IETF 111 Path Computation Element (PCE) WG

Thursday, July 29, 2021 (19:00-20:00 UTC)

Chairs
Julien Meuric (julien.meuric@orange.com)
Dhruv Dhody (dd@dhruvdhody.com)
Secretary
Hariharan Ananthakrishnan (hari@netflix.com)

This session is being recorded

IETF 111
hosted by

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

- BCP 9 (Internet Standards Process)
- BCP 25 (Working Group processes)
- BCP 25 (Anti-Harassment Procedures)
- BCP 54 (Code of Conduct)
- BCP 78 (Copyright)
- BCP 79 (Patents, Participation)
- https://www.ietf.org/privacy-policy/(Privacy Policy)

I E T F®

This session is being recorded

IETF 111 Online Meeting Tips

- Make sure your video is off unless you are presenting during a session
- Mute your microphone unless you are speaking
- Use of a headset is strongly recommended
- Session bluesheet is automatically generated based on IETF Datatracker logins
- Chatrooms in Meetecho are connected to the Jabber chatrooms on IETF Datatracker agenda
- More information and assistance:
  - Participant guide https://www.ietf.org/how/meetings/technology/meetecho-guide-participant/
  - Request assistance and report issues via: http://www.ietf.org/how/meetings/issues/

Administrivia

- Minute taker(s), jabber scribe(s)
- Meetecho Etiquette
  - Join the queue if you would like to speak/present
    - Do not send audio directly
  - Please state your name before speaking
  - Be mindful of the agenda time
    - Longer discussion on mailing list (or jabber)
- Collaborative minutes
  - https://codimd.ietf.org/notes-ietf-111-pce?both

Agenda

Monday, July 26, 2021 14:30-15:30 PST

Introduction
1.1 Administrivia, Agenda Bashing (chairs, 5 min)
1.2 WG Status (chairs, 10 min) [15/60]
1.3 State of WG I-Ds and next steps (chairs, 10 min) [25/60]

Stateful
2.1 Native IP (Aijun, 10 min) [35/60]
draft-ietf-pce-pcep-extension-native-ip-14
2.2 IFIT (Giuseppe, 5 min) [40/60]
draft-chen-pce-pcep-ifit-04
2.3 New TE Constraints (Quan, 5 min) [45/60]
draft-peng-pce-te-constraints-06

New I-Ds
3.1 RSVP Color (Balaji, 5 min) [50/60]
draft-rajagopalan-pcep-rsvp-color-01
~~3.2 VLAN-based Native IP (Yue Wang, 5 min) [55/60]~~
~~draft-wang-pce-vlan-based-traffic-forwarding-00~~

Thursday, July 29, 2021 12:00-13:00 PST (19:00-20:00 UTC)

Segment Routing (SR)
4.1 Algorithm in SID (Samuel, 10 min)
draft-tokar-pce-sid-algo-04
4.2 Entropy Label Position (Quan, 10 min) [20/60]
draft-peng-pce-entropy-label-position-06

Multicast
5.1 SR P2MP Policy (Hooman, 10 min) [30/60]
draft-hsd-pce-sr-p2mp-policy-03
5.2 BIER-TE (Ran, 5 min) [35/60]
draft-chen-pce-bier-09
5.3 PCE based BIER (Huanan, 10 min) [45/60]
draft-li-pce-based-bier-01
5.4 BIER-TE Ingress Protection (Huaimo, 10 min) [55/60]
draft-chen-pce-bier-te-ingress-protect-00

From Session 1

6.1 VLAN-based Native IP (Yue Wang, 5 min) [60/60]
draft-wang-pce-vlan-based-traffic-forwarding-00

Thanks!

IETF 111 – Online
PCE Working Group

Carrying SID Algorithm information in PCE-based Networks

A. Tokar – Cisco Systems (atokar@cisco.com)
S. Sidor – Cisco Systems (ssidor@cisco.com) – Presenter
M. Sivabalan – Ciena Corporation (ssivabal@ciena.com)
S. Peng – Huawei Technologies (pengshuping@huawei.com)
M. Negi – RtBrick Inc (mahend.ietf@gmail.com)

Motivation

- A PCE can compute SR-TE paths using SIDs with different Algorithms depending on the use-case, constraints, etc. While this information is available on the PCE, there is no method of conveying this information to the headend router

- The headend can also compute SR-TE paths using different Algorithms, and this information also needs to be conveyed to the PCE for collection or troubleshooting purposes

- An operator may also want to constrain the path computed by the PCE to a specific SID Algorithm. For example, in order to only use SID Algorithms for a low-latency path

Summary of updates since IETF 110

- Draft update 03 -> 04
- SID Algorithm in ERO
  - Support for SRv6 and Adjacency SID Algorithm
  - New capability negotiated in Open message
  - New flag and Algorithm field in SR-ERO and SRv6-ERO Sub-objects
  - Deprecated SID Algorithm specific NAI types
- SID Algorithm constraint in LSPA – no change

SR PCE Capability sub-TLV

- Exchanged in PATH-SETUP-TYPE-CAPABILITY in Open object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type=26               |             Length=4          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved              |   Flags |S|N|X|      MSD      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- S flag
  - Indicate support for SID Algorithm field in the SR-ERO sub-object

SRv6 PCE Capability sub-TLV

- Exchanged in PATH-SETUP-TYPE-CAPABILITY in Open object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type=TBD1          |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Reserved           |             Flags       |S|N|X|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSD-Type    | MSD-Value     |   MSD-Type    |   MSD-Value   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                             ...                             //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSD-Type    | MSD-Value     |            Padding            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- S flag
  - Indicate support for SID Algorithm field in the SRv6-ERO sub-object

SR-ERO Sub-object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|   Type=36   |     Length    |  NT   |     Flags |A|V|F|S|C|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         SID (optional)                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                   NAI (variable, optional)                  //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Reserved                   |  Algorithm    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

A flag

- If set, then SR-ERO sub-object is increased by 4 and Algorithm field is included

SRv6-ERO Sub-object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|   Type=TBD3 |     Length    |   NT  |     Flags     |A|V|T|F|S|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Reserved           |       Endpoint Behavior       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                      SRv6 SID (optional)                      |
|                           (128-bit)                           |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                    NAI (variable, optional)                 //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    SID Structure (optional)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Reserved                     |  Algorithm    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

