

PCE
Internet-Draft
Intended status: Standards Track
Expires: March 26, 2021

H. Chen
China Telecom
H. Yuan
UnionPay
T. Zhou
W. Li
G. Fioccola
Y. Wang
Huawei
September 22, 2020

Path Computation Element Communication Protocol (PCEP) Extensions to
Enable IFIT
draft-chen-pce-pcep-ifit-01

Abstract

This document defines PCEP extensions to distribute In-situ Flow Information Telemetry (IFIT) information. So that IFIT behavior can be enabled automatically when the path is instantiated. In-situ Flow Information Telemetry (IFIT) refers to network OAM data plane on-path telemetry techniques, in particular the most popular are In-situ OAM (IOAM) and Alternate Marking. The IFIT attributes here described can be generalized for all path types but the application to Segment Routing (SR) is considered in this document. This document extends PCEP to carry the IFIT attributes under the stateful PCE model.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 March 26, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. PCEP Extensions for IFIT Attributes	4
2.1. IFIT for SR Policies	4
3. IFIT capability advertisement TLV	4
4. IFIT Attributes TLV	7
4.1. IOAM Sub-TLVs	8
4.1.1. IOAM Pre-allocated Trace Option Sub-TLV	8
4.1.2. IOAM Incremental Trace Option Sub-TLV	9
4.1.3. IOAM Directly Export Option Sub-TLV	10
4.1.4. IOAM Edge-to-Edge Option Sub-TLV	11
4.2. Enhanced Alternate Marking Sub-TLV	12
5. PCEP Messages	13
5.1. The PCInitiate Message	13
5.2. The PCUpd Message	13
5.3. The PCRpt Message	13
6. Example of application to SR Policy	14
7. IANA Considerations	14
8. Security Considerations	16
9. Contributors	17
10. Acknowledgements	17
11. References	17
11.1. Normative References	17
11.2. Informative References	19
Appendix A.	20
Authors' Addresses	20

1. Introduction

In-situ Flow Information Telemetry (IFIT) refers to network OAM (Operations, Administration, and Maintenance) data plane on-path telemetry techniques, including In-situ OAM (IOAM) [I-D.ietf-ippm-ioam-data] and Alternate Marking [RFC8321]. It can provide flow information on the entire forwarding path on a per-packet basis in real time.

An automatic network requires the Service Level Agreement (SLA) monitoring on the deployed service. So that the system can quickly detect the SLA violation or the performance degradation, hence to change the service deployment.

This document defines extensions to PCEP to distribute paths carrying IFIT information. So that IFIT behavior can be enabled automatically when the path is instantiated.

RFC 5440 [RFC5440] describes the Path Computation Element Protocol (PCEP) as a communication mechanism between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between a PCE and a PCE.

RFC 8231 [RFC8231] specifies extensions to PCEP to enable stateful control and it describes two modes of operation: passive stateful PCE and active stateful PCE. Further, RFC 8281 [RFC8281] describes the setup, maintenance, and teardown of PCE-initiated LSPs for the stateful PCE model.

When a PCE is used to initiate paths using PCEP, it is important that the head end of the path also understands the IFIT behavior that is intended for the path. When PCEP is in use for path initiation it makes sense for that same protocol to be used to also carry the IFIT attributes that describe the IOAM or Alternate Marking procedure that needs to be applied to the data that flow those paths.

The PCEP extension defined in this document allows to signal the IFIT capabilities. In this way IFIT methods are automatically activated and running. The flexibility and dynamicity of the IFIT applications are given by the use of additional functions on the controller and on the network nodes, but this is out of scope here.

The Use Case of Segment Routing (SR) is discussed considering that IFIT methods are becoming mature for Segment Routing over the MPLS data plane (SR-MPLS) and Segment Routing over IPv6 data plane (SRv6). In this way SR policy native IFIT can facilitate the closed loop control and enable the automation of SR service.

Segment Routing (SR) policy [I-D.ietf-spring-segment-routing-policy] is a set of candidate SR paths consisting of one or more segment lists and necessary path attributes. It enables instantiation of an ordered list of segments with a specific intent for traffic steering.

It is to be noted the companion document [I-D.qin-idr-sr-policy-ifit] that proposes the BGP extension to enable IFIT methods for SR policy.

2. PCEP Extensions for IFIT Attributes

This document is to add IFIT attribute TLVs as PCEP Extensions. The following sections will describe the requirement and usage of different IFIT modes, and define the corresponding TLV encoding in PCEP.

The IFIT attributes here described can be generalized and included as TLVs carried inside the LSPA (LSP Attributes) object in order to be applied for all path types, as long as they support the relevant data plane telemetry method. IFIT Attributes TLVs are optional and can be taken into account by the PCE during path computation and by the PCC during path setup. In general, the LSPA object can be carried within a PCInitiate message, a PCUpd message, or a PCRpt message in the stateful PCE model.

In this document it is considered the case of SR Policy since IOAM and Alternate Marking are more mature especially for Segment Routing (SR) and for IPv6.

2.1. IFIT for SR Policies

RFC 8664 [RFC8664] and [I-D.ietf-pce-segment-routing-ipv6] specify extensions to the Path Computation Element Communication Protocol (PCEP) that allow a stateful PCE to compute and initiate Traffic-Engineering (TE) paths, as well as a Path Computation Client (PCC) to request a path subject to certain constraints and optimization criteria in SR networks both for SR-MPLS and SRv6.

IFIT attributes, here defined as TLVs for the LSPA object, complement both RFC 8664 [RFC8664], [I-D.ietf-pce-segment-routing-ipv6] and [I-D.ietf-pce-segment-routing-policy-cp].

3. IFIT capability advertisement TLV

During the PCEP initialization phase, PCEP speakers (PCE or PCC) SHOULD advertise their support of IFIT methods (e.g. IOAM and Alternate Marking).

A PCEP speaker includes the IFIT-CAPABILITY TLVs in the OPEN object to advertise its support for PCEP IFIT extensions. The presence of the IFIT-CAPABILITY TLV in the OPEN object indicates that the IFIT methods are supported.

RFC 8664 [RFC8664] and [I-D.ietf-pce-segment-routing-ipv6] define a new Path Setup Type (PST) for SR and also define the SR-PCE-CAPABILITY sub-TLV. This document defines a new IFIT-CAPABILITY TLV, that is an optional TLV for use in the OPEN Object for IFIT attributes via PCEP capability advertisement.

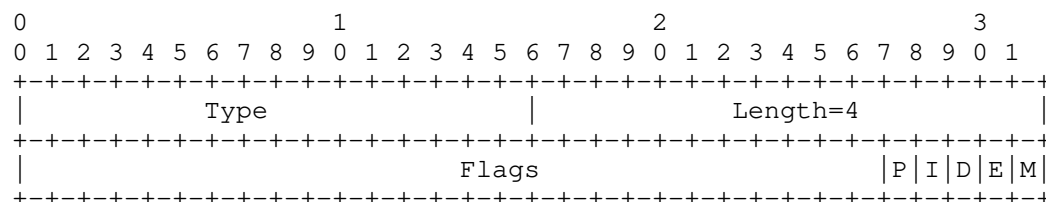


Fig. 1 IFIT-CAPABILITY TLV Format

Where:

Type: to be assigned by IANA.

Length: 4.

Flags: The following flags are defined in this document:

P: IOAM Pre-allocated Trace Option Type-enabled flag [I-D.ietf-ippm-ioam-data]. If set to 1 by a PCC, the P flag indicates that the PCC allows instantiation of the IOAM Pre-allocated Trace feature by a PCE. If set to 1 by a PCE, the P flag indicates that the PCE supports the IOAM Pre-allocated Trace feature instantiation. The P flag MUST be set by both PCC and PCE in order to support the IOAM Pre-allocated Trace instantiation

I: IOAM Incremental Trace Option Type-enabled flag [I-D.ietf-ippm-ioam-data]. If set to 1 by a PCC, the I flag indicates that the PCC allows instantiation of the IOAM Incremental Trace feature by a PCE. If set to 1 by a PCE, the I flag indicates that the PCE supports the relative IOAM Incremental Trace feature instantiation. The I flag MUST be set by both PCC and PCE in order to support the IOAM Incremental Trace feature instantiation

D: IOAM DEX Option Type-enabled flag [I-D.ietf-ippm-ioam-direct-export]. If set to 1 by a PCC, the D

flag indicates that the PCC allows instantiation of the relative IOAM DEX feature by a PCE. If set to 1 by a PCE, the D flag indicates that the PCE supports the relative IOAM DEX feature instantiation. The D flag MUST be set by both PCC and PCE in order to support the IOAM DEX feature instantiation

E: IOAM E2E Option Type-enabled flag [I-D.ietf-ippm-ioam-data]. If set to 1 by a PCC, the E flag indicates that the PCC allows instantiation of the relative IOAM E2E feature by a PCE. If set to 1 by a PCE, the E flag indicates that the PCE supports the relative IOAM E2E feature instantiation. The E flag MUST be set by both PCC and PCE in order to support the IOAM E2E feature instantiation

M: Alternate Marking enabled flag RFC 8321 [RFC8321]. If set to 1 by a PCC, the M flag indicates that the PCC allows instantiation of the relative Alternate Marking feature by a PCE. If set to 1 by a PCE, the M flag indicates that the PCE supports the relative Alternate Marking feature instantiation. The M flag MUST be set by both PCC and PCE in order to support the Alternate Marking feature instantiation

Unassigned bits are considered reserved. They MUST be set to 0 on transmission and MUST be ignored on receipt.

Advertisement of the IFIT-CAPABILITY TLV implies support of IFIT methods (IOAM and/or Alternate Marking) as well as the objects, TLVs, and procedures defined in this document. It is worth mentioning that IOAM and Alternate Marking can be activated one at a time or can coexist; so it is possible to have only IOAM or only Alternate Marking enabled but they are recognized in general as IFIT capability.

The IFIT Capability Advertisement can imply the following cases:

- o The PCEP protocol extensions for IFIT MUST NOT be used if one or both PCEP speakers have not included the IFIT-CAPABILITY TLV in their respective OPEN message.
- o A PCEP speaker that does not recognize the extensions defined in this document would simply ignore the TLVs as per RFC 5440 [RFC5440].
- o If a PCEP speaker supports the extensions defined in this document but did not advertise this capability, then upon receipt of IFIT-ATTRIBUTES TLV in the LSP Attributes (LSPA) object, it SHOULD generate a PCERR with Error-Type 19 (Invalid Operation) with the

relative Error-value "IFIT capability not advertised" and ignore the IFIT-ATTRIBUTES TLV.

4. IFIT Attributes TLV

The IFIT-ATTRIBUTES TLV provides the configurable knobs of the IFIT feature, and it can be included as an optional TLV in the LSPA object (as described in RFC 5440 [RFC5440]).

For a PCE-initiated LSP RFC 8281 [RFC8281], this TLV is included in the LSPA object with the PCInitiate message. For the PCC-initiated delegated LSPs, this TLV is carried in the Path Computation State Report (PCRpt) message in the LSPA object. This TLV is also carried in the LSPA object with the Path Computation Update Request (PCUpd) message to direct the PCC (LSP head-end) to make updates to IFIT attributes.

The TLV is encoded in all PCEP messages for the LSP if IFIT feature is enabled. The absence of the TLV indicates the PCEP speaker wishes to disable the feature. This TLV includes multiple IFIT-ATTRIBUTES sub-TLVs. The IFIT-ATTRIBUTES sub-TLVs are included if there is a change since the last information sent in the PCEP message. The default values for missing sub-TLVs apply for the first PCEP message for the LSP.

The format of the IFIT-ATTRIBUTES TLV is shown in the following figure:

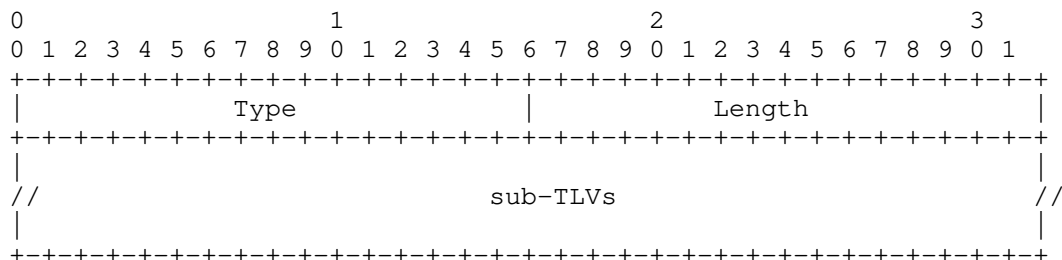


Fig. 2 IFIT-ATTRIBUTES TLV Format

Where:

Type: to be assigned by IANA.

Length: The Length field defines the length of the value portion in bytes as per RFC 5440 [RFC5440].

Value: This comprises one or more sub-TLVs.

The following sub-TLVs are defined in this document:

Type	Len	Name
1	8	IOAM Pre-allocated Trace Option
2	8	IOAM Incremental Trace Option
3	12	IOAM Directly Export Option
4	4	IOAM Edge-to-Edge Option
5	4	Enhanced Alternate Marking

Fig. 3 Sub-TLV Types of the IFIT-ATTRIBUTES TLV

4.1. IOAM Sub-TLVs

In-situ Operations, Administration, and Maintenance (IOAM) [I-D.ietf-ippm-ioam-data] records operational and telemetry information in the packet while the packet traverses a path between two points in the network. In terms of the classification given in RFC 7799 [RFC7799] IOAM could be categorized as Hybrid Type 1. IOAM mechanisms can be leveraged where active OAM do not apply or do not offer the desired results.

For the SR use case, when SR policy enables IOAM, the IOAM header will be inserted into every packet of the traffic that is steered into the SR paths. Since this document aims to define the control plane, it is to be noted that a relevant document for the data plane is [I-D.ietf-ippm-ioam-ipv6-options] for Segment Routing over IPv6 data plane (SRv6).

4.1.1. IOAM Pre-allocated Trace Option Sub-TLV

The IOAM tracing data is expected to be collected at every node that a packet traverses to ensure visibility into the entire path a packet takes within an IOAM domain. The preallocated tracing option will create pre-allocated space for each node to populate its information.

The format of IOAM pre-allocated trace option Sub-TLV is defined as follows:

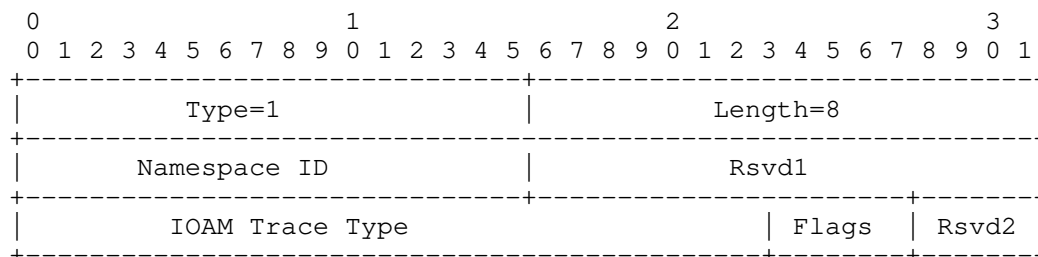


Fig. 4 IOAM Pre-allocated Trace Option Sub-TLV

Where:

Type: 1 (to be assigned by IANA).

Length: 8. It is the total length of the value field not including Type and Length fields.

Namespace ID: A 16-bit identifier of an IOAM-Namespace. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

IOAM Trace Type: A 24-bit identifier which specifies which data types are used in the node data list. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

Flags: A 4-bit field. The definition is the same as described in [I-D.ietf-ippm-ioam-flags] and section 4.4 of [I-D.ietf-ippm-ioam-data].

Rsvd1: A 16-bit field reserved for further usage. It MUST be zero.

Rsvd2: A 4-bit field reserved for further usage. It MUST be zero.

4.1.2. IOAM Incremental Trace Option Sub-TLV

The incremental tracing option contains a variable node data fields where each node allocates and pushes its node data immediately following the option header.

