

Workgroup: PCE Working Group
Internet-Draft: draft-ietf-pce-multipath-07
Published: 14 November 2022
Intended Status: Standards Track
Expires: 18 May 2023
Authors: M. Koldychev S. Sivabalan
 Cisco Systems, Inc. Ciena Corporation
 T. Saad V. Beeram
 Juniper Networks, Inc. Juniper Networks, Inc.
 H. Bidgoli B. Yadav S. Peng
 Nokia Ciena Huawei Technologies
 G. Mishra
 Verizon Inc.

PCEP Extensions for Signaling Multipath Information

Abstract

Path computation algorithms are not limited to return a single optimal path. Multiple paths may exist that satisfy the given objectives and constraints. This document defines a mechanism to encode multiple paths for a single set of objectives and constraints. This is a generic PCEP mechanism, not specific to any path setup type or dataplane. The mechanism is 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 18 May 2023.

Copyright Notice

Copyright (c) 2022 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](#)
- [2. Terminology](#)
 - [2.1. Terms and Abbreviations](#)
- [3. Motivation](#)
 - [3.1. Signaling Multiple Segment-Lists of an SR Candidate-Path](#)
 - [3.2. Splitting of Requested Bandwidth](#)
 - [3.3. Providing Backup path for Protection](#)
 - [3.4. Reverse Path Information](#)
- [4. Protocol Extensions](#)
 - [4.1. Multipath Capability TLV](#)
 - [4.2. Path Attributes Object](#)
 - [4.3. Multipath Weight TLV](#)
 - [4.4. Multipath Backup TLV](#)
 - [4.5. Multipath Opposite Direction Path TLV](#)
 - [4.6. Composite Candidate Path](#)
- [5. Operation](#)
 - [5.1. Capability Negotiation](#)
 - [5.2. Path ID](#)
 - [5.3. Signaling Multiple Paths for Loadbalancing](#)
 - [5.4. Signaling Multiple Paths for Protection](#)
- [6. PCEP Message Extensions](#)
- [7. Examples](#)
 - [7.1. SR Policy Candidate-Path with Multiple Segment-Lists](#)
 - [7.2. Two Primary Paths Protected by One Backup Path](#)
 - [7.3. Composite Candidate Path](#)
 - [7.4. Opposite Direction Tunnels](#)
- [8. Implementation Status](#)
 - [8.1. Cisco Systems](#)
 - [8.2. Ciena Corp](#)
 - [8.3. Huawei Technologies](#)
- [9. IANA Considerations](#)
 - [9.1. PCEP Object](#)
 - [9.2. PCEP TLV](#)
 - [9.3. PCEP-Error Object](#)
 - [9.4. Flags in the Multipath Capability TLV](#)
 - [9.5. Flags in the Path Attribute Object](#)
 - [9.6. Flags in the Multipath Backup TLV](#)
 - [9.7. Flags in the Multipath Opposite Direction Path TLV](#)
- [10. Security Considerations](#)

- [11. Acknowledgement](#)
- [12. Contributors](#)
- [13. References](#)
 - [13.1. Normative References](#)
 - [13.2. Informative References](#)
- [Authors' Addresses](#)

1. Introduction

Path Computation Element (PCE) Communication Protocol (PCEP) [[RFC5440](#)] enables the communication between a Path Computation Client (PCC) and a Path Control Element (PCE), or between two PCEs based on the PCE architecture [[RFC4655](#)].

PCEP Extensions for the Stateful PCE Model [[RFC8231](#)] describes a set of extensions to PCEP that enable active control of Multiprotocol Label Switching Traffic Engineering (MPLS-TE) and Generalized MPLS (GMPLS) tunnels. [[RFC8281](#)] describes the setup and teardown of PCE-initiated LSPs under the active stateful PCE model, without the need for local configuration on the PCC, thus allowing for dynamic centralized control of a network.

PCEP Extensions for Segment Routing [[RFC8664](#)] specifies extensions to the Path Computation Element Protocol (PCEP) that allow a stateful PCE to compute and initiate Traffic Engineering (TE) paths, as well as for a PCC to request a path subject to certain constraint(s) and optimization criteria in SR networks.