A flag

- If set, then SRv6-ERO sub-object is increased by 4 and Algorithm field is included

NAI types deprecated

- Version 03 required duplicating NAI types
  - IPv4 Node ID -> IPv4 Node ID with Algorithm
  - IPv6 Node ID -> IPv6 Node ID with Algorithm
- Extension for SR-ERO and SRv6 ERO
  - Covered Adjacency SID Algorithm
    - draft-peng-lsr-algorithm-related-adjacency-sid
  - Future proof – prepared for extensions introduced later

Next steps

- Comments and discussion are welcome

PCEP Extension for

SR-MPLS Entropy Label Position

draft-peng-pce-entropy-label-position-06

Quan Xiong(ZTE)
Shaofu Peng(ZTE)
Fengwei Qin(China Mobile)

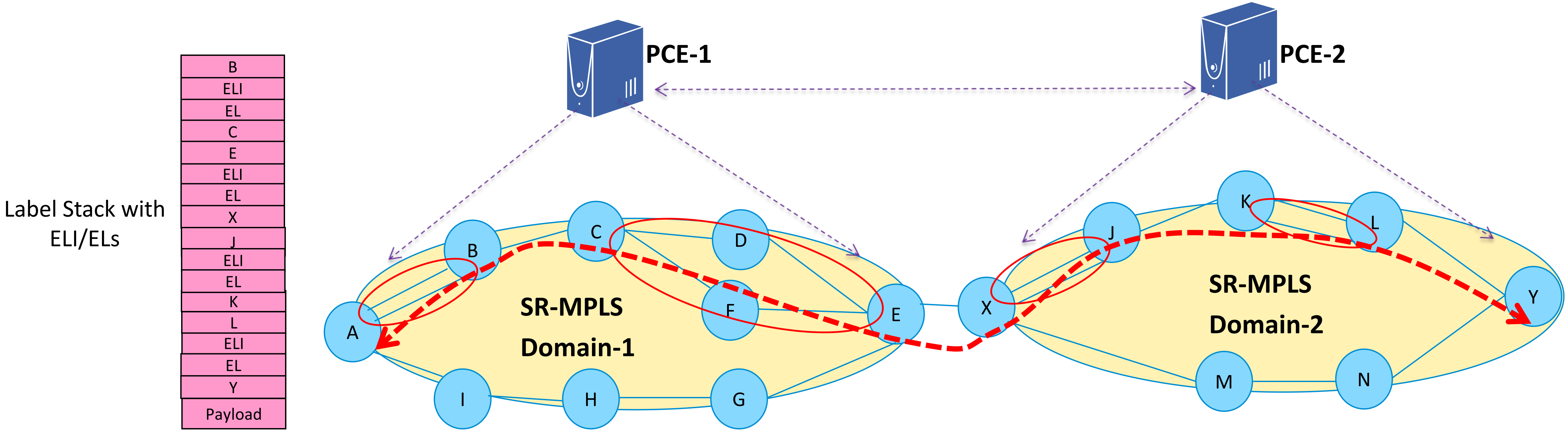
IETF 111 PCE, July 2021, Online

Update from last version

- Presented in IETF#106 and #108 and comments on the mailing list are appreciated from :
  - Stephane Litkowski / Dhruv Dhody / Tarek Saad / Zhenbin Li / Jeff Tantsura
- Change for LSP-EXTENDED-FLAG TLV
  - LSP-EXTENDED-FLAG TLV has been moved into draft-ietf-pce-lsp-extended-flags-00.
  - This document uses a flag from LSP extended flags field in LSP-EXTENDED-FLAG TLV.
- Clarification for the MSD and ERLD limilation and the requirements in PCE scenario
  - As described in RFC8662, the ingress node may not find the minimum ERLD along the path and does not support the computation of the minimum ERLD.
  - Especially in case of inter-domain scenario, PCE would be useful for computing the minimum ERLD and the position of entropy labels as well as SR paths.
- Clarification for ingress capability
  - As defined in RFC8662, multiple <ELI, EL> pairs MAY be inserted in the SR-MPLS label stack.
  - The ingress MAY be required to support the capability of inserting multiple ELI/ELs and it need to be advertised in OPEN message from PCC to PCE.
  - The E (ELP) bit is used to indicate the capability of inserting multiple ELI/EL pairs at PCC and support the SR path with ELP from PCE.
- Clarification for the ELI/ELs positions caculated for a SR-Path
  - The ELI/ELs positions is calculated at PCE for a SR-Path and the values is caculated at PCC for a specific traffic flow.

Overview

- RFC8662 proposes to apply the entropy labels to SR-MPLS networks and provides following criteria to determine the best ELI/ELs placement:
  - a limited number of <ELI, EL> pairs SHOULD be inserted in the SR-MPLS label stack;
  - the inserted positions SHOULD be whithin the Entropy Readable Label Depth (ERLD) of a maximize number of transit LSRs;
  - a minimum number of <ELI, EL> pairs SHOULD be inserted while satisfying the above criteria.
- As described in RFC8662, the ingress may not find the minimum ERLD along the path and does not support the computation of the minimum ERLD.
- The controller (e.g. PCE) MAY perform the end-to-end path computation as well as Entropy Label Position (ELP) including the number and the place of the ELI/ELs based on the minimum ERLD of each segment along the path especailly in inter-domain scenarios.



PCEP Extensions

- SR-PCE-CAPABILITY sub-TLV in Open Object
  - E bit is set to 1.
  - indicates that it supports the SR path computation with ELP configuration.
  - indicates that it supports the capability of inserting multiple ELI/EL pairs at PCC .

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type=TBD11            |            Length=4           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved              |  Flags  |E|N|X|      MSD      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: E-flag in SR-PCE-CAPABILITY sub-TLV

- LSP-EXTENDED-FLAG TLV in LSP Object
  - E bit is set to 1.
  - indicates that the PCC requests PCE to compute the SR path with ELP information.