The format of IOAM incremental trace option Sub-TLV is defined as follows:

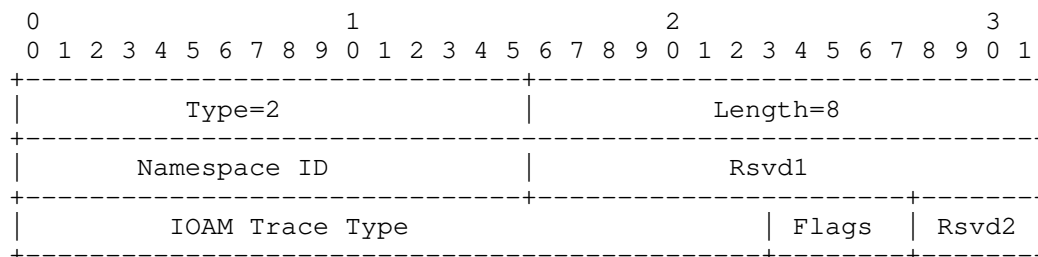


Fig. 5 IOAM Incremental Trace Option Sub-TLV

Where:

Type: 2 (to be assigned by IANA).

Length: 8. It is the total length of the value field not including Type and Length fields.

All the other fields definition is the same as the pre-allocated trace option Sub-TLV in the previous section.

4.1.3. IOAM Directly Export Option Sub-TLV

IOAM directly export option is used as a trigger for IOAM data to be directly exported to a collector without being pushed into in-flight data packets.

The format of IOAM directly export option Sub-TLV is defined as follows:

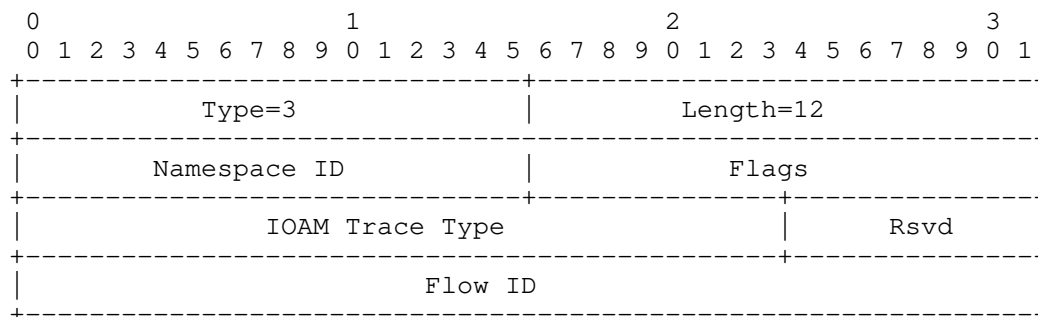


Fig. 6 IOAM Directly Export Option Sub-TLV

Where:

Type: 3 (to be assigned by IANA).

Length: 12. It is the total length of the value field not including Type and Length fields.

Namespace ID: A 16-bit identifier of an IOAM-Namespace. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

IOAM Trace Type: A 24-bit identifier which specifies which data types are used in the node data list. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

Flags: A 16-bit field. The definition is the same as described in section 3.2 of [I-D.ietf-ippm-ioam-direct-export].

Flow ID: A 32-bit flow identifier. The definition is the same as described in section 3.2 of [I-D.ietf-ippm-ioam-direct-export].

Rsvd: A 4-bit field reserved for further usage. It MUST be zero.

4.1.4. IOAM Edge-to-Edge Option Sub-TLV

The IOAM edge to edge option is to carry data that is added by the IOAM encapsulating node and interpreted by IOAM decapsulating node.

The format of IOAM edge-to-edge option Sub-TLV is defined as follows:

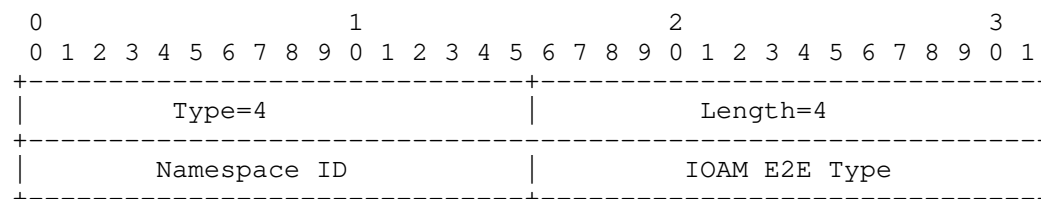


Fig. 7 IOAM Edge-to-Edge Option Sub-TLV

Where:

Type: 4 (to be assigned by IANA).

Length: 4. It is the total length of the value field not including Type and Length fields.

Namespace ID: A 16-bit identifier of an IOAM-Namespace. The definition is the same as described in section 4.6 of [I-D.ietf-ippm-ioam-data].

IOAM E2E Type: A 16-bit identifier which specifies which data types are used in the E2E option data. The definition is the same as described in section 4.6 of [I-D.ietf-ippm-ioam-data].

4.2. Enhanced Alternate Marking Sub-TLV

The Alternate Marking [RFC8321] technique is an hybrid performance measurement method, per RFC 7799 [RFC7799] classification of measurement methods. Because this method is based on marking consecutive batches of packets. It can be used to measure packet loss, latency, and jitter on live traffic.

For the SR use case, since this document aims to define the control plane, it is to be noted that a relevant document for the data plane is [I-D.ietf-6man-ipv6-alt-mark] for Segment Routing over IPv6 data plane (SRv6).

The format of Enhanced Alternate Marking (EAM) Sub-TLV is defined as follows:

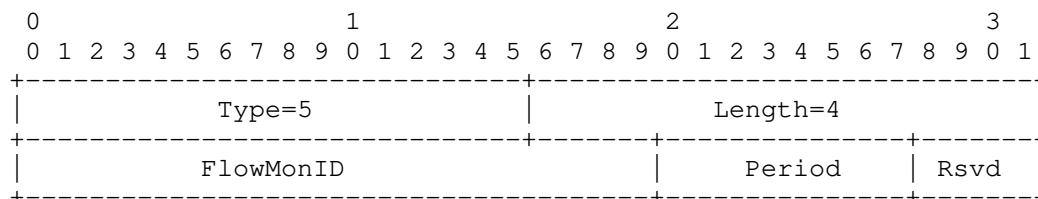


Fig. 8 Enhanced Alternate Marking Sub-TLV

Where:

Type: 5 (to be assigned by IANA).

Length: 4. It is the total length of the value field not including Type and Length fields.

FlowMonID: A 20-bit identifier to uniquely identify a monitored flow within the measurement domain. The definition is the same as described in section 5.3 of [I-D.ietf-6man-ipv6-alt-mark]. It is to be noted that PCE also needs to maintain the uniqueness of FlowMonID as described in [I-D.ietf-6man-ipv6-alt-mark].

Period: Time interval between two alternate marking period. The unit is second.

Rsvd: A 4-bit field reserved for further usage. It MUST be zero.

5. PCEP Messages

5.1. The PCInitiate Message

A PCInitiate message is a PCEP message sent by a PCE to a PCC to trigger LSP instantiation or deletion RFC 8281 [RFC8281].

For the PCE-initiated LSP with the IFIT feature enabled, IFIT-ATTRIBUTES TLV MUST be included in the LSPA object with the PCInitiate message.

The Routing Backus-Naur Form (RBNF) definition of the PCInitiate message RFC 8281 [RFC8281] is unchanged by this document.

5.2. The PCUpd Message

A PCUpd message is a PCEP message sent by a PCE to a PCC to update the LSP parameters RFC 8231 [RFC8231].

For PCE-initiated LSPs with the IFIT feature enabled, the IFIT-ATTRIBUTES TLV MUST be included in the LSPA object with the PCUpd message. The PCE can send this TLV to direct the PCC to change the IFIT parameters.

The RBNF definition of the PCUpd message RFC 8231 [RFC8231] is unchanged by this document.

5.3. The PCRpt Message

The PCRpt message RFC 8231 [RFC8231] is a PCEP message sent by a PCC to a PCE to report the status of one or more LSPs.

For PCE-initiated LSPs RFC 8281 [RFC8281], the PCC creates the LSP using the attributes communicated by the PCE and the local values for the unspecified parameters. After the successful instantiation of the LSP, the PCC automatically delegates the LSP to the PCE and generates a PCRpt message to provide the status report for the LSP.

The RBNF definition of the PCRpt message RFC 8231 [RFC8231] is unchanged by this document.

For both PCE-initiated and PCC-initiated LSPs, when the LSP is instantiated the IFIT methods are applied as specified for the corresponding data plane. [I-D.ietf-ippm-ioam-ipv6-options] and [I-D.ietf-6man-ipv6-alt-mark] are the relevant documents for Segment Routing over IPv6 data plane (SRv6).

6. Example of application to SR Policy

A PCC or PCE sets the IFIT-CAPABILITY TLV in the Open message during the PCEP initialization phase to indicate that it supports the IFIT procedures.

[I-D.ietf-pce-segment-routing-policy-cp] defines the PCEP extension to support Segment Routing Policy Candidate Paths and in this regard the SRPAG Association object is introduced.

The Examples of PCC Initiated SR Policy with single or multiple candidate-paths and PCE Initiated SR Policy with single or multiple candidate-paths are reported in [I-D.ietf-pce-segment-routing-policy-cp].

In case of PCC Initiated SR Policy, PCC sends PCReq message to the PCE, encoding the SRPAG ASSOCIATION object and IFIT-ATTRIBUTES TLV via the LSPA object. This is valid for both single and multiple candidate-paths. Finally PCE returns the path in PCRep message, and echoes back the SRPAG object that were used in the computation and IFIT LSPA TLVs too. Additionally, PCC sends PCRpt message to the PCE, including the LSP object and the SRPAG ASSOCIATION object and IFIT-ATTRIBUTES TLV via the LSPA object. Then PCE computes path and finally PCE updates the SR policy candidate path's ERO using PCUpd message considering the IFIT LSPA TLVs too.

In case of PCE Initiated SR Policy, PCE sends PCInitiate message, containing the SRPAG Association object and IFIT-ATTRIBUTES TLV via the LSPA object. This is valid for both single and multiple candidate-paths. Then PCC uses the color, endpoint and preference from the SRPAG object to create a new candidate path considering the IFIT LSPA TLVs too. Finally PCC sends a PCRpt message back to the PCE to report the newly created Candidate Path. The PCRpt message contains the SRPAG Association object and IFIT-ATTRIBUTES information.

The procedure of enabling/disabling IFIT is simple, indeed the PCE can update the IFIT-ATTRIBUTES of the LSP by sending subsequent Path Computation Update Request (PCUpd) messages. PCE can update the IFIT-ATTRIBUTES of the LSP by sending Path Computation State Report (PCRpt) messages.

7. IANA Considerations

This document defines the new IFIT-CAPABILITY TLV and IFIT-ATTRIBUTES TLV. IANA is requested to make the assignment from the "PCEP TLV Type Indicators" subregistry of the "Path Computation Element Protocol (PCEP) Numbers" registry as follows:

Value	Description	Reference
TBD1	IFIT-CAPABILITY	This document
TBD2	IFIT-ATTRIBUTES	This document

This document specifies the IFIT-CAPABILITY TLV Flags field. IANA is requested to create a registry to manage the value of the IFIT-CAPABILITY TLV's Flags field within the "Path Computation Element Protocol (PCEP) Numbers" registry.

New values are to be assigned by Standards Action RFC 8126 [RFC8126]. Each bit should be tracked with the following qualities:

- * Bit number (count from 0 as the most significant bit)
- * Flag Name
- * Reference

IANA is requested to set 5 new bits in the IFIT-CAPABILITY TLV Flags Field registry, as follows:

Bit no.	Flag Name	Reference
27	P: IOAM Pre-allocated Trace Option flag	This document
28	I: IOAM Incremental Trace Option flag	This document
29	D: IOAM Directly Export Option flag	This document
30	E: IOAM Edge-to-Edge Option	This document
31	M: Alternate Marking Flag	This document

This document also specifies the IFIT-ATTRIBUTES sub-TLVs. IANA is requested to create an "IFIT-ATTRIBUTES Sub-TLV Types" subregistry within the "Path Computation Element Protocol (PCEP) Numbers" registry.

IANA is requested to set the Registration Procedure for this registry to read as follows:

Range	Registration Procedure
0-65503	IETF Review
65504-65535	Experimental Use

This document defines the following types:

Type	Description	Reference
0	Reserved	This document
1	IOAM Pre-allocated Trace Option	This document
2	IOAM Incremental Trace Option	This document
3	IOAM Directly Export Option	This document
4	IOAM Edge-to-Edge Option	This document
5	Enhanced Alternate Marking	This document
6-65503	Unassigned	This document
65504-65535	Experimental Use	This document

This document defines a new Error-value for PCErr message of Error-Type 19 (Invalid Operation). IANA is requested to allocate a new Error-value within the "PCEP-ERROR Object Error Types and Values" subregistry of the "Path Computation Element Protocol (PCEP) Numbers" registry as follows:

Error-Type	Meaning	Error-value	Reference
19	Invalid Operation	TBD3: IFIT capability not advertised	This document

8. Security Considerations

This document defines the new IFIT-CAPABILITY TLV and IFIT Attributes TLVs, which do not add any substantial new security concerns beyond those already discussed in RFC 8231 [RFC8231] and RFC 8281 [RFC8281] for stateful PCE operations. As per RFC 8231 [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security

(TLS) RFC 8253 [RFC8253], as per the recommendations and best current practices in BCP 195 RFC 7525 [RFC7525] (unless explicitly set aside in RFC 8253 [RFC8253]).

Implementation of IFIT methods (IOAM and Alternate Marking) are mindful of security and privacy concerns, as explained in [I-D.ietf-ippm-ioam-data] and RFC 8321 [RFC8321]. Anyway incorrect IFIT parameters in the IFIT-ATTRIBUTES sub-TLVs SHOULD not have an adverse effect on the LSP as well as on the network, since it affects only the operation of the telemetry methodology.

9. Contributors

The following people provided relevant contributions to this document:

Dhruv Doody, Huawei Technologies, dhruv.ietf@gmail.com

10. Acknowledgements

The authors of this document would like to thank Huaimo Chen for the comments and review of this document.

11. References

11.1. Normative References

- [I-D.ietf-6man-ipv6-alt-mark]
Fioccola, G., Zhou, T., Cociglio, M., Qin, F., and R. Pang, "IPv6 Application of the Alternate Marking Method", draft-ietf-6man-ipv6-alt-mark-01 (work in progress), June 2020.
- [I-D.ietf-ippm-ioam-data]
Brockners, F., Bhandari, S., and T. Mizrahi, "Data Fields for In-situ OAM", draft-ietf-ippm-ioam-data-10 (work in progress), July 2020.
- [I-D.ietf-ippm-ioam-direct-export]
Song, H., Gafni, B., Zhou, T., Li, Z., Brockners, F., Bhandari, S., Sivakolundu, R., and T. Mizrahi, "In-situ OAM Direct Exporting", draft-ietf-ippm-ioam-direct-export-01 (work in progress), August 2020.

- [I-D.ietf-ippm-ioam-flags]
Mizrahi, T., Brockners, F., Bhandari, S., Sivakolundu, R., Pignataro, C., Kfir, A., Gafni, B., Spiegel, M., and J. Lemon, "In-situ OAM Flags", draft-ietf-ippm-ioam-flags-02 (work in progress), July 2020.
- [I-D.ietf-ippm-ioam-ipv6-options]
Bhandari, S., Brockners, F., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Kfir, A., Gafni, B., Lapukhov, P., Spiegel, M., Krishnan, S., Asati, R., and M. Smith, "In-situ OAM IPv6 Options", draft-ietf-ippm-ioam-ipv6-options-03 (work in progress), September 2020.
- [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>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC7799] Morton, A., "Active and Passive Metrics and Methods (with Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799, May 2016, <<https://www.rfc-editor.org/info/rfc7799>>.
- [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>>.
- [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>>.

- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8321] Fioccola, G., Ed., Capello, A., Cociglio, M., Castaldelli, L., Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi, "Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8321, DOI 10.17487/RFC8321, January 2018, <<https://www.rfc-editor.org/info/rfc8321>>.
- [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>>.

11.2. Informative References

- [I-D.ietf-pce-segment-routing-ipv6]
Li, C., Negi, M., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", draft-ietf-pce-segment-routing-ipv6-06 (work in progress), July 2020.
- [I-D.ietf-pce-segment-routing-policy-cpl]
Koldychev, M., Sivabalan, S., Barth, C., Peng, S., and H. Bidgoli, "PCEP extension to support Segment Routing Policy Candidate Paths", draft-ietf-pce-segment-routing-policy-cp-00 (work in progress), June 2020.
- [I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-08 (work in progress), July 2020.
- [I-D.qin-idr-sr-policy-ifit]
Qin, F., Yuan, H., Zhou, T., Fioccola, G., and Y. Wang, "BGP SR Policy Extensions to Enable IFIT", draft-qin-idr-sr-policy-ifit-03 (work in progress), September 2020.

Appendix A.

Authors' Addresses

Huanan Chen
China Telecom
Guangzhou
China

Email: chenhuan6@chinatelecom.cn

Hang Yuan
UnionPay
1899 Gu-Tang Rd., Pudong
Shanghai
China

Email: yuanhang@unionpay.com

Tianran Zhou
Huawei
156 Beiqing Rd., Haidian District
Beijing
China

Email: zhoutianran@huawei.com

Weidong Li
Huawei
156 Beiqing Rd., Haidian District
Beijing
China

Email: poly.li@huawei.com

Giuseppe Fioccola
Huawei
Riesstrasse, 25
Munich
Germany

Email: giuseppe.fioccola@huawei.com

Yali Wang
Huawei
156 Beiqing Rd., Haidian District
Beijing
China

Email: wangyalil1@huawei.com

PCE
Internet-Draft
Intended status: Standards Track
Expires: August 8, 2022

H. Yuan
UnionPay
T. Zhou
W. Li
G. Fioccola
Y. Wang
Huawei
February 4, 2022

Path Computation Element Communication Protocol (PCEP) Extensions to
Enable IFIT
draft-chen-pce-pcep-ifit-06

Abstract

This document defines PCEP extensions to distribute In-situ Flow Information Telemetry (IFIT) information. So that IFIT behavior can be enabled automatically when the path is instantiated. In-situ Flow Information Telemetry (IFIT) refers to network OAM data plane on-path telemetry techniques, in particular the most popular are In-situ OAM (IOAM) and Alternate Marking. The IFIT attributes here described can be generalized for all path types but the application to Segment Routing (SR) is considered in this document. This document extends PCEP to carry the IFIT attributes under the stateful PCE model.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 August 8, 2022.

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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. PCEP Extensions for IFIT Attributes	4
2.1. IFIT for SR Policies	5
3. IFIT capability advertisement TLV	5
4. IFIT Attributes TLV	7
4.1. IOAM Sub-TLVs	8
4.1.1. IOAM Pre-allocated Trace Option Sub-TLV	9
4.1.2. IOAM Incremental Trace Option Sub-TLV	10
4.1.3. IOAM Directly Export Option Sub-TLV	10
4.1.4. IOAM Edge-to-Edge Option Sub-TLV	11
4.2. Enhanced Alternate Marking Sub-TLV	12
5. PCEP Messages	13
5.1. The PCInitiate Message	13
5.2. The PCUpd Message	14
5.3. The PCRpt Message	14
6. Example of application to SR Policy	14
7. IANA Considerations	15
7.1. PCEP TLV Type Indicators	15
7.2. IFIT-CAPABILITY TLV Flags field	16
7.3. IFIT-ATTRIBUTES Sub-TLV	16
7.4. Enhanced Alternate Marking Sub-TLV Flags field	17
7.5. PCEP Error Codes	18
8. Security Considerations	18
9. Contributors	19
10. Acknowledgements	19
11. References	19
11.1. Normative References	19
11.2. Informative References	21
Authors' Addresses	22

1. Introduction

In-situ Flow Information Telemetry (IFIT) refers to network OAM (Operations, Administration, and Maintenance) data plane on-path telemetry techniques, including In-situ OAM (IOAM) [I-D.ietf-ippm-ioam-data] and Alternate Marking [RFC8321]. It can provide flow information on the entire forwarding path on a per-packet basis in real time.

An automatic network requires the Service Level Agreement (SLA) monitoring on the deployed service. So that the system can quickly detect the SLA violation or the performance degradation, hence to change the service deployment.

This document defines extensions to PCEP to distribute paths carrying IFIT information. So that IFIT behavior can be enabled automatically when the path is instantiated.

RFC 5440 [RFC5440] describes the Path Computation Element Protocol (PCEP) as a communication mechanism between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between a PCE and a PCE.

RFC 8231 [RFC8231] specifies extensions to PCEP to enable stateful control and it describes two modes of operation: passive stateful PCE and active stateful PCE. Further, RFC 8281 [RFC8281] describes the setup, maintenance, and teardown of PCE-initiated LSPs for the stateful PCE model.

When a PCE is used to initiate paths using PCEP, it is important that the head end of the path also understands the IFIT behavior that is intended for the path. When PCEP is in use for path initiation it makes sense for that same protocol to be used to also carry the IFIT attributes that describe the IOAM or Alternate Marking procedure that needs to be applied to the data that flow those paths.

The PCEP extension defined in this document allows to signal the IFIT capabilities. In this way IFIT methods are automatically activated and running. The flexibility and dynamicity of the IFIT applications are given by the use of additional functions on the controller and on the network nodes, but this is out of scope here.

IFIT is a solution focusing on network domains according to [RFC8799] that introduces the concept of specific domain solutions. A network domain consists of a set of network devices or entities within a single administration. As mentioned in [RFC8799], for a number of reasons, such as policies, options supported, style of network management and security requirements, it is suggested to limit

applications including the emerging IFIT techniques to a controlled domain. Hence, the IFIT methods MUST be typically deployed in such controlled domains.

The Use Case of Segment Routing (SR) is also discussed considering that IFIT methods are becoming mature for Segment Routing over the MPLS data plane (SR-MPLS) and Segment Routing over IPv6 data plane (SRv6). SR policy [I-D.ietf-spring-segment-routing-policy] is a set of candidate SR paths consisting of one or more segment lists and necessary path attributes. It enables instantiation of an ordered list of segments with a specific intent for traffic steering. The PCEP extension defined in this document also enables SR policy with native IFIT, that can facilitate the closed loop control and enable the automation of SR service.

It is to be noted the companion document [I-D.qin-idr-sr-policy-ifit] that proposes the BGP extension to enable IFIT methods for SR policy.

2. PCEP Extensions for IFIT Attributes

This document is to add IFIT attribute TLVs as PCEP Extensions. The following sections will describe the requirement and usage of different IFIT modes, and define the corresponding TLV encoding in PCEP.

The IFIT attributes here described can be generalized and included as TLVs carried inside the LSPA (LSP Attributes) object in order to be applied for all path types, as long as they support the relevant data plane telemetry method. IFIT Attributes TLVs are optional and can be taken into account by the PCE during path computation and by the PCC during path setup. In general, the LSPA object can be carried within a PCInitiate message, a PCUpd message, or a PCRpt message in the stateful PCE model.

In this document it is considered the case of SR Policy since IOAM and Alternate Marking are more mature especially for Segment Routing (SR) and for IPv6.

It is to be noted that, if it is needed to apply different IFIT methods for each Segment List, the IFIT attributes can be added into the PATH-ATTRIB object, instead of the LSPA object, according to [I-D.koldychev-pce-multipath] that defines PCEP Extensions for Signaling Multipath Information.

2.1. IFIT for SR Policies

RFC 8664 [RFC8664] and [I-D.ietf-pce-segment-routing-ipv6] specify extensions to the Path Computation Element Communication Protocol (PCEP) that allow a stateful PCE to compute and initiate Traffic-Engineering (TE) paths, as well as a Path Computation Client (PCC) to request a path subject to certain constraints and optimization criteria in SR networks both for SR-MPLS and SRv6.

IFIT attributes, here defined as TLVs for the LSPA object, complement both RFC 8664 [RFC8664], [I-D.ietf-pce-segment-routing-ipv6] and [I-D.ietf-pce-segment-routing-policy-cp].

3. IFIT capability advertisement TLV

During the PCEP initialization phase, PCEP speakers (PCE or PCC) SHOULD advertise their support of IFIT methods (e.g. IOAM and Alternate Marking).

A PCEP speaker includes the IFIT-CAPABILITY TLVs in the OPEN object to advertise its support for PCEP IFIT extensions. The presence of the IFIT-CAPABILITY TLV in the OPEN object indicates that the IFIT methods are supported.

RFC 8664 [RFC8664] and [I-D.ietf-pce-segment-routing-ipv6] define a new Path Setup Type (PST) for SR and also define the SR-PCE-CAPABILITY sub-TLV. This document defined a new IFIT-CAPABILITY TLV, that is an optional TLV for use in the OPEN Object for IFIT attributes via PCEP capability advertisement.

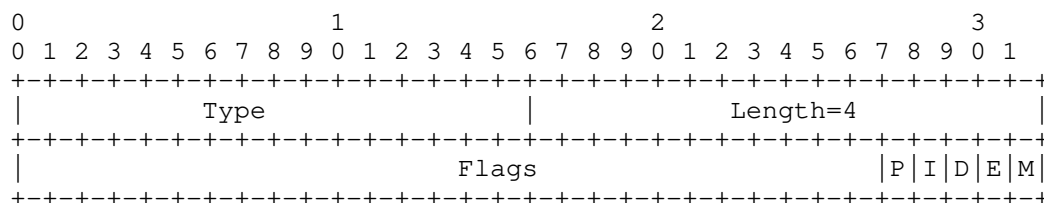


Fig. 1 IFIT-CAPABILITY TLV Format

Where:

Type: to be assigned by IANA.

Length: 4.

Flags: The following flags are defined in this document:

P: IOAM Pre-allocated Trace Option Type-enabled flag [I-D.ietf-ippm-ioam-data]. If set to 1 by a PCC, the P flag indicates that the PCC allows instantiation of the IOAM Pre-allocated Trace feature by a PCE. If set to 1 by a PCE, the P flag indicates that the PCE supports the IOAM Pre-allocated Trace feature instantiation. The P flag MUST be set by both PCC and PCE in order to support the IOAM Pre-allocated Trace instantiation

I: IOAM Incremental Trace Option Type-enabled flag [I-D.ietf-ippm-ioam-data]. If set to 1 by a PCC, the I flag indicates that the PCC allows instantiation of the IOAM Incremental Trace feature by a PCE. If set to 1 by a PCE, the I flag indicates that the PCE supports the relative IOAM Incremental Trace feature instantiation. The I flag MUST be set by both PCC and PCE in order to support the IOAM Incremental Trace feature instantiation

D: IOAM DEX Option Type-enabled flag [I-D.ietf-ippm-ioam-direct-export]. If set to 1 by a PCC, the D flag indicates that the PCC allows instantiation of the relative IOAM DEX feature by a PCE. If set to 1 by a PCE, the D flag indicates that the PCE supports the relative IOAM DEX feature instantiation. The D flag MUST be set by both PCC and PCE in order to support the IOAM DEX feature instantiation

E: IOAM E2E Option Type-enabled flag [I-D.ietf-ippm-ioam-data]. If set to 1 by a PCC, the E flag indicates that the PCC allows instantiation of the relative IOAM E2E feature by a PCE. If set to 1 by a PCE, the E flag indicates that the PCE supports the relative IOAM E2E feature instantiation. The E flag MUST be set by both PCC and PCE in order to support the IOAM E2E feature instantiation

M: Alternate Marking enabled flag RFC 8321 [RFC8321]. If set to 1 by a PCC, the M flag indicates that the PCC allows instantiation of the relative Alternate Marking feature by a PCE. If set to 1 by a PCE, the M flag indicates that the PCE supports the relative Alternate Marking feature instantiation. The M flag MUST be set by both PCC and PCE in order to support the Alternate Marking feature instantiation

Unassigned bits are considered reserved. They MUST be set to 0 on transmission and MUST be ignored on receipt.

Advertisement of the IFIT-CAPABILITY TLV implies support of IFIT methods (IOAM and/or Alternate Marking) as well as the objects, TLVs, and procedures defined in this document. It is worth mentioning that IOAM and Alternate Marking can be activated one at a time or can

coexist; so it is possible to have only IOAM or only Alternate Marking enabled but they are recognized in general as IFIT capability.

The IFIT Capability Advertisement can imply the following cases:

- o The PCEP protocol extensions for IFIT MUST NOT be used if one or both PCEP speakers have not included the IFIT-CAPABILITY TLV in their respective OPEN message.
- o A PCEP speaker that does not recognize the extensions defined in this document would simply ignore the TLVs as per RFC 5440 [RFC5440].
- o If a PCEP speaker supports the extensions defined in this document but did not advertise this capability, then upon receipt of IFIT-ATTRIBUTES TLV in the LSP Attributes (LSPA) object, it SHOULD generate a PCErr with Error-Type 19 (Invalid Operation) with the relative Error-value "IFIT capability not advertised" and ignore the IFIT-ATTRIBUTES TLV.

4. IFIT Attributes TLV

The IFIT-ATTRIBUTES TLV provides the configurable knobs of the IFIT feature, and it can be included as an optional TLV in the LSPA object (as described in RFC 5440 [RFC5440]).

For a PCE-initiated LSP RFC 8281 [RFC8281], this TLV is included in the LSPA object with the PCInitiate message. For the PCC-initiated delegated LSPs, this TLV is carried in the Path Computation State Report (PCRpt) message in the LSPA object. This TLV is also carried in the LSPA object with the Path Computation Update Request (PCUpd) message to direct the PCC (LSP head-end) to make updates to IFIT attributes.

The TLV is encoded in all PCEP messages for the LSP if IFIT feature is enabled. The absence of the TLV indicates the PCEP speaker wishes to disable the feature. This TLV includes multiple IFIT-ATTRIBUTES sub-TLVs. The IFIT-ATTRIBUTES sub-TLVs are included if there is a change since the last information sent in the PCEP message. The default values for missing sub-TLVs apply for the first PCEP message for the LSP.

The format of the IFIT-ATTRIBUTES TLV is shown in the following figure:

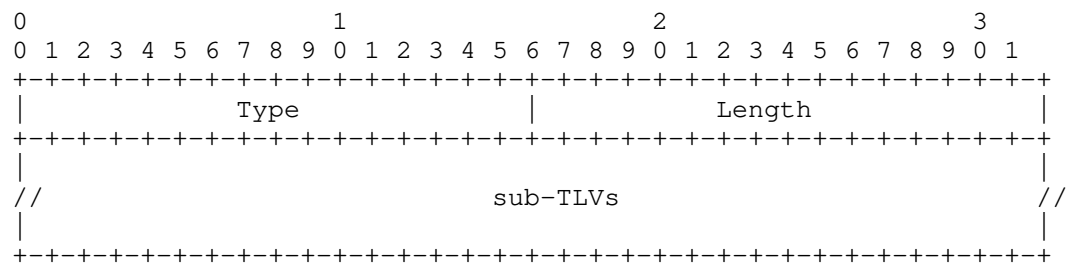


Fig. 2 IFIT-ATTRIBUTES TLV Format

Where:

Type: to be assigned by IANA.

Length: The Length field defines the length of the value portion in bytes as per RFC 5440 [RFC5440].

Value: This comprises one or more sub-TLVs.

The following sub-TLVs are defined in this document:

Type	Len	Name
1	8	IOAM Pre-allocated Trace Option
2	8	IOAM Incremental Trace Option
3	12	IOAM Directly Export Option
4	4	IOAM Edge-to-Edge Option
5	4	Enhanced Alternate Marking

Fig. 3 Sub-TLV Types of the IFIT-ATTRIBUTES TLV

4.1. IOAM Sub-TLVs

In-situ Operations, Administration, and Maintenance (IOAM) [I-D.ietf-ippm-ioam-data] records operational and telemetry information in the packet while the packet traverses a path between two points in the network. In terms of the classification given in RFC 7799 [RFC7799] IOAM could be categorized as Hybrid Type 1. IOAM mechanisms can be leveraged where active OAM do not apply or do not offer the desired results.

