IETF 112 – Online
PCE Working Group

PCEP Extensions for Signaling Multipath Information draft-ietf-pce-multipath

M. Koldychev – Cisco Systems (mkoldych@cisco.com) – Presenter
M. Sivabalan – Ciena Corporation (ssivabal@ciena.com)
T. Saad – Juniper Networks (tsaad@juniper.net)
V. Beeram – Juniper Networks (vbeeram@juniper.net)
H. Bidgoli – Nokia (Hooman.Bidgoli@Nokia.com)
S. Peng – Huawei Technologies (pengshuping@Huawei.com)
B. Yadav – Ciena (byadav@ciena.com)

Review

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.

Mechanisms in this draft are as generic as possible:

- Independent of data-plane/setup-type (RSVP-TE/SR-MPLS/SRv6)
- Independent of stateful/stateless PCEP, i.e., it works with PCReq/PCRply

Forward and Reverse paths

A path computation algorithm that is given a computational problem can output:

- **0 or 1** forward paths (without this draft),
- **N forward paths** (with this draft),
- **N forward paths, plus M reverse paths** (with latest update).

Returning only **0 or 1** forward paths was sufficient for RSVP-TE tunnels.

Returning **N forward** paths is required for SR Policy with multiple Segment Lists (SL) under one Candidate Path.

Returning **N forward paths, plus M reverse paths** is useful for Circuit-Style SR Policy [draft-schmutzer-pce-cs-sr-policy]. It allows the head-end to learn about reverse SL(s) for each forward SL.

OPPDIR-PATH TLV

New TLV in the PATH-ATTRIB object:

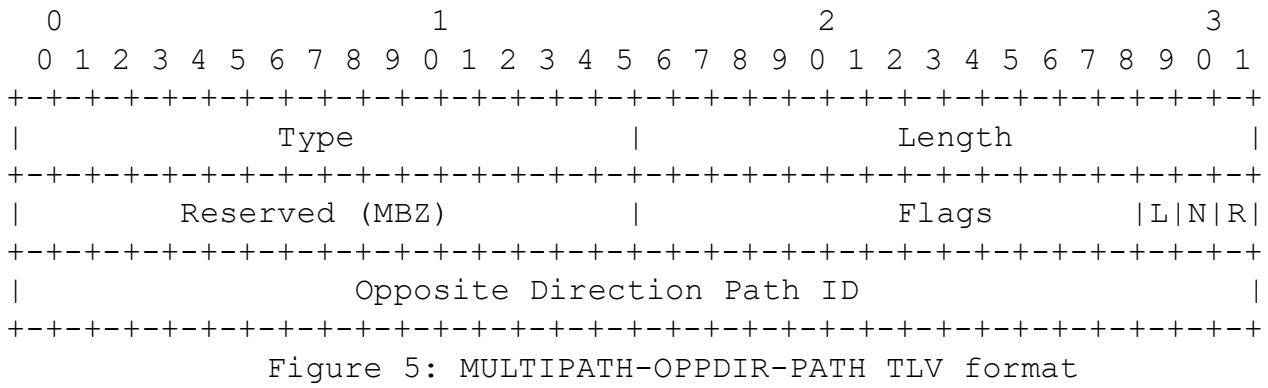


Figure 5: MULTIPATH-OPPDIR-PATH TLV format

Opposite Direction Path ID points to another PATH-ATTRIB object within the same Tunnel. It signifies that the pointed-to path is the opposite of the current path.

R-flag is set to 1 when the current path (described by the current PATH-ATTRIB object) is going in the reverse direction w.r.t the Tunnel, i.e., from the end-point to the head-end.

Each path can have multiple opposite paths.

Oppositeness property is NOT mutual (eg., Path 2 is an opposite of Path1, but Path 1 is NOT an opposite of Path 2).

