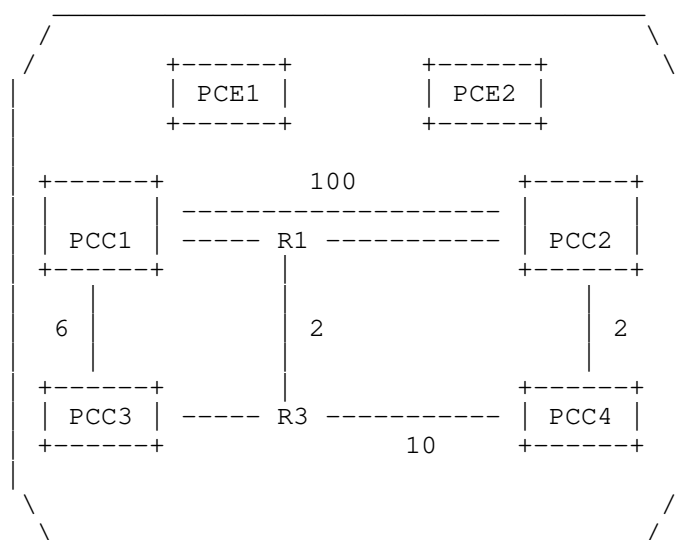
Scenario 4:



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

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

Scenario 5:

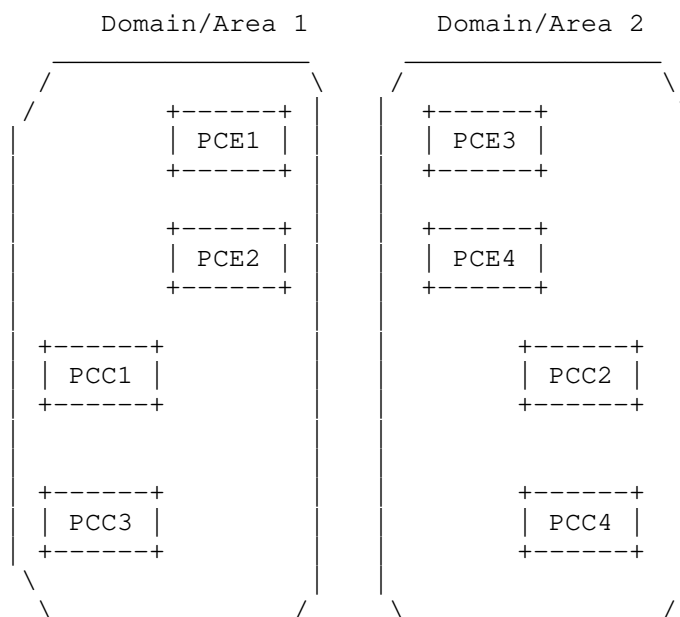


Now, consider a new network topology with the same PCEP sessions as the previous example. Suppose that both LSPs are configured almost at the same time. PCE1 will compute a path for PCC1->PCC2 while PCE2 will compute a path for PCC3->PCC4. As each PCE is not aware of the path of the second LSP in the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

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

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

Scenario 6: multi-domain



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

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

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

### 1.3. Applicability to H-PCE

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

## 2. Proposed solution

Our solution is based on :

- o The creation of the inter-PCE stateful PCEP session with specific procedures.
- o A Primary/Secondary relationship between PCEs.

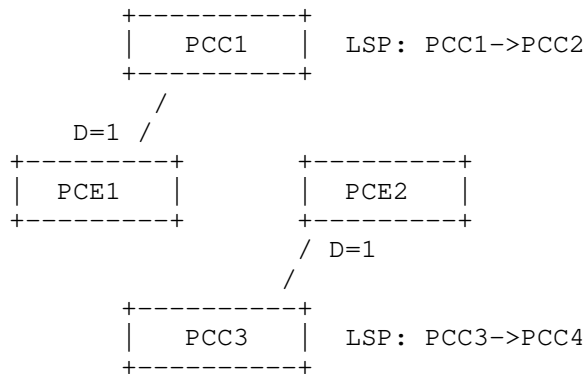
### 2.1. State-sync session

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

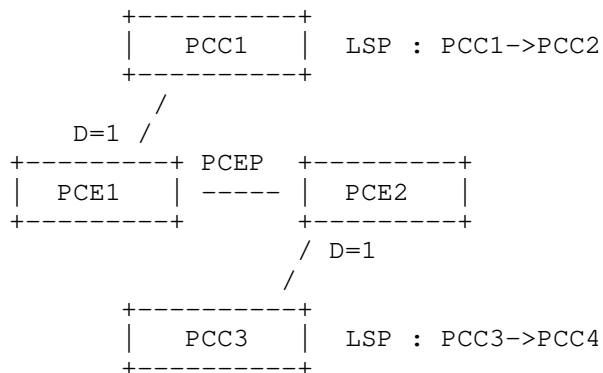
This inter-PCE PCEP session will allow the exchange of LSP states between PCEs that would help some scenarios where PCEP sessions are lost between PCC and PCE. This inter-PCE PCEP session is henceforth called a state-sync session.

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





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



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

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

## 2.2. Primary/Secondary relationship between PCE

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

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

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

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

## 3. Procedures and Protocol Extensions

### 3.1. Opening a state-sync session

#### 3.1.1. Capability Advertisement

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

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

P (INTER-PCE-CAPABILITY - 1 bit - TBD4): If set to 1 by a PCEP Speaker, the PCEP speaker indicates that the session MUST follow the state-sync procedures as described in this document. The P bit MUST be set by both speakers: if a PCEP Speaker receives a STATEFUL-PCE-CAPABILITY TLV with P=0 while it advertised P=1 or if both set P flag to 0, the session SHOULD be set-up but the state-sync procedures MUST NOT be applied on this session.

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

### 3.2. State synchronization

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

- o MUST send a PCRpt message towards its neighbor with S flag set for each LSP in its LSP database learned from a PCC. (PCC role)
- o MUST send the End Of Synchronization Marker towards its neighbor when all LSPs have been reported. (PCC role)
- o MUST wait for the LSP synchronization from its neighbor to end (receiving an End Of Synchronization Marker). (PCE role)

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

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

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

### 3.3. Incremental updates and report forwarding rules

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

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

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

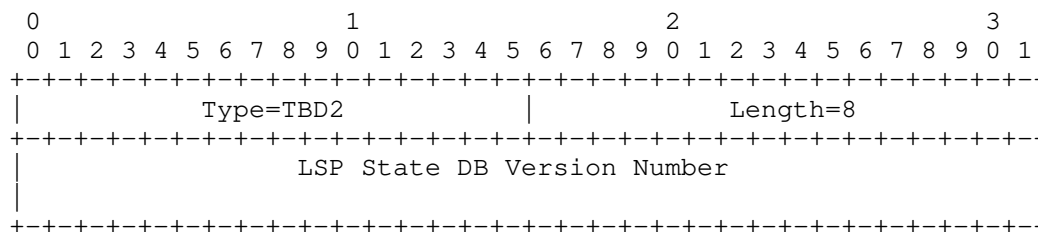
When a PCE receives a new PCRpt from a PCC without the LSP-DB-VERSION, it SHOULD NOT forward the PCRpt on any state-sync sessions and log such an event on the first occurrence.

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

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

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

The ORIGINAL-LSP-DB-VERSION TLV is encoded as follows and MUST always contain the LSP-DB-VERSION received from the owner PCC of the LSP:



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

### 3.4. Maintaining LSP states from different sources

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

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

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

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

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

### 3.5. Computation priority between PCEs and sub-delegation

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

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

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

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

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

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

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

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

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

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

### 3.6. Passive stateful procedures

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

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

### 3.7. PCE initiation procedures

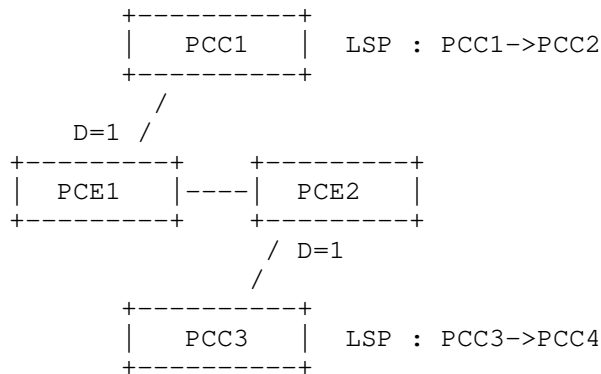
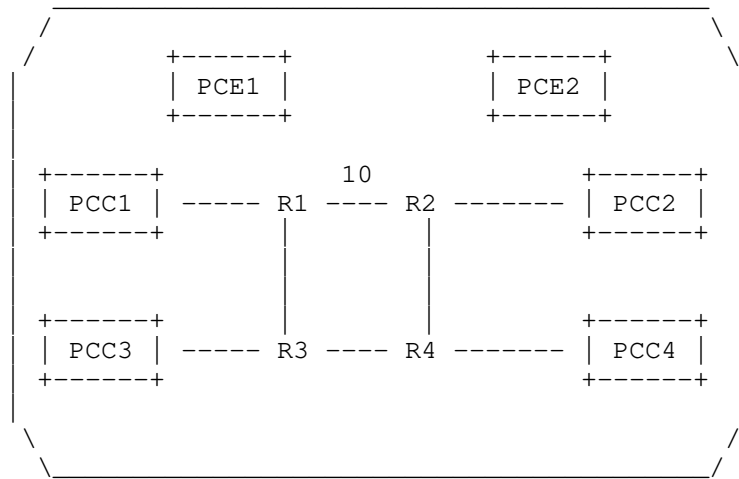
It is possible that a PCE does not have a PCEP session with the headend to initiate a LSP as per [RFC8281]. A PCE could send the PCInitiate message on the state-sync sessions to other PCE to request it to create a PCE-Initiated LSP on its behalf. If the PCE is able to initiate the LSP it would report it on the state-sync session via PCRpt message. If the PCE does not have a session to the headend, it MUST send a PCErr message with Error-type=24 (PCE instantiation error) and Error-value=TBD5 (No PCEP session with the headend). PCE could try to initiate via another state-sync PCE if available.

## 4. Examples

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

### 4.1. Example 1





PCE1 computation priority 100  
PCE2 computation priority 200

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

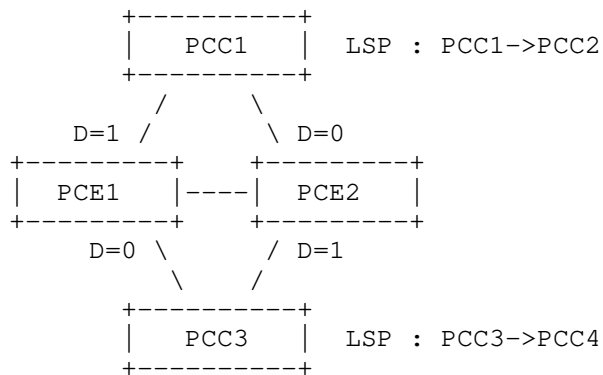
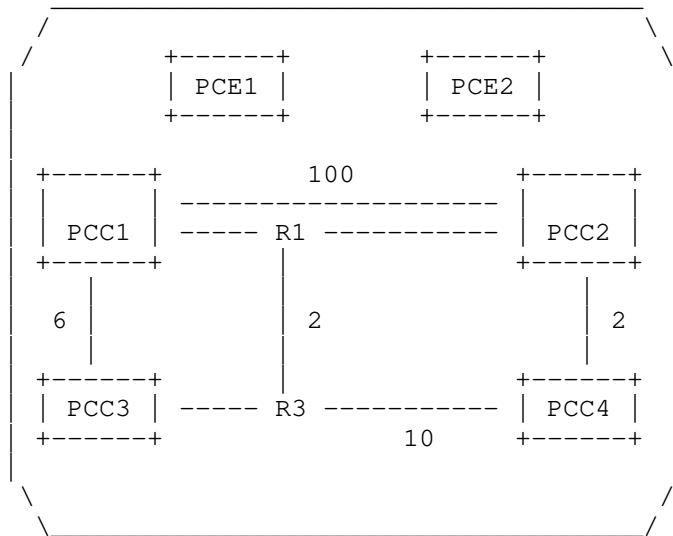
Consider the PCC1->PCC2 is configured first and PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it sub-delegates the LSP to PCE2 by sending a PCRpt with D=1 and including the SPEAKER-IDENTITY-TLV over the state-sync session. PCE2 receives the PCRpt and as it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

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

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

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

#### 4.2. Example 2

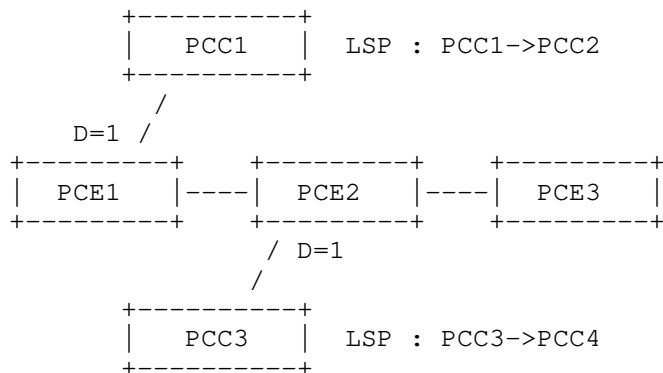
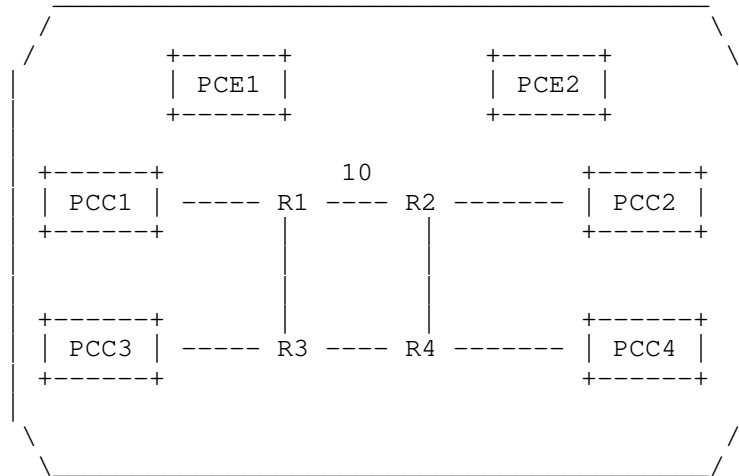


PCE1 computation priority 200

PCE2 computation priority 100

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

## 4.3. Example 3



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

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

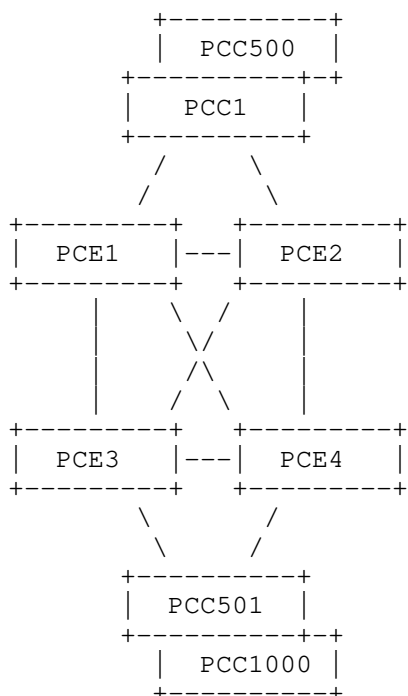
Suppose PCC1->PCC2 is configured first, PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it will sub-delegate the LSP to PCE2 (as it not aware of PCE3 and has no way to reach it). PCE2 cannot compute a path for PCC1->PCC2 as it does not have the highest priority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

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

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

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

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



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

- o PCE1 has priority 200 for association ID 1 up to 300, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.
- o PCE3 has priority 200 for association ID 301 up to 500, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.