For the SR use case, when SR policy enables IOAM, the IOAM header will be inserted into every packet of the traffic that is steered into the SR paths. Since this document aims to define the control plane, it is to be noted that a relevant document for the data plane is [I-D.ietf-ippm-ioam-ipv6-options] for Segment Routing over IPv6 data plane (SRv6).

4.1.1. IOAM Pre-allocated Trace Option Sub-TLV

The IOAM tracing data is expected to be collected at every node that a packet traverses to ensure visibility into the entire path a packet takes within an IOAM domain. The preallocated tracing option will create pre-allocated space for each node to populate its information.

The format of IOAM pre-allocated trace option Sub-TLV is defined as follows:

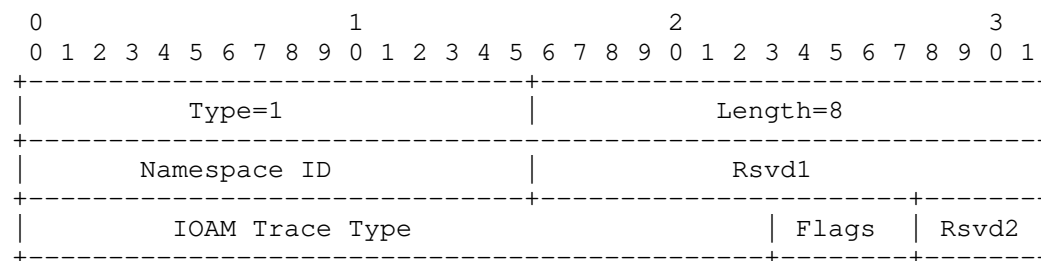


Fig. 4 IOAM Pre-allocated Trace Option Sub-TLV

Where:

Type: 1 (to be assigned by IANA).

Length: 8. It is the total length of the value field not including Type and Length fields.

Namespace ID: A 16-bit identifier of an IOAM-Namespace. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

IOAM Trace Type: A 24-bit identifier which specifies which data types are used in the node data list. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

Flags: A 4-bit field. The definition is the same as described in [I-D.ietf-ippm-ioam-flags] and section 4.4 of [I-D.ietf-ippm-ioam-data].

Rsvd1: A 16-bit field reserved for further usage. It MUST be zero and ignored on receipt.

Rsvd2: A 4-bit field reserved for further usage. It MUST be zero and ignored on receipt.

4.1.2. IOAM Incremental Trace Option Sub-TLV

The incremental tracing option contains a variable node data fields where each node allocates and pushes its node data immediately following the option header.

The format of IOAM incremental trace option Sub-TLV is defined as follows:

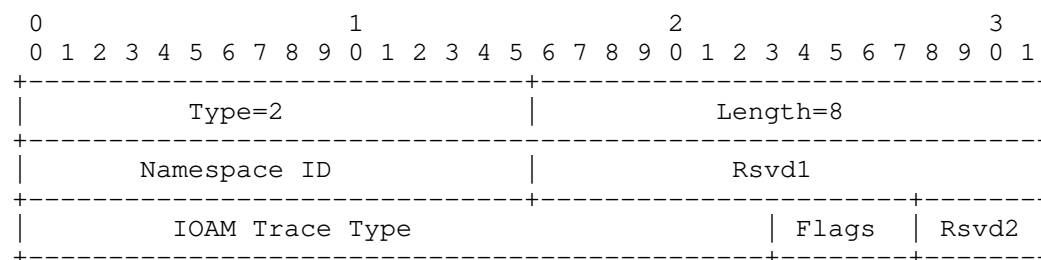


Fig. 5 IOAM Incremental Trace Option Sub-TLV

Where:

Type: 2 (to be assigned by IANA).

Length: 8. It is the total length of the value field not including Type and Length fields.

All the other fields definition is the same as the pre-allocated trace option Sub-TLV in the previous section.

4.1.3. IOAM Directly Export Option Sub-TLV

IOAM directly export option is used as a trigger for IOAM data to be directly exported to a collector without being pushed into in-flight data packets.

The format of IOAM directly export option Sub-TLV is defined as follows:

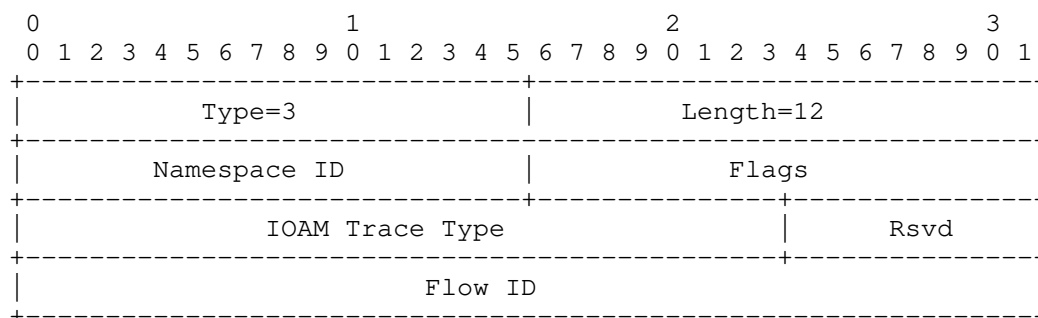


Fig. 6 IOAM Directly Export Option Sub-TLV

Where:

Type: 3 (to be assigned by IANA).

Length: 12. It is the total length of the value field not including Type and Length fields.

Namespace ID: A 16-bit identifier of an IOAM-Namespace. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

IOAM Trace Type: A 24-bit identifier which specifies which data types are used in the node data list. The definition is the same as described in section 4.4 of [I-D.ietf-ippm-ioam-data].

Flags: A 16-bit field. The definition is the same as described in section 3.2 of [I-D.ietf-ippm-ioam-direct-export].

Flow ID: A 32-bit flow identifier. The definition is the same as described in section 3.2 of [I-D.ietf-ippm-ioam-direct-export].

Rsvd: A 4-bit field reserved for further usage. It MUST be zero and ignored on receipt.

4.1.4. IOAM Edge-to-Edge Option Sub-TLV

The IOAM edge to edge option is to carry data that is added by the IOAM encapsulating node and interpreted by IOAM decapsulating node.

The format of IOAM edge-to-edge option Sub-TLV is defined as follows:

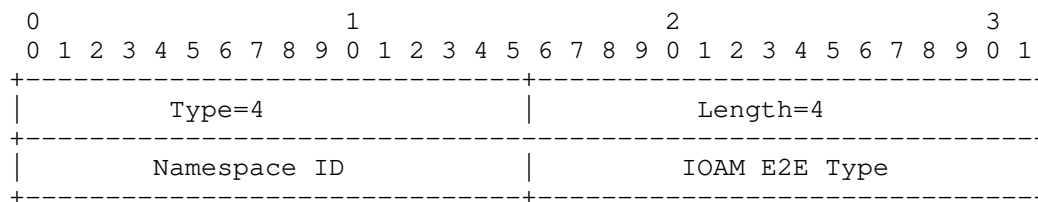


Fig. 7 IOAM Edge-to-Edge Option Sub-TLV

Where:

Type: 4 (to be assigned by IANA).

Length: 4. It is the total length of the value field not including Type and Length fields.

Namespace ID: A 16-bit identifier of an IOAM-Namespace. The definition is the same as described in section 4.6 of [I-D.ietf-ippm-ioam-data].

IOAM E2E Type: A 16-bit identifier which specifies which data types are used in the E2E option data. The definition is the same as described in section 4.6 of [I-D.ietf-ippm-ioam-data].

4.2. Enhanced Alternate Marking Sub-TLV

The Alternate Marking [RFC8321] technique is an hybrid performance measurement method, per RFC 7799 [RFC7799] classification of measurement methods. Because this method is based on marking consecutive batches of packets. It can be used to measure packet loss, latency, and jitter on live traffic.

For the SR use case, since this document aims to define the control plane, it is to be noted that a relevant document for the data plane is [I-D.ietf-6man-ipv6-alt-mark] for Segment Routing over IPv6 data plane (SRv6).

The format of Enhanced Alternate Marking (EAM) Sub-TLV is defined as follows:

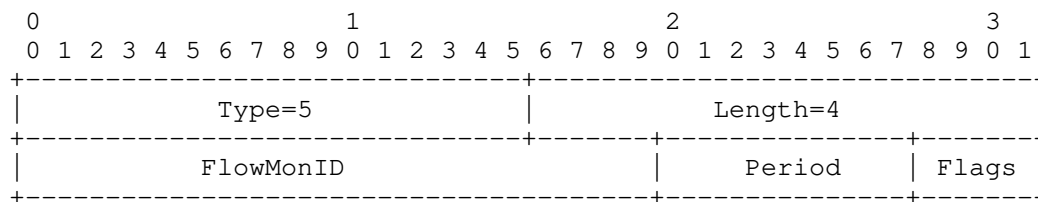


Fig. 8 Enhanced Alternate Marking Sub-TLV

Where:

Type: 5 (to be assigned by IANA).

Length: 4. It is the total length of the value field not including Type and Length fields.

FlowMonID: A 20-bit identifier to uniquely identify a monitored flow within the measurement domain. The definition is the same as described in section 5.3 of [I-D.ietf-6man-ipv6-alt-mark]. It is to be noted that PCE also needs to maintain the uniqueness of FlowMonID as described in [I-D.ietf-6man-ipv6-alt-mark].

Period: Time interval between two alternate marking period. The unit is second.

Flags: A 4-bits field. Two flags are currently assigned:

H: A flag indicating that the measurement is Hop-By-Hop.

E: A flag indicating that the measurement is End-to-End.

Unassigned bits MUST be set to zero on transmission and ignored on receipt.

5. PCEP Messages

5.1. The PCInitiate Message

A PCInitiate message is a PCEP message sent by a PCE to a PCC to trigger LSP instantiation or deletion RFC 8281 [RFC8281].

For the PCE-initiated LSP with the IFIT feature enabled, IFIT-ATTRIBUTES TLV MUST be included in the LSPA object with the PCInitiate message.

The Routing Backus-Naur Form (RBNF) definition of the PCInitiate message RFC 8281 [RFC8281] is unchanged by this document.

5.2. The PCUpd Message

A PCUpd message is a PCEP message sent by a PCE to a PCC to update the LSP parameters RFC 8231 [RFC8231].

For PCE-initiated LSPs with the IFIT feature enabled, the IFIT-ATTRIBUTES TLV MUST be included in the LSPA object with the PCUpd message. The PCE can send this TLV to direct the PCC to change the IFIT parameters.

The RBNF definition of the PCUpd message RFC 8231 [RFC8231] is unchanged by this document.

5.3. The PCRpt Message

The PCRpt message RFC 8231 [RFC8231] is a PCEP message sent by a PCC to a PCE to report the status of one or more LSPs.

For PCE-initiated LSPs RFC 8281 [RFC8281], the PCC creates the LSP using the attributes communicated by the PCE and the local values for the unspecified parameters. After the successful instantiation of the LSP, the PCC automatically delegates the LSP to the PCE and generates a PCRpt message to provide the status report for the LSP.

The RBNF definition of the PCRpt message RFC 8231 [RFC8231] is unchanged by this document.

For both PCE-initiated and PCC-initiated LSPs, when the LSP is instantiated the IFIT methods are applied as specified for the corresponding data plane. [I-D.ietf-ippm-ioam-ipv6-options] and [I-D.ietf-6man-ipv6-alt-mark] are the relevant documents for Segment Routing over IPv6 data plane (SRv6).

6. Example of application to SR Policy

A PCC or PCE sets the IFIT-CAPABILITY TLV in the Open message during the PCEP initialization phase to indicate that it supports the IFIT procedures.

[I-D.ietf-pce-segment-routing-policy-cp] defines the PCEP extension to support Segment Routing Policy Candidate Paths and in this regard the SRPAG Association object is introduced.

The Examples of PCC Initiated SR Policy with single or multiple candidate-paths and PCE Initiated SR Policy with single or multiple candidate-paths are reported in [I-D.ietf-pce-segment-routing-policy-cp].

In case of PCC Initiated SR Policy, PCC sends PCReq message to the PCE, encoding the SRPAG ASSOCIATION object and IFIT-ATTRIBUTES TLV via the LSPA object. This is valid for both single and multiple candidate-paths. Finally PCE returns the path in PCRep message, and echoes back the SRPAG object that were used in the computation and IFIT LSPA TLVs too. Additionally, PCC sends PCRpt message to the PCE, including the LSP object and the SRPAG ASSOCIATION object and IFIT-ATTRIBUTES TLV via the LSPA object. Then PCE computes path and finally PCE updates the SR policy candidate path's ERO using PCUpd message considering the IFIT LSPA TLVs too.

In case of PCE Initiated SR Policy, PCE sends PCInitiate message, containing the SRPAG Association object and IFIT-ATTRIBUTES TLV via the LSPA object. This is valid for both single and multiple candidate-paths. Then PCC uses the color, endpoint and preference from the SRPAG object to create a new candidate path considering the IFIT LSPA TLVs too. Finally PCC sends a PCRpt message back to the PCE to report the newly created Candidate Path. The PCRpt message contains the SRPAG Association object and IFIT-ATTRIBUTES information.

The procedure of enabling/disabling IFIT is simple, indeed the PCE can update the IFIT-ATTRIBUTES of the LSP by sending subsequent Path Computation Update Request (PCUpd) messages. PCE can update the IFIT-ATTRIBUTES of the LSP by sending Path Computation State Report (PCRpt) messages.

7. IANA Considerations

This document defines the new IFIT-CAPABILITY TLV and IFIT-ATTRIBUTES TLV.

7.1. PCEP TLV Type Indicators

IANA is requested to make the assignment from the "PCEP TLV Type Indicators" subregistry of the "Path Computation Element Protocol (PCEP) Numbers" registry as follows:

Value	Description	Reference
TBD1	IFIT-CAPABILITY TLV	This document
TBD2	IFIT-ATTRIBUTES TLV	This document

7.2. IFIT-CAPABILITY TLV Flags field

This document specifies the IFIT-CAPABILITY TLV 32-bits Flags field. IANA is requested to create a registry to manage the value of the IFIT-CAPABILITY TLV's Flags field within the "Path Computation Element Protocol (PCEP) Numbers" registry.

New values are to be assigned by Standards Action RFC 8126 [RFC8126]. Each bit should be tracked with the following qualities:

- * Bit number (count from 0 as the most significant bit)
- * Flag Name
- * Reference

IANA is requested to set 5 new bits in the IFIT-CAPABILITY TLV Flags Field registry, as follows:

Bit no.	Flag Name	Reference
0-26	Unassigned	This document
27	P: IOAM Pre-allocated Trace Option flag	This document
28	I: IOAM Incremental Trace Option flag	This document
29	D: IOAM Directly Export Option flag	This document
30	E: IOAM Edge-to-Edge Option	This document
31	M: Alternate Marking Flag	This document

7.3. IFIT-ATTRIBUTES Sub-TLV

This document also specifies the IFIT-ATTRIBUTES sub-TLVs. IANA is requested to create an "IFIT-ATTRIBUTES Sub-TLV Types" subregistry within the "Path Computation Element Protocol (PCEP) Numbers" registry.

IANA is requested to set the Registration Procedure for this registry to read as follows:

Range	Registration Procedure
0-65503	IETF Review
65504-65535	Experimental Use

This document defines the following types:

Type	Description	Reference
0	Reserved	This document
1	IOAM Pre-allocated Trace Option	This document
2	IOAM Incremental Trace Option	This document
3	IOAM Directly Export Option	This document
4	IOAM Edge-to-Edge Option	This document
5	Enhanced Alternate Marking	This document
6-65503	Unassigned	This document
65504-65535	Experimental Use	This document

7.4. Enhanced Alternate Marking Sub-TLV Flags field

This document specifies the Enhanced Alternate Marking Sub-TLV 4-bits Flags field. IANA is requested to create a registry to manage the value of the Enhanced Alternate Marking Sub-TLV's Flags field within the "Path Computation Element Protocol (PCEP) Numbers" registry.

