

PCE Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: 23 October 2026

M. Koldychev, Ed.  
Ciena Corporation  
S. Sidor, Ed.  
Cisco Systems.  
21 April 2026

Path Computation Element Communication Protocol (PCEP) Extensions for  
Signaling Multipath Information  
draft-ietf-pce-multipath-24

## Abstract

A Segment Routing (SR) Policy Candidate Path can contain multiple Segment Lists, allowing for load-balancing and redundancy across diverse paths. However, current PCEP extensions for SR Policy only allow signaling of a single Segment List per Candidate Path. This document defines PCEP extensions to encode multiple Segment Lists within an SR Policy Candidate Path, enabling multipath capabilities such as weighted or equal-cost load-balancing across Segment Lists. These extensions are designed to be generic and reusable for future path types beyond SR Policy, and are applicable to both stateless and stateful PCEP.

## Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <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 23 October 2026.

## Copyright Notice

Copyright (c) 2026 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Requirements Language . . . . .	3
1.2. Terminology . . . . .	4
2. Motivation . . . . .	5
2.1. Signaling Multiple Segment Lists of an SR Candidate Path . . . . .	5
2.2. Splitting of Requested Bandwidth . . . . .	5
2.3. Reverse Path Information . . . . .	5
3. Protocol Extensions . . . . .	6
3.1. PATH-ATTRIB Object . . . . .	6
3.2. METRIC Object . . . . .	7
3.3. MULTIPATH-WEIGHT TLV . . . . .	7
3.4. MULTIPATH-BACKUP TLV . . . . .	8
3.5. MULTIPATH-OPPDIR-PATH TLV . . . . .	9
3.6. Composite Candidate Path . . . . .	10
3.6.1. Per-Flow Candidate Path . . . . .	11
4. Operation . . . . .	12
4.1. Capability Negotiation . . . . .	12
4.1.1. Multipath Capability TLV . . . . .	12
4.2. Path ID . . . . .	14
4.3. Signaling Multiple Paths for Load-Balancing . . . . .	15
4.4. Signaling Multiple Paths for Protection . . . . .	15
4.5. Signaling Opposite-Direction Path Information . . . . .	16
5. PCEP Message Extensions . . . . .	17
6. Implementation Status . . . . .	18
6.1. Cisco Systems . . . . .	19
6.2. Ciena Corp . . . . .	19
6.3. Huawei Technologies . . . . .	19
7. IANA Considerations . . . . .	19
7.1. PCEP Object . . . . .	19
7.2. PCEP TLV . . . . .	20
7.3. PCEP-Error Object . . . . .	20
7.4. Flags in the MULTIPATH-CAP TLV . . . . .	21
7.5. Flags in the PATH-ATTRIB Object . . . . .	22
7.6. Flags in the MULTIPATH-BACKUP TLV . . . . .	22
7.7. Flags in the MULTIPATH-OPPDIR-PATH TLV . . . . .	23
7.8. Flags in the MULTIPATH-FORWARD-CLASS TLV . . . . .	23

8. Security Considerations . . . . .	23
9. Operational Considerations . . . . .	24
9.1. Control of Function and Policy . . . . .	24
9.2. Information and Data Models . . . . .	24
9.3. Liveness Detection and Monitoring . . . . .	24
9.4. Verify Correct Operations . . . . .	24
9.5. Requirements On Other Protocols . . . . .	25
9.6. Impact On Network Operations . . . . .	25
10. Acknowledgement . . . . .	25
11. Contributors . . . . .	25
11.1. Original Authors . . . . .	25
11.2. Additional Contributors . . . . .	26
12. References . . . . .	27
12.1. Normative References . . . . .	27
12.2. Informative References . . . . .	28
Appendix A. Examples . . . . .	29
A.1. SR Policy Candidate Path with Multiple Segment Lists . . . . .	29
A.2. Two Primary Paths Protected by One Backup Path . . . . .	31
A.3. Composite Candidate Path . . . . .	31
A.4. Opposite Direction Tunnels . . . . .	32
Authors' Addresses . . . . .	34

## 1. Introduction

Segment Routing Policy for Traffic Engineering [RFC9256] details the concepts of Segment Routing (SR) Policy and approaches to steering traffic into an SR Policy. In particular, it describes the SR Candidate Path as a collection of one or more Segment Lists. The current PCEP specifications only allow for signaling of one Segment List per Candidate Path. The PCEP extension to support Segment Routing Policy Candidate Paths [RFC9862] specifically kept the signaling of multiple Segment Lists outside its scope.

This document defines the required extensions that allow the signaling of multipath information via PCEP. Although these extensions are motivated by the SR Policy use case, they are also applicable to other technologies.

### 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.

## 1.2. Terminology

The following terms are used in this document:

ECMP:

Equal Cost Multi Path, equally distributing traffic among multiple paths/links, where each path/link gets the same share of traffic as others.

W-ECMP:

Weighted ECMP, unequally distributing traffic among multiple paths/links, where some paths/links get more traffic than others.

PLSP:

PCE Label Switched Path, a path or set of paths computed or controlled by the PCE. In the context of SR Policy, a PLSP corresponds to a Candidate Path.

Path:

In the context of this document, a path refers to a single forwarding path encoded in an ERO or RRO. For SR Policy, a path corresponds to a Segment List. The mechanisms defined in this document use the generic term "path" to allow applicability beyond SR Policy.

LSP:

Label Switched Path. The base PCEP specification [RFC4655] originally defined the use of the PCE architecture for MPLS and GMPLS networks with LSPs instantiated using the RSVP-TE signaling protocol. Over time, support for additional path setup types such as SRv6 has been introduced [RFC9603]. The term "LSP" is used extensively in PCEP specifications and, while the multipath extensions defined in this document are applicable beyond SR Policy, in the context of PCEP for SR Policy [RFC9862], an LSP object represents an SR Policy Candidate Path, which may be an SRv6 path (still represented using the LSP object as specified in [RFC8231]). A single LSP may contain multiple paths (Segment Lists).

Segment List:

An ordered list of segments that defines a forwarding path in Segment Routing, as defined in [RFC9256]. In PCEP for SR Policy, each Segment List is encoded as an ERO or RRO.

ERO:

Explicit Route Object, defined in [RFC5440], encodes an explicit path. In the context of SR Policy, an ERO encodes a Segment List.

RRO:

Record Route Object, defined in [RFC5440], encodes the actual signaled path. In the context of SR Policy, an RRO reports a Segment List.

## 2. Motivation

This extension is motivated by the use-cases described below.

### 2.1. Signaling Multiple Segment Lists of an SR Candidate Path

The Candidate Path of an SR Policy is the unit of signaling in PCEP [RFC9862]. A single Candidate Path can consist of multiple Segment Lists. Each Segment List is represented by an Explicit Route Object (ERO). In existing PCEP specifications, a PCEP Label Switched Path (LSP) object is associated with exactly one ERO. This restriction prevents the encoding of multiple Segment Lists (i.e., multiple EROs) within the single LSP.

### 2.2. Splitting of Requested Bandwidth

A Path Computation Client (PCC) may request a path with 80 Gbps of bandwidth, but all links in the network have only 60 Gbps capacity. The Path Computation Element (PCE) can return two paths, that can together carry 80 Gbps. The PCC can then equally or unequally split the incoming 80 Gbps of traffic among the two paths. Section 3.3 introduces a new TLV that carries the path weight that facilitates control of load-balancing of traffic among the multiple paths.

