

Supporting explicit-path per destination in PCEP - P2MP Path Request.

draft-dhody-pce-pcep-p2mp-per-destination-00

Dhruv Dhody (dhruvd@huawei.com)

Udayasree Palle (udayasreepalle@huawei.com)

Note: *Explicit Path here refers to the configured list of network elements that MUST be traversed or MUST be excluded in the final path computation.*

Introduction

- PCEP is extended to support P2MP path computations in intra- & inter-domain scenarios.
- Path Request message is defined as follows:

```
<PCReq Message> ::= <Common Header>  
                    <request>
```

where:

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

<IRO> is attached to all leaves and thus to only the common path.

where:

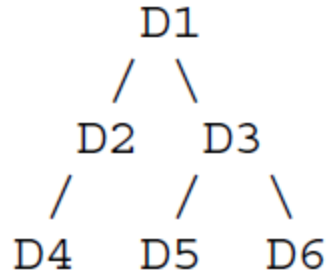
```
<end-point-rro-pair-list> ::=  
    <END-POINTS> [<RRO-List>] [<BANDWIDTH>]  
    [<end-point-rro-pair-list>]  
  
<RRO-List> ::= <RRO> [<BANDWIDTH>] [<RRO-List>]  
<metric-list> ::= <METRIC> [<metric-list>]
```

- Extend PCEP to define explicit-path per destination in P2MP context.

Motivation for this work

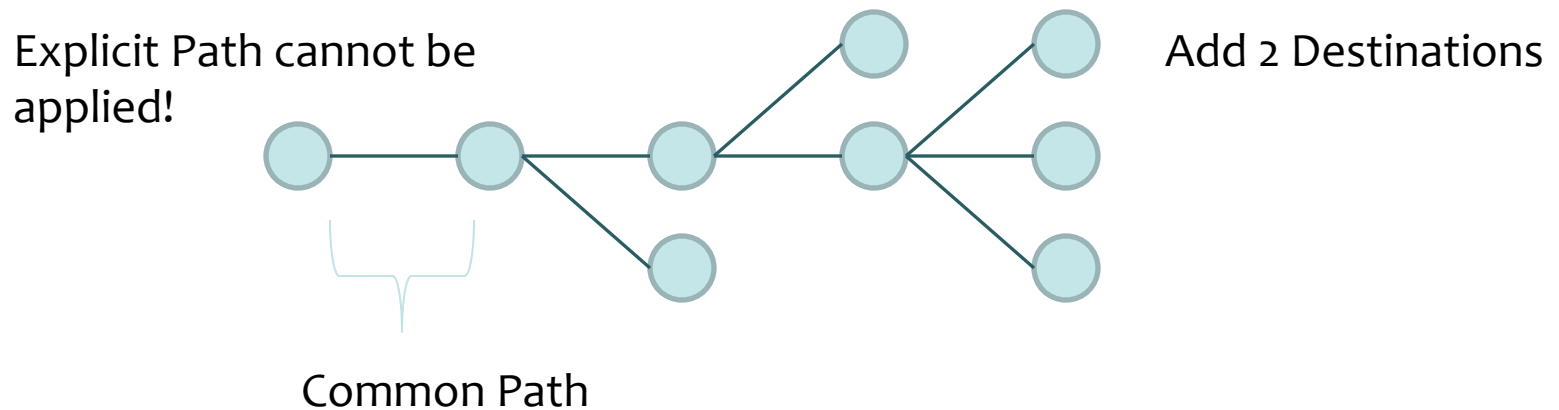
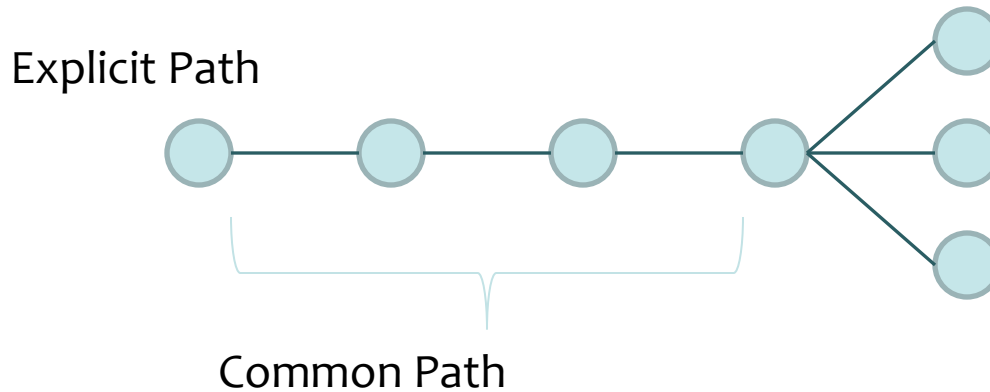
- In INTER-DOMAIN case different destinations will have different domain paths within the domain tree, it requires domain-sequence encoded in form of <IRO> to be attached per destination. It cannot be encoded for all destinations.

D1-D3-D6, D1-D3-D5 and D1-D2-D4.



- Administrator at the source can exert stronger control by providing explicit path (include, exclude, loose etc) per destination.
- Compatibility: Basic MPLS TE P2MP Tunnel configurations for various operators support the configuration of explicit-path per destination.

Example



- Supporting configuration of <IRO> / <XRO> only for the common path is a disadvantage for the administrator in specifying network element and computation of final P2MP tree.

Path Request Message Format

```
<PCReq Message> ::= <Common Header>  
                    <request>
```

where:

```
<request> ::= <RP>  
             <end-point-iro-xro-rro-metric-list>  
             [<OF>]  
             [<LSPA>]  
             [<BANDWIDTH>]  
             [<metric-list>]  
             [<IRO>]  
             [<LOAD-BALANCING>]
```

Attach <IRO-List> / <XRO-List> to END-POINTS

where:

```
<end-point-iro-xro-rro-metric-list> ::=  
    <END-POINTS>  
    [<IRO-List>]  
    [<XRO-List>]  
    [<RRO-List>]  
    [<metric-list>]  
    [<end-point-iro-xro-rro-metric-list >]
```

<IRO> / <XRO>
matches to
destination in
END_POINTS in
ordered
sequence

```
<RRO-List> ::= <RRO> [<BANDWIDTH>] [<RRO-List>]  
<metric-list> ::= <METRIC> [<metric-list>]  
<IRO-List> ::= <IRO> [<IRO-List>]  
<XRO-List> ::= <XRO> [<XRO-List>]
```

Ordering Destinations in END-POINTS

- The rule of the thumb is first <IRO> in <IRO-list> belongs to first destination in END-POINTS, the first <XRO> in <XRO-List> belongs to first destinations in END-POINTS etc..
- Thus the order of the destinations in the END-POINTS is important.
- Sometimes there may-be a need to split the destinations into multiple END-POINTS while encoding path-request message.

Next Steps

- Evaluate the need within the WG and extend PCEP accordingly.
- IANA and security considerations needs analysis.

Questions
&
Comments?

Thanks!