New values are to be assigned by Standards Action RFC 8126 [RFC8126]. Each bit should be tracked with the following qualities:

- * Bit number (count from 0 as the most significant bit)
- * Flag Name
- * Reference

IANA is requested to set 2 new bits in the IFIT-CAPABILITY TLV Flags Field registry, as follows:

Bit no.	Flag Name	Reference
3	H: Hop-By-Hop flag	This document
2	E: End-to-End flag	This document
0-1	Unassigned	

7.5. PCEP Error Codes

This document defines a new Error-value for PCErr message of Error-Type 19 (Invalid Operation). IANA is requested to allocate a new Error-value within the "PCEP-ERROR Object Error Types and Values" subregistry of the "Path Computation Element Protocol (PCEP) Numbers" registry as follows:

Error-Type	Meaning	Error-value	Reference
19	Invalid Operation	TBD3: IFIT capability not advertised	This document

8. Security Considerations

This document defines the new IFIT-CAPABILITY TLV and IFIT Attributes TLVs, which do not add any substantial new security concerns beyond those already discussed in RFC 8231 [RFC8231] and RFC 8281 [RFC8281] for stateful PCE operations. As per RFC 8231 [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) RFC 8253 [RFC8253], as per the recommendations and best current practices in BCP 195 RFC 7525 [RFC7525] (unless explicitly set aside in RFC 8253 [RFC8253]).

Implementation of IFIT methods (IOAM and Alternate Marking) are mindful of security and privacy concerns, as explained in [I-D.ietf-ippm-ioam-data] and RFC 8321 [RFC8321]. Anyway incorrect IFIT parameters in the IFIT-ATTRIBUTES sub-TLVs SHOULD NOT have an adverse effect on the LSP as well as on the network, since it affects only the operation of the telemetry methodology.

IFIT data MUST be propagated in a limited domain in order to avoid malicious attacks and solutions to ensure this requirement are respectively discussed in [I-D.ietf-ippm-ioam-data] and [I-D.ietf-6man-ipv6-alt-mark].

IFIT methods (IOAM and Alternate Marking) are applied within a controlled domain where the network nodes are locally administered. A limited administrative domain provides the network administrator with the means to select, monitor and control the access to the network, making it a trusted domain also for the PCEP extensions defined in this document.

9. Contributors

The following people provided relevant contributions to this document:

Huanan Chen, independent, -

Dhruv Doody, Huawei Technologies, dhruv.ietf@gmail.com

10. Acknowledgements

The authors of this document would like to thank Huaimo Chen for the comments and review of this document.

11. References

11.1. Normative References

[I-D.ietf-6man-ipv6-alt-mark]

Fioccola, G., Zhou, T., Cociglio, M., Qin, F., and R. Pang, "IPv6 Application of the Alternate Marking Method", draft-ietf-6man-ipv6-alt-mark-12 (work in progress), October 2021.

[I-D.ietf-ippm-ioam-data]

Brockners, F., Bhandari, S., and T. Mizrahi, "Data Fields for In-situ OAM", draft-ietf-ippm-ioam-data-17 (work in progress), December 2021.

[I-D.ietf-ippm-ioam-direct-export]

Song, H., Gafni, B., Zhou, T., Li, Z., Brockners, F., Bhandari, S., Sivakolundu, R., and T. Mizrahi, "In-situ OAM Direct Exporting", draft-ietf-ippm-ioam-direct-export-07 (work in progress), October 2021.

[I-D.ietf-ippm-ioam-flags]

Mizrahi, T., Brockners, F., Bhandari, S., Sivakolundu, R., Pignataro, C., Kfir, A., Gafni, B., Spiegel, M., and J. Lemon, "In-situ OAM Loopback and Active Flags", draft-ietf-ippm-ioam-flags-07 (work in progress), October 2021.

- [I-D.ietf-ippm-ioam-ipv6-options]
Bhandari, S. and F. Brockners, "In-situ OAM IPv6 Options",
draft-ietf-ippm-ioam-ipv6-options-06 (work in progress),
July 2021.
- [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>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre,
"Recommendations for Secure Use of Transport Layer
Security (TLS) and Datagram Transport Layer Security
(DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May
2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC7799] Morton, A., "Active and Passive Metrics and Methods (with
Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799,
May 2016, <<https://www.rfc-editor.org/info/rfc7799>>.
- [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>>.
- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody,
"PCEPS: Usage of TLS to Provide a Secure Transport for the
Path Computation Element Communication Protocol (PCEP)",
RFC 8253, DOI 10.17487/RFC8253, October 2017,
<<https://www.rfc-editor.org/info/rfc8253>>.

- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8321] Fioccola, G., Ed., Capello, A., Cociglio, M., Castaldelli, L., Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi, "Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8321, DOI 10.17487/RFC8321, January 2018, <<https://www.rfc-editor.org/info/rfc8321>>.
- [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>>.
- [RFC8799] Carpenter, B. and B. Liu, "Limited Domains and Internet Protocols", RFC 8799, DOI 10.17487/RFC8799, July 2020, <<https://www.rfc-editor.org/info/rfc8799>>.

11.2. Informative References

- [I-D.ietf-pce-segment-routing-ipv6]
Li, C., Negl, M., Sivabalan, S., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", draft-ietf-pce-segment-routing-ipv6-11 (work in progress), January 2022.
- [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", draft-ietf-pce-segment-routing-policy-cp-06 (work in progress), October 2021.
- [I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-16 (work in progress), January 2022.
- [I-D.koldychev-pce-multipath]
Koldychev, M., Sivabalan, S., Saad, T., Beeram, V. P., Bidgoli, H., Yadav, B., and S. Peng, "PCEP Extensions for Signaling Multipath Information", draft-koldychev-pce-multipath-05 (work in progress), February 2021.

[I-D.qin-idr-sr-policy-ifit]

Qin, F., Yuan, H., Zhou, T., Fioccola, G., and Y. Wang,
"BGP SR Policy Extensions to Enable IFIT", draft-qin-idr-
sr-policy-ifit-04 (work in progress), October 2020.

Authors' Addresses

Hang Yuan
UnionPay
1899 Gu-Tang Rd., Pudong
Shanghai
China

Email: yuanhang@unionpay.com

Tianran Zhou
Huawei
156 Beiqing Rd., Haidian District
Beijing
China

Email: zhoutianran@huawei.com

Weidong Li
Huawei
156 Beiqing Rd., Haidian District
Beijing
China

Email: poly.li@huawei.com

Giuseppe Fioccola
Huawei
Riesstrasse, 25
Munich
Germany

Email: giuseppe.fioccola@huawei.com

Yali Wang
Huawei
156 Beiqing Rd., Haidian District
Beijing
China

Email: wangyalil1@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 5, 2021

Z. Li
S. Peng
X. Geng
Huawei Technologies
M. Negi
RtBrick Inc
November 1, 2020

PCEP Procedures and Protocol Extensions for Using PCE as a Central
Controller (PCECC) for P2MP LSPs
draft-dhody-pce-pcep-extension-pce-controller-p2mp-05

Abstract

The Path Computation Element (PCE) is a core component of Software-Defined Networking (SDN) systems. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands.

The PCE has been identified as an appropriate technology for the determination of the paths of point-to-multipoint (P2MP) TE Label Switched Paths (LSPs).

PCE was developed to derive paths for MPLS P2MP LSPs, which are supplied to the head end (root) of the LSP using PCEP. PCEP has been proposed as a control protocol to allow the PCE to be fully enabled as a central controller.

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. Thus, the P2MP LSP can be calculated/set up/initiated and the label forwarding entries can also be downloaded through a centralized PCE server to each network device along the P2MP path, while leveraging the existing PCE technologies as much as possible.

This document specifies the procedures and PCEP extensions for using the PCE as the central controller for P2MP TE LSP.

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 May 5, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	5
2. Terminology	5
3. Basic PCECC Mode	5
4. Procedures for Using the PCE as a Central Controller (PCECC) for P2MP	6
4.1. Stateful PCE Model	6
4.2. PCECC Capability Advertisement	6
4.3. LSP Operations	6
4.3.1. PCE-Initiated PCECC LSP	7
4.3.2. PCC-Initiated PCECC LSP	7
4.3.3. Central Control Instructions	8
4.3.3.1. Label Download CCI	8
4.3.3.2. Label Clean up CCI	9
4.3.4. PCECC LSP Update	10
4.3.5. Re-Delegation and Clean up	10
4.3.6. Synchronization of Central Controllers Instructions	10
4.3.7. PCECC LSP State Report	10
4.3.8. PCC-Based Allocations	10
5. PCEP Messages	10
6. PCEP Objects	10

6.1.	OPEN Object	10
6.1.1.	PCECC Capability sub-TLV	10
6.2.	PATH-SETUP-TYPE TLV	11
6.3.	CCI Object	11
7.	Security Considerations	12
8.	Manageability Considerations	12
8.1.	Control of Function and Policy	12
8.2.	Information and Data Models	12
8.3.	Liveness Detection and Monitoring	12
8.4.	Verify Correct Operations	12
8.5.	Requirements On Other Protocols	12
8.6.	Impact On Network Operations	13
9.	IANA Considerations	13
9.1.	PCECC-CAPABILITY sub-TLV	13
9.2.	PCEP-Error Object	13
10.	Acknowledgments	13
11.	References	13
11.1.	Normative References	13
11.2.	Informative References	14
Appendix A.	Contributor Addresses	17
Authors'	Addresses	17

1. Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload the path computation function from routers in an MPLS traffic-engineered network. Since then, the role and function of the PCE has grown to cover a number of other uses (such as GMPLS [RFC7025]) and to allow delegated control [RFC8231] and PCE-initiated use of network resources [RFC8281].

According to [RFC7399], Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system, called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place traffic flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths for MPLS Label Switched Paths (LSPs), paths were requested by network elements (known as Path Computation Clients (PCCs)), and the results

of the path computations were supplied to network elements using the Path Computation Element Communication Protocol (PCEP) [RFC5440]. This protocol was later extended to allow a PCE to send unsolicited requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE and PCC. [RFC8283] further examines the motivations and applicability for PCEP as a Southbound Interface (SBI), and introduces the implications for the protocol.

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. Thus, the LSP can be calculated/setup/initiated and the label forwarding entries can also be downloaded through a centralized PCE server to each network devices along the path while leveraging the existing PCE technologies as much as possible.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify the procedures and PCEP extensions for using the PCE as the central controller for static P2P LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what label-forwarding instructions to program and what resources to reserve. The PCE-based controller keeps a view of the network and determines the paths of the end-to-end LSPs, and the controller uses PCEP to communicate with each router along the path of the end-to-end LSP.

[RFC4857] describes how to set up point-to-multipoint (P2MP) Traffic Engineering Label Switched Paths (TE LSPs) for use in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks. The PCE has been identified as a suitable application for the computation of paths for P2MP TE LSPs ([RFC5671]). The extensions of PCEP to request path computation for P2MP TE LSPs are described in [RFC8306]. Further [RFC8623] specify the extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs as well as the setup, maintenance and teardown of PCE-initiated P2MP LSPs under the stateful PCE model.

This document extends

[I-D.ietf-pce-pcep-extension-for-pce-controller] to specify the procedures and PCEP extensions for using the PCE as the central controller for static P2MP LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path with an added functionality of a P2MP branch node. As per [RFC4875], a branch node is an LSR that replicates the incoming data on to one or

more outgoing interfaces. [I-D.ietf-teas-pcecc-use-cases] describes the use cases for P2MP in PCECC architecture.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

Terminologies used in this document is the same as described in the draft [RFC8283] and [I-D.ietf-teas-pcecc-use-cases].

3. Basic PCECC Mode

As described in [I-D.ietf-pce-pcep-extension-for-pce-controller], in this mode LSPs are provisioned as explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what label forwarding instructions to program and what resources to reserve. The controller uses PCEP to communicate with each router along the path of the end-to-end LSP.

Note that the PCE-based controller will take responsibility for managing some part of the MPLS label space for each of the routers that it controls, and may take wider responsibility for partitioning the label space for each router and allocating different parts for different uses. This is also described in section 3.1.2. of [RFC8283]. For the purpose of this document, it is assumed that the label range to be used by a PCE is known and set on both PCEP peers. A future extension could add the capability to advertise the range via possible PCEP extensions as well (see [I-D.li-pce-controlled-id-space]). The rest of the processing is similar to the existing stateful PCE mechanism.

This document extends the functionality to include support for central control instruction for replication at the branch nodes.

The rest of the processing is similar to the existing stateful PCE mechanism for P2MP [RFC8623].

4. Procedures for Using the PCE as a Central Controller (PCECC) for P2MP

4.1. Stateful PCE Model

Active stateful PCE is described in [RFC8231] and extended for P2MP [RFC8623]. PCE as a Central Controller (PCECC) reuses the existing active stateful PCE mechanism as much as possible to control the LSPs.

[I-D.ietf-pce-pcep-extension-for-pce-controller] extends PCEP messages - PCRpt, PCInitiate message for the Central Controller's Instructions (CCI) (label forwarding instructions in the context of this document). This document specifies the procedure for additional instruction for branch node needed for P2MP.

4.2. PCECC Capability Advertisement

As per [I-D.ietf-pce-pcep-extension-for-pce-controller], during PCEP Initialization Phase, PCEP Speakers (PCE or PCC) advertise their support of the PCECC extensions by sending a PATH-SETUP-TYPE-CAPABILITY TLV in the OPEN object with this PST=TBD [I-D.ietf-pce-pcep-extension-for-pce-controller] included in the PST list.

[I-D.ietf-pce-pcep-extension-for-pce-controller] also defines the PCECC Capability sub-TLV. A new M-bit is added in the PCECC-CAPABILITY sub-TLV to indicate support for PCECC-P2MP. A PCC MUST set M-bit in the PCECC-CAPABILITY sub-TLV and include STATEFUL-PCE-CAPABILITY TLV with the P2MP bits set ([RFC8623]) in the OPEN Object to support the PCECC P2MP extensions defined in this document. If the M-bit is set in PCECC-CAPABILITY sub-TLV and N-bit in the STATEFUL-PCE-CAPABILITY TLV is not set in the OPEN Object, the PCE SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD2 (P2MP capability was not advertised) and terminate the session.

The rest of the processing is as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

4.3. LSP Operations

The PCEP messages pertaining to a PCECC includes the PATH-SETUP-TYPE TLV [RFC8408] with PST=TBD [I-D.ietf-pce-pcep-extension-for-pce-controller] in the SRP object to identify the PCECC LSP is intended as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

4.3.1. PCE-Initiated PCECC LSP

The LSP Instantiation operation is the same as defined in [RFC8281] and [RFC8623].

In order to set up a PCE-Initiated P2MP LSP based on the PCECC mechanism, a PCE sends PCInitiate message with Path Setup Type set to TBD for PCECC ([I-D.ietf-pce-pcep-extension-for-pce-controller]) to the ingress PCC (root node).

P2MP-LSP-IDENTIFIER TLV [RFC8623] MUST be included for the PCECC P2MP LSPs, the tuple uniquely identifies the P2MP LSP in the network. As per [I-D.ietf-pce-pcep-extension-for-pce-controller], the LSP object is included in the central controller's instructions (label download) to identify the PCECC P2MP LSP for this instruction. The handling of PLSP-ID is as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

The ingress PCC MUST also set D (Delegate) flag (see [RFC8231]) and C (Create) flag (see [RFC8281]) in the LSP object of the PCRpt message. The PCC responds with a PCRpt message with the status set to "GOING-UP" and carrying the assigned PLSP-ID. As per [I-D.ietf-pce-pcep-extension-for-pce-controller] when the PCE receives this PCRpt message with the PLSP-ID, it assigns labels along the path; and sets up the path by sending a PCInitiate message to each node along the path of the P2MP Tree as per the PCECC technique. The CC-ID uniquely identifies the central controller instruction within a PCEP session. Each PCC further responds with the PCRpt messages including the central controller instruction (CCI) and the LSP objects.