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=TBD            |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                             |E|
//                      LSP Extended Flags                     //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: E-flag in LSP-EXTENDED-FLAG TLV

PCEP Extensions

- PATH-MINIMUM-ERLD TLV in LSP Object
  - Path Minimum ERLD: indicates the minimum ERLD value of the nodes along the path.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type              |             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Reserved                |Path Minimum ERLD|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: The PATH-MINIMUM-ERLD TLV

- SR-ERO Subobject
  - E bit is set to 1.
  - indicates that the position after this SR-ERO subobject is the position to insert <ELI, EL>, otherwise it cannot insert <ELI, EL> after this segment.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|   Type=36   |     Length    |  NT   |     Flags     |E|F|S|C|M|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         SID (optional)                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                   NAI (variable, optional)                  //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: E-flag in SR-ERO subobject

Next Step

- This document has been discussed many times in details at the meetings and on the mailing list.
- Thanks for all your comments and suggestions!
- Ready for adoption!

Thank you!

P2MP Policy
draft-hsd-pce-sr-p2mp-policy

Authors:

Hooman Bidgoli, Nokia
Daniel Voyer, Bell Canada
Ehsan Hemmati, Cisco
Saranya Rajarathinam, Nokia
Tarek Saad, Juniper
Siva Sivabalan, Ciena

IETF 111, Nov 2020 - Singapore
PCEP Working Group

Draft evolution as it stands now

- Discussions with Dhruv, Thank you for taking time! My apologies for not adding you to the contributor section will correct that.
  - Agreed to the CCI Object for replication segments. It is used to identify the entire cross connect of incoming segment and the set of outgoing Interfaces and their corresponding SIDs/SIDList.
  - Any modification to the cross connect should use this CCI ID to identify the cross connect uniquely

- Two separate messages for crafting a P2MP policy and Replication segment
  - Previously they could be created via the same PC-Init message
  - Draft has been updated to describe message use case workflows, incl. CCI usage

- Everything else as it was before
  - ERO is still used for forwarding instruction
  - Multipath-backup TLV is used for protection

New Replication Segment

```
<Common Header>
    <SRP>
    <P2MP LSP>
    (<cci-list>|
    (<CCI><intended-path>))
    <cci-list> ::=  <CCI>
                [<cci-list>]
   <intended-path> ::= ((<PATH-ATTRIB><ERO>)
                  [<intended-path>])
```

Next Steps

- Asking for adoption of this draft

Thank you!

PCEP extensions for BIER-TE

draft-chen-pce-bier-09

Ran Chen, Zheng Zhang(ZTE)

Huaimo Chen , Senthil Dhanaraj (futurewei)

Fengwei Qin(China Mobile)

Aijun Wang (China Telecom)

Virtual PCE WG IETF-111 Meeting, July 2021

Introduction

- This document specifies extensions to the Path Computation Element Protocol (PCEP) that allow a stateful PCE to compute and initiate the path for the BIER-TE.

Extensions

- BIER Capability Advertisement.
  - Defines a new Path Setup Type (PST) for BIER.
  - Defines the BIER-TE-PCE-CAPABILITY sub-TLV to exchange BIER capability.
- The SRP Object
  - Defines a new Path Setup Type (PST=TBD2) for BIER-TE.
- END-POINTS Object, two options:
  - Reuses the P2MP END-POINTS object body for IPv4 and END-POINTS object body for IPv6 (Object-Type 4) which is defined in [RFC8306].
- Objective Functions
  - Defines a new Objective Function for path calculation.
- ERO Object
  - Defines an BIER-TE-ERO subobjects to carry a adjacencies BitStrings, BSL,subdomain and SI.
- RRO Object
  - Defines an BIER-TE-RRO subobjects to reports an BIER-TE to PCE .

Update

- Merge draft-chen-pce-bier-te-path-01, and add Huaimo Chen and Aijun Wang as co-authors.
- Add the Objective Functions

```
Objective Function Code: TBD3
Name: Minimum Bit Sets (MBS)
Description: Find a path represented by BitPositions that has
             the minimum number of bit sets.
```

- Add the RRO Object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Type=TBD5  |       Length  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    BS Length  | subdomain-id  |       SI      |   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Adjacency BitString  (first 32 bits)               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~            Adjacency BitString  (last 32 bits)                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Next Step

- Comments welcome.
- It’s in WG adoption queue. WG adoption 😊

Thanks!

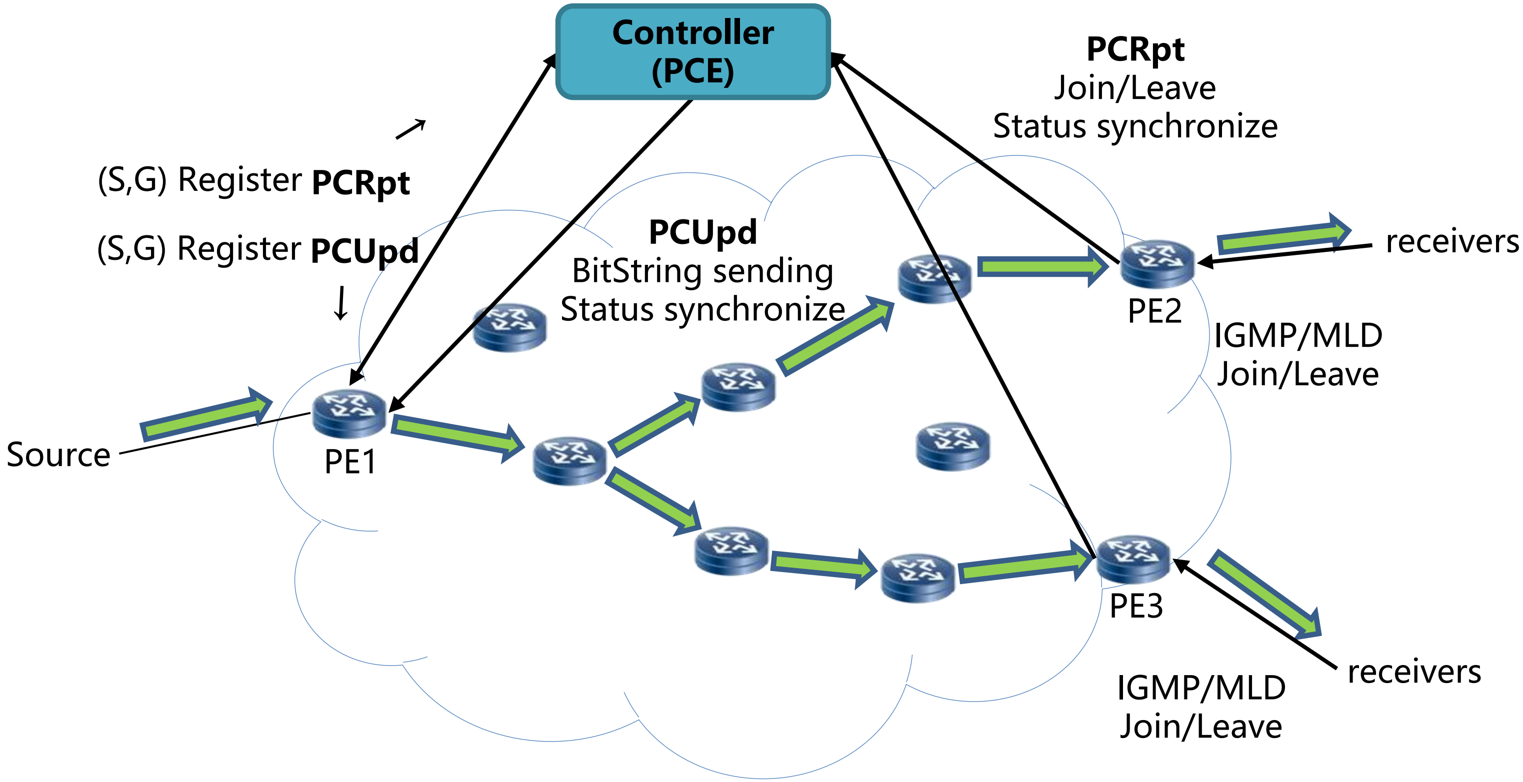
PCE based BIER Procedures and Protocol Extensions