### 2.3. Reverse Path Information

Path Computation Element Communication Protocol (PCEP) Extensions for Associated Bidirectional LSPs [RFC9059] defines a mechanism in PCEP to associate two opposite direction SR Policy Candidate Paths. However, within each Candidate Path there can be multiple Segment Lists, and [RFC9059] does not define a mechanism to specify mapping between Segment Lists of the forward and reverse Candidate Paths.

Certain applications such as Circuit Style SR Policy [I-D.ietf-spring-cs-sr-policy], require the knowledge of reverse paths per Segment List, not just per Candidate Path. For example, when the headend knows the reverse Segment List for each forward Segment List, then Performance Measurement (PM)/Bidirectional Forwarding Detection (BFD) can run a separate session on every Segment List, by imposing a double stack (forward stack followed by reverse stack) onto the packet. If the reverse Segment List is co-routed with the forward Segment List, then the PM/BFD session would traverse the same links in the forward and reverse directions, thus allowing detection of link/node failures in both directions.

### 3. Protocol Extensions

### 3.1. PATH-ATTRIB Object

This document defines the PATH-ATTRIB object that is used to carry per-path information and to act as a separator between EROs/RROs in the <intended-path>/<actual-path> Routing Backus-Naur Form (RBNF) [RFC5511] element. The PATH-ATTRIB object always precedes the ERO or RRO that it applies to. If multiple EROs or RROs are present, then each ERO or RRO MUST be preceded by an PATH-ATTRIB object that describes it.

The PATH-ATTRIB Object-Class value is 45.

The PATH-ATTRIB Object-Type value is 1.

The format of the PATH-ATTRIB object is shown in Figure 1.

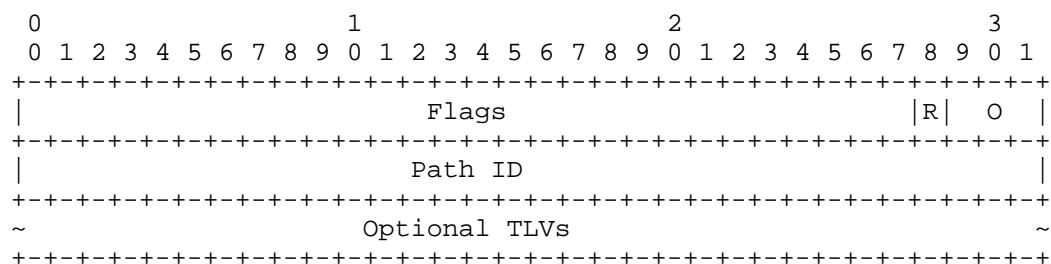


Figure 1: PATH-ATTRIB object format

Flags (32 bits):

- \* 0 (Operational - 3 bits): operational state of the path, same values as the identically named field in the LSP object [RFC8231].

- \* R (Reverse - 1 bit): Indicates this path is reverse, i.e., it originates on the LSP destination and terminates on the LSP source (usually the PCC headend itself). Paths with this flag set serve only informational purpose to the PCC.
- \* Unassigned bits MUST be set to 0 on transmission and MUST be ignored on receipt.

Path ID (32 bits): 4-octet identifier that identifies a path (encoded in the ERO/RRO) within the set of multiple paths under the PCEP LSP. See Section 4.2 for details.

Optional TLVs: Variable length field that can contain one or more TLVs that carry additional per-path information. The specific TLVs that can be included are defined in subsequent sections of this document.

### 3.2. METRIC Object

The PCEP METRIC object can continue to be used at the LSP level to describe properties of the overall LSP. Mechanisms for encoding per-path metrics (e.g., a separate METRIC for each path) are outside the scope of this document and would require further extensions.

### 3.3. MULTIPATH-WEIGHT TLV

New MULTIPATH-WEIGHT TLV is optional in the PATH-ATTRIB object.

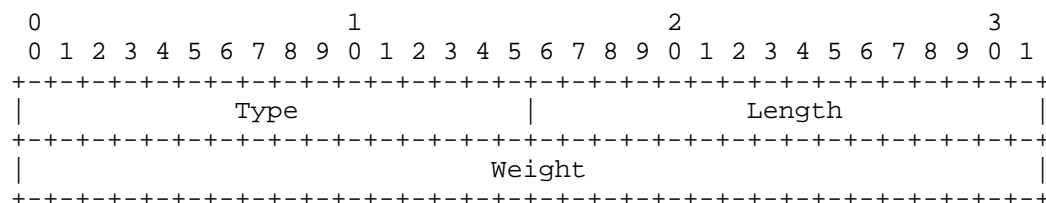


Figure 2: MULTIPATH-WEIGHT TLV format

Type (16 bits): 61 for "MULTIPATH-WEIGHT" TLV.

Length (16 bits): 4 bytes.

Weight (32 bits): unsigned integer weight of this path within the multipath, if W-ECMP is desired. The fraction of flows that a specific ERO/RRO carries is derived from the ratio of its weight to the sum of the weights of all other paths: see Section 4.3 for details.

When the MULTIPATH-WEIGHT TLV is absent from the PATH-ATTRIB object, or the PATH-ATTRIB object is absent from the <intended-path>/<actual-path>, then the Weight of the corresponding path is taken to be 1.

### 3.4. MULTIPATH-BACKUP TLV

New MULTIPATH-BACKUP TLV is optional in the PATH-ATTRIB object.

This TLV is used to describe a set of backup paths protecting a primary path within a PCEP LSP: see Section 4.4 for details. This is similar to path protection, but works at the ECMP path level instead of at the PCEP LSP level.

This functionality is not part of the SR Policy Architecture [RFC9256], but is something optional that may be implemented for certain specialized use cases. This document specifies the TLV format and PCEP encoding; the applicability to specific use cases is defined by other documents. For example, [I-D.draft-ietf-pce-sr-p2mp-policy] describes the application of this TLV in Point-to-Multipoint (P2MP) SR Policy.

Support for the MULTIPATH-BACKUP TLV is currently defined only for P2MP paths. Support for Point-to-Point (P2P) paths is out of scope for this document and may be defined in future documents.

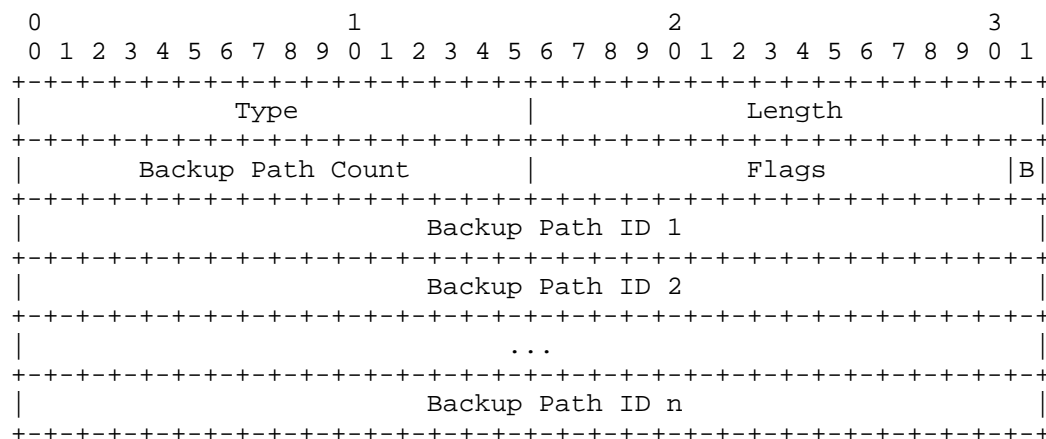


Figure 3: MULTIPATH-BACKUP TLV format

Type (16 bits): 62 for "MULTIPATH-BACKUP" TLV

Length (16 bits): 4 + (N \* 4) bytes (where N is the Backup Path Count)



Backup Path Count (16 bits): Number of backup paths.