As described in [I-D.ietf-pce-pcep-extension-for-pce-controller], the label forwarding instructions from PCECC are sent after the initial PCInitiate and PCRpt exchange. This is done so that the PLSP-ID and other LSP identifiers can be obtained from the ingress and can be included in the label forwarding instruction in the next PCInitiate message.

4.3.2. PCC-Initiated PCECC LSP

In order to set up a P2MP LSP based on the PCECC mechanism where the LSP is configured at the PCC, a PCC MUST delegate the P2MP LSP by sending a PCRpt message with PST set for PCECC and D (Delegate) flag (see [RFC8623]) set in the LSP object.

When a PCE receives the initial PCRpt message with the D flags and PST Type set to TBD, it SHOULD calculate the P2MP tree and assigns labels along the P2MP tree; and set up the P2MP LSP by sending PCInitiate message to each node along the path of the P2MP LSP as per

[I-D.ietf-pce-pcep-extension-for-pce-controller]. The new extension required is the instructions on the branch nodes for replications to more than one outgoing interface with the respective label. The rest of the operations remains the same as [I-D.ietf-pce-pcep-extension-for-pce-controller] and [RFC8623].

4.3.3. Central Control Instructions

The new central controller's instructions (CCI) for the label operations in PCEP is done via the PCInitiate message as described in [I-D.ietf-pce-pcep-extension-for-pce-controller], by defining a new PCEP Objects for CCI operations. The local label range of each PCC is assumed to be known by both the PCC and the PCE.

4.3.3.1. Label Download CCI

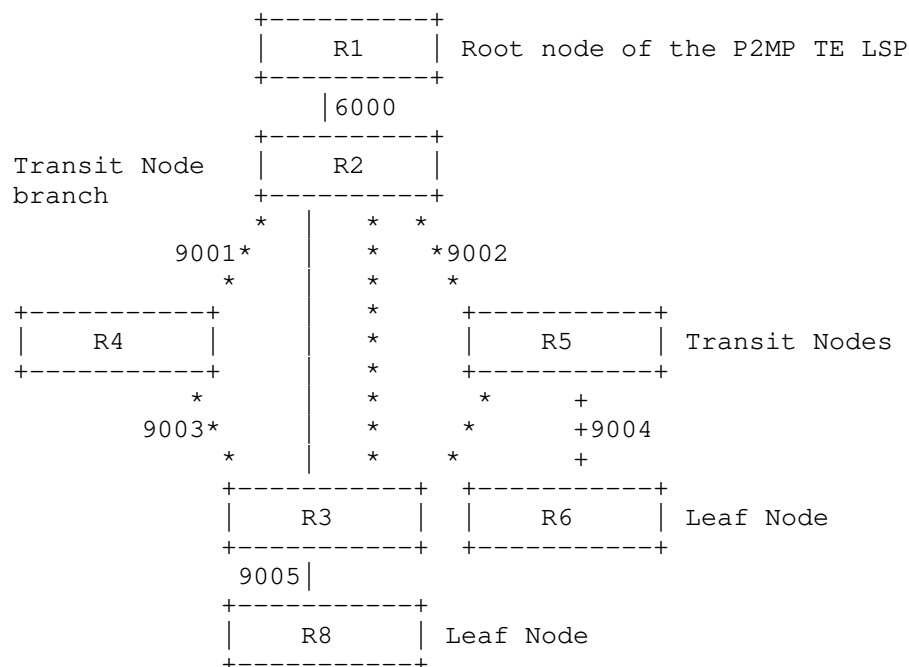
In order to set up an LSP based on PCECC, the PCE sends a PCInitiate message to each node along the path to download the Label instruction as described in Section 4.3.1 and Section 4.3.2.

The CCI object MUST be included, along with the LSP object in the PCInitiate message. As per [I-D.ietf-pce-pcep-extension-for-pce-controller], there are at most 2 instances of CCI object in the PCInitiate message. For PCECC-P2MP operations, multiple instances of CCI object for out-labels is allowed. Similarly to acknowledge the central controller instructions, the PCRpt message allows multiple instances of CCI object for PCECC-P2MP operations.

The LSP-IDENTIFIER TLV MUST be included in the LSP object. The SPEAKER-ENTITY-ID TLV SHOULD be included in LSP object.

As described in [I-D.ietf-pce-pcep-extension-for-pce-controller], if a node (PCC) receives a PCInitiate message which includes a Label to download, as part of CCI, that is out of the range set aside for the PCE, it send a PCErr message with Error-type=TBD (PCECC failure) and Error-value=TBD (Label out of range) (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]). If a PCC receives a PCInitiate message but fails to download the Label entry, it sends a PCErr message with Error-type=TBD (PCECC failure) and Error-value=TBD (Instruction failed) (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]).

Consider the example in the [I-D.ietf-teas-pcecc-use-cases] -



PCECC would provision each node along the path and assign incoming and outgoing labels from R1 to {R6, R8} with the path: {R1, 6000}, {6000, R2, {9001,9002}}, {9001, R4, 9003}, {9002, R5, 9004} {9003, R3, 9005}, {9004, R6}, {9005, R8}. The operations on all nodes except R2 are same as [I-D.ietf-pce-pcep-extension-for-pce-controller]. The branch node (R2) needs to be instructed to replicate two copies of the incoming packet, and sent towards R4 and R5 with 9001 and 9002 labels respectively). This done via including 3 instances of CCI objects in the PCEP messages, one for each label in the example, 6000 for incoming and 9001/9002 for outgoing (along with remote nexthop). The message and procedure remains exactly as [I-D.ietf-pce-pcep-extension-for-pce-controller] with only distinction that more than one outgoing CCI MAY be present for the P2MP LSP.

4.3.3.2. Label Clean up CCI

In order to delete a P2MP LSP based on PCECC, the PCE sends a central controller instructions via a PCInitiate message to each node along the path of the P2MP tree to clean up the Label forwarding instruction as per [I-D.ietf-pce-pcep-extension-for-pce-controller]. In case of branch nodes, all instances of CCIs needs to be present in the PCEP message.

4.3.4. PCECC LSP Update

In case of a modification of PCECC P2MP LSP with a new path, the procedure, and instructions as described in [I-D.ietf-pce-pcep-extension-for-pce-controller] apply.

4.3.5. Re-Delegation and Clean up

In case of a re-delegation and clean up of PCECC P2MP LSP, the procedure, and instructions as described in [I-D.ietf-pce-pcep-extension-for-pce-controller] apply.

4.3.6. Synchronization of Central Controllers Instructions

The procedure and instructions are as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

4.3.7. PCECC LSP State Report

An ingress PCC MAY choose to apply any OAM mechanism to check the status of LSP in the Data plane and MAY further send its status in PCRpt message (as per [RFC8623]) to the PCE.

4.3.8. PCC-Based Allocations

The PCE can request the PCC to allocate the label using the PCInitiate message. The procedure and instructions are as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

5. PCEP Messages

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify the extension to PCInitiate and PCRpt message for PCECC. For P2P LSP, only two instances of CCI objects can be included. In the case of the P2MP LSP, multiple CCI objects are allowed. The message format and other procedures continue to apply.

6. PCEP Objects

6.1. OPEN Object

6.1.1. PCECC Capability sub-TLV

The PCECC-CAPABILITY sub-TLV is an optional TLV for use in the OPEN Object for PCECC capability advertisement in PATH-SETUP-TYPE-CAPABILITY TLV as specified in [I-D.ietf-pce-pcep-extension-for-pce-controller].

This document adds a new flag (M-bit) in the PCECC-CAPABILITY sub-TLV to indicate the support for P2MP in PCECC.

M (PCECC-P2MP-CAPABILITY - 1 bit - TBD1): If set to 1 by a PCEP speaker, it indicates that the PCEP speaker is capable of PCECC-P2MP capability.

A PCC MUST set the M-bit in the PCECC-CAPABILITY sub-TLV and set the N (P2MP-CAPABILITY), the M (P2MP-LSP-UPDATE-CAPABILITY), and the P (P2MP-LSP-INSTITUTION-CAPABILITY) bits (as per [RFC8623]) in the STATEFUL-PCE-CAPABILITY TLV [RFC8231] to support the PCECC-P2MP extensions defined in this document. If the M-bit is set in PCECC-CAPABILITY sub-TLV and the P2MP bits (in the STATEFUL-PCE-CAPABILITY TLV) are not set in the OPEN Object, a PCEP speaker SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD2 (P2MP capability was not advertised) and terminate the session.

6.2. PATH-SETUP-TYPE TLV

The PATH-SETUP-TYPE TLV is defined in [RFC8408]; [I-D.ietf-pce-pcep-extension-for-pce-controller] defines a PST value for PCECC, which is applicable for P2MP LSP as well.

6.3. CCI Object

The Central Control Instructions (CCI) Object [I-D.ietf-pce-pcep-extension-for-pce-controller] is used by the PCE to specify the forwarding instructions (Label information in the context of this document) to the PCC, and optionally carried within PCInitiate or PCRpt message for label download/report. The CCI Object Type 1 for MPLS Label is defined in [I-D.ietf-pce-pcep-extension-for-pce-controller], which is used for the P2MP LSPs as well. The address TLVs are defined in [I-D.ietf-pce-pcep-extension-for-pce-controller], they associate the next-hop information in case of an outgoing label.

If a node (PCC) receives a PCInitiate message with more than one CCI with O-bit set for the outgoing label and the node does not support the P2MP branch/replication capability, it MUST respond with PCErr message with Error-Type=2 (Capability not supported) (defined in [RFC5440]).

The rest of the processing is same as [I-D.ietf-pce-pcep-extension-for-pce-controller].

7. Security Considerations

The security considerations described in [RFC8231], [RFC8281], [RFC8623], and [I-D.ietf-pce-pcep-extension-for-pce-controller] apply to the extensions described in this document.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

8. Manageability Considerations

8.1. Control of Function and Policy

A PCE or PCC implementation SHOULD allow to configure to enable/disable PCECC-P2MP capability as a global configuration.

8.2. Information and Data Models

[RFC7420] describes the PCEP MIB, this MIB can be extended to get the PCECC capability status.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to enable/disable PCECC-P2MP capability.

8.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

8.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

8.5. Requirements On Other Protocols

PCEP extensions defined in this document do not put new requirements on other protocols.

8.6. Impact On Network Operations

PCEP extensions defined in this document do not put new requirements on network operations.

9. IANA Considerations

9.1. PCECC-CAPABILITY sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defines the PCECC-CAPABILITY sub-TLV and requests that IANA creates a registry to manage the value of the PCECC-CAPABILITY sub-TLV's Flag field. IANA is requested to allocate a new bit in the PCECC-CAPABILITY sub-TLV Flag Field registry, as follows:

Bit	Description	Reference
TBD1	P2MP	This document

9.2. PCEP-Error Object

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

Error-Type	Meaning
-----	-----
19	Invalid operation.
Error-value = TBD2 :	
	P2MP capability was not advertised

10. Acknowledgments

11. References

11.1. Normative References

- [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>>.

- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8623] Palle, U., Dhody, D., Tanaka, Y., and V. Beeram, "Stateful Path Computation Element (PCE) Protocol Extensions for Usage with Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 8623, DOI 10.17487/RFC8623, June 2019, <<https://www.rfc-editor.org/info/rfc8623>>.
- [I-D.ietf-pce-pcep-extension-for-pce-controller]
Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-07 (work in progress), September 2020.

11.2. Informative References

- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC4857] Fogelstroem, E., Jonsson, A., and C. Perkins, "Mobile IPv4 Regional Registration", RFC 4857, DOI 10.17487/RFC4857, June 2007, <<https://www.rfc-editor.org/info/rfc4857>>.

- [RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <<https://www.rfc-editor.org/info/rfc4875>>.
- [RFC5671] Yasukawa, S. and A. Farrel, Ed., "Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)", RFC 5671, DOI 10.17487/RFC5671, October 2009, <<https://www.rfc-editor.org/info/rfc5671>>.
- [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C. Margaria, "Requirements for GMPLS Applications of PCE", RFC 7025, DOI 10.17487/RFC7025, September 2013, <<https://www.rfc-editor.org/info/rfc7025>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <<https://www.rfc-editor.org/info/rfc7420>>.
- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.

- [RFC8306] Zhao, Q., Dhody, D., Ed., Palleti, R., and D. King, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 8306, DOI 10.17487/RFC8306, November 2017, <<https://www.rfc-editor.org/info/rfc8306>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [I-D.ietf-teas-pcecc-use-cases]
Li, Z., Khasanov, B., Dhody, D., Zhao, Q., Ke, Z., Fang, L., Zhou, C., Communications, T., Rachitskiy, A., and A. Gulida, "The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).", draft-ietf-teas-pcecc-use-cases-06 (work in progress), September 2020.
- [I-D.ietf-pce-pcep-yang]
Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-14 (work in progress), July 2020.
- [I-D.li-pce-controlled-id-space]
Li, C., Chen, M., Wang, A., Cheng, W., and C. Zhou, "PCE Controlled ID Space", draft-li-pce-controlled-id-space-07 (work in progress), October 2020.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Udayasree Palle

EMail: udayasreereddy@gmail.com

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: pengshuping@huawei.com

Xuesong Geng
Huawei Technologies
China

EMail: gengxuesong@huawei.com

Mahendra Singh Negi
RtBrick Inc
N-17L, 18th Cross Rd, HSR Layout
Bangalore, Karnataka 560102
India

EMail: mahend.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 7 September 2022

Z. Li
S. Peng
X. Geng
Huawei Technologies
M. Negi
RtBrick Inc
6 March 2022

Path Computation Element Communication Protocol (PCEP) Procedures and
Extensions for Using the PCE as a Central Controller (PCECC) of point-
to-multipoint (P2MP) LSPs
draft-dhody-pcep-extension-pce-controller-p2mp-08

Abstract

The Path Computation Element (PCE) is a core component of Software-Defined Networking (SDN) systems.

The PCE has been identified as an appropriate technology for the determination of the paths of point-to-multipoint (P2MP) TE Label Switched Paths (LSPs).

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. Thus, the P2MP LSP can be calculated/set up/initiated and the label-forwarding entries can also be downloaded through a centralized PCE server to each network device along the P2MP path, while leveraging the existing PCE technologies as much as possible.

This document specifies the procedures and Path Computation Element Communication Protocol (PCEP) extensions for using the PCE as the central controller for provisioning labels along the path of the static P2MP LSP.

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 7 September 2022.

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	3
2. Terminology	5
2.1. Requirements Language	5
3. Basic PCECC Mode	5
4. Procedures for Using the PCE as a Central Controller (PCECC) for P2MP	5
4.1. Stateful PCE Model	5
4.2. PCECC Capability Advertisement	6
4.3. LSP Operations	6
4.3.1. PCE-Initiated PCECC LSP	6
4.3.2. PCC-Initiated PCECC LSP	7
4.3.3. Central Control Instructions	7
4.3.3.1. Label Download CCI	7
4.3.3.2. Label Cleanup CCI	9
4.3.4. PCECC LSP Update	9
4.3.5. Re-delegation and Cleanup	9
4.3.6. Synchronization of Central Controllers Instructions	9
4.3.7. PCECC LSP State Report	9
4.3.8. PCC-Based Allocations	9
5. PCEP Messages	10
6. PCEP Objects	10
6.1. OPEN Object	10
6.1.1. PCECC Capability sub-TLV	10
6.2. PATH-SETUP-TYPE TLV	10

6.3. CCI Object	10
7. Security Considerations	11
8. Manageability Considerations	11
8.1. Control of Function and Policy	11
8.2. Information and Data Models	11
8.3. Liveness Detection and Monitoring	12
8.4. Verify Correct Operations	12
8.5. Requirements On Other Protocols	12
8.6. Impact On Network Operations	12
9. IANA Considerations	12
9.1. PCECC-CAPABILITY sub-TLV	12
9.2. PCEP-Error Object	12
10. References	13
10.1. Normative References	13
10.2. Informative References	14
Appendix A. Contributor Addresses	15
Authors' Addresses	16

1. Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload the path computation function from routers in an MPLS traffic-engineered (TE) network. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands. Since then, the role and function of the PCE have grown to cover a number of other uses (such as GMPLS [RFC7025]) and to allow delegated control [RFC8231] and PCE-initiated use of network resources [RFC8281].