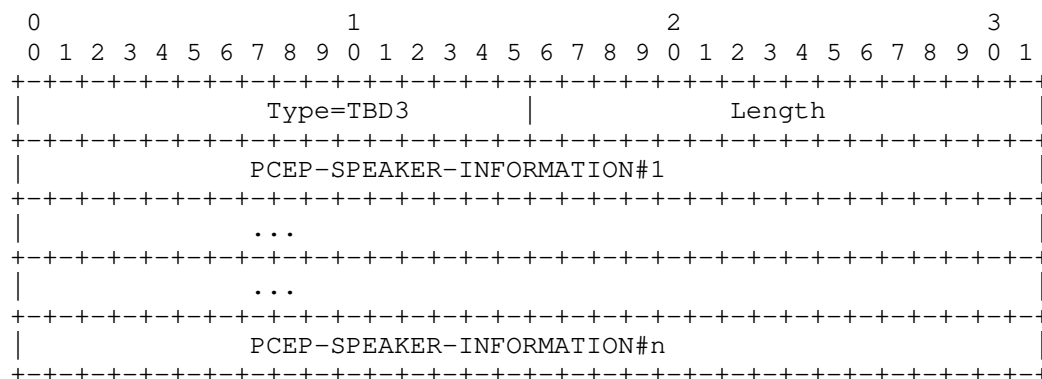
If some PCCs delegate LSPs with association ID 1 up to 300 and association source 0.0.0.0, the receiving PCE (if not PCE1) will sub-delegate the LSPs to PCE1. PCE1 becomes responsible for the computation of these LSP associations while PCE3 is responsible for the computation of another set of associations.

The procedures described in this document could help greatly in load-sharing between a group of stateful PCEs.

## 6. PCEP-PATH-VECTOR TLV

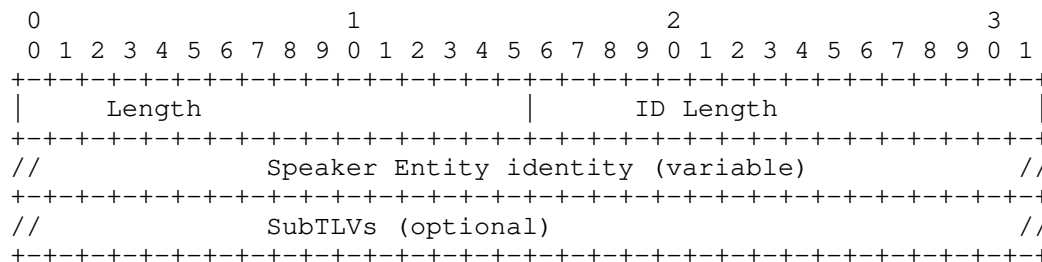
This document allows PCEP messages to be propagated among PCEP speaker. It may be useful to track information about the propagation of the messages. One of the use cases is a message loop detection mechanism, but other use cases like hop by hop information recording may also be implemented.

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



The TLV format and padding rules are as per [RFC5440].

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



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

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

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

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

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

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

## 7. Security Considerations

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

## 8. Acknowledgements

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

## 9. IANA Considerations

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

### 9.1. PCEP-Error Object

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



Error-Type	Meaning	Reference
6	Mandatory Object Missing Error-value=TBD1: SPEAKER-IDENTITY-TLV missing	[RFC5440] This document
24	LSP instantiation error Error-value=TBD5: No PCEP session with the headend	[RFC8281] This document

## 9.2. PCEP TLV Type Indicators

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

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

## 9.3. STATEFUL-PCE-CAPABILITY TLV

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

Bit	Description	Reference
TBD4	INTER-PCE-CAPABILITY	This document

## 10. References

### 10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.

## 10.2. Informative References

- [I-D.knodel-terminology] Knodel, M. and N. Oever, "Terminology, Power, and Inclusive Language in Internet-Drafts and RFCs", draft-knodel-terminology-04 (work in progress), August 2020.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <<https://www.rfc-editor.org/info/rfc4655>>.
- [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <<https://www.rfc-editor.org/info/rfc6805>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.

- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8751] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., and D. King, "Hierarchical Stateful Path Computation Element (PCE)", RFC 8751, DOI 10.17487/RFC8751, March 2020, <<https://www.rfc-editor.org/info/rfc8751>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.

#### Appendix A. Contributors

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

Email: [dhruv.ietf@gmail.com](mailto:dhruv.ietf@gmail.com)

#### Authors' Addresses

Stephane Litkowski  
Cisco

Email: [slitkows.ietf@gmail.com](mailto:slitkows.ietf@gmail.com)

Siva Sivabalan  
Ciena Corporation

Email: msiva282@gmail.com

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

Email: c.l@huawei.com

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

Email: zhenghaomian@huawei.com

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 21, 2018

M. Negi  
P. Kaladharan  
D. Dhody  
Huawei Technologies  
S. Sivabalan  
Cisco Systems  
October 18, 2017

PCEP Extensions for Segment Routing leveraging the IPv6 data plane  
draft-negi-pce-segment-routing-ipv6-00

Abstract

The Source Packet Routing in Networking (SPRING) architecture describes how Segment Routing (SR) can be used to steer packets through an IPv6 or MPLS network using the source routing paradigm. Segment Routing (SR) enables any head-end node to select any path without relying on a hop-by-hop signaling technique (e.g., LDP or RSVP-TE).

It depends only on "segments" that are advertised by Link- State Interior Gateway Protocols (IGPs). A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE).

Since, Segment Routing can be applied to both MPLS and IPv6 forwarding plane, a PCE should be able to compute SR-Path for both MPLS and IPv6 forwarding plane. This draft describes the extensions required for Segment Routing support for IPv6 data plane in PCEP.

Requirements Language

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

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute

working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 21, 2018.

#### Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Overview of PCEP Operation in SRv6 Networks . . . . .	5
3.1. Operation Overview . . . . .	6
3.2. SRv6-Specific PCEP Message Extensions . . . . .	6
3.3. Object Formats . . . . .	6
3.3.1. The OPEN Object . . . . .	6
3.3.1.1. The SRv6 PCE Capability TLV . . . . .	6
3.3.1.2. Exchanging SRv6 Capability . . . . .	7
3.3.2. The RP/SRP Object . . . . .	7
3.3.3. ERO Object . . . . .	8
3.3.3.1. SR-ERO Subobject . . . . .	8
3.3.3.2. ERO Processing . . . . .	10
3.3.4. RRO Object . . . . .	11
3.3.4.1. SR-RRO Subobject . . . . .	11
3.4. Security Considerations . . . . .	11
3.5. IANA Considerations . . . . .	12
3.5.1. PCEP Objects . . . . .	12
3.5.1.1. ERROR Objects . . . . .	12
3.5.1.2. TLV Type Indicators . . . . .	12
3.5.1.3. New Path Setup Type . . . . .	12

4. References . . . . .	13
4.1. Normative References . . . . .	13
4.2. Informative References . . . . .	14
Appendix A. Contributor . . . . .	16
Authors' Addresses . . . . .	16

## 1. Introduction

As per [I-D.ietf-spring-segment-routing], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based. A segment can have a semantic local to an SR node or global within an SR domain. SR allows to enforce a flow through any path and service chain while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locators, etc. The list of segment forming the path is called the Segment List and is encoded in the packet header. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [I-D.ietf-6man-segment-routing-header]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment.

Segment Routing use cases are described in [RFC7855] and [I-D.ietf-spring-ipv6-use-cases]. Segment Routing protocol extensions are defined in [I-D.ietf-isis-segment-routing-extensions], and [I-D.ietf-ospf-ospfv3-segment-routing-extensions].

As per [I-D.ietf-6man-segment-routing-header], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in An illustration is provided in [I-D.ietf-spring-srv6-network-programming].

The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). The SR is applied to IPV6 forwarding plane using SRH. A SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool or an and provisioned on the ingress node.

[RFC5440] describes Path Computation Element Protocol (PCEP) for communication between a Path Computation Client (PCC) and a Path Computation Element (PCE) or between one a pair of PCEs. A PCE or a

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

A mechanism to dynamically initiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE is specified in [I-D.ietf-pce-pce-initiated-lsp]. As per [I-D.ietf-pce-segment-routing], it is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [I-D.ietf-pce-pce-initiated-lsp] using the SR specific PCEP extensions specified in [I-D.ietf-pce-segment-routing]. [I-D.ietf-pce-segment-routing] specifies PCEP extensions for supporting a SR-TE LSP for MPLS data plane. This document extends [I-D.ietf-pce-segment-routing] to support SR for IPv6 data plane. Additionally, using procedures described in this document, a PCC can request an SRv6 path from either stateful or a stateless PCE. This specification relies on the PATH-SETUP-TYPE TLV and procedures specified in [I-D.ietf-pce-lsp-setup-type].

## 2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [RFC8051]: Stateful PCE, Delegation.

The message formats in this document are specified using Routing Backus-Naur Format (RBNF) encoding as specified in [RFC5511].

PCC: Path Computation Client.

PCE: Path Computation Element.

PCEP: Path Computation Element Protocol.

SR: Segment Routing.



SID: Segment Identifier.

SRv6: Segment Routing for IPv6 forwarding plane.

SRH: IPv6 Segment Routing Header.

SR Path: IPv6 Segment (List of IPv6 SID representing a path in IPv6 SR domain)

### 3. Overview of PCEP Operation in SRv6 Networks

Basic operations for PCEP speakers is as per [I-D.ietf-pce-segment-routing]. SRv6 Paths computed by a PCE can be represented as an ordered list of SRv6 segments of 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment".

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

This document extends the "SR-ERO subobject" defined in [I-D.ietf-pce-segment-routing] to carry IPv6 SID(s) (IPv6 Addresses). SRv6-capable PCEP speakers should be able to generate and/or process this.

When a PCEP session between a PCC and a PCE is established, both PCEP speakers exchange their capabilities to indicate their ability to support SRv6 specific functionality.

In summary, this document defines new path setup type carried in the PATH-SETUP-TYPE TLV for SRv6 path.

In summary, this document:

- o Defines a new PCEP capability for SRv6
- o Update the SR-ERO and SR-RR0 sub-object for SRv6
- o Defines a new path setup type carried in the PATH-SETUP-TYPE TLV for SR-TE LSP.

### 3.1. Operation Overview

In SR networks, an ingress node of an SR path appends all outgoing packets with an SR header consisting of a list of SIDs (IPv6 Prefix in case of SRv6). The header has all necessary information to guide the packets from the ingress node to the egress node of the path, and hence there is no need for any signaling protocol.

For IPv6 in control plane with MPLS data-plane, mechanism remains same as [I-D.ietf-pce-segment-routing]

This document describes extensions to SR path for IPv6 data plane. SRv6 Path (i.e. ERO) consists of an ordered set of SIDs (see details in Figure 2).

A PCC or PCE indicates its ability to support SRv6 during the PCEP session Initialization Phase via a new SRv6-PCE-CAPABILITY TLV (see details in Section 3.3.1.1).

### 3.2. SRv6-Specific PCEP Message Extensions

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable length body made up of mandatory and/or optional objects. This document does not require any changes in the format of PCReq and PCRep messages specified in [RFC5440], PCInitiate message specified in [I-D.ietf-pce-pce-initiated-lsp], and PCRpt and PCUpd messages specified in [RFC8231]. However, PCEP messages pertaining to SRv6 MUST include PATH-SETUP-TYPE TLV in the RP or SRP object to clearly identify that SRv6 is intended. In other words, a PCEP speaker MUST NOT infer whether or not a PCEP message pertains to SRv6 from any other object or TLV.

### 3.3. Object Formats

#### 3.3.1. The OPEN Object

This document defines a new optional TLV for use in the OPEN Object.

##### 3.3.1.1. The SRv6 PCE Capability TLV

The SRv6-PCE-CAPABILITY TLV is an optional TLV associated with the OPEN Object to exchange SRv6 capability of PCEP speakers. The format of the SR-PCE-CAPABILITY TLV is shown in the following figure:

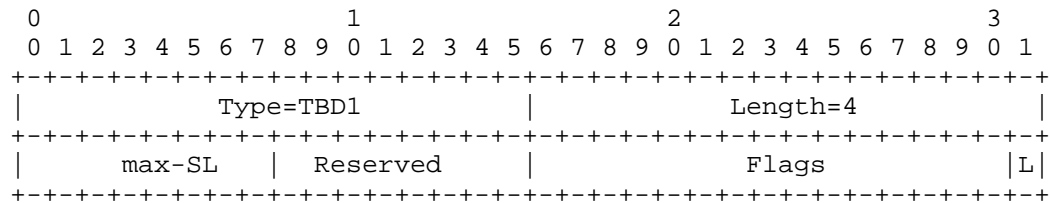


Figure 1: SRv6-PCE-CAPABILITY TLV format

The code point for the TLV type is to be defined by IANA. The TLV length is 4 octets.

The 4-octet value comprise of -

max-SL: 1 octet, this field specifies the maximum value of the "Segments Left (SL)" in the SRH [I-D.ietf-6man-segment-routing-header].

Reserved: 1 octet, this field MUST be set to 0 on transmission, and ignored on receipt.

Flags: 2 octet, one flag is currently defined in this document.

L bit: A PCC sets this bit to 1 to indicate that it does not impose any limit on SL.

### 3.3.1.2. Exchanging SRv6 Capability

By including the SRv6-PCE-CAPABILITY TLV in the OPEN message destined to a PCE, a PCC indicates that it is capable of supporting the head-end functions for SRv6. By including the TLV in the OPEN message destined to a PCC, a PCE indicates that it is capable of computing SRv6 paths.

### 3.3.2. The RP/SRP Object

In order to indicate the SRv6 path, RP or SRP object MUST include the PATH-SETUP-TYPE TLV specified in [I-D.ietf-pce-lsp-setup-type]. This document defines a new Path Setup Type (PST) for SRv6 as follows:

- o PST = TBD2: Path is setup using SRv6 technique.

### 3.3.3. ERO Object

In order to support SRv6, the SR-ERO subobject is used [I-D.ietf-pce-segment-routing]. All other processing rules remains the same.