[draft-li-pce-based-bier]

Huanan Li(China Telecom)
Aijun Wang (China Telecom)
Huaimo Chen(Futurewei)
Ran Chen(ZTE Corporation)
IETF 111, July. 2021

- Overview of PCE based BIER solution
- Extensions to PCE
- Further Action

Overview of PCE based BIER solution



Main flow for PCE based BIER multicast

1. PCE receives the registration information from ingress and responds.
2. PCE gets reports about egresses in PCRpt.
3. PCE generates BitString and sends it to ingress via PCUpd.
4. Ingress encapsulate BIER header and forward multicast packets.
5. The number of receivers is regularly synchronized between egress and PCE, and between PCE and ingress, using PCRpt and PCUpd respectively.

Extensions to PCE

Newly defined Objects

1. BIER-MULTICAST-CAPABILITY flag in STATEFUL-PCE-CAPABILITY TLV in the OPEN object
2. Multicast Source Registration Object
3. Multicast Receiver Information Object
4. Forwarding Indication Object
5. Multicast Receiver Status Object

Extensions to PCEP Message

1. Open Message
2. PCRpt Message
3. PCUpd Message

Multicast Source Registration Object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Subdomain-id  |            BFR-ID             |  BSL  |   |R|A|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              RD                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Address Type  |Auxliary Length|           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                    Multicast Source Address                   ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                    Multicast Group Address                    ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                        Auxliary Data                          ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Application scenarios:

  (S,G) registration/revocation in PCRpt and response in PCUpd

  - ✓ Flag “R” bit indicates whether the action of the message is a registration request.
  - ✓ Flag “A” bit indicates whether the request is successful.

Multicast Receiver Information Object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Subdomain-id  |            BFR-ID             |  BSL  |     |S|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              RD                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Address Type  | Label Length  |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                   Multicast Source Address                    ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                   Multicast Group Address                     ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                       Forwarding Label                        ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Application scenarios:

  Multicast joining or leaving in PCRpt

  - ✓ Flag “S” bit indicates whether to join a multicast group.
  - ✓ Forwarding Label is used for egress to distinguish receivers at the forwarding layer.

Forwarding Indication Object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Subdomain-id  |      SI       |  BSL  |      Reserved       |F|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               RD                              |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Address Type  | Label Length  |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                   Multicast Source Address                    ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                   Multicast Group Address                     ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                       Forwarding Label                        ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  BitString (first 32 bits)                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                               .                               |
|                               .                               |
|                               .                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  BitString  (last 32 bits)                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Application scenarios:

  Informing ingress to forward packets of a specific (S,G) to receivers in PCUpd.

  - ✓ Flag “F” bit indicates whether to start forwarding multicast data.

Multicast Receiver Status Object

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Subdomain-id  |      SI       |  BSL  | Address Type  |       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Number of Receivers                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                    Multicast Source Address                   ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                    Multicast Group Address                    ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- Application scenarios:

  Egress synchronizes receiver information to PCE in PCRpt

  PCE synchronizes receiver information to ingress in PCUpd

  - ✓ Flag “F” bit indicates whether to start forwarding multicast data.
  - ✓ In PCRpt messages, the number of receivers refers to connected to the egress in a specific (S,G) .
  - ✓ In PCUpd messages, the number of receivers refers to all receivers in a specific (S,G).

Extensions to PCEP Message

Open Message

- BIER-MULTICAST-CAPABILITY flag in STATEFUL-PCE-CAPABILITY TLV in the OPEN object is set to support BIER multicast.

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

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

   <state-report> ::= [<SRP>]
                      <LSP>
                      <path>
                      [<MSR>]
                      [<MRI>]
                      [<MRS>]
```

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

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

   <update-request> ::= <SRP>
                        <LSP>
                        <path>
                        [<MSR>]
                        [<FI>]
                        [<MRS>]
```

PCRpt Message includes

- MSR object for registration or revocation
- MRI object for receivers' joining or leaving(not the last one)
- MRS object for periodic synchronization of receivers' number
- MRI and MRS objects for receiver's leaving(the last one)

PCUpd Message includes

- MSR object for response to registration or revocation
- FI object for indicating data forwarding
- MRS object for periodic synchronization of receivers' number

Next Step

- Comments

lihn6@chinatelecom.cn
wangaj3@chinatelecom.cn
Huaimo.chen@futurewei.com
chen.ran@zte.com.cn
IETF111