Segment Routing Policy for Traffic Engineering [[I-D.ietf-spring-segment-routing-policy](#)] details the concepts of 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 standards only allow for signaling of one Segment-List per Candidate-Path. PCEP extension to support Segment Routing Policy Candidate Paths [[I-D.ietf-pce-segment-routing-policy-cp](#)] specifically avoids defining how to signal multipath information, and states that this will be defined in another document.

This document defines the required extensions that allow the signaling of multipath information via PCEP.

2. Terminology

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.

2.1. Terms and Abbreviations

The following terms are used in this document:

PCEP Tunnel:

The object identified by the PLSP-ID, see [\[I-D.koldychev-pce-operational\]](#) for more details.

3. Motivation

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

3.1. Signaling Multiple Segment-Lists of an SR Candidate-Path

The Candidate-Path of an SR Policy is the unit of report/update in PCEP, see [\[I-D.ietf-pce-segment-routing-policy-cp\]](#). Each Candidate-Path can contain multiple Segment-Lists and each Segment-List is encoded by one ERO. However, each PCEP LSP can contain only a single ERO, which prevents us from encoding multiple Segment-Lists within the same SR Candidate-Path.

With the help of the protocol extensions defined in this document, this limitation is overcome.

3.2. Splitting of Requested Bandwidth

A PCC may request a path with 80 Gbps of bandwidth, but all links in the network have only 50 Gbps capacity. The 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 4.3](#) introduces a new TLV that carries the path weight that allows for distribution of incoming traffic on to the multiple paths.

3.3. Providing Backup path for Protection

It is desirable for the PCE to compute and signal to the PCC a backup path that is used to protect a primary path within the multipaths in a given LSP.

Note that [\[RFC8745\]](#) specify the Path Protection association among LSPs. The use of [\[RFC8745\]](#) with multipath is out of scope of this document and is for future study.

When multipath is used, a backup path may protect one or more primary paths. For this reason, primary and backup path identifiers are needed to indicate which backup path(s) protect which primary path(s). [Section 4.4](#) introduces a new TLV that carries the required information.

3.4. Reverse Path Information

Certain applications, such as Circuit Style SR Policy [[I-D.schmutzer-pce-cs-sr-policy](#)], require the head-end to know both forward and reverse paths for each of the segment lists of an SR Policy in order to run OAM/PM/BFD protocols on each Segment List as a separate circuit.

4. Protocol Extensions

4.1. Multipath Capability TLV

We define the MULTIPATH-CAP TLV that MAY be present in the OPEN object and/or the LSP object. The purpose of this TLV is two-fold:

1. From PCC: it tells how many multipaths per PCEP Tunnel, the PCC can install in forwarding.
2. From PCE: it tells that the PCE supports this standard and how many multipaths per PCEP Tunnel, the PCE can compute.

Only the first instance of this TLV can be processed, subsequent instances SHOULD be ignored.

[Section 5](#) specify the usage of this TLV with Open message (within the OPEN object) and other PCEP messages (within the LSP object).

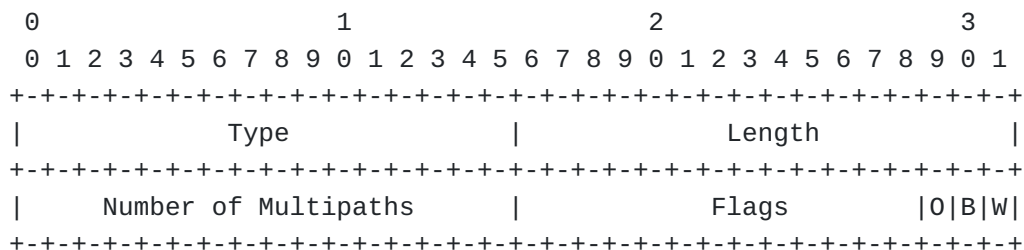


Figure 1: MULTIPATH-CAP TLV format

Type: TBD1 for "MULTIPATH-CAP" TLV.

Length: 4.

Number of Multipaths: the maximum number of multipaths per PCEP Tunnel. The value 0 indicates unlimited number.

W-flag: whether MULTIPATH-WEIGHT-TLV is supported.

B-flag: whether MULTIPATH-BACKUP-TLV is supported.

0-flag: whether MULTIPATH-OPPPDIR-PATH-TLV is supported.