Flags (16 bits):

- \* B (Pure Backup): If set, indicates the path is a backup path (e.g., for protection) and not used for load balancing under normal conditions. A pure backup path only carries rerouted traffic after the protected paths fail. If this flag is not set, or if the MULTIPATH-BACKUP TLV is absent, then the path is assumed to be primary that carries normal traffic.
- \* Unassigned bits MUST be set to 0 on transmission and MUST be ignored on receipt.

Backup Path IDs: A series of 4-octet identifiers that reference the Path ID field (see Section 4.2) of other PATH-ATTRIB objects within the same PCEP LSP. These referenced paths act as backup paths that protect this primary path. Each Backup Path ID value MUST match the Path ID of a PATH-ATTRIB object in the same LSP that has the B-flag set (indicating it is a pure backup path).

If a PCEP speaker receives a MULTIPATH-BACKUP TLV applied to a P2P path, it MUST reject the path and send a PCError message with Error-Type = 19 ("Invalid Operation") and Error-Value = 20 ("Not supported path backup").

### 3.5. MULTIPATH-OPPDIR-PATH TLV

New MULTIPATH-OPPDIR-PATH TLV is optional in the PATH-ATTRIB object. Multiple instances of the TLV are allowed in the same PATH-ATTRIB object. Each TLV instance identifies one opposite-direction path for the path described by this PATH-ATTRIB object. This provides per-path level opposite-direction mapping within an LSP. In the context of SR Policy, this corresponds to per-Segment List mapping within a Candidate Path, complementing the Candidate Path level bidirectional association defined in [I-D.ietf-pce-sr-bidir-path], which also describes the usage of this TLV in the context of associated bidirectional SR Paths. See Section 4.5 for operational details.

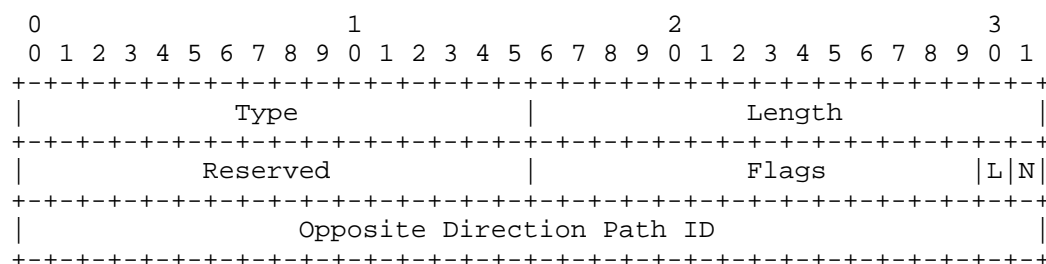


Figure 4: MULTIPATH-OPPDIR-PATH TLV format

Type (16 bits): 63 for "MULTIPATH-OPPDIR-PATH" TLV

Length (16 bits): 8 bytes.

Reserved: This field MUST be set to zero on transmission and MUST be ignored on receipt.

Flags (16 bits):

- \* N (Node co-routed): If set, indicates this path is node co-routed with its opposite direction path, specified in this TLV. Two opposite direction paths are node co-routed if they traverse the same nodes, but MAY traverse different links. If not set, the paths are not guaranteed to be node co-routed (they may or may not traverse the same set of nodes).
- \* L (Link co-routed): If set, indicates this path is link co-routed with its opposite direction path, specified in this TLV. Two opposite direction paths are link co-routed if they traverse the same links (but in opposite directions). Link co-routing implies node co-routing; therefore, it is not necessary to set the N flag when the L flag is set.
- \* Unassigned bits MUST be set to 0 on transmission and MUST be ignored on receipt.

Opposite Direction Path ID (32 bits): References the Path ID field (see Section 4.2) of a PATH-ATTRIB object that identifies a path going in the opposite direction to this path. If no opposite-direction path exists, then this field MUST be set to 0, a value reserved to indicate the absence of a Path ID.

### 3.6. Composite Candidate Path

SR Policy Architecture [RFC9256] defines the concept of a Composite Candidate Path. A regular SR Policy Candidate Path outputs traffic to a set of Segment Lists, while an SR Policy Composite Candidate Path outputs traffic recursively to a set of SR Policies on the same headend. In PCEP, the Composite Candidate Path still consists of PATH-ATTRIB objects, but ERO is replaced by Color of the recursively used SR Policy.

To signal the Composite Candidate Path, we make use of the COLOR TLV, defined in [RFC9863]. For a Composite Candidate Path, the COLOR TLV is included in the PATH-ATTRIB Object, thus allowing each Composite Candidate Path to do ECMP/W-ECMP among SR Policies identified by its

constituent Colors. To achieve W-ECMP, the MULTIPATH-WEIGHT TLV (Section 3.3) is included alongside the COLOR TLV in each PATH-ATTRIB object. If multiple COLOR TLVs are contained in the PATH-ATTRIB object, the first one is processed and the others MUST be ignored.

An ERO MUST be included as per the existing RBNF, this ERO MUST contain no sub-objects. This empty ERO serves as a placeholder to maintain compatibility with existing implementations based on the RBNF defined in [RFC8231]. If the head-end receives a non-empty ERO for a Composite Candidate Path, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = 21 ("Non-empty path").

See Appendix A.3 for an example of the encoding.

### 3.6.1. Per-Flow Candidate Path

Per-Flow Candidate Path builds on top of the concept of the Composite Candidate Path. Each Path in a Per-Flow Candidate Path is assigned a 3-bit forwarding class value, which allows Quality of Service (QoS) classified traffic to be steered depending on the forwarding class.

New MULTIPATH-FORWARD-CLASS TLV is optional in the PATH-ATTRIB object.

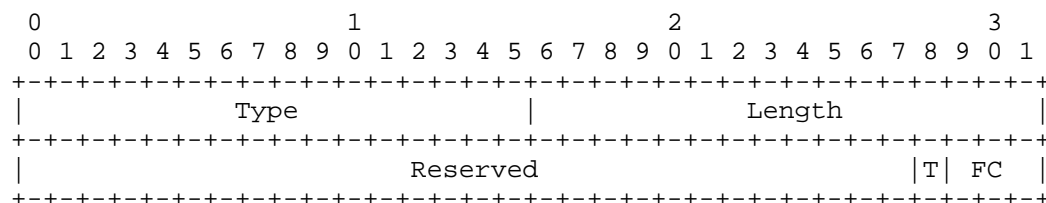


Figure 5: MULTIPATH-FORWARD-CLASS TLV format

Type (16 bits): TBD1 for "MULTIPATH-FORWARD-CLASS" TLV.