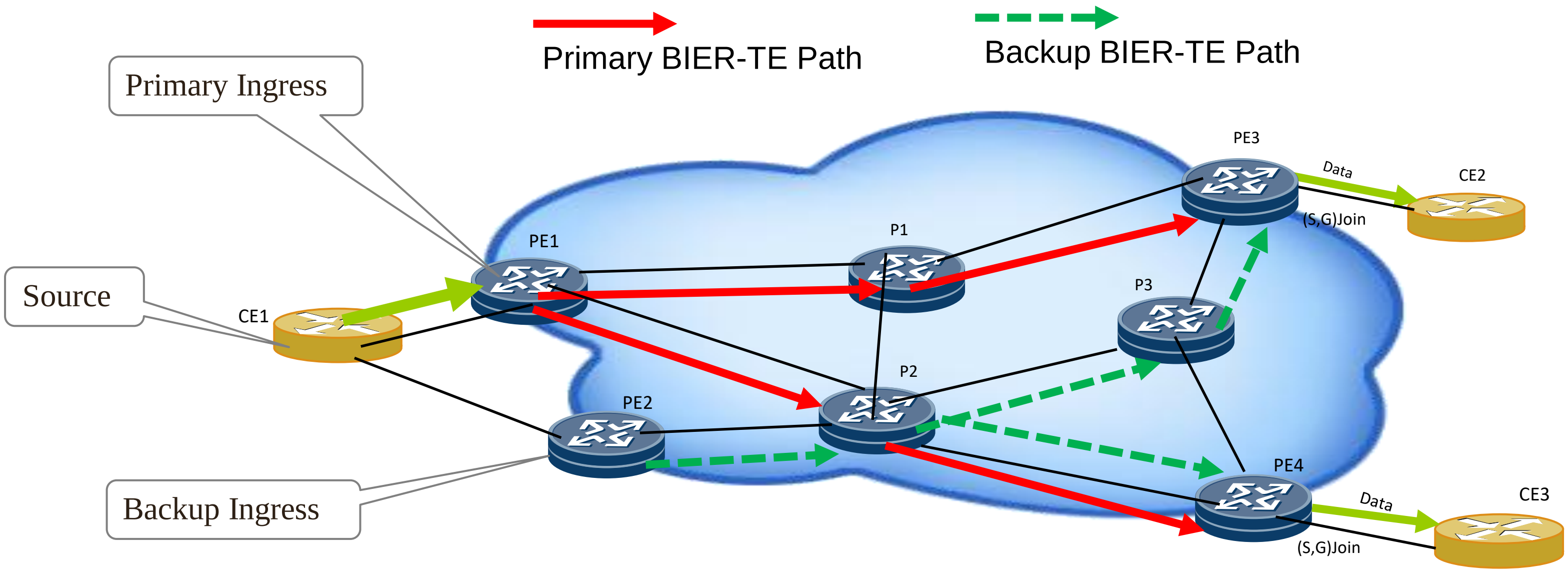
PCE for BIER-TE Ingress Protection

draft-chen-pce-bier-te-ingress-protect-00

Huaimo Chen, Mike McBride(Futurewei)
Gyan S. Mishra (Verizon Inc.)
Yisong Liu (China Mobile)
Aijun Wang (China Telecom)
Lei Liu (Fujitsu)
Xufeng Liu (Volta Networks)

IETF 111

BIER-TE Ingress Protection Overview

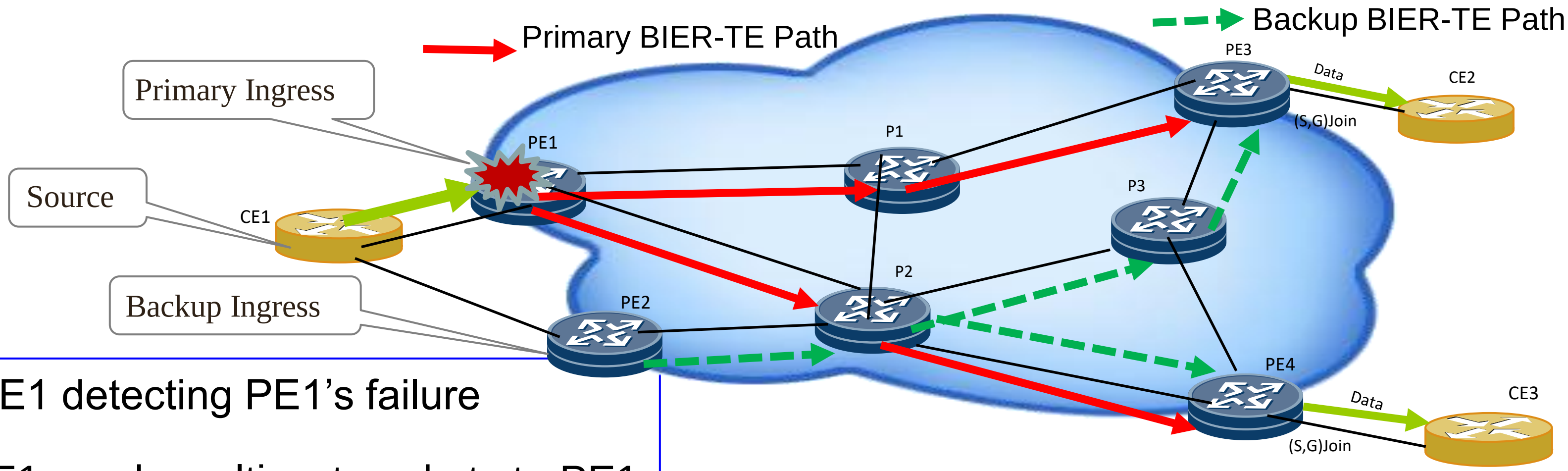


In Normal Operations
1. CE1 sends multicast packets to primary ingress PE1
2. PE1 encapsulates the packets with BIER-TE header encoding primary path from PE1 to PE3 and PE4

When PE1 fails
1. CE1 sends multicast packets to backup ingress PE2
2. PE2 encapsulates the packets with BIER-TE header encoding backup path from PE2 to PE3 and PE4

To support ingress protection,
PCE sends information to backup ingress: backup BIER-TE path + others

Behavior around Ingress Failure



A. Source CE1 detecting PE1's failure

Before failure:

1. CE1 sends multicast packets to PE1
2. PE2 is ready to encapsulate packets with backup path

After CE1 detects failure,

1. CE1 sends multicast packets to PE2
2. (PE2 encapsulates packets with backup path)

B. Backup Ingress PE2 detecting PE1's failure

Before failure:

1. CE1 sends multicast packets to both PE1 and PE2
2. PE2 drops the packets

After PE2 detects failure,

1. (CE1 sends multicast packets to PE2)
2. PE2 encapsulates packets with backup path

C. Both CE1 and PE2 detecting PE1's failure

Before failure:

1. CE1 sends multicast packets to PE1
2. PE2 drops packets

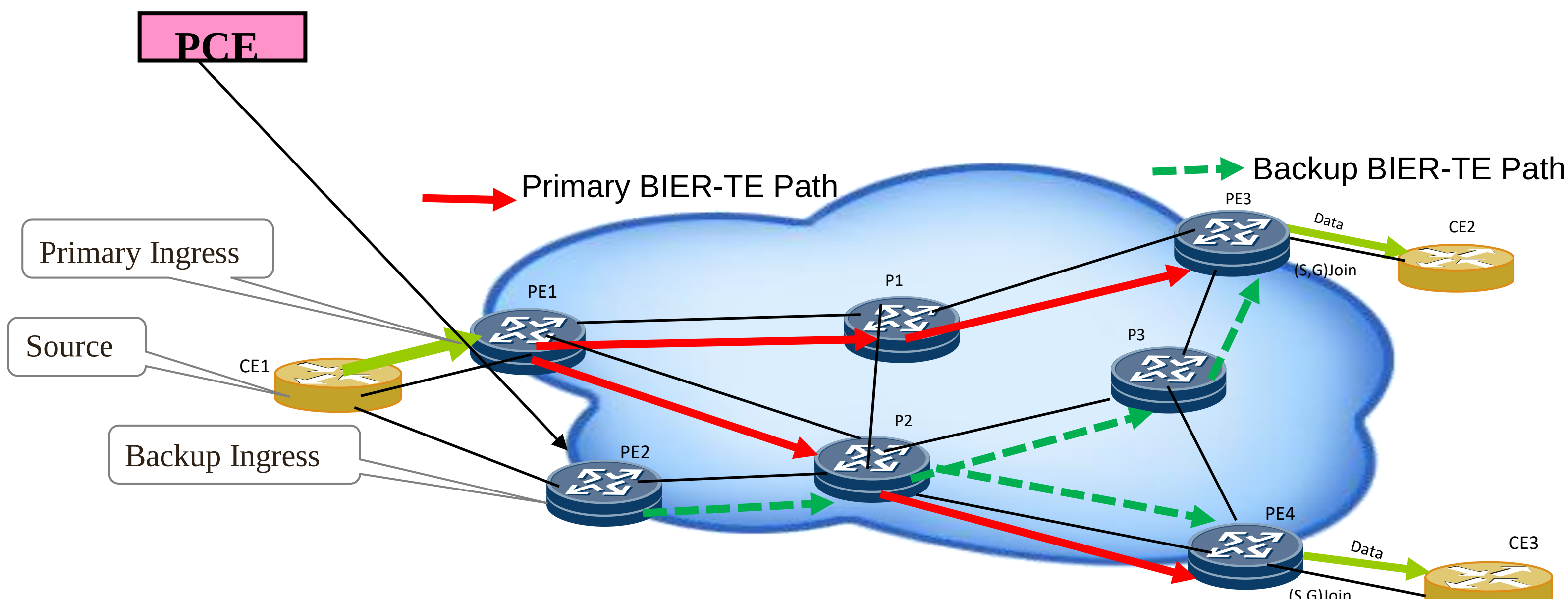
After CE1 and PE2 detect failure,

1. CE1 sends multicast packets to PE2
2. PE2 encapsulates packets with backup path

PCE sends PCC on PE2:
backup path + PE1's Address + others
PCE sends PCC on CE1:
instructions

Information sent to Backup Ingress by PCE

- **Backup BIER-TE Path** (can be encoded in the same way as primary path)
- **Primary Ingress** Address if backup ingress detects failure of primary ingress
- **Description of Traffic** carried by BIER-TE path
- **Service** Label/ID carried by BIER-TE path



Capability for BIER-TE Path Ingress Protection

PCC on backup ingress and PCE exchange capabilities of protecting ingress of BIER-TE path
Sub-TLV below is included in the PATH_SETUP_TYPE_CAPABILITY TLV with PST = TBD1 in Open

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |          Type = TBD2          |           Length=4            |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |           Reserved            |        Flags              |D|A|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
         BIER-TE-PATH_INGRESS_PROTECTION_CAPABILITY sub-TLV

o  D flag: A PCC sets this flag to 1 to indicate that it is able
        to detect its adjacent node's failure quickly
o  A flag: A PCE sets this flag to 1 to request a PCC to let the
        forwarding entry for the backup BIER-TE path be Active.
```

PCC on source and PCE exchange capabilities of supporting protection for ingress of BIER-TE path
PCECC-CAPABILITY Sub-TLV included in the PATH_SETUP_TYPE_CAPABILITY TLV in Open

A new flag bit P is defined in the Flags field of the PCECC-CAPABILITY sub-TLV

- P flag (for Ingress Protection): if set to 1 by a PCEP speaker, the P flag indicates that the PCEP speaker supports and is willing to handle the PCECC based central controller instructions for ingress protection. The bit MUST be set to 1 by both a PCC and a PCE for the PCECC ingress protection instruction download/report on a PCEP session.

BIER-TE Path Ingress Protection TLV for Backup Ingress

- BIER-TE-PATH_INGRESS_PROTECTION TLV is defined below
- When PCE sends PCC on backup ingress a PCInitiate for initiating a backup BIER-TE path to protect ingress of Primary BIER-TE path, containing it

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBD3           |        Length (variable)      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Reserved            |        Flags                |A|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                                                               ~
~                        sub-TLVs (optional)                    ~
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

              BIER-TE-PATH_INGRESS_PROTECTION TLV
```

```
o  A flag: A PCE sets this flag to 1 to request a PCC to let the
        forwarding entry for the backup BIER-TE path be Active.
```

Two optional sub-TLVs are defined

1. Primary-Ingress sub-TLV
2. Service sub-TLV

Primary Ingress IPv4/6 Address Sub-TLV

Primary-Ingress IPv4 sub-TLV indicates the IPv4 address of the primary ingress of a BIER-TE path

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBDa           |          Length (4)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Primary Ingress IPv4 Address (4 octets)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