4.2. Path Attributes Object

We define the PATH-ATTRIB object that is used to carry per-path information and to act as a separator between several ERO/RR0 objects in the <intended-path>/<actual-path> RBNF element. The PATH-ATTRIB object always precedes the ERO/RR0 that it applies to. If multiple ERO/RR0 objects are present, then each ERO/RR0 object MUST be preceded by an PATH-ATTRIB object that describes it.

The PATH-ATTRIB Object-Class value is TBD2.

The PATH-ATTRIB Object-Type value is 1.

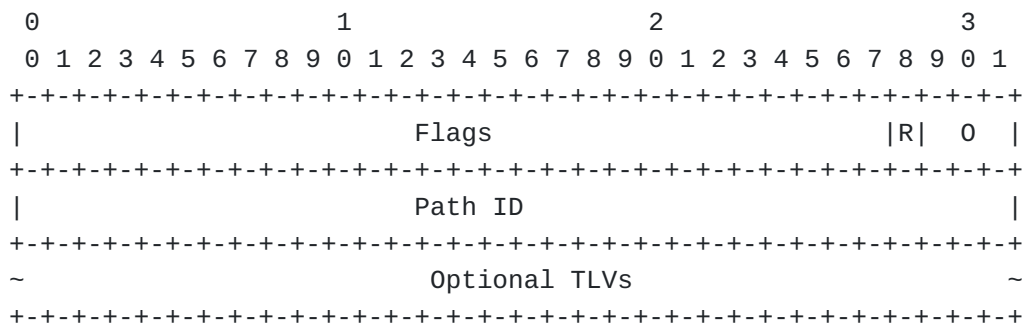


Figure 2: PATH-ATTRIB object format

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

R (Reverse): Indicates this path is reverse, i.e., it originates on the Tunnel destination and terminates on the Tunnel source (usually the PCC headend itself). Paths with this flag set MUST NOT be installed into forwarding, they serve only informational purposes.

Path ID: 4-octet identifier that identifies a path (encoded in the ERO/RR0) within the set of multiple paths under the PCEP LSP. See [Section 5.2](#) for details.

TLVs that may be included in the PATH-ATTRIB object are described in the following sections. Other optional TLVs could be defined by future documents to be included within the PATH-ATTRIB object body.

4.3. Multipath Weight TLV

We define the MULTIPATH-WEIGHT TLV that MAY be present in the PATH-ATTRIB object.

Figure 4: MULTIPATH-BACKUP TLV format

Type: TBD4 for "MULTIPATH-BACKUP" TLV

Length: $4 + (N * 4)$ (where N is the Backup Path Count)

Backup Path Count: Number of backup path(s).

B: If set, indicates a pure backup path. This is a path that only carries rerouted traffic after the protected path fails. 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.

Backup Path ID(s): a series of 4-octet identifier(s) that identify the backup path(s) in the set that protect this primary path.

4.5. Multipath Opposite Direction Path TLV

This document introduces a new MULTIPATH-OPPDIR-PATH TLV that MAY be present in the PATH-ATTRIB object. This TLV encodes a many-to-many mapping between forward and reverse paths within a PCEP Tunnel.

Many-to-many mapping means that a single forward path MAY map to multiple reverse paths and conversely that a single reverse path MAY map to multiple forward paths. Many-to-many mapping can happen for an SR Policy, when a Segment List contains Node Segment(s) which traverse parallel links at the midpoint. The reverse of this Segment List may not be able to be expressed as a single Reverse Segment List, but need to return multiple Reverse Segment Lists to cover all the parallel links at the midpoint.

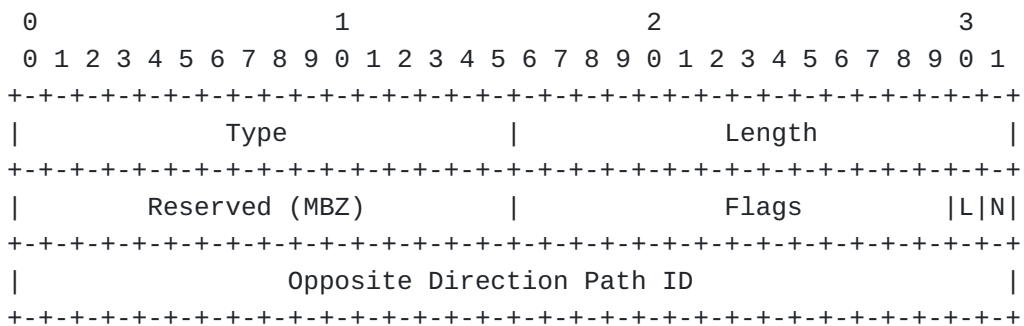


Figure 5: MULTIPATH-OPPDIR-PATH TLV format

Type: TBD9 for "MULTIPATH-OPPDIR-PATH" TLV

Length: 16.