Length (16 bits): 4 bytes.

Reserved: This field MUST be set to zero on transmission and MUST be ignored on receipt.

T (1 bit): MPLS TC type. When set, indicates that the FC value is derived from the MPLS Traffic Class (TC) bits as described in Section 8.6 of [RFC9256]. When not set, the interpretation of the FC value is reserved for future use.

FC (3 bits): Forwarding class value. When the T flag is set, this carries the MPLS TC-based forwarding class value as defined in Section 8.6 of [RFC9256]. This value is given by the QoS classifier to traffic entering the given Candidate Path. Different classes of traffic that enter the given Candidate Path can be differentially steered into different Colors. The FC field allows up to 8 different forwarding classes (values 0-7). The semantics of specific FC values are significant at the headend node (PCC) that implements the SR Policy and are determined by that node's local QoS policy or configuration. Coordination of FC value meanings between PCEP peers (e.g., through out-of-band configuration management or operational procedures) is outside the scope of this document.

## 4. Operation

### 4.1. Capability Negotiation

#### 4.1.1. Multipath Capability TLV

New MULTIPATH-CAP TLV is defined. This TLV MAY be present in the OPEN object during PCEP session establishment. It MAY also be present in the LSP object for each individual LSP from the PCC.

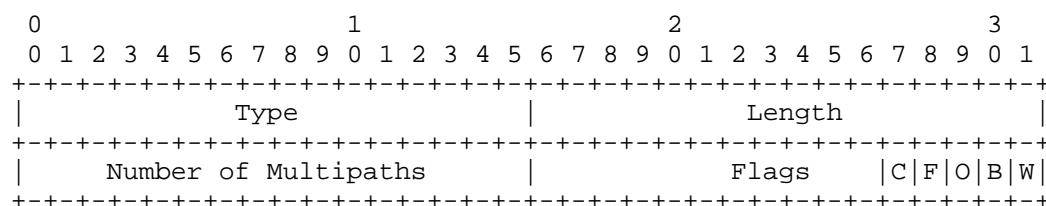


Figure 6: MULTIPATH-CAP TLV format

Type (16 bits): 60 for "MULTIPATH-CAP" TLV.

Length (16 bits): 4 bytes.

Number of Multipaths (16 bits): When sent from a PCC, it indicates how many forward primary multipaths the PCC can install in forwarding. From a PCE, it indicates how many forward primary multipaths the PCE can compute. This count does not include reverse paths (R-flag=1) or pure backup paths (B-flag=1), which are not installed in forwarding for load-balancing purposes. Therefore, the total number of PATH-ATTRIB objects in an LSP may exceed this value when reverse or pure backup paths are also signaled. The value 0 indicates an unlimited number.

Flags (16 bits):

- \* W-flag: whether MULTIPATH-WEIGHT TLV is supported. This flag covers the use of MULTIPATH-WEIGHT for both regular and Composite Candidate Paths.
- \* B-flag: whether MULTIPATH-BACKUP TLV is supported.
- \* O-flag: In the OPEN object, this flag indicates whether the MULTIPATH-OPPDIR-PATH TLV is supported. In the LSP object, this flag indicates that opposite-direction path information is requested or provided for that specific LSP. When set by the PCC (in PCRpt/PCReq), it requests the PCE to provide reverse path information. When set by the PCE (in PCInit/PCUpd/PCRep), it indicates the PCE is providing or will provide reverse path information. In both cases, the PCE SHOULD provide the reverse path information, if it is able to.
- \* F-flag: whether MULTIPATH-FORWARD-CLASS TLV is supported.
- \* C-flag: whether Composite Candidate Path (Section 3.6) is supported, including the use of the COLOR TLV in the PATH-ATTRIB object.
- \* Unassigned bits MUST be set to 0 on transmission and MUST be ignored on receipt.

Note that F-flag and C-flag can be set independently for capability negotiation purposes. While Per-Flow Candidate Path (Section 3.6.1) builds on top of Composite Candidate Path, the F-flag reflects whether the MULTIPATH-FORWARD-CLASS TLV is supported, and the C-flag reflects whether Composite Candidate Path signaling is supported. A peer that supports Per-Flow Candidate Path MUST set both C-flag and F-flag. Note that the F-flag is defined independently of the C-flag to allow for future use cases that may use the MULTIPATH-FORWARD-CLASS TLV for purposes other than Per-Flow Candidate Path; in such cases, the F-flag MAY be set without the C-flag.

When a PCE computes an LSP path, it MUST NOT return more forward multipaths than the minimum of the effective "Number of Multipaths" values of both the PCE and PCC. The effective value for a given LSP is determined by the per-LSP MULTIPATH-CAP TLV in the LSP object if present; otherwise, it defaults to the value from the MULTIPATH-CAP TLV in the OPEN object. This ensures the PCE does not exceed either its own computation capability or the PCC's installation capability. If this TLV is absent from both OPEN and LSP objects, the PCEP speaker does not support multipath and the behavior is consistent with existing PCEP specifications, where a single path is associated with each LSP.

If a PCC receives more paths than it advertised support for, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD3 ("Unsupported multipath capability").

From the PCC, the MULTIPATH-CAP TLV MAY also be present in the LSP object for each individual LSP, to specify per-LSP values. The PCC MUST NOT include this TLV in the LSP object if the TLV was not present in the OPEN objects of both PCEP peers. TLV values in the LSP object override the session default values in the OPEN object. If a PCEP speaker receives a PATH-ATTRIB object but the multipath capability was not successfully negotiated during session establishment, it MUST treat this as an error. The PCEP speaker MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD2 ("Unexpected PATH-ATTRIB object").

For example, the PCC includes this TLV in the OPEN object at session establishment, setting "Number of Multipaths" to 4 and "O-flag" to 0. The PCC also includes this TLV in the LSP object for a particular LSP, setting "Number of Multipaths" to 16 and "O-flag" to 1. This indicates that the PCC only wants to receive the reverse path information for that particular LSP and that this LSP can have up to 16 multipaths, while other LSPs can only have up to 4 multipaths.

Additionally, if a PCEP speaker receives a TLV within the PATH-ATTRIB object (such as MULTIPATH-WEIGHT, MULTIPATH-BACKUP, MULTIPATH-OPPDIR-PATH, or MULTIPATH-FORWARD-CLASS) but the corresponding capability flag was not set in the negotiated MULTIPATH-CAP TLV, it MUST treat this as an error. The PCEP speaker MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD3 ("Unsupported multipath capability").

#### 4.2. Path ID

The Path ID uniquely identifies a Path within the context of an LSP. A single Path ID space is shared among all paths within the LSP, including forward paths, reverse paths (R-flag=1), and pure backup paths (B-flag=1). Path IDs MUST be unique across all these path types within the same LSP. The meaning of "Path" depends on the type of LSP:

- \* For a regular SR Policy Candidate Path, the Paths within that LSP are the Segment Lists.
- \* For a Composite Candidate Path (Section 3.6), the Paths within that LSP are the constituent SR Policies, each of which is identified by its Color (carried in the COLOR TLV within the corresponding PATH-ATTRIB object).