#### 3.3.3.1. SR-ERO Subobject

For supporting SRv6, a new SID Type (ST) is defined, the format of SR-ERO sub object remains the same as defined in [I-D.ietf-pce-segment-routing].

When the SID Type (ST) indicates SRv6, then the SR-ERO subobject represent a SRv6 segment and include a field SRv6I (SRv6 Identifier) in place of NAI (Node or Adjacency Identifier) defined in [I-D.ietf-pce-segment-routing]. The 32 bit SID MUST be set to zero on transit and ignored on receipt. The format of SR-ERO subobject is reproduced with the SRv6I field as shown below:

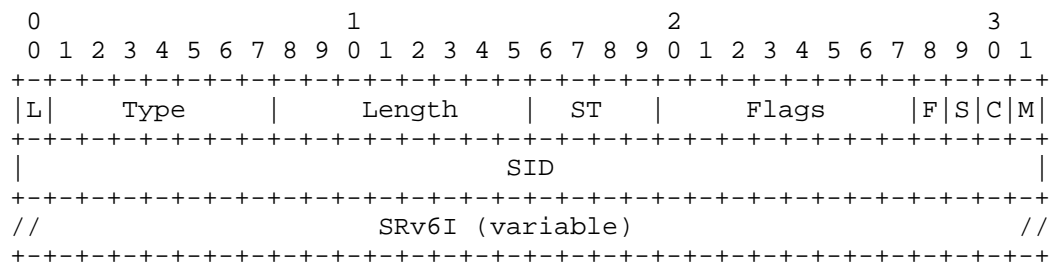


Figure 2: SR-ERO Subobject Format

The description of all the flags and fields is as per [I-D.ietf-pce-segment-routing].

For SRv6 segments, a new ST (SID Type) is assigned by IANA as TBD.

For SRv6 segments (when ST is TBD), M and C flag MUST NOT be set. The S flag MUST be set and SID field MUST be set to 0. The F bit MUST NOT be set. The Length is 28.

The SRv6I format is as shown below:

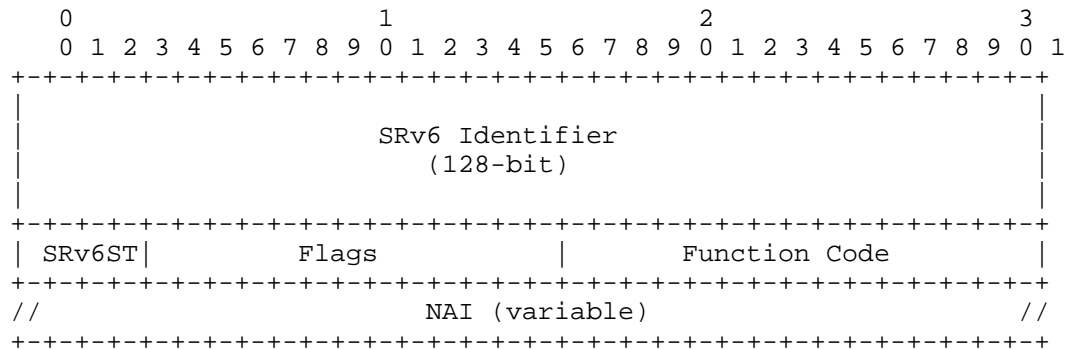


Figure 3: SR-ERO Subobject's SRv6I Format

SRv6 Identifier is the 128 bit IPv6 addresses representing SRv6 segment.

SRv6ST is the SRv6 SID Type which indicates the interpretation for NAI (Node or Adjacency Identifier) as per [I-D.ietf-pce-segment-routing].

Flags is the 12 bit field, no flag bits are currently defined in this document.

Function Code is is the 16 bit field representing supported functions. associated with SRv6 SIDs. See [I-D.filsfils-spring-srv6-network-programming]. Following function codes are currently defined -

- 0: Reserved
- 1: End Function
- 2: End.DX6 Function
- 3: End.DT6 Function
- 4: End.X Function

NAI field [I-D.ietf-pce-segment-routing] contains the NAI associated with the SRv6 Identifier. Depending on the value of SRv6ST, the NAI can have different formats.

When SRv6ST value is 1, the NAI is as per the 'IPv6 Node ID' format defined in [I-D.ietf-pce-segment-routing], which specify

an IPv6 address. This is used to identify the owner of the SRv6 Identifier.

When SRv6ST value is 2, the NAI is as per the 'IPv6 Adjacency' format defined in [I-D.ietf-pce-segment-routing], which specify a pair of IPv6 addresses. This is used to identify the IPv6 Adjacency and used with the SRv6 Adj-SID.

Note that when SRv6ST value is 0, NAI is not included and MUST be NULL.

### 3.3.3.2. ERO Processing

The ERO and SR-ERO subobject processing remains as per [RFC5440] and [I-D.ietf-pce-segment-routing].

The ST MUST only be TDB, if the SRv6-PCE-CAPABILITY TLV is exchanged with the PCEP peer. In case a PCEP speaker receives the SR-ERO subobject with ST indicating SRv6 segment, when the SRv6-PCE-CAPABILITY TLV was not exchanged, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD3 ("Attempted SRv6 when the capability was not advertised"). A PCEP speaker that does not recognize the ST value, it would behave as per [I-D.ietf-pce-segment-routing].

[Editor's Note - this is missing from [I-D.ietf-pce-segment-routing].]

If a PCC receives a list of SRv6 segments, and the number of SRv6 segments exceeds the max-SL that the PCC can impose on the packet (SRH), it MAY send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Unsupported number of Segment ERO subobjects") as per [I-D.ietf-pce-segment-routing].

When a PCEP speaker detects that all subobjects of ERO are not identical to SRv6, and if it does not handle such ERO, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Non-identical ERO subobjects") as per [I-D.ietf-pce-segment-routing].

When a PCEP speaker receives an SR-ERO subobject for SRv6 segment, M, C and F flag MUST NOT be set and S flag MUST be set. Otherwise, it MUST consider the entire ERO object invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Malformed object") as per [I-D.ietf-pce-segment-routing]. The PCEP speaker MAY include the malformed SR-ERO object in the PCErr message as well.

### 3.3.4. RRO Object

In order to support SRv6, the SR-ERO Subobject is used [I-D.ietf-pce-segment-routing]. All other processing rules remains the same.

#### 3.3.4.1. SR-RRO Subobject

For SRv6 segments, a new ST (SID Type) is assigned by IANA as TBD, the format of SR-ERO sub object remains the same as defined in [I-D.ietf-pce-segment-routing].

When the SID Type (ST) indicates SRv6, then the SR-RRO subobject represent a SRv6 segment and include a field SRv6I (SRv6 Identifier) in place of NAI (Node or Adjacency Identifier) defined in [I-D.ietf-pce-segment-routing]. The 32 bit SID MUST be set to zero on transit and ignored on receipt. The format of SR-RRO subobject is reproduced with the SRv6I field as shown below:

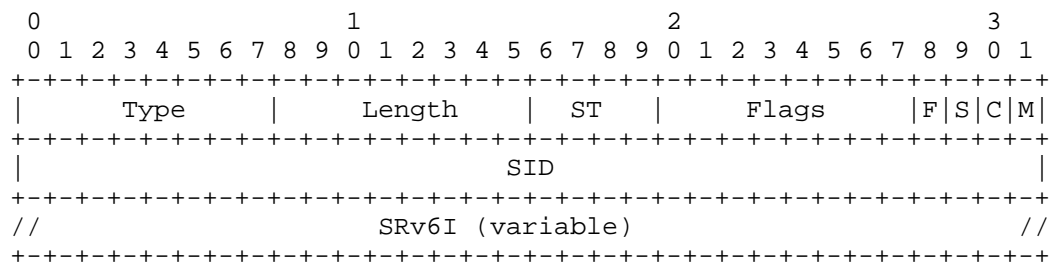


Figure 4: SR-RRO Subobject Format

The description of all fields and flags is as per SR-ERO subobject.

Processing rules of SR-RRO subobject are identical to those of SR-ERO subobject.

### 3.4. Security Considerations

The security considerations described in [RFC5440], [RFC8231] and [I-D.ietf-pce-pce-initiated-lsp] are applicable to this specification. No additional security measure is required.

### 3.5. IANA Considerations

This document requests IANA to include (I) bit in flags registry for SR-ERO and SR-RRO sub-objects. Other changes are defined as:

#### 3.5.1. PCEP Objects

##### 3.5.1.1. ERROR Objects

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

Error-Type	Meaning
-----	-----
19	Invalid Operation Error-value = TBD3 (Attempted SRv6 when the capability was not advertised)

##### 3.5.1.2. TLV Type Indicators

IANA is requested to make the assignment of the new code points for the existing "PCEP TLV Type Indicators" registry as follows:

Value	Meaning	Reference
-----	-----	-----
TBD1	SRv6-PCE-CAPABILITY TLV	This Document

##### 3.5.1.3. New Path Setup Type

[I-D.ietf-pce-lsp-setup-type] defines the PATH-SETUP-TYPE TLV and requests that IANA creates a registry to manage the value of the PATH-SETUP-TYPE TLV's PST field. IANA is requested to allocate a new code point in the PCEP PATH\_SETUP\_TYPE TLV PST field registry, as follows:

Value	Description	Reference
-----	-----	-----
TBD2	SRv6 (SRH) technique	This Document



## 4. References

### 4.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [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>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [I-D.ietf-pce-segment-routing]  
Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-10 (work in progress), October 2017.
- [I-D.ietf-spring-segment-routing]  
Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-12 (work in progress), June 2017.
- [I-D.ietf-pce-pce-initiated-lsp]  
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-11 (work in progress), October 2017.

[I-D.ietf-pce-lsp-setup-type]

Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying path setup type in PCEP messages", draft-ietf-pce-lsp-setup-type-04 (work in progress), April 2017.

#### 4.2. Informative References

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

[RFC7855] Previdi, S., Ed., Filsfils, C., Ed., Decraene, B., Litkowski, S., Horneffer, M., and R. Shakir, "Source Packet Routing in Networking (SPRING) Problem Statement and Requirements", RFC 7855, DOI 10.17487/RFC7855, May 2016, <<https://www.rfc-editor.org/info/rfc7855>>.

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

[I-D.ietf-6man-segment-routing-header]

Previdi, S., Filsfils, C., Raza, K., Leddy, J., Field, B., daniel.voyer@bell.ca, d., daniel.bernier@bell.ca, d., Matsushima, S., Leung, I., Linkova, J., Aries, E., Kosugi, T., Vyncke, E., Lebrun, D., Steinberg, D., and R. Raszuk, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-07 (work in progress), July 2017.

[I-D.ietf-spring-ipv6-use-cases]

Brzozowski, J., Leddy, J., Filsfils, C., Maglione, R., and M. Townsley, "IPv6 SPRING Use Cases", draft-ietf-spring-ipv6-use-cases-11 (work in progress), June 2017.

[I-D.ietf-isis-segment-routing-extensions]

Previdi, S., Filsfils, C., Bashandy, A., Gredler, H., Litkowski, S., Decraene, B., and j. jeffrant@gmail.com, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-13 (work in progress), June 2017.

[I-D.ietf-ospf-ospfv3-segment-routing-extensions]

Psenak, P., Previdi, S., Filsfils, C., Gredler, H.,  
Shakir, R., Henderickx, W., and J. Tantsura, "OSPFv3  
Extensions for Segment Routing", draft-ietf-ospf-ospfv3-  
segment-routing-extensions-10 (work in progress),  
September 2017.

[I-D.filsfils-spring-srv6-network-programming]

Filsfils, C., Leddy, J., daniel.voyer@bell.ca, d.,  
daniel.bernier@bell.ca, d., Steinberg, D., Raszuk, R.,  
Matsushima, S., Lebrun, D., Decraene, B., Peirens, B.,  
Salsano, S., Naik, G., Elmalky, H., Jonnalagadda, P.,  
Sharif, M., Ayyangar, A., Mynam, S., Henderickx, W.,  
Bashandy, A., Raza, K., Dukes, D., Clad, F., and P.  
Camarillo, "SRv6 Network Programming", draft-filsfils-  
spring-srv6-network-programming-01 (work in progress),  
June 2017.

## Appendix A. Contributor

The following persons contributed to this document:

Huang Wumin  
Huawei Technologies  
Huawei Building, No. 156 Beiqing Rd.  
Beijing 100095  
China

Email: [huangwumin@huawei.com](mailto:huangwumin@huawei.com)

## Authors' Addresses

Mahendra Singh Negi  
Huawei Technologies  
Divyashree Techno Park, Whitefield  
Bangalore, Karnataka 560066  
India

Email: [mahendrasingh@huawei.com](mailto:mahendrasingh@huawei.com)

Prejeeth Kaladharan  
Huawei Technologies  
Divyashree Techno Park, Whitefield  
Bangalore, Karnataka 560066  
India

Email: [prejeeth.k@huawei.com](mailto:prejeeth.k@huawei.com)

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

Email: [dhruv.ietf@gmail.com](mailto:dhruv.ietf@gmail.com)

Siva Sivabalan  
Cisco Systems

Email: [msiva@cisco.com](mailto:msiva@cisco.com)

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: August 10, 2019

M. Negi  
C. Li  
Huawei Technologies  
S. Sivabalan  
Cisco Systems  
P. Kaladharan  
RtBrick Inc  
February 6, 2019

PCEP Extensions for Segment Routing leveraging the IPv6 data plane  
draft-negi-pce-segment-routing-ipv6-04

## Abstract

The Source Packet Routing in Networking (SPRING) architecture describes how Segment Routing (SR) can be used to steer packets through an IPv6 or MPLS network using the source routing paradigm. Segment Routing (SR) enables any head-end node to select any path without relying on a hop-by-hop signaling technique (e.g., LDP or RSVP-TE).

It depends only on "segments" that are advertised by Link- State IGPs. A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE).

Since, Segment Routing can be applied to both MPLS and IPv6 forwarding plane, a PCE should be able to compute SR-Path for both MPLS and IPv6 forwarding plane. This draft describes the extensions required for Segment Routing support for IPv6 data plane in Path Computation Element communication Protocol (PCEP).

## Requirements Language

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

## Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute

working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 10, 2019.

#### Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Overview of PCEP Operation in SRv6 Networks . . . . .	5
3.1. Operation Overview . . . . .	6
3.2. SRv6-Specific PCEP Message Extensions . . . . .	6
3.3. Object Formats . . . . .	6
3.3.1. The OPEN Object . . . . .	6
3.3.1.1. The SRv6 PCE Capability sub-TLV . . . . .	6
3.3.1.2. Exchanging the SRv6 Capability . . . . .	8
3.3.2. The RP/SRP Object . . . . .	9
3.3.3. ERO Object . . . . .	9
3.3.3.1. SR-ERO Subobject . . . . .	10
3.3.3.2. ERO Processing . . . . .	12
3.3.4. RRO Object . . . . .	13
3.3.4.1. SR-RRO Subobject . . . . .	13
3.4. Security Considerations . . . . .	14
3.5. IANA Considerations . . . . .	14
3.5.1. PCEP Objects . . . . .	14
3.5.1.1. ERROR Objects . . . . .	14
3.5.1.2. TLV Type Indicators . . . . .	14
3.5.1.3. New Path Setup Type . . . . .	15

3.6. The NAI Type field . . . . .	15
4. Acknowledgements . . . . .	15
5. References . . . . .	15
5.1. Normative References . . . . .	15
5.2. Informative References . . . . .	16
Appendix A. Contributor . . . . .	19
Authors' Addresses . . . . .	19

## 1. Introduction

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based. A segment can have a semantic local to an SR node or global within an SR domain. SR allows to enforce a flow through any path and service chain while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locator, etc. The list of segment forming the path is called the Segment List and is encoded in the packet header. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [I-D.ietf-6man-segment-routing-header]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment.

Segment Routing use cases are described in [RFC7855] and [RFC8354]. Segment Routing protocol extensions are defined in [I-D.ietf-isis-segment-routing-extensions], and [I-D.ietf-ospf-ospfv3-segment-routing-extensions].

As per [I-D.ietf-6man-segment-routing-header], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in an illustration provided in [I-D.filsfils-spring-srv6-network-programming].

The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). The SR is applied to IPV6 forwarding plane using SRH. A SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node.

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

A mechanism to dynamically initiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE is specified in [RFC8281]. As per [I-D.ietf-pce-segment-routing], it is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [RFC8281] using the SR specific PCEP extensions specified in [I-D.ietf-pce-segment-routing]. [I-D.ietf-pce-segment-routing] specifies PCEP extensions for supporting a SR-TE LSP for MPLS data plane. This document extends [I-D.ietf-pce-segment-routing] to support SR for IPv6 data plane. Additionally, using procedures described in this document, a PCC can request an SRv6 path from either stateful or a stateless PCE. This specification relies on the PATH-SETUP-TYPE TLV and procedures specified in [RFC8408].

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

## 2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [RFC8051]: Stateful PCE, Delegation.

The message formats in this document are specified using Routing Backus-Naur Format (RBNF) encoding as specified in [RFC5511].



NAI: Node or Adjacency Identifier.

PCC: Path Computation Client.

PCE: Path Computation Element.

PCEP: Path Computation Element Protocol.

SR: Segment Routing.

SID: Segment Identifier.

SRv6: Segment Routing for IPv6 forwarding plane.

SRH: IPv6 Segment Routing Header.

SR Path: IPv6 Segment (List of IPv6 SID representing a path in IPv6 SR domain)

### 3. Overview of PCEP Operation in SRv6 Networks

Basic operations for PCEP speakers is as per [I-D.ietf-pce-segment-routing]. SRv6 Paths computed by a PCE can be represented as an ordered list of SRv6 segments of 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment" in this document.

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

This document extends the "SR-ERO subobject" defined in [I-D.ietf-pce-segment-routing] to carry IPv6 SID(s) (IPv6 Addresses). SRv6-capable PCEP speakers MUST be able to generate and/or process this.

When a PCEP session between a PCC and a PCE is established, both PCEP speakers exchange their capabilities to indicate their ability to support SRv6 specific functionality.

In summary, this document:

- o Defines a new PCEP capability for SRv6
- o Update the SR-ERO and SR-RRO sub-object for SRv6
- o Defines a new path setup type carried in the PATH-SETUP-TYPE and PATH-SETUP-TYPE-CAPABILITY TLVs.

### 3.1. Operation Overview

In SR networks, an ingress node of an SR path appends all outgoing packets with an SR header consisting of a list of SIDs (IPv6 Prefix in case of SRv6). The header has all necessary information to guide the packets from the ingress node to the egress node of the path, and hence there is no need for any signaling protocol.

For IPv6 in control plane with MPLS data-plane, mechanism remains same as [I-D.ietf-pce-segment-routing]

This document describes extensions to SR path for IPv6 data plane. SRv6 Path (i.e. ERO) consists of an ordered set of SRv6 SIDs (see details in Figure 2).

A PCC or PCE indicates its ability to support SRv6 during the PCEP session Initialization Phase via a new SRv6-PCE-CAPABILITY sub-TLV (see details in Section 3.3.1.1).

### 3.2. SRv6-Specific PCEP Message Extensions

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable length body made up of mandatory and/or optional objects. This document does not require any changes in the format of PCReq and PCRep messages specified in [RFC5440], PCInitiate message specified in [RFC8281], and PCRpt and PCUpd messages specified in [RFC8231]. However, PCEP messages pertaining to SRv6 MUST include PATH-SETUP-TYPE TLV in the RP or SRP object to clearly identify that SRv6 is intended.

### 3.3. Object Formats

#### 3.3.1. The OPEN Object

##### 3.3.1.1. The SRv6 PCE Capability sub-TLV

This document defines a new Path Setup Type (PST) [RFC8408] for SRv6, as follows:

- o PST = TBD2: Path is setup using SRv6.

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

This document also defines the SRv6-PCE-CAPABILITY sub-TLV. PCEP speakers use this sub-TLV to exchange information about their SRv6 capability. If a PCEP speaker includes PST=TBD2 in the PST List of the PATH-SETUP-TYPE-CAPABILITY TLV then it MUST also include the SRv6-PCE-CAPABILITY sub-TLV inside the PATH-SETUP-TYPE-CAPABILITY TLV.

The format of the SRv6-PCE-CAPABILITY sub-TLV is shown in the following figure:

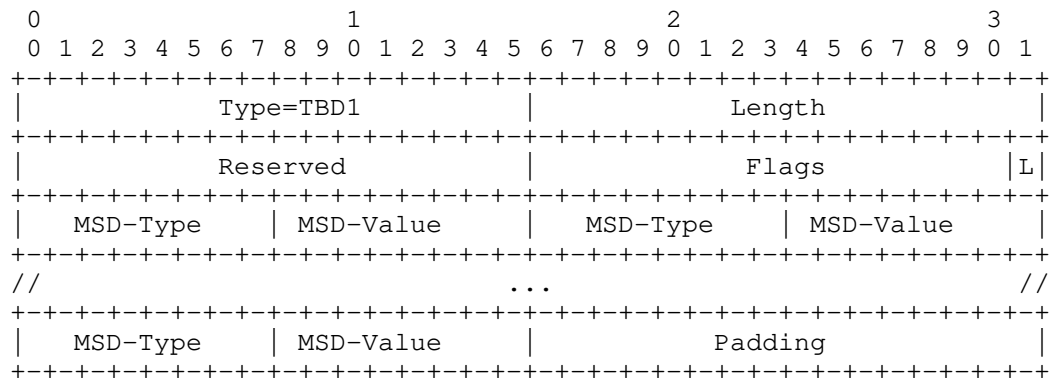


Figure 1: SRv6-PCE-CAPABILITY sub-TLV format

The code point for the TLV type (TBD1) is to be defined by IANA. The TLV length is variable.

The value comprise of -

Reserved: 2 octet, this field MUST be set to 0 on transmission, and ignored on receipt.

Flags: 2 octet, one bit is currently assigned in this document.

L bit: A PCC sets this bit to 1 to indicate that it does not impose any limit on MSD (irrespective of the MSD-Type).

Unassigned bits MUST be set to 0 and ignored on receipt.

A pair of (MSD-Type,MSD-Value): Where MSD-Type (1 octet) is as per the IGP MSD Type registry created by [RFC8491] and populated with SRv6 MSD types as per [I-D.bashandy-isis-srv6-extensions]; MSD-Value (1 octet) is as per [RFC8491].

The TLV format is compliant with the PCEP TLV format defined in [RFC5440]. That is, the TLV is composed of 2 octets for the type, 2 octets specifying the TLV length, and a Value field. The Length field defines the length of the value portion in octets. The TLV is padded to 4-octet alignment, and padding is not included in the Length field. The number of (MSD-Type,MSD-Value) pairs can be determined from the Length field of the TLV.

#### 3.3.1.2. Exchanging the SRv6 Capability

A PCC indicates that it is capable of supporting the head-end functions for SRv6 by including the SRv6-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCE. A PCE indicates that it is capable of computing SRv6 paths by including the SRv6-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCC.

If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a PST list containing PST=TBD2, but the SRv6-PCE-CAPABILITY sub-TLV is absent, then the PCEP speaker MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value TBD5 (to be assigned by IANA) (Missing PCE-SRv6-CAPABILITY sub-TLV) and MUST then close the PCEP session. If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a SRv6-PCE-CAPABILITY sub-TLV, but the PST list does not contain PST=TBD2, then the PCEP speaker MUST ignore the SRv6-PCE-CAPABILITY sub-TLV.

The number of SRv6 SIDs that can be imposed on a packet depends on the PCC's IPv6 data plane's capability. If a PCC sets the L flag to 1 then the MSD is not used and MUST not be included. If a PCE receives an SRv6-PCE-CAPABILITY sub-TLV with the L flag set to 1 then it MUST ignore any MSD-Type, MSD-Value fields and MUST assume that the sender can impose any length of SRH. If a PCC sets the L flag to zero, then it sets the SRv6 MSD-Type, MSD-Value fields that it can impose on a packet. If a PCE receives an SRv6-PCE-CAPABILITY sub-TLV with the L flag and SRv6 MSD-Type, MSD-Value fields both set to zero then it is considered as an error and the PCE MUST respond with a PCErr message (Error-Type=1 "PCEP session establishment failure" and Error-Value=1 "reception of an invalid Open message or a non Open message."). In case the MSD-Type in SRv6-PCE-CAPABILITY sub-TLV received by the PCE does not correspond to one of the SRv6 MSD types, the PCE MUST respond with a PCErr message (Error-Type=1 "PCEP session establishment failure" and Error-Value=1 "reception of an invalid Open message or a non Open message.").

Note that the MSD-Type, MSD-Value exchanged via the SRv6-PCE-CAPABILITY sub-TLV indicates the SRv6 SID imposition limit for the PCC node. However, if a PCE learns these via different means, e.g routing protocols, as specified in:  
[I-D.li-ospf-ospfv3-srv6-extensions];  
[I-D.bashandy-isis-srv6-extensions]; [I-D.dawra-idr-bgpls-srv6-ext], then it ignores the values in the SRv6-PCE-CAPABILITY sub-TLV. Furthermore, whenever a PCE learns the other advanced SRv6 MSD via different means, it MUST use that value regardless of the values exchanged in the SRv6-PCE-CAPABILITY sub-TLV.

Once an SRv6-capable PCEP session is established with a non-zero SRv6 MSD value, the corresponding PCE MUST NOT send SRv6 paths with a number of SIDs exceeding that SRv6 MSD value (based on the SRv6 MSD Type). If a PCC needs to modify the SRv6 MSD value, it MUST close the PCEP session and re-establish it with the new value. If a PCEP session is established with a non-zero SRv6 MSD value, and the PCC receives an SRv6 path containing more SIDs than specified in the SRv6 MSD value (based on the SRv6 MSD type), the PCC MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value 3 (Unsupported number of Segment ERO subobjects). If a PCEP session is established with an SRv6 MSD value of zero, then the PCC MAY specify an SRv6 MSD for each path computation request that it sends to the PCE, by including a "maximum SID depth" metric object on the request similar to [I-D.ietf-pce-segment-routing].

The L flag and (MSD-Type,MSD-Value) pair inside the SRv6-PCE-CAPABILITY sub-TLV are meaningful only in the Open message sent from a PCC to a PCE. As such, a PCE MUST set the L flag and not include (MSD-Type,MSD-Value) in an outbound message to a PCC. Similarly, a PCC MUST ignore any (MSD-Type,MSD-Value) received from a PCE. If a PCE receives multiple SRv6-PCE-CAPABILITY sub-TLVs in an Open message, it processes only the first sub-TLV received.

### 3.3.2. The RP/SRP Object

In order to indicate the SRv6 path, RP or SRP object MUST include the PATH-SETUP-TYPE TLV specified in [RFC8408]. This document defines a new Path Setup Type (PST=TBD2) for SRv6.

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

### 3.3.3. ERO Object

In order to support SRv6, the SR-ERO subobject is used [I-D.ietf-pce-segment-routing]. This documents extends the SR-ERO subobject. All the processing rules remains the same.

### 3.3.3.1. SR-ERO Subobject

For supporting SRv6, a new NAI Type (NT) is defined, the format of SR-ERO sub object remains the same as defined in [I-D.ietf-pce-segment-routing].

When the NAI Type (NT) indicates SRv6, then the SR-ERO subobject represent a SRv6 segment and include a field SRv6I (SRv6 Identifier) in place of NAI (Node or Adjacency Identifier) defined in [I-D.ietf-pce-segment-routing]. The 32 bit SID is not used for SRv6 and MUST NOT be included. The format of SR-ERO subobject is reproduced with the SRv6I field as shown below:

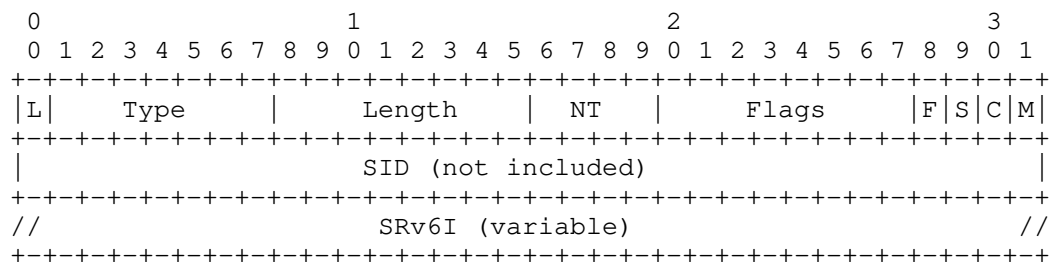


Figure 2: SR-ERO Subobject Format

The description of all the flags and fields is as per [I-D.ietf-pce-segment-routing].

For SRv6 segments, a new NT (NAI Type) is assigned by IANA as TBD3.

For SRv6 segments (when NT is TBD3), M and C flag MUST NOT be set. The S flag MUST be set and SID field MUST NOT be included. The F bit MUST NOT be set.

If these flags are not set properly, the subobject MUST be considered malformed and the PCEP speaker react as per the error handling described in Section 3.3.3.2.