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.

L (Link co-routed): If set, indicates this path is link co-routed with its opposite directions path, specified in this TLV. Two opposite direction paths are link co-routed if they traverse the same links (but in the opposite directions).

Opposite Direction Path ID: Identifies a path that goes in the opposite direction to this path. If no such path exists, then this field MUST be set to 0x0, which is reserved to indicate the absence of a Path ID.

Multiple instances of this TLV present in the same PATH-ATTRIB object indicate that there are multiple opposite-direction paths corresponding to the given path. This allows for many-to-many relationship among the paths of two opposite direction Tunnels.

Whenever path A references another path B as being the opposite-direction path, then path B typically also reference path A as its own opposite-direction path.

See [Section 7.4](#) for an example of usage.

4.6. Composite Candidate Path

SR Policy Architecture [[I-D.ietf-spring-segment-routing-policy](#)] defines the concept of a Composite Candidate Path. Unlike a Non-Composite Candidate Path, which contains Segment Lists, the Composite Candidate Path contains Colors of other policies. The traffic that is steered into a Composite Candidate Path is split among the policies that are identified by the Colors contained in the Composite Candidate Path. The split can be either ECMP or UCMP by adjusting the weight of each color in the Composite Candidate Path, in the same manner as the weight of each Segment List in the Non-Composite Candidate Path is adjusted.

To signal the Composite Candidate Path, we make use of the COLOR TLV, defined in [[I-D.draft-rajagopalan-pce-pcep-color](#)]. For a Composite Candidate Path, the COLOR TLV is included in the PATH-ATTRIB Object, thus allowing each Composite Candidate Path to do ECMP/UCMP among SR Policies or Tunnels identified by its constituent Colors. Only one COLOR TLV SHOULD be included into the PATH-ATTRIB object. If multiple COLOR TLVs are contained in the PATH-ATTRIB object, only the first one MUST be processed and the others SHOULD be ignored.

An empty ERO object MUST be included as per the existing RBNF, i.e., ERO MUST contain no sub-objects. If the head-end receives a non-empty ERO, then it MUST send PCError message with Error-Type 19 ("Invalid Operation") and Error-Value = TBD8 ("Non-empty path").

See [Section 7.3](#) for an example of the encoding.

5. Operation

5.1. Capability Negotiation

When the PCC wants to indicate to the PCE that it wants to get multipaths for a PCEP Tunnel, instead of a single path, it can do either (1) or both (1) and (2) of the following:

(1) Send the MULTIPATH-CAP TLV in the OPEN object during session establishment. This applies to all PCEP Tunnels on the PCC, unless overridden by PCEP Tunnel specific information.

(2) Additionally send the MULTIPATH-CAP TLV in the LSP object for a particular PCEP Tunnel in the PCRpt or PCReq message. This applies to the specified PCEP Tunnel and overrides the information from the OPEN object.

When PCE computes the path for a PCEP Tunnel, it MUST NOT return more multipaths than the corresponding value of "Number of Multipaths" from the MULTIPATH-CAP TLV. If this TLV is absent (from both OPEN and LSP objects), then the "Number of Multipaths" is assumed to be 1.

If the PCE supports this standard, then it MUST include the MULTIPATH-CAP TLV in the OPEN object. This tells the PCC that it can report multiple ERO/RR0 objects per PCEP Tunnel to this PCE. If the PCE does not include the MULTIPATH-CAP TLV in the OPEN object, then the PCC MUST assume that the PCE does not support this standard and fall back to reporting only a single ERO/RR0.

5.2. Path ID

The Path ID uniquely identifies a Path within the context of a PCEP Tunnel. Note that when the PCEP Tunnel is an SR Policy Candidate Path, the Paths within that tunnel are the Segment Lists of that Candidate Path.