Value 0 indicates an unallocated Path ID. The value of 0 MAY be used when this Path is not referenced and the allocation of a Path ID is not necessary.

Path IDs are allocated by the PCEP peer that owns the LSP. If the LSP is delegated to the PCE, then the PCE allocates the Path IDs and sends them in the PCReply/PCUpd/PCInitiate messages. If the LSP is locally computed on the PCC, then the PCC allocates the Path IDs and sends them in the PCReq/PCRpT messages.

If a PCEP speaker detects that there are two Paths with the same non-zero Path ID, then the PCEP speaker MUST send PCErr message with Error-Type = 1 ("Reception of an invalid object") and Error-Value = 38 ("Conflicting Path ID"). Multiple paths MAY have Path ID set to 0, as this value indicates those paths are not referenced and do not require unique identification.

#### 4.3. Signaling Multiple Paths for Load-Balancing

The PATH-ATTRIB object can be used to signal multiple paths and indicate (un)equal load-balancing amongst the set of multipaths. In this case, the PATH-ATTRIB is populated for each ERO as follows:

1. The PCE MAY assign a unique Path ID to each ERO path and populate it inside the PATH-ATTRIB object. The Path ID is unique within the context of a PLSP (PCE Label Switched Path) (when non-zero).
2. The PCE MAY include the MULTIPATH-WEIGHT TLV inside the PATH-ATTRIB object, populating a weight value to reflect the relative share of traffic to be carried by the path. If the MULTIPATH-WEIGHT is not carried inside a PATH-ATTRIB object, the PCC MUST assume the default weight of 1 when computing the traffic share.
3. The PCC derives the fraction of flows carried by a specific primary path from the ratio of its weight to the sum of all other multipath weights. For SR Policy, the use of weights for load-balancing between Segment Lists of a Candidate Path is described in Section 2.11 of [RFC9256].

#### 4.4. Signaling Multiple Paths for Protection

The PATH-ATTRIB object can be used to describe a set of backup paths protecting a primary path within a PCEP LSP. This capability is currently defined only for P2MP paths. Support for P2P paths with the MULTIPATH-BACKUP TLV is out of scope for this document. In this case, the PATH-ATTRIB is populated for each ERO as follows:

1. The PCE assigns a unique Path ID to each ERO path and populates it inside the PATH-ATTRIB object. The Path ID is unique within the context of a PLSP (PCE Label Switched Path).
2. The PCE MAY include the MULTIPATH-BACKUP TLV inside the PATH-ATTRIB object for each ERO that is protected, specifying the backup path IDs to reflect the set of backup paths protecting the primary path. The PCE updates the Length field and Backup Path Count in the MULTIPATH-BACKUP according to the number of backup path IDs included.
3. The PCE MAY include the MULTIPATH-BACKUP TLV inside the PATH-ATTRIB object for each ERO that is unprotected. In this case, MULTIPATH-BACKUP does not carry any backup path IDs in the TLV. If the path acts as a pure backup (i.e., the path only carries rerouted traffic after the protected paths fail), then the B flag MUST be set.

Primary paths which do not include the MULTIPATH-BACKUP TLV are assumed to be protected by all the backup paths (i.e., omitting the TLV is equivalent to including the TLV with all the backup path IDs filled in).

Note that a given PCC may not support certain backup combinations, such as a backup path that is itself protected by another backup path, etc. If a PCC does not support a requested backup scenario, the PCC MUST send a PCErrror message with Error-Type = 19 ("Invalid Operation") and Error-Value = 20 ("Not supported path backup"). Additionally, if a P2P path is sent with a MULTIPATH-BACKUP TLV, the PCC or PCE SHOULD reject it with the same PCErrror as above.

#### 4.5. Signaling Opposite-Direction Path Information

The PATH-ATTRIB object can be used to signal opposite-direction path associations within a PCEP LSP. This capability is used to establish bidirectional path relationships where forward and reverse paths can be explicitly mapped to each other. In this case, the PATH-ATTRIB is populated for each ERO as follows:

1. The PCEP peer (PCC or PCE) allocates a unique Path ID to each path and populates it inside the PATH-ATTRIB object. The Path ID is unique within the context of a PLSP (PCE Label Switched Path).
2. For paths that have opposite-direction counterparts, the MULTIPATH-OPPDIR-PATH TLV is added to the PATH-ATTRIB object. The Opposite Direction Path ID field is set to reference the Path ID of the corresponding opposite-direction path.



3. Multiple instances of the MULTIPATH-OPPDIR-PATH TLV MAY be present in the same PATH-ATTRIB object to support many-to-many mappings between forward and reverse paths. This allows a single forward path to map to multiple reverse paths and vice versa. Many-to-many mapping can occur when a Segment List contains Node Segment(s) that traverse parallel links at a midpoint. The reverse of this Segment List may require multiple Reverse Segment Lists to cover all the parallel links at the midpoint.
4. The N-flag and L-flag in the MULTIPATH-OPPDIR-PATH TLV MAY be set to indicate node co-routing or link co-routing respectively. These flags inform the receiver about the relationship between the forward and reverse paths.
5. For paths that have no opposite-direction counterpart, the MULTIPATH-OPPDIR-PATH TLV is omitted from the PATH-ATTRIB object.

Forward paths (R-flag=0) and reverse paths (R-flag=1) are included in the same PCEP LSP, allowing bidirectional relationships to be established atomically. The opposite-direction path associations MUST be symmetric within the same LSP. When path A references path B as its opposite-direction path, path B MUST also reference path A as its opposite-direction path. Additionally, their R-flags in the PATH-ATTRIB object MUST have opposite values (one set to 0, the other to 1).

If a PCEP speaker receives an opposite-direction path mapping that is asymmetric or where the R-flags are inconsistent, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD4 ("Invalid opposite-direction path mapping").

See Appendix A.4 for an example of usage.

## 5. PCEP Message Extensions

The RBNF of PCRpt and PCUpd messages, as defined in [RFC8231], use a combination of <intended-path> and/or <actual-path>. PCReq and PCRep messages, as defined in [RFC5440] and extended by [RFC8231], directly include ERO and RRO within their respective message structures rather than encapsulating them within <intended-path> or <actual-path> constructs.

As specified in Section 6.1 of [RFC8231], within the context of messages that use these constructs, <intended-path> is represented by the ERO and <actual-path> is represented by the RRO:

<intended-path> ::= <ERO>

<actual-path> ::= <RRO>

This document extends [RFC8231] by allowing multiple EROs/RROs to be present in the <intended-path>/<actual-path>:

<intended-path> ::= <ERO> |  
                  <PATH-ATTRIB><ERO>[<intended-path-multipath>]

<intended-path-multipath> ::= <PATH-ATTRIB><ERO>  
                              [<intended-path-multipath>]

<actual-path> ::= <RRO> |  
                  <PATH-ATTRIB><RRO>[<actual-path-multipath>]

<actual-path-multipath> ::= <PATH-ATTRIB><RRO>  
                              [<actual-path-multipath>]

Similarly, this document extends [RFC8281] by allowing multiple paths in the PCInitiate message by allowing multiple EROs with their associated path attributes. The PCE-initiated LSP instantiation format is updated to:

<PCE-initiated-lsp-instantiation> ::= <SRP>  
                                      <LSP>  
                                      [<END-POINTS>]  
                                      <intended-path>  
                                      [<attribute-list>]

where <intended-path> follows the recursive definition above, allowing multiple paths to be signaled in a single PCInitiate message. When multiple paths are present, each ERO MUST be preceded by a PATH-ATTRIB object that describes it. A single path MAY be sent as a bare ERO without PATH-ATTRIB for backward compatibility.