            Primary Ingress IPv4 Address sub-TLV
```

Primary-Ingress IPv6 sub-TLV indicates the IPv6 address of the primary ingress of a BIER-TE path.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBDb           |         Length (16)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Primary Ingress IPv6 Address (16 octets)           |
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

            Primary Ingress IPv6 Address sub-TLV
```

Service Sub-TLVs

A Service sub-TLV contains a service ID or label to be added into a packet to be carried by a BIER-TE path. It has two formats: one for the service identified by a label and the other for the service identified by a service identifier (ID) of 32 or 128 bits.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBDc           |           Length (4)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        zero           |       Service Label (20 bits)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

                  Service Label sub-TLV
```

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBDd           |         Length (4/16)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Service ID (4 or 16 octets)              |
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

                      Service ID sub-TLV
```

BIER-TE Path Ingress Protection Object for Source

- a new object-type (TBDt) for BIER-TE ingress protection based on CCI object
- The body of the object with the new object-type is illustrated below. The object may be in PCRpt, PCUpd, or PCInitiate message.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            CC-ID                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Reserved             |              Flags        |B|D|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                        Optional TLV                         //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
              BIER-TE-PATH_INGRESS-PROTECTION Object Body
```

Flags: Two flag bits D and B are defined as follows:

- D: D = 1 instructs the PCC of the traffic source to Detect the failure of the primary ingress and switch the traffic to the backup ingress when it detects the failure.
- B: B = 1 instructs the PCC of the traffic source to send the traffic to Both the primary ingress and the backup ingress.

Optional TLV: Primary ingress TLV, backup ingress TLV and/or Multicast Flow Specification TLV.

Backup Ingress IPv4/6 Address TLV

Backup-Ingress IPv4 TLV indicates the IPv4 address of the backup ingress of a BIER-TE path

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBD8           |          Length (4)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Backup Ingress IPv4 Address (4 octets)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

            Backup Ingress IPv4 Address TLV
```

Backup-Ingress IPv6 TLV indicates the IPv6 address of the backup ingress of a BIER-TE path.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type = TBD9           |         Length (16)           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Backup Ingress IPv6 Address (16 octets)           |
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

             Backup Ingress IPv6 Address TLV
```

Next Step

Comments

PCEP Procedures and Extension for VLAN-based Traffic Forwarding

[draft-wang-pce-vlan-based-traffic-forwarding]

Yue Wang (China Telecom)

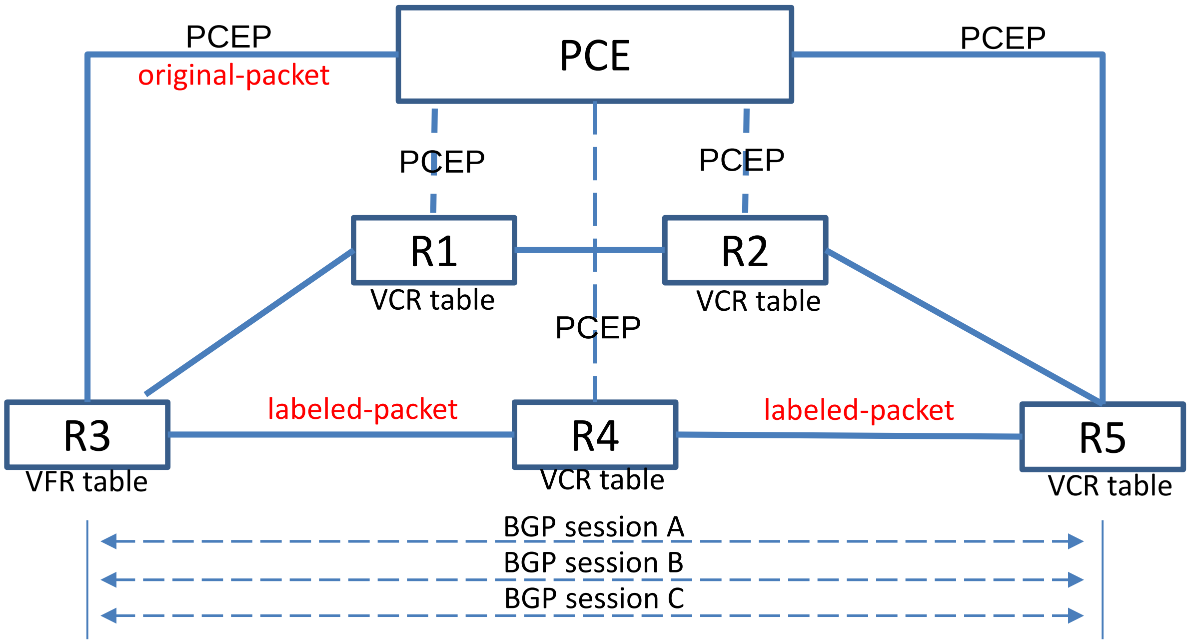
Aijun Wang (China Telecom)

IETF 111, July. 2021

Motivation

- Draft-ietf-pce-pcep-extension-native-ip describes the PCEP extensions and procedures to practically build a PCE-based central control mechanism.
- With the large scale deployment of Ethernet interface, it is possible to use the info contained in the Layer2 frame to simplify the E2E packet forwarding procedure.
- This document defines PCEP extension for VLAN-based traffic forwarding in native IP network and describes the processes of the data packet forwarding system based on VLAN info.
- This mechanism uses a completely new address space and is suitable for ipv4 and ipv6 networks and can leverage the existing PCE technologies as much as possible.

Procedures for VLAN-based Traffic Forwarding



1. The PCE calculates the explict route and sends the route information to the PCCs through PCInitiate messages.
2. The ingress PCC forms a VLAN-Forwarding routing(VFR) table, the transit PCC and the egress PCC forms a VLAN-Crossing routing(VCR) table.
3. The packet to be guaranteed matches the table and then be labeled with corresponding VLAN tag.
4. The labeled packet will be further sent to the PCC's specific subinterface identified by the VLAN tag and then be forwarded.

Capability Advertisement

- RFC8408 defines the Path Setup Type Capability TLV to indicate the path type supported by the PCE and PCC

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type (34)           |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Reserved                      |  Num of PSTs  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     PST#1     |      ...      |     PST#N     |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//              Optional sub-TLVs (variable)                   //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: PATH-SETUP-TYPE-CAPABILITY TLV

- New PST(TBD) is defined for VLAN-based traffic forwarding

- Draft-ietf-pce-pcep-extension-native-ip describes the PCECC capability sub-TLV to indicate the capability for TE in Native IP network.

- V (VLAN-based-forwarding-CAPABILITY - 1 bit - TBD2) is defined to indicate the PCEP speaker’s capability of VLAN based traffic

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Type=TBD2        |            Length=4           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                       |V|N|L|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: PCECC Capability sub-TLV

Updated PCEP Messages