Value 0x0 is reserved to indicate the absence of a Path ID. The value of 0x0 MAY be used when this Path is not being referenced and the allocation of a Path ID is not necessary.

Path IDs are allocated by the PCEP peer that currently owns the Tunnel. If the Tunnel is delegated to the PCE, then the PCE

allocates the Path IDs and sends them in the PCReply/PCUpd/PCInit messages. If the Tunnel 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 Path ID, then the PCEP speaker SHOULD send PCError message with Error-Type = 1 ("Reception of an invalid object") and Error-Value = TBD5 ("Conflicting Path ID").

5.3. Signaling Multiple Paths for Loadbalancing

The PATH-ATTRIB object can be used to signal multiple path(s) and indicate (un)equal loadbalancing amongst the set of multipaths. 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 or PCEP Tunnel.
2. The MULTIPATH-WEIGHT TLV MAY be carried inside the PATH-ATTRIB object. A weight is populated to reflect the relative loadshare that is to be carried by the path. If the MULTIPATH-WEIGHT is not carried inside a PATH-ATTRIB object, the default weight 1 MUST be assumed when computing the loadshare.
3. The fraction of flows carried by a specific primary path is derived from the ratio of its weight to the sum of all other multipath weights.

5.4. Signaling Multiple Paths for Protection

The PATH-ATTRIB object can be used to describe a set of backup path(s) protecting a primary path within a PCEP Tunnel. 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 or PCEP Tunnel.
2. The MULTIPATH-BACKUP TLV MAY be added inside the PATH-ATTRIB object for each ERO that is protected. The backup path ID(s) are populated in the MULTIPATH-BACKUP TLV to reflect the set of backup path(s) protecting the primary path. The Length field and Backup Path Number in the MULTIPATH-BACKUP are updated according to the number of backup path ID(s) included.
3. The MULTIPATH-BACKUP TLV MAY be added 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 path(s) fail- then the B flag MUST be set.

Note that 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 is not able to implement a requested backup scenario, the PCC SHOULD send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD7 ("Not supported path backup").

6. PCEP Message Extensions

The RBNF of PCReq, PCRep, PCRpt, PCUpd and PCInit messages currently use a combination of <intended-path> and/or <actual-path>. As specified in Section 6.1 of [\[RFC8231\]](#), <intended-path> is represented by the ERO object and <actual-path> is represented by the RRO object:

<intended-path> ::= <ERO>

<actual-path> ::= <RRO>

In this standard, we extend these two elements to allow multiple ERO/RRO objects to be present in the <intended-path>/<actual-path>:

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

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

7. Examples

7.1. SR Policy Candidate-Path with Multiple Segment-Lists

Consider the following sample SR Policy, taken from [\[I-D.ietf-spring-segment-routing-policy\]](#).

```

SR policy POL1 <headend, color, endpoint>
  Candidate-path CP1 <protocol-origin = 20, originator =
    100:1.1.1.1, discriminator = 1>
    Preference 200
    Weight W1, SID-List1 <SID11...SID1i>
    Weight W2, SID-List2 <SID21...SID2j>
  Candidate-path CP2 <protocol-origin = 20, originator =
    100:2.2.2.2, discriminator = 2>
    Preference 100
    Weight W3, SID-List3 <SID31...SID3i>
    Weight W4, SID-List4 <SID41...SID4j>

```

As specified in [[I-D.ietf-pce-segment-routing-policy-cp](#)], CP1 and CP2 are signaled as separate state-report elements and each has a unique PLSP-ID, assigned by the PCC. Let us assign PLSP-ID 100 to CP1 and PLSP-ID 200 to CP2.

The state-report for CP1 can be encoded as:

```

<state-report> =
  <LSP PLSP_ID=100>
  <ASSOCIATION>
  <END-POINT>
  <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>
  <END-POINT>
  <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>>

```

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

7.2. Two Primary Paths Protected by One Backup Path

Suppose there are 3 paths: A, B, C. Where A,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>  
  <END-POINT>  
  <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]>>
```

, simply 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.

7.3. Composite Candidate Path

Consider the following Composite Candidate Path, taken from [\[I-D.ietf-spring-segment-routing-policy\]](#).