## 6. Implementation Status

Note to the RFC Editor - remove this section before publication, as well as remove the reference to [RFC7942].

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort

has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [RFC7942], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

#### 6.1. Cisco Systems

Organization: Cisco Systems  
Implementation: IOS-XR PCC and PCE  
Description: Circuit-Style SR Policies  
Maturity Level: Supported feature  
Coverage: Multiple Segment Lists and reverse paths in SR Policy  
Contact: mkoldych@cisco.com

#### 6.2. Ciena Corp

Organization: Ciena Corp  
Implementation: Head-end and controller  
Maturity Level: Proof of concept  
Coverage: Partial  
Contact: byadav@ciena.com

#### 6.3. Huawei Technologies

Organization: Huawei Technologies Co.,Ltd.  
Implementation: Huawei's Router and Controller  
Maturity Level: Proof of concept  
Coverage: Partial  
Contact: tanren@huawei.com

### 7. IANA Considerations

All IANA actions in this section pertain to the "Path Computation Element Protocol (PCEP) Numbers" registry group.

#### 7.1. PCEP Object

IANA is requested to confirm the following allocation in the "PCEP Objects" registry:

Object-Class Value	Name	Object-Type Value	Reference
45	PATH-ATTRIB	0: Reserved 1: PATH-ATTRIB 2-15: Unassigned	This document

Object-Type values are managed via the IETF Review policy as per [RFC8126].

## 7.2. PCEP TLV

IANA is requested to confirm the following allocations in the "PCEP TLV Type Indicators" registry:

TLV Type Value	TLV Name	Reference
60	MULTIPATH-CAP	This document
61	MULTIPATH-WEIGHT	This document
62	MULTIPATH-BACKUP	This document
63	MULTIPATH-OPPDIR-PATH	This document

IANA is requested to make new allocations in the "PCEP TLV Type Indicators" registry:

TLV Type Value	TLV Name	Reference
TBD1	MULTIPATH-FORWARD-CLASS	This document

## 7.3. PCEP-Error Object

IANA is requested to confirm the following allocations in the "PCEP-ERROR Object Error Types and Values" registry:

Error-Type	Error-Value	Reference
10	38 - Conflicting Path ID	This document
19	20 - Not supported path backup	This document
19	21 - Non-empty path	This document

IANA is requested to make new allocations in the "PCEP-ERROR Object Error Types and Values" registry:

Error-Type	Error-Value	Reference
10	TBD2 - Unexpected PATH-ATTRIB Object	This document
19	TBD3 - Unsupported multipath capability	This document
19	TBD4 - Invalid opposite-direction path mapping	This document

#### 7.4. Flags in the MULTIPATH-CAP TLV

IANA is requested to create a new registry called "Flags in MULTIPATH-CAP TLV" to manage the Flag field of the MULTIPATH-CAP TLV. New values are to be assigned by "IETF review" [RFC8126]

Bit	Description	Reference
0-10	Unassigned	This document
11	C-flag: Composite Candidate Path support	This document
12	F-flag: MULTIPATH-FORWARD-CLASS TLV support	This document
13	O-flag: MULTIPATH-OPPDIR-PATH TLV support	This document
14	B-flag: MULTIPATH-BACKUP TLV support	This document
15	W-flag: MULTIPATH-WEIGHT TLV support	This document

#### 7.5. Flags in the PATH-ATTRIB Object

IANA is requested to create a new registry called "Flags in PATH-ATTRIB Object" to manage the Flag field of the PATH-ATTRIB object. New values are to be assigned by "IETF review" [RFC8126]

Bit	Description	Reference
0-27	Unassigned	This document
28	R-flag: Reverse path	This document
29-31	O-flag: Operational state	This document

#### 7.6. Flags in the MULTIPATH-BACKUP TLV

IANA is requested to create a new registry called "Flags in MULTIPATH-BACKUP TLV" to manage the Flag field of the MULTIPATH-BACKUP TLV. New values are to be assigned by "IETF review" [RFC8126]

Bit	Description	Reference
0-14	Unassigned	This document
15	B-flag: Pure backup	This document

#### 7.7. Flags in the MULTIPATH-OPPDIR-PATH TLV

IANA is requested to create a new registry called "Flags in MULTIPATH-OPPDIR-PATH TLV" to manage the Flag field of the MULTIPATH-OPPDIR-PATH TLV. New values are to be assigned by "IETF review" [RFC8126]

Bit	Description	Reference
0-12	Unassigned	This document
14	L-flag: Link co-routed	This document
15	N-flag: Node co-routed	This document

#### 7.8. Flags in the MULTIPATH-FORWARD-CLASS TLV

IANA is requested to create a new registry called "Flags in MULTIPATH-FORWARD-CLASS TLV" to manage the Flag field of the MULTIPATH-FORWARD-CLASS TLV. New values are to be assigned by "IETF review" [RFC8126]

Bit	Description	Reference
0-27	Unassigned	This document
28	T-flag: MPLS TC type	This document
29-31	FC: Forwarding class	This document

### 8. Security Considerations

The security considerations described in [RFC5440], [RFC8231], [RFC8281], [RFC8664], [RFC9256], [RFC9862] and [RFC9863] are applicable to this specification.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions can only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] [I-D.ietf-pce-pceps-tls13] as per the recommendations and best current practices in [RFC9325].

## 9. Operational Considerations

All manageability requirements and considerations listed in [RFC5440], [RFC8231], [RFC8664], and [RFC9256] apply to the PCEP protocol extensions defined in this document. In addition, the requirements and considerations listed in this section apply.

### 9.1. Control of Function and Policy

A PCEP speaker (PCC or PCE) implementation SHOULD allow an operator to enable or disable the multipath capabilities advertised in the MULTIPATH-CAP TLV (see Section 4).

### 9.2. Information and Data Models

It is expected that a future version of the PCEP YANG module [RFC9826] will be extended to include the PCEP extensions defined in this document.

### 9.3. Liveness Detection and Monitoring

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

### 9.4. Verify Correct Operations

In addition to the verification requirements in [RFC5440] and [RFC8231], the following considerations apply:

- \* An implementation SHOULD allow an operator to view the capabilities advertised in the MULTIPATH-CAP TLV by each PCEP peer for a session and for individual LSPs.
- \* An implementation SHOULD allow an operator to view the PATH-ATTRIB object and all its associated TLVs for each path within an LSP. This includes the Path ID, weight, backup information, and opposite-direction path associations.
- \* An implementation SHOULD provide a mechanism to log and display the new PCEP errors defined in this document



### 9.5. Requirements On Other Protocols

The PCEP extensions defined in this document do not impose any new requirements on other protocols.

### 9.6. Impact On Network Operations

The mechanisms in this document allow for more complex LSP structures with multiple paths. Network operators should be aware of the potential increase in PCEP message sizes and the additional state that must be maintained by PCEP speakers. The "Number of Multipaths" field in the MULTIPATH-CAP TLV can be used to control the scale of multipath computations and state.