According to [RFC7399], Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system, called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place traffic flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths for MPLS Label Switched Paths (LSPs), paths were requested by network elements (known as Path Computation Clients (PCCs)), and the results of the path computations were supplied to network elements using the Path Computation Element Communication Protocol (PCEP) [RFC5440]. This protocol was later extended to allow a PCE to send unsolicited requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE and PCC. [RFC8283] further examines the motivations and applicability for PCEP as a Southbound Interface (SBI), and introduces the implications for the protocol.

A PCECC can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. Thus, the LSP can be calculated/set up/initiated and the label-forwarding entries can also be downloaded through a centralized PCE server to each network device along the path while leveraging the existing PCE technologies as much as possible.

[RFC9050] specify the procedures and PCEP extensions for using the PCE as the central controller for static P2P LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what label-forwarding instructions to program and what resources to reserve. The PCE-based controller keeps a view of the network and determines the paths of the end-to-end LSPs, and the controller uses PCEP to communicate with each router along the path of the end-to-end LSP.

[RFC4857] describes how to set up point-to-multipoint (P2MP) Traffic Engineering Label Switched Paths (TE LSPs) for use in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks. The PCE has been identified as a suitable application for the computation of paths for P2MP TE LSPs ([RFC5671]). The extensions of PCEP to request path computation for P2MP TE LSPs are described in [RFC8306]. Further [RFC8623] specify the extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs as well as the setup, maintenance and teardown of PCE-initiated P2MP LSPs under the stateful PCE model.

This document extends [RFC9050] to specify the procedures and PCEP extensions for using the PCE as the central controller for static P2MP LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path with an added functionality of a P2MP branch node. As per [RFC4875], a branch node is an LSR that replicates the incoming data on to one or more outgoing interfaces. [I-D.ietf-teas-pcecc-use-cases] describes the use cases for P2MP in PCECC architecture.

2. Terminology

Terminologies used in this document is the same as described in the draft [RFC8283].

2.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Basic PCECC Mode

Section 3 of [RFC9050] describe the PCECC model of operation.

This document extends the functionality to include support for central control instruction for replication at the branch nodes for the P2MP LSP.

The rest of the processing at the root node is similar to the existing stateful PCE mechanism for P2MP [RFC8623].

4. Procedures for Using the PCE as a Central Controller (PCECC) for P2MP

4.1. Stateful PCE Model

Active stateful PCE is described in [RFC8231] and extended for P2MP [RFC8623]. A PCE as a Central Controller (PCECC) reuses the existing active stateful PCE mechanism as much as possible to control the LSPs.

[RFC9050] extends PCEP messages - PCInitiate, PCRpt, and PCUpd message for the Central Controller's Instructions (CCI) (label-forwarding instructions in the context of this document). This document specify the procedure for additional instruction for branch node needed for P2MP.

4.2. PCECC Capability Advertisement

As per Section 5.4 of [RFC9050], during the PCEP initialization phase, PCEP Speakers (PCE or PCC) advertise their support of and willingness to use PCEP extension for the PCECC using a new Path Setup Type (PST) in PATH-SETUP-TYPE-CAPABILITY TLV and a new PCECC-CAPABILITY sub-TLV.

A new M bit is added in the PCECC-CAPABILITY sub-TLV to indicate support for PCECC-P2MP. A PCC MUST set the M bit in the PCECC-CAPABILITY sub-TLV and include STATEFUL-PCE-CAPABILITY TLV with the P2MP bits set (as per [RFC8623]) in the OPEN object to support the PCECC P2MP extensions defined in this document.

If the M bit is set in PCECC-CAPABILITY sub-TLV and the STATEFUL-PCE-CAPABILITY TLV is not advertised, or is advertised without the N bit set, in the OPEN object, the receiver MUST:

- * send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD2 (P2MP capability was not advertised) and
- * terminate the session.

The rest of the processing is as per [RFC9050].

4.3. LSP Operations

The PCEP messages pertaining to a PCECC includes the PATH-SETUP-TYPE TLV [RFC8408] in the SRP object [RFC8231] with the PST set to '2' to clearly identify the the PCECC LSP is intended as per [RFC9050].

4.3.1. PCE-Initiated PCECC LSP

The LSP instantiation operation is the same as defined in [RFC8281] and [RFC8623].

In order to set up a PCE-Initiated P2MP LSP based on the PCECC mechanism, a PCE sends a PCInitiate message with the PST set to '2' for the PCECC ([RFC9050]) to the ingress PCC (root node).

As described in [RFC9050], the label-forwarding instructions from PCECC are sent after the initial PCInitiate and PCRpt exchange. This is done so that the PCEP-specific identifier for the LSP (PLSP-ID) and other LSP identifiers can be obtained from the ingress and can be included in the label-forwarding instruction in the next set of PCInitiate message along the path.

An P2MP-LSP-IDENTIFIER TLV [RFC8623] MUST be included for the PCECC P2MP LSPs, it uniquely identifies the P2MP LSP in the network. As per [RFC9050], the LSP object is included in the central controller's instructions (label download) to identify the PCECC P2MP LSP for this instruction. The handling of PLSP-ID is as per [RFC9050].

The ingress PCC (root) also sets the D (Delegate) flag (see [RFC8231]) and C (Create) flag (see [RFC8281]) in the LSP object of the PCRpt message. As per [RFC9050], when the PCE receives this PCRpt message with the PLSP-ID, it assigns labels along the path and sets up the path by sending a PCInitiate message to each node along the path of the P2MP Tree as per the PCECC technique. The CC-ID uniquely identifies the central controller instruction within a PCEP session. Each node along the path (PCC) responds with the PCRpt messages to acknowledge the CCI with the PCRpt messages including the CCI and the LSP objects. The only new extension required is the instructions on the branch nodes for replications to more than one outgoing interface with the respective label. The rest of the operations remains the same as [RFC9050] and [RFC8623].

4.3.2. PCC-Initiated PCECC LSP

In order to set up a P2MP LSP based on the PCECC mechanism where the LSP is configured at the PCC, a PCC MUST delegate the P2MP LSP by sending a PCRpt message with the PST set for the PCECC and D (Delegate) flag (see [RFC8623]) set in the LSP object.

When a PCE receives the initial PCRpt message with the D flags and PST Type set to '2', it SHOULD calculate the P2MP tree and assign labels along the P2MP tree in addition to setting up the P2MP LSP by sending PCInitiate message to each node along the path of the P2MP LSP as per [RFC9050]. The only new extension required is the instructions on the branch nodes for replications to more than one outgoing interface with the respective label. The rest of the operations remains the same as [RFC9050] and [RFC8623].

4.3.3. Central Control Instructions

The CCI for the label operations in PCEP are done via the PCInitiate message as described in [RFC9050], by defining a PCEP Objects for CCI operations. The local label range of each PCC is assumed to be known by both the PCC and the PCE.

4.3.3.1. Label Download CCI

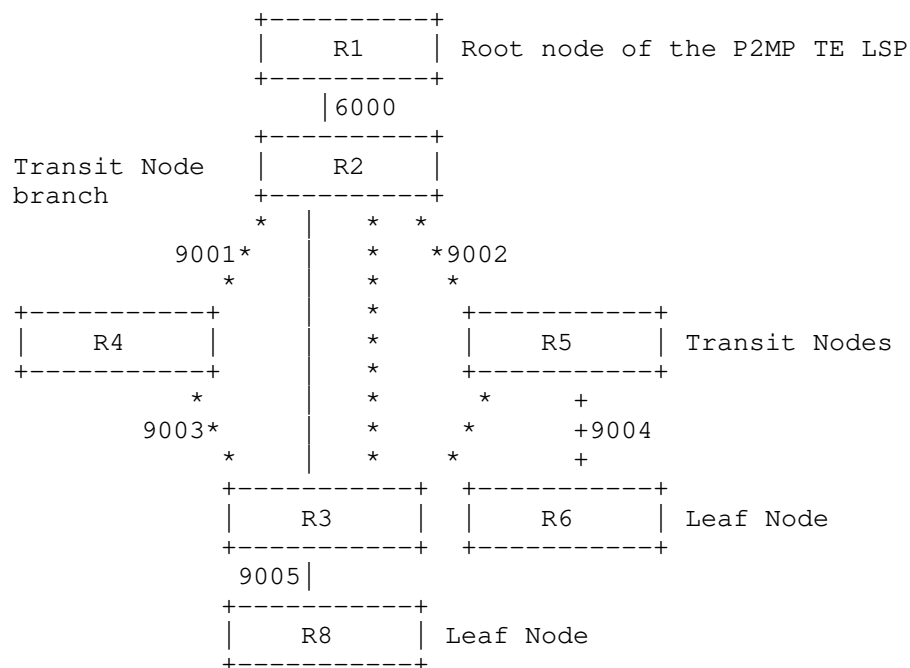
In order to set up an LSP based on the PCECC, the PCE sends a PCInitiate message to each node along the path to download the label instructions, as described in Section 4.3.1 and Section 4.3.2.

The CCI object MUST be included, along with the LSP object in the PCInitiate message. As per [RFC9050], there are at most 2 instances of CCI object in the PCInitiate message. For PCECC-P2MP operations, multiple instances of CCI object for out-labels is allowed. Similarly to acknowledge the central controller instructions, the PCRpt message allows multiple instances of CCI object for PCECC-P2MP operations.

The P2MP-LSP-IDENTIFIERS TLV MUST be included in the LSP object for the PCECC based P2MP LSP. The SPEAKER-ENTITY-ID TLV SHOULD be included in LSP object.

As described in [RFC9050], if a node (PCC) receives a PCInitiate message that includes a label to download (as part of CCI) that is out of the range set aside for the PCE, it send a PCErr message with Error-type=3 (PCECC failure) and Error-value=1 (Label out of range) ([RFC9050]). If a PCC receives a PCInitiate message but fails to download the label entry, it sends a PCErr message with Error-type=3 (PCECC failure) and Error-value=2 (Instruction failed) ([RFC9050]).

Consider the example in the [I-D.ietf-teas-pcecc-use-cases] -



PCECC would provision each node along the path and assign incoming and outgoing labels from R1 to {R6, R8} with the path: {R1, 6000}, {6000, R2, {9001,9002}}, {9001, R4, 9003}, {9002, R5, 9004} {9003, R3, 9005}, {9004, R6}, {9005, R8}. The operations on all nodes except R2 are same as [RFC9050]. The branch node (R2) needs to be instructed to replicate two copies of the incoming packet, and sent towards R4 and R5 with 9001 and 9002 labels respectively). This done via including 3 instances of CCI objects in the PCEP messages, one for each label in the example, 6000 for incoming and 9001/9002 for outgoing (along with remote nexthop). The message and procedure remains exactly as [RFC9050] with only distinction that more than one outgoing CCI MAY be present for the P2MP LSP.

4.3.3.2. Label Cleanup CCI

In order to delete a P2MP LSP based on the PCECC, the PCE sends a Central Controller Instructions via a PCInitiate message to each node along the path of the P2MP tree to clean up the label-forwarding instruction as per [RFC9050]. In case of branch nodes, all instances of CCIs needs to be present in the PCEP message.

4.3.4. PCECC LSP Update

In case of a modification of PCECC P2MP LSP with a new path, the procedure, and instructions as described in [RFC9050] apply.

4.3.5. Re-delegation and Cleanup

In case of a re-delegation and clean up of PCECC P2MP LSP, the procedure, and instructions as described in [RFC9050] apply.

4.3.6. Synchronization of Central Controllers Instructions

The procedure and instructions are as per [RFC9050].

4.3.7. PCECC LSP State Report

An ingress PCC MAY choose to apply any Operations, Administration, and Maintenance (OAM) mechanism to check the status of the LSP in the data plane and MAY further send its status in the PCRpt message (as per [RFC8623]) to the PCE.

4.3.8. PCC-Based Allocations

The PCE can request the PCC to allocate the label using the PCInitiate message. The procedure and instructions are as per Section 5.5.8 of [RFC9050].

5. PCEP Messages

[RFC9050] specify the extension to PCInitiate and PCRpt message for PCECC. For P2P LSP, only two instances of CCI objects can be included. In the case of the P2MP LSP, multiple CCI objects are allowed. The message format and other procedures continue to apply.

6. PCEP Objects

6.1. OPEN Object

6.1.1. PCECC Capability sub-TLV

The PCECC-CAPABILITY sub-TLV is an optional TLV for use in the OPEN Object for PCECC capability advertisement in PATH-SETUP-TYPE-CAPABILITY TLV as specified in [RFC9050].

This document adds a new flag (M Bit) in the PCECC-CAPABILITY sub-TLV to indicate the support for P2MP in PCECC.

M (PCECC-P2MP-CAPABILITY - 1 bit - TBD1): If set to 1 by a PCEP speaker, it indicates that the PCEP speaker is capable of PCECC-P2MP capability.

A PCC MUST set the M Bit in the PCECC-CAPABILITY sub-TLV and set the N (P2MP-CAPABILITY), the M (P2MP-LSP-UPDATE-CAPABILITY), and the P (P2MP-LSP-INstantiation-CAPABILITY) bits (as per [RFC8623]) in the STATEFUL-PCE-CAPABILITY TLV [RFC8231] to support the PCECC-P2MP extensions defined in this document. If the M Bit is set in PCECC-CAPABILITY sub-TLV and the P2MP bits (in the STATEFUL-PCE-CAPABILITY TLV) are not set in the OPEN Object, a PCEP speaker SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD2 (P2MP capability was not advertised) and terminate the session.

6.2. PATH-SETUP-TYPE TLV

The PATH-SETUP-TYPE TLV is defined in [RFC8408]; [RFC9050] defines a PST value for PCECC as '2', which is applicable for P2MP LSP as well.

6.3. CCI Object

The CCI object [RFC9050] is used by the PCE to specify the forwarding instructions (label information in the context of this document) to the PCC, and optionally carried within PCInitiate or PCRpt message for label download/report. The CCI Object Type 1 for MPLS Label is defined in [RFC9050], which is used for the P2MP LSPs as well. The address TLVs are defined in [RFC9050], they associate the next-hop

information in case of an outgoing label.

If a node (PCC) receives a PCInitiate message with more than one CCI with O-bit set for the outgoing label and the node does not support the P2MP branch/replication capability, it MUST respond with PCErr message with Error-Type=2 (Capability not supported) (defined in [RFC5440]).

The rest of the processing is same as [RFC9050].

7. Security Considerations

As per [RFC8283], the security considerations for a PCE-based controller are a little different from those for any other PCE system. That is, the operation relies heavily on the use and security of PCEP, so consideration should be given to the security features discussed in [RFC5440] and the additional mechanisms described in [RFC8253]. It further lists the vulnerability of a central controller architecture, such as a central point of failure, denial of service, and a focus for interception and modification of messages sent to individual Network Elements (NEs).

The security considerations described in [RFC8231], [RFC8281], [RFC8623], and [RFC9050] apply to the extensions described in this document.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

8. Manageability Considerations

8.1. Control of Function and Policy

A PCE or PCC implementation SHOULD allow to configure to enable/disable PCECC-P2MP capability as a global configuration.

8.2. Information and Data Models

[RFC7420] describes the PCEP MIB, this MIB can be extended to get the PCECC capability status.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to enable/disable PCECC-P2MP capability.

8.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

8.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

8.5. Requirements On Other Protocols

PCEP extensions defined in this document do not put new requirements on other protocols.