```
SR policy POL100 <headend = H1, color = 100, endpoint = E1>  
  Candidate-path CP1 <protocol-origin = 20, originator =  
    100:1.1.1.1, 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>
  <END-POINT>
  <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)>

```

7.4. Opposite Direction Tunnels

Consider the two opposite-direction SR Policies between end-points 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 PathID=1 R-flag=0
    <OPPDIR-PATH-TLV OppositePathID=3>>
  <ERO <H1,M1,M2,E1>>
  <PATH-ATTRIB PathID=2 R-flag=0
    <OPPDIR-PATH-TLV OppositePathID=4>>
  <ERO <H1,M3,M4,E1>>
  <PATH-ATTRIB PathID=3 R-flag=1
    <OPPDIR-PATH-TLV OppositePathID=1>>
  <ERO <E1,M2,M1,H1>>
  <PATH-ATTRIB PathID=4 R-flag=1
    <OPPDIR-PATH-TLV OppositePathID=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 PathID=1 R-flag=0
    <OPPDIR-PATH-TLV OppositePathID=3>>
  <ERO <H1,M5,N6,E1>>
  <PATH-ATTRIB PathID=2 R-flag=0
    <OPPDIR-PATH-TLV OppositePathID=0>>
  <ERO <H1,M7,M8,E1>>
  <PATH-ATTRIB PathID=3 R-flag=1
    <OPPDIR-PATH-TLV OppositePathID=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 PathID=1 R-flag=0
    <OPPDIR-PATH-TLV OppositePathID=3>>
  <ERO <E1,M2,M1,H1>>
  <PATH-ATTRIB PathID=2 R-flag=0
    <OPPDIR-PATH-TLV OppositePathID=4>>
  <ERO <E1,M4,M3,H1>>
  <PATH-ATTRIB PathID=3 R-flag=1
    <OPPDIR-PATH-TLV OppositePathID=1>>
  <ERO <H1,M1,M2,E1>>
  <PATH-ATTRIB PathID=4 R-flag=1
    <OPPDIR-PATH-TLV OppositePathID=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 PathID=1 R-flag=0  
    <OPPDIR-PATH-TLV OppositePathID=3>>  
  <ERO <E1,M6,M5,H1>>  
  <PATH-ATTRIB PathID=2 R-flag=1  
    <OPPDIR-PATH-TLV OppositePathID=0>>  
  <ERO <H1,M7,M8,E1>>  
  <PATH-ATTRIB PathID=3 R-flag=1  
    <OPPDIR-PATH-TLV OppositePathID=1>>  
  <ERO <H1,M5,N6,E1>>
```

8. 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".

8.1. 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

8.2. Ciena Corp

Organization: Ciena Corp

Implementation: Head-end and controller

Maturity Level: Proof of concept

Coverage: Full

Contact: byadav@ciena.com

8.3. Huawei Technologies

Organization: Huawei Technologies Co.,Ltd.

Implementation: Huawei's Router and Controller

Maturity Level: Proof of concept

Coverage: Partial

Contact: tanren@huawei.com

9. IANA Considerations

9.1. PCEP Object

IANA is requested to make the assignment of a new value for the existing "PCEP Objects" registry as follows:

Object-Class	Name	Object-Type	Reference
Value		Value	
45	PATH-ATTRIB	1	This document

9.2. PCEP TLV

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

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

9.3. PCEP-Error Object

IANA is requested to make the assignment of a new value for the existing "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry for the following errors:

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

9.4. Flags in the Multipath Capability TLV

IANA is requested to create a new sub-registry to manage the Flag field of the MULTIPATH-CAP TLV, called "Flags in MULTIPATH-CAP TLV". New values are to be assigned by Standards Action [[RFC8126](#)]

Bit	Description	Reference
0-12	Unassigned	This document
13	0-flag: support for processing MULTIPATH-OPPDIR-PATH TLV	This document
14	B-flag: support for processing MULTIPATH-BACKUP TLV	This document
15	W-flag: support for processing MULTIPATH-WEIGHT TLV	This document

9.5. Flags in the Path Attribute Object

IANA is requested to create a new sub-registry to manage the Flag field of the PATH-ATTRIBUTE object, called "Flags in PATH-ATTRIBUTE Object". New values are to be assigned by Standards Action [[RFC8126](#)]