The SRv6I format is as shown below:

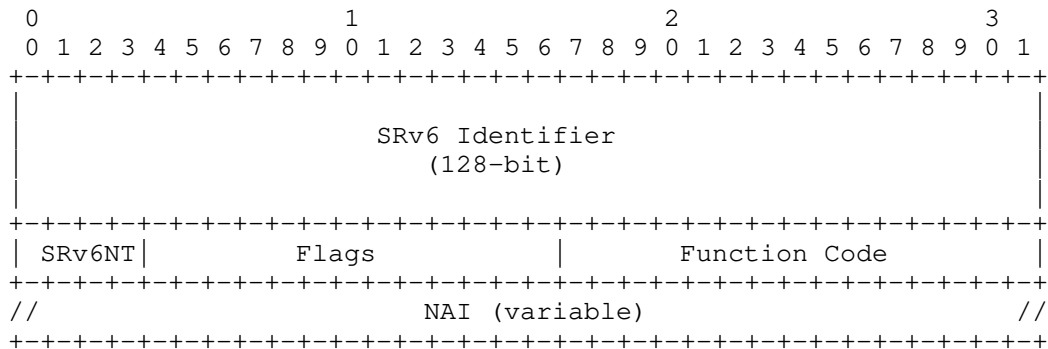


Figure 3: SR-ERO Subobject's SRv6I Format

SRv6 Identifier is the 128 bit IPv6 addresses representing SRv6 segment.

SRv6NT is the SRv6 NAI Type which indicates the interpretation for NAI (Node or Adjacency Identifier) as per [I-D.ietf-pce-segment-routing].

Flags is the 12 bit field, no flag bits are currently defined in this document. This MUST be set to 0 and ignored on receipt.

Function Code is the 16 bit field representing supported functions associated with SRv6 SIDs. This information is optional and included only for maintainability. Following function codes are currently defined -

- 0: Reserved
- 1: End Function
- 2: End.DX6 Function
- 3: End.DT6 Function
- 4: End.X Function

NAI field [I-D.ietf-pce-segment-routing] contains the NAI associated with the SRv6 Identifier. Depending on the value of SRv6NT, the NAI can have different formats.

When SRv6NT value is 1, the NAI is as per the 'IPv6 Node ID' format defined in [I-D.ietf-pce-segment-routing], which specify an IPv6 address. This is used to identify the owner of the

SRv6 Identifier. This is optional, as the LOC (the locator portion) of the SRv6 SID serves a similar purpose.

When SRv6NT value is 2, the NAI is as per the 'IPv6 Adjacency' format defined in [I-D.ietf-pce-segment-routing], which specify a pair of IPv6 addresses. This is used to identify the IPv6 Adjacency and used with the SRv6 Adj-SID.

Note that when SRv6NT value is 0, NAI is not included and MUST be NULL.

[Editor's Note - Add IPv6 unnumbered adjacency, once done by [I-D.ietf-pce-segment-routing]]

### 3.3.3.2. ERO Processing

The ERO and SR-ERO subobject processing remains as per [RFC5440] and [I-D.ietf-pce-segment-routing].

The NT MUST only be TBD3, if the PST=TBD3 is set in the PCEP message and SRv6-PCE-CAPABILITY sub-TLV is exchanged with the PCEP peer. In case a PCEP speaker receives the SR-ERO subobject with NT indicating SRv6 segment, when the PST is not set to TBD3 or SRv6-PCE-CAPABILITY sub-TLV was not exchanged, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD4 ("Attempted SRv6 when the capability was not advertised"). A PCEP speaker that does not recognize the NT value, it would behave as per [I-D.ietf-pce-segment-routing].

If a PCC receives a list of SRv6 segments, and the number of SRv6 segments exceeds the SRv6 MSD that the PCC can impose on the packet (SRH), it MAY send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Unsupported number of Segment ERO subobjects") as per [I-D.ietf-pce-segment-routing].

When a PCEP speaker detects that all subobjects of ERO are not identical to SRv6, and if it does not handle such ERO, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Non-identical ERO subobjects") as per [I-D.ietf-pce-segment-routing].

When a PCEP speaker receives an SR-ERO subobject for SRv6 segment, M, C and F flag MUST NOT be set and S flag MUST be set. Otherwise, it MUST consider the entire ERO object invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Malformed object") as per [I-D.ietf-pce-segment-routing]. The PCEP speaker MAY include the malformed SR-ERO object in the PCErr message as well.



### 3.3.3.2.1. Interpreting the SR-ERO

The SR-ERO contains a sequence of subobjects. According to [I-D.ietf-spring-segment-routing-policy], each SR-ERO subobject in the sequence identifies a segment that the traffic will be directed to, in the order given. That is, the first subobject identifies the first segment the traffic will be directed to, the second SR-ERO subobject represents the second segment, and so on.

The PCC interprets the SR-ERO by converting it to an SRv6 SRH plus a next hop. The PCC sends packets along the segment routed path by prepending the SRH onto the packets and sending the resulting, modified packet to the next hop.

### 3.3.4. RRO Object

In order to support SRv6, the SR-RRO Subobject is used [I-D.ietf-pce-segment-routing]. All other processing rules remains the same.

#### 3.3.4.1. SR-RRO Subobject

For SRv6 segments, a new NT (NAI Type) is assigned by IANA as TBD3, the format of SR-RRO sub object remains the same as the SR-ERO subobject, but without the L flag [I-D.ietf-pce-segment-routing].

When the NAI Type (NT) indicates SRv6, then the SR-RRO subobject represent a SRv6 segment and include a field SRv6I (SRv6 Identifier) in place of NAI (Node or Adjacency Identifier) defined in [I-D.ietf-pce-segment-routing]. The 32 bit SID MUST NOT be included. The format of SR-RRO subobject is reproduced with the SRv6I field as shown below:

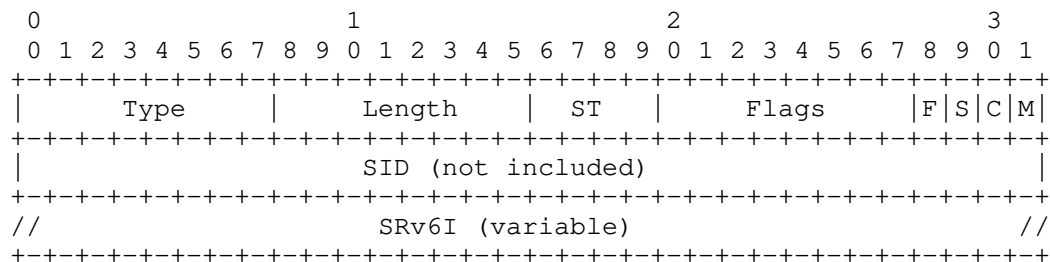


Figure 4: SR-RRO Subobject Format

The description of all fields and flags is as per SR-ERO subobject.

Processing rules of SR-RRO subobject are identical to those of SR-ERO subobject.

If a PCE detects that all subobjects of RRO are not identical, and if it does not handle such RRO, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 10 ("Non-identical RRO subobjects").

### 3.4. Security Considerations

The security considerations described in [RFC5440], [RFC8231] and [RFC8281], [I-D.ietf-pce-segment-routing], are applicable to this specification. No additional security measure is required.

### 3.5. IANA Considerations

This document requests IANA to include (I) bit in flags registry for SR-ERO and SR-RRO sub-objects. Other changes are defined as:

#### 3.5.1. PCEP Objects

##### 3.5.1.1. ERROR Objects

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

Error-Type	Meaning
-----	-----
10	Reception of an invalid object Error-value = TBD5 (Missing PCE-SRv6-CAPABILITY sub-TLV)
19	Invalid Operation Error-value = TBD4 (Attempted SRv6 when the capability was not advertised)

##### 3.5.1.2. TLV Type Indicators

IANA is requested to make the assignment of the new code points for the existing "PCEP TLV Type Indicators" registry as follows:

Value	Meaning	Reference
-----	-----	-----
TBD1	SRv6-PCE-CAPABILITY	This Document

### 3.5.1.3. New Path Setup Type

[RFC8408] defines the PATH-SETUP-TYPE TLV and requests that IANA creates a registry to manage the value of the PATH-SETUP-TYPE TLV's PST field. IANA is requested to allocate a new code point in the PCEP PATH\_SETUP\_TYPE TLV PST field registry, as follows:

Value -----	Description -----	Reference -----
TBD2	SRv6 (SRH) technique	This Document

### 3.6. The NAI Type field

As per [I-D.ietf-pce-segment-routing], a new subregistry for "PCEP SR-ERO NAI Types" was created. IANA is requested to make the assignment of the new code points in the afore-mentioned registry as follows:

Value -----	Description -----	Reference -----
TBD3	NAI is an SRv6 segment	This Document

## 4. Acknowledgements

The authors would like to thank Jeff Tentsura for suggestions regarding SRv6 MSD Types.

## 5. References

### 5.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [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>>.

- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491, DOI 10.17487/RFC8491, November 2018, <<https://www.rfc-editor.org/info/rfc8491>>.
- [I-D.ietf-pce-segment-routing]  
Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-14 (work in progress), October 2018.

## 5.2. Informative References

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

- [RFC7855] Previdi, S., Ed., Filsfils, C., Ed., Decraene, B., Litkowski, S., Horneffer, M., and R. Shakir, "Source Packet Routing in Networking (SPRING) Problem Statement and Requirements", RFC 7855, DOI 10.17487/RFC7855, May 2016, <<https://www.rfc-editor.org/info/rfc7855>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8354] Brzozowski, J., Leddy, J., Filsfils, C., Maglione, R., Ed., and M. Townsley, "Use Cases for IPv6 Source Packet Routing in Networking (SPRING)", RFC 8354, DOI 10.17487/RFC8354, March 2018, <<https://www.rfc-editor.org/info/rfc8354>>.
- [I-D.ietf-6man-segment-routing-header]  
Filsfils, C., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-16 (work in progress), February 2019.
- [I-D.ietf-isis-segment-routing-extensions]  
Previdi, S., Ginsberg, L., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-22 (work in progress), December 2018.
- [I-D.ietf-ospf-ospfv3-segment-routing-extensions]  
Psenak, P. and S. Previdi, "OSPFv3 Extensions for Segment Routing", draft-ietf-ospf-ospfv3-segment-routing-extensions-23 (work in progress), January 2019.
- [I-D.ietf-spring-segment-routing-policy]  
Filsfils, C., Sivabalan, S., daniel.voyer@bell.ca, d., bogdanov@google.com, b., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-02 (work in progress), October 2018.
- [I-D.filsfils-spring-srv6-network-programming]  
Filsfils, C., Camarillo, P., Leddy, J., daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-filsfils-spring-srv6-network-programming-06 (work in progress), October 2018.

[I-D.li-ospf-ospfv3-srv6-extensions]

Li, Z., Hu, Z., Cheng, D., Talaulikar, K., and P. Psenak,  
"OSPFv3 Extensions for SRv6", draft-li-ospf-  
ospfv3-srv6-extensions-02 (work in progress), September  
2018.

[I-D.bashandy-isis-srv6-extensions]

Psenak, P., Filsfils, C., Bashandy, A., Decraene, B., and  
Z. Hu, "IS-IS Extensions to Support Routing over IPv6  
Dataplane", draft-bashandy-isis-srv6-extensions-04 (work  
in progress), October 2018.

[I-D.dawra-idr-bgpls-srv6-ext]

Dawra, G., Filsfils, C., Talaulikar, K., Chen, M.,  
daniel.bernier@bell.ca, d., Uttaro, J., Decraene, B., and  
H. Elmalky, "BGP Link State extensions for IPv6 Segment  
Routing(SRv6)", draft-dawra-idr-bgpls-srv6-ext-04 (work in  
progress), September 2018.

## Appendix A. Contributor

The following persons contributed to this document:

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

EMail: dhruv.ietf@gmail.com

Huang Wumin  
Huawei Technologies  
Huawei Building, No. 156 Beiqing Rd.  
Beijing 100095  
China

Email: huangwumin@huawei.com

## Authors' Addresses

Mahendra Singh Negi  
Huawei Technologies  
Divyashree Techno Park, Whitefield  
Bangalore, Karnataka 560066  
India

EMail: mahendrasingh@huawei.com

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

EMail: chengli13@huawei.com

Siva Sivabalan  
Cisco Systems

EMail: msiva@cisco.com

Prejeeth Kaladharan  
RtBrick Inc  
Bangalore, Karnataka  
India

EMail: prejeeth@rtbrick.com



PCE Working Group  
Internet Draft  
Category: Standards track

Xian Zhang  
Haomian Zheng  
Huawei Technologies  
Oscar Gonzales de Dios  
Victor Lopez  
Telefonica I+D

Expires: April 30, 2018

October 30, 2017

Extensions to Path Computation Element Protocol (PCEP) to Support  
Resource Sharing-based Path Computation

draft-zhang-pce-resource-sharing-05.txt

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on April 30, 2018.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Requirements Language

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

#### Abstract

Resource sharing in a network means two or more Label Switched Paths (LSPs) use common piece(s) of resource along their paths. This can help save network resource and is useful in scenarios such as LSP recovery or when two LSPs do not need to be active at the same time. A Path Computation Element (PCE) is responsible for path computation with such requirement. Given this feature and its access to the network resource information and possibly active LSPs information, it can be used to support resource-sharing-based path computation with better efficiency.

This document extends the Path Computation Element Protocol (PCEP) in order to support resource sharing-based path computation.

#### Table of Contents

1. Introduction .....	3
2. Motivation .....	4
2.1. Single PCE Use Case .....	4
2.2. Multiple PCEs Use Case .....	6
3. Extensions to PCEP .....	8
3.1. Association group and type .....	8
3.2. Resource Sharing TLV .....	9
3.3. Processing Rules .....	10
4. Security Considerations .....	10
5. IANA Considerations .....	11
5.1. Association Object Type Indicators .....	11
6. References .....	12
6.1. Normative References .....	12
6.2. Informative References .....	12
7. Authors' Addresses .....	13

## 1. Introduction

A Path Computation Element (PCE) provides an alternative way for providing path computation function, and it is especially useful in the scenarios where complex constraints and/or a demanding amount of computation resource are required [RFC4655]. The development of PCE standardization has evolved from stateless to stateful. A stateful PCE has access to the LSP database information of the network(s) it serves as a computation engine [RFC8231]. Unless specified, this document assumes a PCE mentioned is a stateful PCE (either passive or active).

Resource sharing denotes that two or more Label Switched Paths (LSPs) share common piece(s) of resource, (such as a common time slot of a link in an Optical Transport Network (OTN)). This is usually useful in the scenario where only one LSP is active and the benefit herein is to save network resources. A simple example of this is dynamically calculating a LSP for an existing LSP undergoing a link failure. Note that the resource sharing can be worked out using a stateless PCE, but the mechanism may be complex and is out the scope of this draft.

This document considers the following requirement: new LSP may request for resource sharing with one or multiple existing LSPs. Furthermore, if there is resource sharing between new LSP and existing LSP, the two LSPs cannot exist simultaneously, the new LSP will replace the existing LSP(s).

In a single domain, this is a common requirement in the recovery cases especially in order to increase traffic resilience against failure while reducing the amount of network resource used for recovery purpose [RFC4428].