```
<PCInitiate Message> ::= <Common Header>
                         <PCE-initiated-lsp-list>
   Where:
      <Common Header> is defined in [RFC5440]

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

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

      <PCE-initiated-lsp-central-control> ::= <SRP>
                                              <LSP>
                                              <cci-list>|
                                              ((<BPI>|<PPA>)
                                              <new-CCI>)

      <cci-list> ::=  <new-CCI>
                      [<cci-list>]

Where:
       <cci-list> is as per
       [I-D.ietf-pce-pcep-extension-for-pce-controller].
       <PCE-initiated-lsp-instantiation> and
       <PCE-initiated-lsp-deletion> are as per [RFC8281].
       <BPI> and <PPA> are as per
       [draft-ietf-pce-pcep-extension-native-ip-09]
```

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

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

        <state-report> ::= (<lsp-state-report>|
                            <central-control-report>)

        <lsp-state-report> ::= [<SRP>]
                               <LSP>
                               <path>

        <central-control-report> ::= [<SRP>]
                                     <LSP>
                                     <cci-list>|
                                     ((<BPI>|<PPA>)
                                     (<new-CCI>)

      Where:
        <path> is as per [RFC8231] and the LSP and SRP object are
        also defined in [RFC8231].
        <BPI> and <PPA> are as per
        [draft-ietf-pce-pcep-extension-native-ip-09]
```

- When PCInitiate message is used to create VLAN-based forwarding instructions, the SRP, LSP and CCI objects MUST be present.
- Only one of BPI, PPA or one type of CCI objects MUST be present.

New PCEP Objects(1/2)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             CC-ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Reserved1          |             Flags             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          VLAN-ID      |             Reserved2                 |
+---------------------------------------------------------------+
|                                                               |
//                      Peer IP address TLV                     //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: VLAN forwarding CCI Object

- VLAN ID(12 bits):the ID of the VLAN forwarding path that the PCC will set up on its logical subinterface in order to transfer the packet to the specific hop.

- [RFC8779] defines IPV4-ADDRESS, IPV6-ADDRESS, and UNNUMBERED-ENDPOINT TLVs for the use of Generalized Endpoint. The same TLVs can also be used in the CCI object to find the Peer address that matches egress PCC and further identify the packet to be guaranteed.

New PCEP Objects(2/2)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             CC-ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Reserved1            |             Flags           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       IN-VLAN-ID      |       OUT-VLAN-ID      |   Reserved2   |
+---------------------------------------------------------------+
|                                                               |
//                         Additional TLVs                     //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: VLAN Crossing CCI Object

- IN-VLAN ID(12 bits): The ID of the VLAN forwarding path which is used to identify the traffic that needs to be protected.

- OUT-VLAN ID(12 bits):The ID of the VLAN forwarding path that the PCC will set up on its logical subinterface in order to transfer the packet labeld with this VLAN ID to the specific hop.
  transit PCC - the value must not be 0.
  egress PCC - the value must be 0.

VLAN-Based forwarding info Advertisement Procedures

```
                  +----------------------+
        +---------+          PCE          + ---------+
        |         +----------^-----------+          |
        |          |         |          |           |
      M1&M1-R      |         |          |           |
        |          |         |          |           |
        |          |         |          |           |
        |        +--+      +--+       +--+          |
        |------- +R2+ ------+R3+--------+R4+ --------
        |        +--+      +--+       +--+          |
        |                                           |
        +--+               +--+                  +--+
        +R1+---------------+R5+-----------------+R6+
        +--+               +--+                  +--+
```

Figure 3: VLAN-Based forwarding info Advertisement
Procedures for Ingress PCC

Table 1: Message Information

No.	Peers	Type	Message Key Parameters
M1 M1-R	PCE/R1	PCInitiate PCRpt	CC-ID=X1 VLAN Forwarding CCI Object (Peer_IP=R6_A,VLAN_ID=VLAN_R1_R2)

VLAN-Based crossing info Advertisement Procedures

```
                  +----------------------+
         +---------+        PCE           +  ---------+
         |         +----------^-----------+           |
         |          |         |          |            |
         |       M1&M1-R   M2&M2-R    M3&M3-R     M4&M4-R
         |          |         |          |            |
         |          |         |          |            |
         |         +--+      +--+       +--+          |
         |-------  +R2+ -----+R3+-------+R4+  --------|
         |         +--+      +--+       +--+          |
         |                                            |
        +--+                 +--+                    +--+
        +R1+-----------------+R5+--------------------+R6+
        +--+                 +--+                    +--+
```

Figure 4: VLAN-Based crossing info Advertisement Procedures
for transit PCC and egress PCC

Table 2: Message Information

No.	Peers	Type	Message Key Parameters
M1 M1-R	PCE/R2	PCInitiate PCRpt	CC-ID=X1 VLAN crossing CCI Object (IN_VLAN_ID=VLAN_R1_R2,OUT_VLAN_ID=VLAN_R2_R3)
M2 M2-R	PCE/R3	PCInitiate PCRpt	CC-ID=X1 VLAN crossing CCI Object (IN_VLAN_ID=VLAN_R2_R3,OUT_VLAN_ID=VLAN_R3_R4)
M3 M3-R	PCE/R4	PCInitiate PCRpt	CC-ID=X1 VLAN crossing CCI Object (IN_VLAN_ID=VLAN_R3_R4,OUT_VLAN_ID=VLAN_R4_R6)
M4 M4-R	PCE/R6	PCInitiate PCRpt	CC-ID=X1 VLAN crossing CCI Object (IN_VLAN_ID=VLAN_R4_R6,OUT_VLAN_ID=0)

Next Step

- Comments

wangy73@chinatelecom.cn
wangaj3@chinatelecom.cn
IETF111