## 10. Acknowledgement

Thanks to Adrian Farrel for shepherding this document, Ketan Talaulikar for his thorough AD review, Dhruv Dhody for ideas and discussion, and Diego Achaval, Quan Xiong, Giuseppe Fioccola, Italo Busi, Yuan Yaping, and Cheng Li for their reviews.

## 11. Contributors

### 11.1. Original Authors

The following individuals are the original authors who initiated and developed the core work of this document. Mike Koldychev is also listed as editor in the Authors' Addresses section. The remaining individuals appear here rather than in the Authors' Addresses section due to the IETF guidelines on the maximum number of listed authors, but should be considered co-authors of this document. Samuel Sidor joined the effort at a later stage as an additional editor.

Mike Koldychev (also listed as editor)  
Ciena Corporation  
Email: mkoldych@ciena.com

Siva Sivabalan  
Ciena Corporation  
Email: ssivabal@ciena.com

Tarek Saad  
Cisco Systems  
Email: tsaad@cisco.com

Vishnu Pavan Beeram  
Juniper Networks, Inc.  
Email: vbeeram@juniper.net

Hooman Bidgoli  
Nokia  
Email: hooman.bidgoli@nokia.com

Shuping Peng  
Huawei Technologies  
Email: pengshuping@huawei.com

Bhupendra Yadav  
Ciena  
Email: byadav@ciena.com

Gyan Mishra  
Verizon Inc.  
Email: hayabusagsm@gmail.com

#### 11.2. Additional Contributors

The following individuals made contributions to this document:

Zafar Ali  
Cisco Systems  
Email: zali@cisco.com

Andrew Stone  
Nokia  
Email: andrew.stone@nokia.com

Chen Ran  
ZTE  
Email: chen.ran@zte.com.cn

## 12. References

### 12.1. Normative References

- [I-D.ietf-pce-pceps-tls13]  
Dhody, D., Turner, S., and R. Housley, "Updates for PCEPS: TLS Connection Establishment Restrictions", Work in Progress, Internet-Draft, draft-ietf-pce-pceps-tls13-04, 9 January 2024, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-pceps-tls13-04>>.
- [I-D.ietf-pce-sr-bidir-path]  
Li, C., Chen, M., Cheng, W., Gandhi, R., and Q. Xiong, "Path Computation Element Communication Protocol (PCEP) Extensions for Associated Bidirectional Segment Routing (SR) LSPs", Work in Progress, Internet-Draft, draft-ietf-pce-sr-bidir-path-25, 6 March 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-sr-bidir-path-25>>.
- [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/rfc/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/rfc/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/rfc/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/rfc/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/rfc/rfc8231>>.

- [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/rfc/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/rfc/rfc8281>>.
- [RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing", RFC 8664, DOI 10.17487/RFC8664, December 2019, <<https://www.rfc-editor.org/rfc/rfc8664>>.
- [RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <<https://www.rfc-editor.org/rfc/rfc9256>>.
- [RFC9325] Sheffer, Y., Saint-Andre, P., and T. Fossati, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 9325, DOI 10.17487/RFC9325, November 2022, <<https://www.rfc-editor.org/rfc/rfc9325>>.
- [RFC9603] Li, C., Ed., Kaladharan, P., Sivabalan, S., Koldychev, M., and Y. Zhu, "Path Computation Element Communication Protocol (PCEP) Extensions for IPv6 Segment Routing", RFC 9603, DOI 10.17487/RFC9603, July 2024, <<https://www.rfc-editor.org/rfc/rfc9603>>.
- [RFC9862] Koldychev, M., Sivabalan, S., Sidor, S., Barth, C., Peng, S., and H. Bidgoli, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing (SR) Policy Candidate Paths", RFC 9862, DOI 10.17487/RFC9862, October 2025, <<https://www.rfc-editor.org/rfc/rfc9862>>.
- [RFC9863] Rajagopalan, B., Beeram, V., Peng, S., Koldychev, M., and G. Mishra, "Path Computation Element Protocol (PCEP) Extension for Color", RFC 9863, DOI 10.17487/RFC9863, October 2025, <<https://www.rfc-editor.org/rfc/rfc9863>>.

## 12.2. Informative References

[I-D.draft-ietf-pce-sr-p2mp-policy]

Bidgoli, H., Voyer, D., Budhiraja, A., Parekh, R., and S. Sivabalan, "PCEP extensions for SR P2MP Policy", Work in Progress, Internet-Draft, draft-ietf-pce-sr-p2mp-policy-14, 23 February 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-sr-p2mp-policy-14>>.

[I-D.ietf-spring-cs-sr-policy]

Schmutzer, C., Ali, Z., Maheshwari, P., Rokui, R., and A. Stone, "Circuit Style Segment Routing Policy", Work in Progress, Internet-Draft, draft-ietf-spring-cs-sr-policy-17, 12 March 2026, <<https://datatracker.ietf.org/doc/html/draft-ietf-spring-cs-sr-policy-17>>.

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

[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/rfc/rfc7942>>.

[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/rfc/rfc8126>>.

[RFC9059] Gandhi, R., Ed., Barth, C., and B. Wen, "Path Computation Element Communication Protocol (PCEP) Extensions for Associated Bidirectional Label Switched Paths (LSPs)", RFC 9059, DOI 10.17487/RFC9059, June 2021, <<https://www.rfc-editor.org/rfc/rfc9059>>.

[RFC9826] Dhody, D., Ed., Beeram, V., Hardwick, J., and J. Tantsura, "A YANG Data Model for the Path Computation Element Communication Protocol (PCEP)", RFC 9826, DOI 10.17487/RFC9826, September 2025, <<https://www.rfc-editor.org/rfc/rfc9826>>.

## Appendix A. Examples

### A.1. SR Policy Candidate Path with Multiple Segment Lists

Consider the following sample SR Policy.

```
SR policy POL1 <headend, color, endpoint>
  Candidate Path CP1 <protocol-origin = 20, originator-asn = 100,
    originator-address = 192.0.2.1,
    discriminator = 1>
    Preference 200
    Weight W1, SID-List1 <SID11...SID1i>
    Weight W2, SID-List2 <SID21...SID2j>
  Candidate Path CP2 <protocol-origin = 20, originator-asn = 100,
    originator-address = 198.51.100.1,
    discriminator = 2>
    Preference 100
    Weight W3, SID-List3 <SID31...SID3i>
    Weight W4, SID-List4 <SID41...SID4j>
```

As specified in [RFC9862], CP1 and CP2 are signaled as separate state-report elements and each has a unique PLSP-ID, assigned by the PCC. For this example, PLSP-ID 100 is assigned to CP1 and PLSP-ID 200 to CP2.