The current protocol supporting the communication between a PCE and a Path Computation Client (PCC), i.e. PCE Protocol (PCEP), allows for re-optimization of an existing LSP [RFC5440]. This is achieved by setting R bit in the Request Parameter (RP) object, together with some additional information if applicable, in the Path Computation Request (PCReq) message sent from a PCC to the PCE. To support this type of resource sharing, a PCC needs to ask a PCE to compute a new path with the constraints of sharing resource with one or multiple existing LSPs. It is worth noting the "resource sharing" in this draft not only means one LSP re-using the same link(s) of another LSP, but also the same slice of bandwidth. This may occur when an LSP is required for re-routing, or online re-optimization. Current PCEP specifications do not provide such function. More specifically, this draft describes the resource sharing issue during the procedure

when a new LSP is required to replace an existing LSP, which can be used together with Make-before-break (MBB) described in [RFC3209]. There are a few objects which indicate the resource sharing/disjoint relationships, such as SRLG and ASSOCIATE. However, these objects are used to describe the relationship with two simultaneous LSPs, instead of a new one and an old one, which is different with the object proposed in this draft.

As mentioned in [RFC8231], the PLSP-ID is unique during a PCEP session between PCC and PCE. Such identification is helpful in supporting the above resource sharing requirement for standardization of stateful PCEs. With a unique identifier, the configuration of PCCs is greatly simplified. Instead of determining all the resources to be shared, the PCC could request resource sharing directly from PCE.

The resource sharing can also be required in an inter-layer PCEP session. This is similar to the previous requirement. However, it is more complex and therefore deserves a more detailed explanation here.

In a multi-layer network, Label Switched Paths (LSPs) in a lower layer are used to carry higher-layer LSPs across the lower-layer network [RFC5623]. Therefore, the resource sharing constraints in the higher layer might actually relate to the resource sharing in the lower layer. Thus, it is useful to consider how this can be achieved and whether additional extensions are needed using the models defined in [RFC5623].

In the next sections, use cases are provided to show what information needs to be exchanged to fulfill these requirements. This memo then provides extensions to PCEP to enable this function.

## 2. Motivation

### 2.1. Single PCE Use Case

Figure 1 shows a single domain network with a stateful PCE. Assume a working LSP (N1-N2-N3) exists in the network, when there is failure on the link N2-N3, it is desired to set up a restoration path for this working LSP. Suppose N1 serves as the PCC and sends a request to the stateful PCE for such an LSP. Before sending the request, N1 may need to check what policy should be applied for the path re-computation. For example, it might value resource sharing and prefer to share as much resource with the working LSP as possible and specify this policy in the PCReq message. If resources are shared between the old and new LSPs, there will be some 'interruption' when the traffic is switched from the old LSP to the new LSP. Here the

resources to be shared mean the LSP information, which includes the node, link and corresponding SRLG information, etc.

On the other hand, in some scenarios there are different policies, for example the LSP should be restored without any interruption with best effort. An example can be found in Fig. 1 without failure on N2-N3 link, instead, an online re-optimization is needed for the working LSP (N1-N2-N3) from the stateful PCE. In such cases, the best choice is to set up a backup LSP for the working LSP with totally separate routing (for example N1-N5-N4-N3), and move the traffic to that backup LSP. After that the working LSP can be torn down, which will not result in any interruption during the optimization procedure. This can actually be implemented with existing PCEP mechanism. However, if there is no such separate path, existing PCEP will reply error. A secondary option for this case is to set up an LSP and complete such re-optimization with resource sharing, even if some interruption introduced. Given the resource from the LSP to be interrupted, there may be some solutions instead of Path Compute error due to the lack of resource.

A simple illustration is provided below:

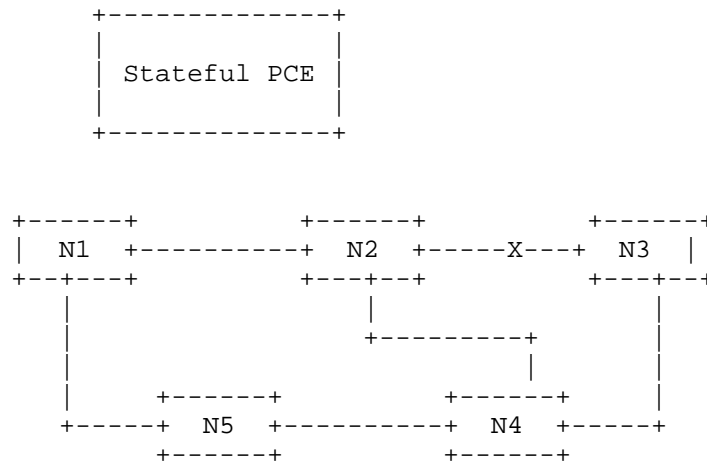


Figure 1: A Single Domain Example

Available recovery paths computed by the stateful PCE:

LSP1: N1-N2-N4-N3

LSP2: N1-N5-N4-N3

If resource sharing is preferred, the stateful PCE will reply with LSP1 information. Instead, if PCC prefer to have less interruption, PCE will reply with LSP2 information.

Another piece of information that needs to be conveyed to the PCE is the information about the working path LSP. Note this simple use case assumes end-to-end recovery. But in order to be applicable to use cases such as shared mesh protection purpose, where the head-end or tail-end nodes may be different, this information is necessary in the message exchange between PCCs and PCEs, so that the stateful PCE knows which LSP the path computation request wants to share the resource.

Besides, parameter changes during the resource sharing computation also need to be considered. For example, the bandwidth of the request LSP may be different with the existing LSP, while resource sharing is still preferred by the PCC. PCE should consider the sharing request together with the policy and available resource(s) in the network. Details can be found in Section 3.3.

## 2.2. Multiple PCEs Use Case

Figure 2 shows a two-layer network example, with each layer managed by a PCE. As Discussed in Section 3 of [RFC5623], there are three models for inter-layer path computation. They are single PCE computation, multiple PCE with inter-PCE communication and multiple PCE without inter-PCE communication, respectively. For the single PCE computation, the process would be similar to that of the use case in Section 2.1. Thus, this model is not discussed further.

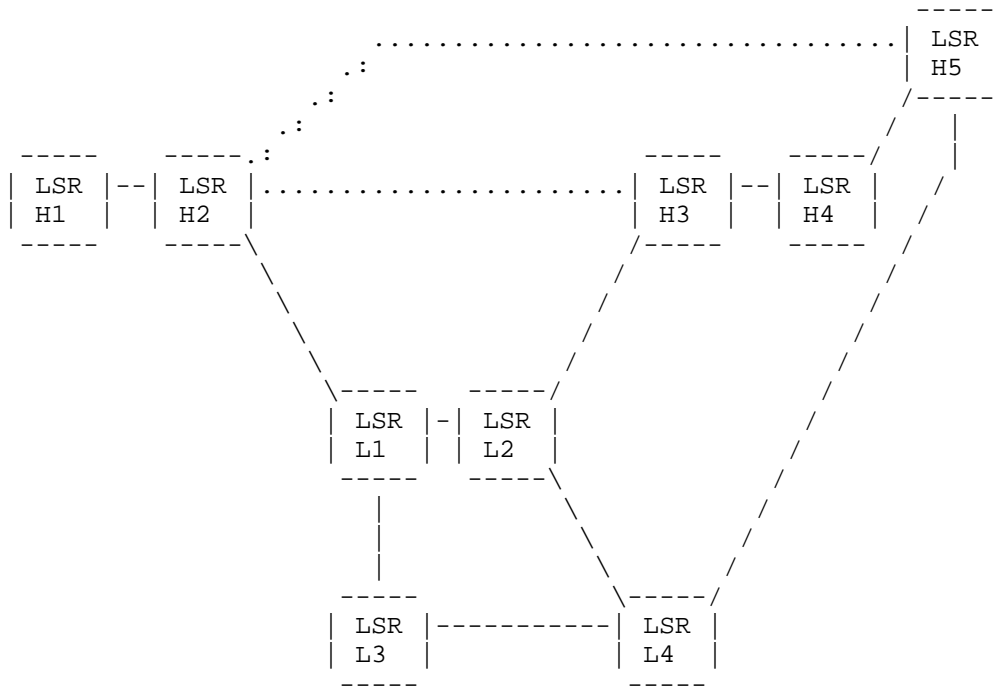


Figure 2: A Two-layer Network Example

An inter-layer path computation example is shown in Fig. 2, assume a LSP (LSP1: H2-H3) has been established already, visible as H2-H3 from view of higher-layer PCE and H2-L1-L2-H3 from the global view (or from the view of lower-layer PCE). A new request comes at H2 to establish a new LSP (LSP2: from H2 to H5), given the constraint it can share resource with LSP1. This requirement is possible if only one of the LSPs needs to be active and resource sharing is the target.

If multiple PCE with inter-PCE communication model is employed, the path computation request sent by H2 to higher-layer PCE will be forwarded to lower-layer PCE since there is no resource readily available in the higher layer. So it leaves the lower-layer PCE to compute a path in the lower layer in order to support the higher layer request. In this case, lower-layer PCE is required to compute a path between H2 and H5 under the constraint that it can share the resource with that of the LSP1. At this moment the lower-layer PCE has the knowledge on the explicit routing that LSP1 go through (H2-L1-L2-H3), and therefore can map the lower layer LSP with the higher-layer one. So when lower-layer PCE computes the path for LSP2, it can consider the resource used by LSP1 as available with higher priority. For example, lower-layer PCE may choose H2-L1-L2-L4-H5 as

the computation result. On the other hand, if the path computation policy is to have a separate path with LSP1, the lower-layer PCE may choose H2-L1-L3-L4-H5.

During this procedure higher-layer PCE can only use LSP1 information (such as its five-tuple LSP information) as the information, an issue to solve is how lower-layer PCE can resolve this information to the actual resource usage in its own layer, i.e. lower layer. This could be solved by edge LSR L1 reporting this higher-lower layer LSP correlation to the lower-layer PCE as part of the LSP information during the LSP state synchronization process. If needed, it can be later updated when there is a change in this information. Alternatively, the lower-layer PCE can get this information from other sources, such as network management system, where this information should be stored.

If multiple PCE without inter-PCE communication model is employed, the path computation request in the lower layer will be initiated the border LSR node, i.e., L1. The process would be similar to that of the previous scenario. A point worth noting is that the border LSR node may be able to resolve the higher layer LSP information itself, such as mapping it to the corresponding LSP in the lower layer, in this way lower-layer PCE does not need to perform this function. Otherwise, the mapping method mentioned above can still be used.

### 3. Extensions to PCEP

This section provides PCEP extensions. Currently the text focuses only on passive stateful PCE and corresponding PCReq. But if active stateful PCE delegation is used, we would like to convey the same information in PCRpt. In the passive stateful PCE architecture, a PCC is allowed to specify resource sharing when sending a PCReq message. It also details the processing rule and error codes needed.

#### 3.1. Association group and type

According to the definition in [ietf-pce-association-group], the association group is used to associate multiple LSPs into one group for further path computation considerations, such as disjointness and resource sharing. An association ID will be used to identify the resource sharing group. In this draft, a new association type is defined as:

Association type = TBD1 ("Sharing Association Type").



A sharing group should have multiple LSPs. The number of LSPs and the criteria for how LSPs share among each other are implementation dependent. Local path computation policies apply to different PCE and PCC, some examples can be found in section 2.