Bit	Description	Reference
0-12	Unassigned	This document
13-15	0-flag: Operational state	This document

9.6. Flags in the Multipath Backup TLV

IANA is requested to create a new sub-registry to manage the Flag field of the MULTIPATH-BACKUP TLV, called "Flags in MULTIPATH-BACKUP TLV". New values are to be assigned by Standards Action [[RFC8126](#)]

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

9.7. Flags in the Multipath Opposite Direction Path TLV

IANA is requested to create a new sub-registry to manage the flag fields of the MULTIPATH-OPPDIR-PATH TLV, called "Flags in the MULTIPATH-OPPDIR-PATH TLV". New values are to be assigned by Standards Action [[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

10. Security Considerations

None at this time.

11. Acknowledgement

Thanks to Dhruv Dhody for ideas and discussion.

12. Contributors

Andrew Stone
Nokia

Email: andrew.stone@nokia.com

13. References

13.1. Normative References

[I-D.draft-rajagopalan-pce-pcep-color]

Rajagopalan, B., Beeram, V. P., Peng, S., Xiong, Q., Koldychev, M., and G. S. Mishra, "Path Computation Element Protocol(PCEP) Extension for Color", Work in Progress, Internet-Draft, draft-rajagopalan-pce-pcep-color-02, 6 July 2022, <<https://www.ietf.org/archive/id/draft-rajagopalan-pce-pcep-color-02.txt>>.

[I-D.ietf-pce-segment-routing-policy-cp]

Koldychev, M., Sivabalan, S., Barth, C., Peng, S., and H. Bidgoli, "PCEP extension to support Segment Routing Policy Candidate Paths", Work in Progress, Internet-Draft, draft-ietf-pce-segment-routing-policy-cp-08, 24 October 2022, <<https://www.ietf.org/archive/id/draft-ietf-pce-segment-routing-policy-cp-08.txt>>.

[I-D.ietf-spring-segment-routing-policy]

Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", Work in Progress, Internet-Draft, draft-ietf-spring-segment-routing-policy-22, 22 March 2022, <<https://www.ietf.org/archive/id/draft-ietf-spring-segment-routing-policy-22.txt>>.

[I-D.koldychev-pce-operational]

Koldychev, M., Sivabalan, S., Peng, S., Achaval, D., and H. Kotni, "PCEP Operational Clarification", Work in Progress, Internet-Draft, draft-koldychev-pce-operational-06, 4 July 2022, <<https://www.ietf.org/archive/id/draft-koldychev-pce-operational-06.txt>>.

[I-D.schmutzer-pce-cs-sr-policy]

Schmutzer, C., Filsfils, C., Ali, Z., Clad, F., Maheshwari, P., Rokui, R., Stone, A., Jalil, L., Peng, S., Saad, T., and D. Voyer, "Circuit Style Segment Routing Policies", Work in Progress, Internet-Draft, draft-schmutzer-pce-cs-sr-policy-02, 5 May 2022, <<https://www.ietf.org/archive/id/draft-schmutzer-pce-cs-sr-policy-02.txt>>.

[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>>.

[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>>.

[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>>.

[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/info/rfc8664>>.

13.2. Informative References

[RFC4655]

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

[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>>.

[RFC8745]

Ananthakrishnan, H., Sivabalan, S., Barth, C., Minei, I., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extensions for Associating Working and Protection Label Switched Paths (LSPs) with Stateful PCE", RFC 8745, DOI 10.17487/RFC8745, March 2020, <<https://www.rfc-editor.org/info/rfc8745>>.

Authors' Addresses

Mike Koldychev

Cisco Systems, Inc.

Email: mkoldych@cisco.com

Siva Sivabalan
Ciena Corporation

Email: ssivabal@ciena.com

Tarek Saad
Juniper Networks, Inc.

Email: tsaad@juniper.net

Vishnu Pavan Beeram
Juniper Networks, Inc.

Email: vbeeram@juniper.net

Hooman Bidgoli
Nokia

Email: hooman.bidgoli@nokia.com

Bhupendra Yadav
Ciena

Email: byadav@ciena.com

Shuping Peng
Huawei Technologies

Email: pengshuping@huawei.com

Gyan Mishra
Verizon Inc.

Email: gyan.s.mishra@verizon.com