The state-report (as defined in [RFC8231]) for CP1 can be encoded as:

```
<state-report> =
  <LSP PLSP-ID=100>
  <ASSOCIATION>
  <PATH-ATTRIB Path ID=1 <WEIGHT-TLV Weight=W1>>
  <ERO SID-List1>
  <PATH-ATTRIB Path ID=2 <WEIGHT-TLV Weight=W2>>
  <ERO SID-List2>
```

The state-report for CP2 can be encoded as:

```
<state-report> =
  <LSP PLSP-ID=200>
  <ASSOCIATION>
  <PATH-ATTRIB Path ID=1 <WEIGHT-TLV Weight=W3>>
  <ERO SID-List3>
  <PATH-ATTRIB Path ID=2 <WEIGHT-TLV Weight=W4>>
  <ERO SID-List4>
```

The above sample state-report elements only specify the minimum mandatory objects, of course other objects like SRP, LSPA, METRIC, etc., are allowed to be inserted.

Note that the syntax

```
<PATH-ATTRIB Path ID=1 <WEIGHT-TLV Weight=W1>>
```

means that this is PATH-ATTRIB object with Path ID field set to 1 and with a MULTIPATH-WEIGHT TLV carrying weight of "W1".

#### A.2. Two Primary Paths Protected by One Backup Path

Suppose there are 3 paths: A, B, C. Where A and B are primary and C is to be used only when A or B fail. Suppose the Path IDs for A, B, C are respectively 1, 2, 3. This would be encoded in a state-report as:

```
<state-report> =  
  <LSP>  
  <ASSOCIATION>  
  <PATH-ATTRIB Path ID=1 <BACKUP-TLV B=0, Backup_Paths=[3]>>  
  <ERO A>  
  <PATH-ATTRIB Path ID=2 <BACKUP-TLV B=0, Backup_Paths=[3]>>  
  <ERO B>  
  <PATH-ATTRIB Path ID=3 <BACKUP-TLV B=1, Backup_Paths=[]>>  
  <ERO C>
```

Note that the syntax

```
<PATH-ATTRIB Path ID=1 <BACKUP-TLV B=0, Backup_Paths=[3]>>
```

means that this is PATH-ATTRIB object with Path ID field set to 1 and with a MULTIPATH-BACKUP TLV that has B-flag cleared and contains a single backup path with Backup Path ID of 3.

#### A.3. Composite Candidate Path

Consider the following Composite Candidate Path.

```
SR policy POL100 <headend = H1, color = 100, endpoint = E1>  
  Candidate Path CP1 <protocol-origin = 20, originator-asn = 100,  
    originator-address = 192.0.2.1,  
    discriminator = 1>  
    Preference 200  
    Weight W1, SR policy <color = 1>  
    Weight W2, SR policy <color = 2>
```

This is signaled in PCEP as:

```
<LSP PLSP-ID=100>
  <ASSOCIATION>
    <PATH-ATTRIB Path ID=1
      <WEIGHT-TLV Weight=W1>
      <COLOR-TLV Color=1>>
    <ERO (empty)>
    <PATH-ATTRIB Path ID=2
      <WEIGHT-TLV Weight=W2>
      <COLOR-TLV Color=2>>
    <ERO (empty)>
```

#### A.4. Opposite Direction Tunnels

Consider the two opposite-direction SR Policies between endpoints H1 and E1.

SR policy POL1 <headend = H1, color, endpoint = E1>

```
Candidate Path CP1
  Preference 200
  Bidirectional Association = A1
  SID-List = <H1,M1,M2,E1>
  SID-List = <H1,M3,M4,E1>
Candidate Path CP2
  Preference 100
  Bidirectional Association = A2
  SID-List = <H1,M5,M6,E1>
  SID-List = <H1,M7,M8,E1>
```

SR policy POL2 <headend = E1, color, endpoint = H1>

```
Candidate Path CP1
  Preference 200
  Bidirectional Association = A1
  SID-List = <E1,M2,M1,H1>
  SID-List = <E1,M4,M3,H1>
Candidate Path CP2
  Preference 100
  Bidirectional Association = A2
  SID-List = <E1,M6,M5,H1>
```

The state-report for POL1, CP1 can be encoded as:



```

<state-report> =
  <LSP PLSP-ID=100>
  <BIDIRECTIONAL ASSOCIATION = A1>
  <PATH-ATTRIB Path ID=1 R-flag=0
    <OPPDIR-PATH-TLV OppositePath ID=3>>
  <ERO <H1,M1,M2,E1>>
  <PATH-ATTRIB Path ID=2 R-flag=0
    <OPPDIR-PATH-TLV OppositePath ID=4>>
  <ERO <H1,M3,M4,E1>>
  <PATH-ATTRIB Path ID=3 R-flag=1
    <OPPDIR-PATH-TLV OppositePath ID=1>>
  <ERO <E1,M2,M1,H1>>
  <PATH-ATTRIB Path ID=4 R-flag=1
    <OPPDIR-PATH-TLV OppositePath ID=2>>
  <ERO <E1,M4,M3,H1>>

```

The state-report for POL1, CP2 can be encoded as:

```

<state-report> =
  <LSP PLSP-ID=200>
  <BIDIRECTIONAL ASSOCIATION = A2>
  <PATH-ATTRIB Path ID=1 R-flag=0
    <OPPDIR-PATH-TLV OppositePath ID=3>>
  <ERO <H1,M5,M6,E1>>
  <PATH-ATTRIB Path ID=2 R-flag=0
    <OPPDIR-PATH-TLV OppositePath ID=0>>
  <ERO <H1,M7,M8,E1>>
  <PATH-ATTRIB Path ID=3 R-flag=1
    <OPPDIR-PATH-TLV OppositePath ID=1>>
  <ERO <E1,M6,M5,H1>>

```

The state-report for POL2, CP1 can be encoded as:

```

<state-report> =
  <LSP PLSP-ID=100>
  <BIDIRECTIONAL ASSOCIATION = A1>
  <PATH-ATTRIB Path ID=1 R-flag=0
    <OPPDIR-PATH-TLV OppositePath ID=3>>
  <ERO <E1,M2,M1,H1>>
  <PATH-ATTRIB Path ID=2 R-flag=0
    <OPPDIR-PATH-TLV OppositePath ID=4>>
  <ERO <E1,M4,M3,H1>>
  <PATH-ATTRIB Path ID=3 R-flag=1
    <OPPDIR-PATH-TLV OppositePath ID=1>>
  <ERO <H1,M1,M2,E1>>
  <PATH-ATTRIB Path ID=4 R-flag=1
    <OPPDIR-PATH-TLV OppositePath ID=2>>
  <ERO <H1,M3,M4,E1>>

```

The state-report for POL2, CP2 can be encoded as:

```
<state-report> =  
  <LSP PLSP-ID=200>  
  <BIDIRECTIONAL ASSOCIATION = A2>  
  <PATH-ATTRIB Path ID=1 R-flag=0  
    <OPPDIR-PATH-TLV OppositePath ID=3>>  
  <ERO <E1,M6,M5,H1>>  
  <PATH-ATTRIB Path ID=2 R-flag=1  
    <OPPDIR-PATH-TLV OppositePath ID=0>>  
  <ERO <H1,M7,M8,E1>>  
  <PATH-ATTRIB Path ID=3 R-flag=1  
    <OPPDIR-PATH-TLV OppositePath ID=1>>  
  <ERO <H1,M5,M6,E1>>
```

#### Authors' Addresses

Mike Koldychev (editor)  
Ciena Corporation  
Email: mkoldych@ciena.com

Samuel Sidor (editor)  
Cisco Systems.  
Email: ssidor@cisco.com