### 3.2. Resource Sharing TLV

The PCEP Resource Sharing group MUST carry the following TLV. It MAY be carried within a PCReq message from the network element (or other PCCs) so as to indicate the desired resource sharing requirements to be applied by the stateful PCE during path computation.

```

      0               1               2               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     |                                     |
|      Type = TBD2                   |      Length                       |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     |                                     |
|      Flags                         |      S | N | L                    |
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     |                                     |
|      Optional TLVs                 |                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Currently the following flags have been defined:

- \* L (Link share) bit: when set, this flag indicates that the PCE should prioritize the links that shared by existing LSPs within the sharing group for path computation.
- \* N (Node share) bit: when set, this flag indicates that the PCE should prioritize the nodes that shared by existing LSPs within the sharing group for path computation.
- \* S (SRLG share) bit: when set, this flag indicates that the PCE should set the SRLG (Shared Risk Link Group) of the computed LSP to the same as existing LSPs within the sharing group for path computation.

Optional TLVs may be needed to indicate the LSP(s) with which the resource is shared. If multiple LSPs are required, the PCE may need to consider different sharing policies, which is implementation dependent and may result in a different computing result. The selection policy among multiple computation result is out of the scope of this draft.

### 3.3. Processing Rules

To request a path allowing sharing resource with one or multiple existing LSPs, a PCC includes a Resource Sharing TLV in the association group object in the PCReq message.

On receipt of a PCReq message with a Resource Sharing TLV, a stateful PCE MUST proceed as follows:

- If the Resource Sharing TLV is unknown/unsupported, the PCE will follow procedures defined in [RFC5440]. That is, the PCE sends a PCErr message with error type 3 or 4 (Unknown / Not supported object) and error value 1 or 2 (unknown / unsupported object class / object type), and the related path computation request is discarded.
- If Resource Sharing TLV are unknown/unsupported and the P bit is set, the PCE MUST send a PCErr message with error type 3 or 4 (Unknown / Not supported object) and error value 4 (Unrecognized/Unsupported parameter), and the related path computation request MUST be discarded as defined in [RFC5440].
- If the resource sharing TLV is extracted correctly, the PCE MUST apply the requested resource sharing requirement.

The procedure of setting flags follows the rules defined in Section 3.1. The RSO flags may be locally configured on the requesting nodes via external entities, such as a network management system or the entity that impose the resource sharing requirement.

It is worth noting that the Resource Sharing TLV can be used together with other path indication objects like IRO/XRO. The difference is, the use of Resource Sharing TLV is to setup an alternative path, instead a new path. It is also dependent on the knowledge of PCC, e.g., if the PCC have a full knowledge of the path information and have strong preference on the route, it may send the PCReq with IRO message to specify the route. On the other hand, if the PCC does not know how the path should go but just want to set up a new LSP to replace the old one, it may use the Resource Sharing TLV instead of IRO.

### 4. Security Considerations

Security of PCEP is discussed in [RFC5440] and [RFC6952]. The extensions in this document do not change the fundamentals of security for PCEP.

However, the introduction of the Resource Sharing TLV in association group object provides a vector that may be used to probe for information from a network. For example, a PCC that wants to discover the path of an LSP with which it is not involved can issue a PCReq with a Resource sharing TLV and may be able to get back quite a lot of information about the path of the LSP through issuing multiple such requests for different endpoints and analyzing the received results. To protect against this, a PCE should be configured with access and authorization controls such that only authorized PCCs (for example, those within the network) can make computation requests, only specifically authorized PCCs can make requests for resource sharing, and such requests relating to specific LSPs are further limited to a select few PCCs. How such access controls and authorization is managed is outside the scope of this document, but it will at the least include Access Control Lists.

Furthermore, a PCC must be aware that setting up an LSP that share resources with another LSP may be a way of attacking the other LSP, for example by depriving it of the resources it needs to operate correctly. Thus it is important that, both in PCEP and the associated signaling protocols, only authorized resource sharing is allowed.

## 5. IANA Considerations

### 5.1. Association Object Type Indicators

This document defines a new association type, with the following information:

Object Class	Name	Object Type	Reference
TBA1	Sharing-group	Association Type	[this document]

### 5.2 PCEP TLV Definitions

This document defines the following TLVs to support the resource sharing scenario:

Value	Name	Reference
TBA2	Resource-sharing TLV	[this document]

IANA is requested to allocate the following bit numbers in the flag spaces of Resource-sharing TLV:

Bit	Flag name	Reference
0	Link Share	[this document]
1	Node Share	[this document]
2	SRLG Share	[this document]

## 6. References

### 6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to indicate requirements levels", RFC 2119, March 1997.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and Swallow, G., "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC5440] Vasseur, J.-P., and Le Roux, JL., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.
- [RFC8231] Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", RFC8231, June 2017.
- [ietf-pce-association-group] Minei, I., Crabbe E., Sivabalan S., Ananthakrishnan H., Dhody D., Tanaka Y., "PCEP Extensions for Establishing Relationships Between Sets of LSPs".

### 6.2. Informative References

- [RFC4428] Papadimitriou, D., Mannie., E., "Analysis of Generalized Multi-Protocol Label Switching (GMPLS)-based Recovery Mechanisms (including Protection and Restoration)", RFC4428, March 2006.
- [RFC4655] Farrel, A., Vasseur, J.-P., and Ash, J., "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.
- [RFC5623] Oki., E., Takeda, T., Le Roux, JL., Farrel, A., "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC5623, September 2009.
- [RFC6952] Jethanandani, M., Patel, K., Zheng, L., "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC6952, May 2013.

## 7. Authors' Addresses

Xian Zhang  
Huawei Technologies

Email: zhang.xian@huawei.com

Haomian Zheng  
Huawei Technologies

Email: zhenghaomian@huawei.com

Oscar Gonzalez de Dios  
Telefonica I+D  
Don Ramon de la Cruz 82-84  
Madrid 28045  
Spain  
EMail: ogondio@tid.es

Victor Lopez  
Telefonica I+D  
Don Ramon de la Cruz 82-84  
Madrid 28045  
Spain  
EMail: vlopez@tid.es

### Contributor's Address :

Dhruv Dhody  
Huawei Technologies  
Email: dhruv.dhody@huawei.com

Igor Bryskin  
Huawei Technologies  
Email: Igor.Bryskin@huawei.com



PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: April 27, 2022

X. Zhang  
H. Zheng  
Huawei Technologies  
O. Gonzales de Dios  
Telefonica  
V. Lopez  
Nokia  
Y. Xu  
CAICT  
October 24, 2021

Extensions to the Path Computation Element Protocol (PCEP) to Support  
Resource Sharing-based Path Computation  
draft-zhang-pce-resource-sharing-15

Abstract

Resource sharing in a network means two or more Label Switched Paths (LSPs) use common pieces of resource along their paths. This can help save network resources and is useful in scenarios such as LSP recovery or when two LSPs do not need to be active at the same time. A Path Computation Element (PCE) is responsible for path computation with such requirement.

Existing extensions to the Path Computation Element Protocol (PCEP) allow one path computation request for an LSP to be associated with other (existing) LSPs through the use of the PCEP Association Object.

This document extends PCEP in order to support resource-sharing-based path computation as another use of the Association Object to enable better efficiency in the computation and in the resultant paths and network resource usage.

Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 27, 2022.

#### Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	3
1.1. Requirements Language . . . . .	4
2. Motivation . . . . .	4
2.1. Single Domain Use Case . . . . .	4
2.2. Multiple Layers/Domains Use Case . . . . .	6
2.3. Bulk Path Computation Use Case . . . . .	8
3. Extensions to PCEP . . . . .	10
3.1. Association Group and Type . . . . .	10
3.2. Resource Sharing TLV . . . . .	10
3.3. Processing Rules . . . . .	11
4. Implementation Status . . . . .	12
5. Manageability Considerations . . . . .	12
5.1. Control of Function and Policy . . . . .	12
5.2. Information and Data Models . . . . .	12
5.3. Liveness Detection and Monitoring . . . . .	13
5.4. Verify Correct Operations . . . . .	13
5.5. Requirements on Other Protocols . . . . .	13
5.6. Impact on Network Operations . . . . .	13
6. Security Considerations . . . . .	13
7. IANA Considerations . . . . .	14
7.1. Association Object Type Indicators . . . . .	14
7.2. PCEP TLV Definitions . . . . .	14
8. References . . . . .	15
8.1. Normative References . . . . .	15
8.2. Informational References . . . . .	15
Authors' Addresses . . . . .	16



## 1. Introduction

A Path Computation Element (PCE) is a way to provide path computation function, and it is especially useful in the scenarios where complex constraints and/or a demanding amount of computation resource are required [RFC4655]. The development of PCE standardization has evolved from stateless to stateful. A stateful PCE has access to the LSP database information of the networks it serves as a computation engine [RFC8231]. Unless specified, this document assumes a PCE mentioned is a stateful PCE..

Resource sharing denotes that two or more Label Switched Paths (LSPs) share common pieces of resource, (such as a common time slot of a link in an Optical Transport Network (OTN)). This is usually useful in the scenario where only one of the LSPs is active and the benefit is to save network resources. A simple example of this is dynamically calculating a recovery LSP for an existing LSP undergoing a link failure. Note that resource sharing can be worked out using a stateless PCE, but the mechanism may be complex and is out the scope of this document.

This document considers the requirement that a new LSP may request for resource sharing with one or multiple existing LSPs. Furthermore, if there is resource sharing between a new LSP and existing an LSP, the two LSPs cannot be used to carry traffic simultaneously, the new LSP will take over the traffic from the existing LSP.

In a single domain, this is a common requirement in the recovery cases especially in order to increase traffic resilience against failure while reducing the amount of network resource used for recovery purposes [RFC4428]

The current protocol supporting the communication between a PCE and a Path Computation Client (PCC), i.e. PCE Protocol (PCEP), allows for re-optimization of an existing LSP [RFC5440]. This is achieved by setting the R bit in the Request Parameter (RP) object, together with some additional information if applicable, in the Path Computation Request (PCReq) message sent from a PCC to the PCE. To support this type of resource sharing, a PCC needs to ask a PCE to compute a new path with the constraints of sharing resource with one or multiple existing LSPs. It is worth noting the "resource sharing" in this draft not only means one LSP re-using the same links of another LSP, but also the same slice of bandwidth in the network. This may occur when an LSP is required for re-routing, or online re-optimization. Current PCEP specifications do not provide such function. More specifically, this document describes the resource sharing issue during the procedure when a new LSP is required to replace an

existing LSP for use together with Make-before-break (MBB) described in [RFC3209].

As mentioned in [RFC8231], the PLSP-ID provides a unique identifier for an LSP during a PCEP session between PCC and PCE. Such identification is helpful in supporting the resource sharing requirement for stateful PCEs because it greatly simplifies the operation of a PCC. Instead of the PCC determining all the resources to be shared, the PCC can request that the PCE share the resources of a specific LSP: the stateful PCE is able to determine those resource itself.

Resource sharing can also be required in an inter-layer PCEP session. This is similar to the previous requirement. However, it is more complex and therefore deserves a more detailed explanation here.

In a multi-layer network, LSPs in a lower layer are used to carry higher-layer LSPs across the lower-layer network [RFC5623]. Therefore, the resource sharing constraints in the higher layer might actually relate to resource sharing in the lower layer. Thus, it is useful to consider how this can be achieved and whether additional extensions are needed using the models defined in [RFC5623].

In the next sections, use cases are provided to show what information needs to be exchanged to fulfill these requirements. This memo then provides extensions to PCEP to enable this function.

### 1.1. Requirements Language

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

## 2. Motivation

### 2.1. Single Domain Use Case

There are two potential cases that request resource to be shared: restoration and re-optimization. Figure 1 shows a single domain network with a stateful PCE, and is used as an example for the resource sharing application.

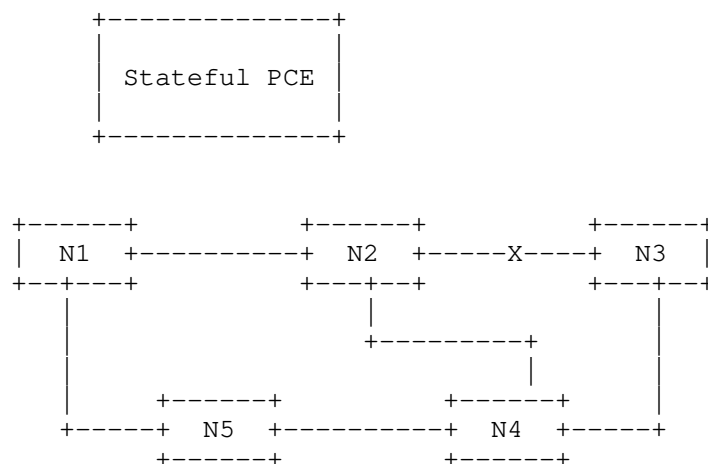


Figure 1: A Single Domain Example

LSP0 (existing): N1-N2-N3.

LSP1 (restoration): N1-N2-N4-N3.

LSP2 (re-optimization): N1-N5-N4-N3.

For the failure restoration, we can assume a working LSP (LSP0) exists in the network. When there is failure on the link N2-N3, it is desired to set up a restoration path for this working LSP. Suppose N1 serves as the PCC and sends a request to the stateful PCE for such an LSP. Besides the head-end and tail-end node of the working LSP, N1 may also need to check what policy should be applied for the restoration. For example, it may evaluate resource sharing and prefer to share as much resource with the working LSP as possible and specify this policy as a special object in the PCReq message. Given such policy, a probable outcome from the path computation would be LSP1, which shares the link 'N1-N2' with the existing LSP. The LSP1 will be set up by PCC via either PCInitiate or RSVP.

Re-optimization does not usually result from a specific failure in the network, but takes place on a stable network when more optimal paths may have become available. Thus switching from the existing LSP to the new LSP happens with live traffic. An example can be found in Figure 1 without failure on the link N2-N3. Instead, an online re-optimization is needed for the working LSP (LSP0) from the stateful PCE. In such cases, the best choice is to set up a backup

LSP for the working LSP with totally separate routing (for example, LSP2), and move the traffic to that backup LSP. After that, the working LSP can be torn down, which will not result in any interruption during the optimization procedure. This can actually be implemented with existing PCEP mechanisms. However, if there is no such separate path, existing PCEP mechanisms will return an error. A secondary option for this case is to set up an LSP and complete re-optimization with resource sharing, even if some interruption is introduced.

In the example from Figure 1 it is assumed that the restored LSP or re-optimized LSP have the same source and destination nodes. But in some applications there is no restriction for this assumption, i.e., after an LSP is failed, it can be restored as a new LSP with different source/destination.