Example: Circuit Style SR Policies

Router H1:

```
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>
```

```
<state-report> =
   <LSP PLSP_ID=100>
   <BIDIRECTIONAL ASSOCIATION = A1>
   <PATH-ATTRIB PathID=1
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=3>>
   <ERO <H1,M1,M2,E1>>
   <PATH-ATTRIB PathID=2
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=4>>
   <ERO <H1,M3,M4,E1>>
   <PATH-ATTRIB PathID=3
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=1>>
   <ERO <E1,M2,M1,H1>>
   <PATH-ATTRIB PathID=4
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=2>>
   <ERO <E1,M4,M3,H1>>

<state-report> =
   <LSP PLSP_ID=200>
   <BIDIRECTIONAL ASSOCIATION = A2>
   <PATH-ATTRIB PathID=1
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=3>>
   <ERO <H1,M5,N6,E1>>
   <PATH-ATTRIB PathID=2
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=0>>
   <ERO <H1,M7,M8,E1>>
   <PATH-ATTRIB PathID=3
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=1>>
   <ERO <E1,M6,M5,H1>>
```

Router 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>
```

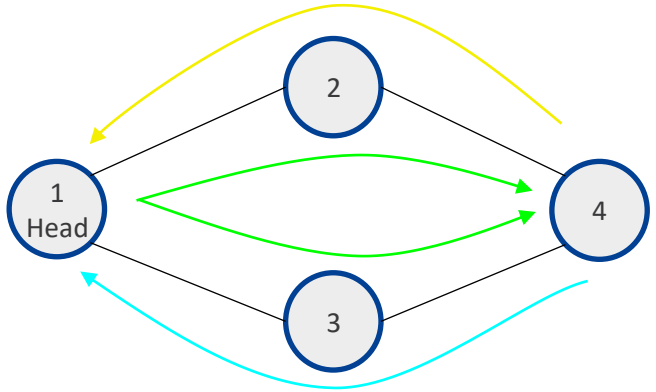
```
<state-report> =
   <LSP PLSP_ID=100>
   <BIDIRECTIONAL ASSOCIATION = A1>
   <PATH-ATTRIB PathID=1
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=3>>
   <ERO <E1,M2,M1,H1>>
   <PATH-ATTRIB PathID=2
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=4>>
   <ERO <E1,M4,M3,H1>>
   <PATH-ATTRIB PathID=3
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=1>>
   <ERO <H1,M1,M2,E1>>
   <PATH-ATTRIB PathID=4
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=2>>
   <ERO <H1,M3,M4,E1>>

<state-report> =
   <LSP PLSP_ID=200>
   <BIDIRECTIONAL ASSOCIATION = A2>
   <PATH-ATTRIB PathID=1
     <OPPDIR-PATH-TLV R-flag=0 OppositePathID=3>>
   <ERO <E1,M6,M5,H1>>
   <PATH-ATTRIB PathID=2
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=0>>
   <ERO <H1,M7,M8,E1>>
   <PATH-ATTRIB PathID=3
     <OPPDIR-PATH-TLV R-flag=1 OppositePathID=1>>
   <ERO <H1,M5,N6,E1>>
```

Data model

PCEP signaling

Example: forward path with 2 reverse paths



Node(X) -> node segment of router X
Adj(X,Y) -> adjacency segment from router X to router Y
Where X, Y ∈ {1,2,3,4}

In Segment Routing, node segments can send traffic along multiple links. Thus, multiple segment lists may be required to express all the reverse paths.

Multiple instances of the OPPDIR-PATH-TLV encode multiple reverse paths. For example, the following encodes that Segment List <Node(4)> has two reverse Segment Lists: <Adj(42),Adj(21)> and <Adj(43),Adj(31)>.

```
<PATH-ATTRIB PathID=1
  <OPPDIR-PATH-TLV R-flag=0 OppositePathID=2>
  <OPPDIR-PATH-TLV R-flag=0 OppositePathID=3>>
<ERO <Node(4)>>
<PATH-ATTRIB PathID=2
  <OPPDIR-PATH-TLV R-flag=1 OppositePathID=0>>
<ERO <Adj(4,2),Adj(2,1)>>
<PATH-ATTRIB PathID=3
  <OPPDIR-PATH-TLV R-flag=1 OppositePathID=0>>
<ERO <Adj(4,3),Adj(3,1)>>
```

Note that just because Path 2 is a reverse of Path 1, does NOT mean that Path 1 is a reverse of Path 2.

Value of R-flag among all instances of the OPPDIR-PATH-TLV MUST be the same. We could also put this R-flag into the PATH-ATTRIB object, instead of the OPPDIR-PATH-TLV?

Next steps

- Get feedback from WG
- Request IANA code point allocation