In the use cases above it is also assumed that the characteristics of the restored LSP or re-optimized LSP are unchanged. However, it is possible to have parameter changes during the resource sharing computation. For example, the bandwidth of the request LSP may be different from the existing LSP, while resource sharing is still preferred by the PCC. The PCE should consider the sharing request together with the policy and available resources in the network. Details can be found in Section 3.3.

Conversely to resource sharing, it may also be required to apply a disjoint constraint for the path computation. [RFC8800] discusses the solution under such a scenario, which is a companion work to this document.

## 2.2. Multiple Layers/Domains Use Case

As Discussed in Section 3 of [RFC5623], there are three models for inter-layer path computation. They are single PCE computation, multiple PCE with inter-PCE communication, and multiple PCE without inter-PCE communication. For the single PCE computation, the process would be similar to that of the use case in Section 2.1.

An inter-layer path computation example is shown in Figure 2. Assume an LSP (LSP1: H2-H3) has been established already, visible as H2-H3 from the view of the higher-layer PCE, and as H2-L1-L2-H3 from the global view (or from the view of the lower-layer PCE). A new request is received by H2 to establish a new LSP (LSP2: from H2 to H5), given the constraint that it can share resources with LSP1. This requirement is possible if only one of the LSPs needs to be active and resource sharing is the target.

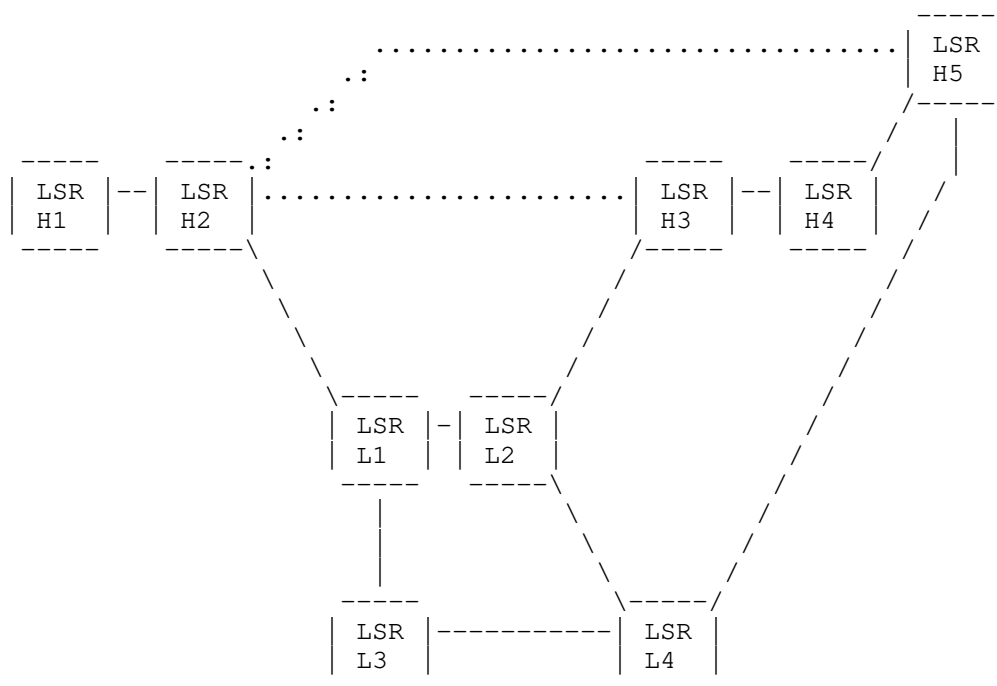


Figure 2: A Two-layer Network Example

If the model of multiple PCEs with inter-PCE communication is employed, the path computation request sent by H2 to higher-layer PCE will be forwarded to lower-layer PCE since there is no resource readily available in the higher layer. So it leaves the lower-layer PCE to compute a path in the lower layer in order to support the higher layer request. In this case, the lower-layer PCE is required to compute a path between H2 and H5 under the constraint that it can share the resource with that of LSP1. At this moment the lower-layer PCE has knowledge of the mapping relationship between the higher-layer link H2-H3 and the lower layer link L1-L2, and therefore can convert the resource to be shared from higher layer to lower layer. So when the lower-layer PCE computes the path for LSP2, it can consider the resource used by L1-L2 as available with higher priority. For example, the lower-layer PCE may choose H2-L1-L2-L4-H5 as the computation result. On the other hand, if the path computation policy is to have a separate path with LSP1, the lower-layer PCE may choose H2-L1-L3-L4-H5.

During this procedure the higher-layer PCE can only use information about LSP1 (such as its five-tuple LSP information). An issue to solve is how the lower-layer PCE can resolve this information to the actual resource usage in its own layer, i.e. the lower layer. This could be solved by the edge LSR (L1) reporting this higher-lower LSP correlation to the lower-layer PCE as part of the LSP information during the LSP state synchronization process. If needed, it can be updated later when there is a change in this information. Alternatively, the lower-layer PCE can get this information from other sources, such as a network management system, where this information should be stored.

If the model of multiple PCEs without inter-PCE communication is employed, the path computation request in the lower layer will be initiated by the border LSR node, i.e., L1. The process would be similar to that of the previous scenario. A point worth noting is that the border LSR node may be able to resolve the higher layer LSP information itself, such as by mapping it to the corresponding LSP in the lower layer, in this way the lower-layer PCE does not need to perform this function. Otherwise, the mapping method mentioned above can still be used.

### 2.3. Bulk Path Computation Use Case

There is a potential need for resource sharing during bulk path computation, especially the processing of the "sticky resources" in [RFC7399]. It would be useful to specify the resources that can be shared among different paths, i.e., the bandwidth information.

Considering the H-PCE architecture in [RFC8751], when the parent PCE asks for a single path across a few domains, such a request may become a bulk path computation to a certain child PCE. Figure 3 shows an example of 3 domains. The parent PCE will select one of these path for establishment.

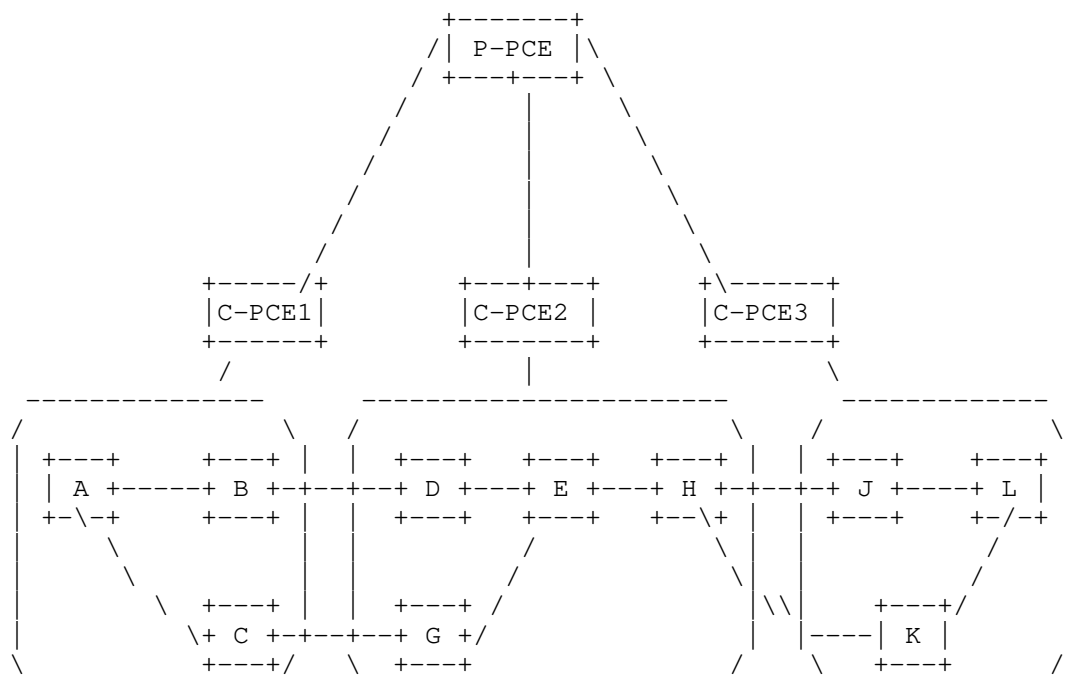


Figure 3: Bulk Request example with Hierarchical PCEs

A 3-domain example is shown in Figure 3, with the hierarchical PCE architecture. In this example nodes A/B/C belong to domain 1, nodes D/E/G/H belong to domain 2, and nodes J/K/L belong to domain 3. Inter-domain links are B-D/C-G between domains 1 and 2, and H-J/H-K between domains 2 and 3. Given a path computation request from A to L, a bulk request from P-PCE would be helpful to understand whether it is possible to have different combinations on the inter-domain links. However, the resources on some specific links become 'sticky' and have to be indicated as 'sharing allowed' to avoid unnecessary resource competition. For example, both the route A-B-D-E-H-J-L and A-C-G-E-H-K-L are qualified, but these routes are competing for the resource on the link E-H and cannot be established simultaneously, so there must be one route failed to be reported to P-PCE. Given the indication of allowing resource sharing on the link E-H, both of these routes can be reported for P-PCE's decision, and there will not be any competition as the P-PCE understands that only one path needs to be set up.

### 3. Extensions to PCEP

#### 3.1. Association Group and Type

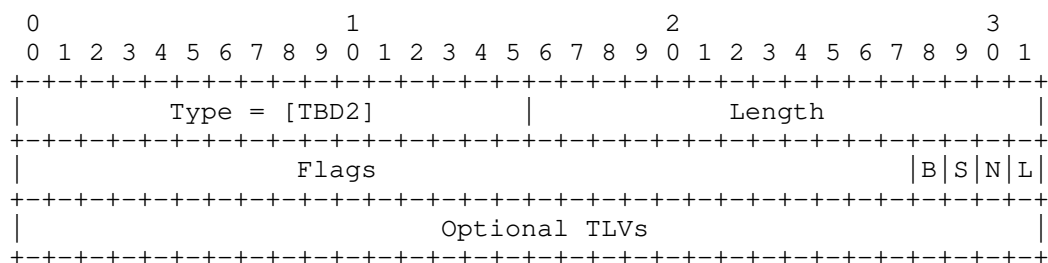
According to the definition in [RFC8697], the association group is used to associate multiple LSPs into one group for further path computation considerations, such as disjointness and resource sharing, in the messages when requesting path computation. An association ID will be used to identify the resource sharing group. An association type that described disjointness has been defined in [RFC8800]. In this document, a new association type is defined as follows:

- o Association type = TBD1 ("Sharing Association Type").

A sharing group should have multiple LSPs. The number of LSPs and the criteria for how LSPs share among each other are dependent on local policy.

#### 3.2. Resource Sharing TLV

The PCEP Resource Sharing group MAY carry the following TLV. It MAY be carried within a PCReq message from the network element (or other PCCs) so as to indicate the desired resource sharing requirements to be applied by the stateful PCE during path computation.



The following flags are defined:

- o L (Link share) bit: when set, this flag indicates that the PCE should prioritize the links shared by existing LSPs within the sharing group for path computation. The existing LSP identifier and its available link identifiers can be contained in the optional TLVs.



- o N (Node share) bit: when set, this flag indicates that the PCE should prioritize the nodes shared by existing LSPs within the sharing group for path computation. The existing LSP identifier and its available node identifiers can be contained in the optional TLVs.
- o S (SRLG share) bit: when set, this flag indicates that the PCE should set the SRLG (Shared Risk Link Group) of the computed LSP to the same as existing LSPs within the sharing group for path computation. The existing LSP identifier and SRLG information can be contained in the optional TLVs.
- o B (Bandwidth share) bit: when set, this flag indicates that the PCE should prioritize the bandwidth to be shared by LSPs within the sharing group for bulk path computation. The LSP identifiers can be contained in the optional TLVs.

It is worth noting that there can be multiple flags set which may conflict with each other. In this scenario, the result for path computation may not be unique, and is dependent on the implementation. The selection among multiple computation results is out of the scope of this document.

### 3.3. Processing Rules

To request a path allowing resource sharing with one or multiple existing LSPs, a PCC includes a Resource Sharing TLV in the Association Group Object in any kind of path computation request message, such as the PCReq, PCUpd, or PCInitiate messages specified in [RFC8231] and [RFC8281].

On receipt of a PCEP message with a Resource Sharing TLV, a stateful PCE MUST proceed as follows:

- o If the Resource Sharing TLV is unknown/unsupported, the PCE will follow procedures defined in [RFC5440]. That is, the PCE sends a PCErr message with error type 26 (Association Error) and error value 6 (Association Information Mismatch), and the related path computation request is discarded.
- o If the Resource Sharing TLV is extracted correctly, the PCE MUST apply the requested resource sharing requirement, i.e., try to share as much resource as possible with the LSP specified in Resource Sharing TLV.

The procedure of setting flags follows the rules defined in Section 3.1. The flags in the Resource Sharing TLV may be locally configured on the requesting nodes via external entities, such as a

network management system or the entity that imposes the resource sharing requirement.

It is worth noting that the Resource Sharing TLV can be used together with other path indication objects like the IRO/XRO, with different objectives. The first difference is, the use of the Resource Sharing TLV is to set up an alternative path, instead a new path. It is also dependent on the knowledge held by the PCC, e.g., if the PCC has full knowledge of the path information and has a strong preference on the route, it may send the request message with an IRO to specify the route. On the other hand, if the PCC does not know how the path should go but just wants to set up a new LSP to replace the old one, it may use the Resource Sharing TLV instead of an IRO. The second difference is that the Resource Sharing TLV is a loose requirement. For example, if the constraint specified in an IRO/XRO in an A-Z path computation request cannot be satisfied, the reply message from PCE to PCC would be unsuccessful. However it is still possible to have a path from the A-Z. If the target node/link/SRLG/Bandwidth is set in the Resource Sharing TLV rather than an IRO, the PCE may feedback a path from A-Z that does not share the target specified in the Resource Sharing TLV.

#### 4. Implementation Status

[Note to the RFC Editor - remove this section before publication, as well as remove the reference to [RFC7942].

Currently the authors are not aware of any implementations.

#### 5. Manageability Considerations

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

##### 5.1. Control of Function and Policy

A PCE or PCC implementation MUST allow operator-configured associations and SHOULD allow setting of the resource sharing TLV (Section 3.2) as described in this document.

##### 5.2. Information and Data Models

An implementation SHOULD allow the operator to view the resource sharing configured or created dynamically. Further implementation SHOULD allow to view resource sharing associations reported by each peer, and the current set of LSPs in the association. The PCEP YANG

module [I-D.ietf-pce-pcep-yang] includes association groups information.

### 5.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

### 5.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

### 5.5. Requirements on Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols. The configuration on local policy may be accomplished by other protocols, such as Netconf.

### 5.6. Impact on Network Operations

Mechanisms defined in [RFC5440] and [RFC8231] also apply to PCEP extensions defined in this document.

## 6. Security Considerations

Security of PCEP is discussed in [RFC5440] and [RFC6952]. The extensions in this document do not change the fundamentals of security for PCEP.

However, the introduction of the Resource Sharing TLV in the Association Group Object provides a vector that may be used to probe for information from a network. For example, a PCC that wants to discover the path of an LSP with which it is not involved can issue a request message with a Resource Sharing TLV and may be able to get back quite a lot of information about the path of the LSP through issuing multiple such requests for different endpoints and analyzing the received results. To protect against this, a PCE SHOULD be configured with access and authorization controls such that only authorized PCCs (for example, those within the network) can make computation requests, only specifically authorized PCCs can make requests for resource sharing, and such requests relating to specific LSPs are further limited to a select few PCCs. How such access controls and authorization is managed is outside the scope of this document, but it will at the least include Access Control Lists.

Furthermore, a PCC must be aware that setting up an LSP that shares resources with another LSP may be a way of attacking the other LSP, for example by depriving it of the resources it needs to operate correctly. Thus it is important that, both in PCEP and the associated signaling protocols, only authorized resource sharing is allowed.

## 7. IANA Considerations

### 7.1. Association Object Type Indicators

IANA maintains a registry called the "Path Computation Element Protocol (PCEP) Numbers" registry with a subregistry called the "Association Type Field" subregistry. IANA is requested to make an assignment from that subregistry as follows:

Object Class	Name	Object Type	Reference
TBD1	Sharing-group	Association Type	[this document]

### 7.2. PCEP TLV Definitions

This document defines the following TLVs to support the resource sharing scenario:

Value	Name	Reference
TBD2	Resource-sharing TLV	[this document]

IANA is requested to allocate the following bit numbers in the flag spaces of Resource-sharing TLV:

Bit	Flag name	Reference
31	Link Share	[this document]
30	Node Share	[this document]
29	SRLG Share	[this document]
28	Bandwidth Share	[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, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <<https://www.rfc-editor.org/info/rfc5440>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <<https://www.rfc-editor.org/info/rfc8231>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8697] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "Path Computation Element Communication Protocol (PCEP) Extensions for Establishing Relationships between Sets of Label Switched Paths (LSPs)", RFC 8697, DOI 10.17487/RFC8697, January 2020, <<https://www.rfc-editor.org/info/rfc8697>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.

### 8.2. Informational References

- [I-D.ietf-pce-pcep-yang]  
Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura,  
"A YANG Data Model for Path Computation Element  
Communications Protocol (PCEP)", draft-ietf-pce-pcep-  
yang-16 (work in progress), February 2021.
- [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>>.
- [RFC4428] Papadimitriou, D., Ed. and E. Mannie, Ed., "Analysis of  
Generalized Multi-Protocol Label Switching (GMPLS)-based  
Recovery Mechanisms (including Protection and  
Restoration)", RFC 4428, DOI 10.17487/RFC4428, March 2006,  
<<https://www.rfc-editor.org/info/rfc4428>>.
- [RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation  
Element (PCE)-Based Architecture", RFC 4655,  
DOI 10.17487/RFC4655, August 2006,  
<<https://www.rfc-editor.org/info/rfc4655>>.
- [RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel,  
"Framework for PCE-Based Inter-Layer MPLS and GMPLS  
Traffic Engineering", RFC 5623, DOI 10.17487/RFC5623,  
September 2009, <<https://www.rfc-editor.org/info/rfc5623>>.
- [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>>.
- [RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running  
Code: The Implementation Status Section", BCP 205,  
RFC 7942, DOI 10.17487/RFC7942, July 2016,  
<<https://www.rfc-editor.org/info/rfc7942>>.

Authors' Addresses

Xian Zhang  
Huawei Technologies  
China

Email: [zhang.xian@huawei.com](mailto:zhang.xian@huawei.com)

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

Email: zhenghaomian@huawei.com

Oscar Gonzales de Dios  
Telefonica  
Spain

Email: oscar.gonzalezdedios@telefonica.com

Victor Lopez  
Nokia  
Spain

Email: victor.lopez@nokia.com

Yunbin Xu  
CAICT  
China

Email: xuyunbin@caict.ac.cn