8.6. Impact On Network Operations

PCEP extensions defined in this document do not put new requirements on network operations.

9. IANA Considerations

9.1. PCECC-CAPABILITY sub-TLV

[RFC9050] defines the PCECC-CAPABILITY sub-TLV and requests that IANA creates a registry to manage the value of the PCECC-CAPABILITY sub-TLV's Flag field. IANA is requested to allocate a new bit in the PCECC-CAPABILITY sub-TLV Flag Field registry, as follows:

Bit	Description	Reference
TBD1	P2MP	This document

Table 1

9.2. PCEP-Error Object

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

Error-Type	Meaning
19	Invalid operation.

Error-value = TBD2 : P2MP capability was
not advertised

The Reference is marked as "This document".

10. References

10.1. Normative References

- [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>>.
- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8623] Palle, U., Dhody, D., Tanaka, Y., and V. Beeram, "Stateful Path Computation Element (PCE) Protocol Extensions for Usage with Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 8623, DOI 10.17487/RFC8623, June 2019, <<https://www.rfc-editor.org/info/rfc8623>>.

- [RFC9050] Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) Procedures and Extensions for Using the PCE as a Central Controller (PCECC) of LSPs", RFC 9050, DOI 10.17487/RFC9050, July 2021, <<https://www.rfc-editor.org/info/rfc9050>>.

10.2. 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>>.
- [RFC4857] Fogelstroem, E., Jonsson, A., and C. Perkins, "Mobile IPv4 Regional Registration", RFC 4857, DOI 10.17487/RFC4857, June 2007, <<https://www.rfc-editor.org/info/rfc4857>>.
- [RFC4875] Aggarwal, R., Ed., Papadimitriou, D., Ed., and S. Yasukawa, Ed., "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, DOI 10.17487/RFC4875, May 2007, <<https://www.rfc-editor.org/info/rfc4875>>.
- [RFC5671] Yasukawa, S. and A. Farrel, Ed., "Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)", RFC 5671, DOI 10.17487/RFC5671, October 2009, <<https://www.rfc-editor.org/info/rfc5671>>.
- [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C. Margaria, "Requirements for GMPLS Applications of PCE", RFC 7025, DOI 10.17487/RFC7025, September 2013, <<https://www.rfc-editor.org/info/rfc7025>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <<https://www.rfc-editor.org/info/rfc7420>>.

- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.
- [RFC8306] Zhao, Q., Dhody, D., Ed., Palleti, R., and D. King, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 8306, DOI 10.17487/RFC8306, November 2017, <<https://www.rfc-editor.org/info/rfc8306>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [I-D.ietf-teas-pcecc-use-cases]
Li, Z. (., Dhody, D., Zhao, Q., Ke, K., Khasanov, B., Fang, L., Zhou, C., Zhang, B., Rachitskiy, A., and A. Gulida, "The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).", Work in Progress, Internet-Draft, draft-ietf-teas-pcecc-use-cases-08, 25 October 2021, <<https://datatracker.ietf.org/doc/html/draft-ietf-teas-pcecc-use-cases-08>>.
- [I-D.ietf-pce-pcep-yang]
Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-yang-18, 25 January 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-yang-18>>.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Udayasree Palle

Email: udayasreereddy@gmail.com

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: pengshuping@huawei.com

Xuesong Geng
Huawei Technologies
China
Email: gengxuesong@huawei.com

Mahendra Singh Negi
RtBrick Inc
N-17L, 18th Cross Rd, HSR Layout
Bangalore 560102
Karnataka
India
Email: mahend.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 5, 2021

Z. Li
S. Peng
X. Geng
Huawei Technologies
M. Negi
RtBrick Inc
November 1, 2020

PCEP Procedures and Protocol Extensions for Using PCE as a Central
Controller (PCECC) for SRv6
draft-dhody-pce-pcep-extension-pce-controller-srv6-05

Abstract

The Path Computation Element (PCE) is a core component of Software-Defined Networking (SDN) systems. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands.

PCE was developed to derive paths for MPLS Label Switched Paths (LSPs), which are supplied to the head end of the LSP using the Path Computation Element Communication Protocol (PCEP). But SDN has a broader applicability than signaled (G)MPLS traffic-engineered (TE) networks, and the PCE may be used to determine paths in a range of use cases. PCEP has been proposed as a control protocol for use in these environments to allow the PCE to be fully enabled as a central controller.

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. This document specifies the procedures and PCEP protocol extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers for Segment Routing (SR) in IPv6 (SRv6), in addition to computing the SRv6 paths for packet flows and telling the edge routers what instructions to attach to packets as they enter the network.

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 May 5, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	5
2. Terminology	5
3. PCECC SRv6	5
4. PCEP Requirements	6
5. Procedures for Using the PCE as a Central Controller (PCECC) in SRv6	6
5.1. Stateful PCE Model	6
5.2. New Functions	6
5.3. PCECC Capability Advertisement	7
5.4. PCEP session IP address and TED Router ID	7
5.5. SRv6 Path Operations	7
5.5.1. PCECC Segment Routing in IPv6 (SRv6)	7
5.5.1.1. PCECC SRv6 Node/Prefix SID allocation	8
5.5.1.2. PCECC SRv6 Adjacency SID allocation	8
5.5.1.3. Redundant PCEs	9
5.5.1.4. Re-Delegation and Clean up	9
5.5.1.5. Synchronization of SRv6 SID Allocations	9
6. PCEP Messages	9
7. PCEP Objects	9
7.1. OPEN Object	9
7.1.1. PCECC Capability sub-TLV	9
7.2. SRv6 Path Setup	10
7.3. CCI Object	10

7.4. FEC Object	11
8. Security Considerations	12
9. Manageability Considerations	12
9.1. Control of Function and Policy	12
9.2. Information and Data Models	12
9.3. Liveness Detection and Monitoring	12
9.4. Verify Correct Operations	12
9.5. Requirements On Other Protocols	12
9.6. Impact On Network Operations	13
10. IANA Considerations	13
10.1. PCECC-CAPABILITY sub-TLV	13
10.2. PCEP Object	13
10.3. PCEP-Error Object	13
11. Acknowledgments	13
12. References	14
12.1. Normative References	14
12.2. Informative References	15
Appendix A. Contributor Addresses	18
Authors' Addresses	18

1. Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload the path computation function from routers in an MPLS traffic-engineered network. Since then, the role and function of the PCE has grown to cover a number of other uses (such as GMPLS [RFC7025]) and to allow delegated control [RFC8231] and PCE-initiated use of network resources [RFC8281].

According to [RFC7399], Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system, called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place traffic flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths for MPLS Label Switched Paths (LSPs), paths were requested by network elements (known as Path Computation Clients (PCCs)), and the results of the path computations were supplied to network elements using the Path Computation Element Communication Protocol (PCEP) [RFC5440].

This protocol was later extended to allow a PCE to send unsolicited requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE and PCC. [RFC8283] further examines the motivations and applicability for PCEP as a Southbound Interface (SBI), and introduces the implications for the protocol.

[I-D.ietf-teas-pcecc-use-cases] describes the use cases for the PCE-based Central Controller (PCECC) architecture.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify the procedures and PCEP extensions for using the PCE as the central controller for static LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path.

Segment Routing (SR) technology leverages the source routing and tunneling paradigms. A source node can choose a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as a set of "segments" advertised by link-state routing protocols (IS-IS or OSPF). [RFC8402] provides an introduction to SR architecture. The corresponding IS-IS and OSPF extensions are specified in [RFC8667] and [RFC8665], respectively. It relies on a series of forwarding instructions being placed in the header of a packet. The list of segments forming the path is called the Segment List and is encoded in the packet header. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [RFC8754]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment. The segment routing architecture supports operations that can be used to steer packet flows in a network, thus providing a form of traffic engineering. [RFC8664] and [I-D.ietf-pce-segment-routing-ipv6] specify the SR specific PCEP extensions.

PCECC may further use PCEP for SR SID (Segment Identifier) distribution on the SR nodes with some benefits.

[I-D.zhao-pce-pcep-extension-pce-controller-sr] specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers (SR-MPLS SID distribution in this case), in addition to computing the paths for packet flows in a segment routing network and telling the edge routers what instructions to attach to packets as they enter the

network. This document extends this to include SRv6 SID distribution as well.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

Terminologies used in this document is the same as described in the draft [RFC8283] and [I-D.ietf-teas-pcecc-use-cases].

3. PCECC SRv6

[RFC8664] specifies extensions to PCEP that allow a stateful PCE to compute, update, or initiate SR-TE paths for MPLS dataplane. An ingress node of an SR-TE path appends all outgoing packets with a list of MPLS labels (SIDs). This is encoded in SR-ERO subobject, capable of carrying a label (SID) as well as the identity of the node/adjacency label (SID). [I-D.ietf-pce-segment-routing-ipv6] extends the procedure to include support for SRv6 paths.

As per [RFC8754], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in an illustration provided in [I-D.ietf-spring-srv6-network-programming]. The SR is applied to IPV6 data plane using SRH. An SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node. [I-D.ietf-pce-segment-routing-ipv6] specify the SRv6-ERO subobject capable of carrying an SRv6 SID as well as the identity of the node/adjacency represented by the SID.

As per [RFC8283], PCECC can allocate and provision the node/prefix/adjacency label (SID) via PCEP. As per [I-D.ietf-teas-pcecc-use-cases] this is also applicable to SRv6 SIDs.

The rest of the processing is similar to existing stateful PCE for SRv6 [I-D.ietf-pce-segment-routing-ipv6].

4. PCEP Requirements

Following key requirements for PCECC-SRv6 should be considered when designing the PCECC-based solution:

- o A PCEP speaker supporting this draft needs to have the capability to advertise its PCECC-SRv6 capability to its peers.
- o PCEP procedures need to allow for PCC-based SRv6 SID allocations.
- o PCEP procedures need means to update (or clean up) the SRv6 SID to the PCC.
- o PCEP procedures need to provide a mean to synchronize the SRv6 SID allocations between the PCE to the PCC in the PCEP messages.

5. Procedures for Using the PCE as a Central Controller (PCECC) in SRv6

5.1. Stateful PCE Model

Active stateful PCE is described in [RFC8231]. PCE as a Central Controller (PCECC) reuses the existing active stateful PCE mechanism as much as possible to control the LSPs.

5.2. New Functions

This document uses the same PCEP messages and its extensions which are described in [I-D.ietf-pce-pcep-extension-for-pce-controller] and [I-D.zhao-pce-pcep-extension-pce-controller-sr] for PCECC-SRv6 as well.

The PCEP messages PCRpt, PCInitiate, PCUpd are used to send LSP Reports, LSP setup, and LSP update respectively. The extended PCInitiate message described in [I-D.ietf-pce-pcep-extension-for-pce-controller] is used to download or clean up central controller's instructions (CCIs) (SRv6 SID in the scope of this document). The extended PCRpt message described in [I-D.ietf-pce-pcep-extension-for-pce-controller] is also used to report the CCIs (SRv6 SIDs) from PCC to PCE.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify an object called CCI for the encoding of the central controller's instructions. [I-D.zhao-pce-pcep-extension-pce-controller-sr] defined a CCI object-type for segment routing. This document further defines a new CCI object-type for SRv6.

5.3. PCECC Capability Advertisement

During PCEP Initialization Phase, PCEP Speakers (PCE or PCC) advertise their support of PCECC extensions. A PCEP Speaker includes the "PCECC Capability" sub-TLV, described in [I-D.ietf-pce-pcep-extension-for-pce-controller].

A new S-bit is added in the PCECC-CAPABILITY sub-TLV to indicate support for PCECC-SR in [I-D.zhao-pce-pcep-extension-pce-controller-sr]. This document adds another I-bit to indicate support for SR in IPv6. A PCC MUST set the I-bit in the PCECC-CAPABILITY sub-TLV and include the SRv6-PCE-CAPABILITY sub-TLV ([I-D.ietf-pce-segment-routing-ipv6]) in the OPEN Object (inside the PATH-SETUP-TYPE-CAPABILITY TLV) to support the PCECC SRv6 extensions defined in this document. If I-bit is set in PCECC-CAPABILITY sub-TLV and the SRv6-PCE-CAPABILITY sub-TLV is not advertised in the OPEN Object, PCE SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD4 (SRv6 capability was not advertised) and terminate the session.

The rest of the processing is as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

5.4. PCEP session IP address and TED Router ID

As described in [I-D.zhao-pce-pcep-extension-pce-controller-sr], it is important to link the session IP address with the Router ID in TED for successful PCECC-SRv6 operations.

5.5. SRv6 Path Operations

[RFC8664] specify the PCEP extension to allow a stateful PCE to compute and initiate SR-TE paths, as well as a PCC to request a path subject to certain constraint(s) and optimization criteria in SR networks. [I-D.ietf-pce-segment-routing-ipv6] extends it to support SRv6.

The Path Setup Type for SRv6 (PST=TBD) is used on the PCEP session with the Ingress as per [I-D.ietf-pce-segment-routing-ipv6].

5.5.1. PCECC Segment Routing in IPv6 (SRv6)

Segment Routing (SR) as described in [RFC8402] depends on "segments" that are advertised by Interior Gateway Protocols (IGPs). The SR-node allocates and advertises the SID (node, adj, etc) and floods them via the IGP. This document proposes a new mechanism where PCE allocates the SRv6 SID centrally and uses PCEP to advertise them. In some deployments, PCE (and PCEP) are better suited than IGP because

of the centralized nature of PCE and direct TCP based PCEP sessions to the node.

5.5.1.1. PCECC SRv6 Node/Prefix SID allocation

Each node (PCC) is allocated a node SRv6 SID by the PCECC. The PCECC sends the PCInitiate message to update the SRv6 SID table of each node. The TE router ID is determined from the TED or from "IPv4/IPv6 Router-ID" Sub-TLV [I-D.dhodylee-pce-pcep-ls], in the OPEN Object.

On receiving the SRv6 node SID allocation, each node (PCC) uses the local routing information to determine the next-hop and download the forwarding instructions accordingly. The PCInitiate message uses the FEC object [I-D.zhao-pce-pcep-extension-pce-controller-sr].

On receiving the SRv6 node SID allocation:

For the local SID, the node (PCC) needs to update SID with associated function (END function in this case) in "My Local SID Table" ([I-D.ietf-spring-srv6-network-programming]).

For the non-local SID, the node (PCC) uses the local routing information to determine the next-hop and download the forwarding instructions accordingly.

The forwarding behavior and the end result is similar to IGP based "Node-SID" in SRv6. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node as per [RFC8402].

PCE relies on the Node/Prefix SRv6 SID clean up using the same PCInitiate message as per [RFC8281].

5.5.1.2. PCECC SRv6 Adjacency SID allocation

For PCECC-SRv6, apart from node-SID, Adj-SID is used where each adjacency is allocated an Adj-SID by the PCECC. The PCECC sends PCInitiate message to update the SRv6 SID entry for each adjacency to the corresponding nodes in the domain. Each node (PCC) download the SRv6 SID instructions accordingly. Similar to SRv6 Node/Prefix Label allocation, the PCInitiate message in this case uses the FEC object.

The forwarding behavior and the end result is similar to IGP based "Adj-SID" in SRv6 as per [RFC8402].

The handling of adjacencies on the LAN subnetworks is specified in [RFC8402]. PCECC MUST assign Adj-SID for every pair of routers in the LAN. The rest of the protocol mechanism remains the same.

PCE relies on the Adj label clean up using the same PCInitiate message as per [RFC8281].

5.5.1.3. Redundant PCEs

[I-D.litkowski-pce-state-sync] describes the synchronization mechanism between the stateful PCEs. The SRv6 SIDs allocated by a PCE MUST also be synchronized among PCEs for PCECC-SRv6 state synchronization. Note that the SRv6 SIDs are independent of the SRv6 paths, and remains intact till any topology change. The redundant PCEs MUST have a common view of all SRv6 SIDs allocated in the domain.

5.5.1.4. Re-Delegation and Clean up

[I-D.ietf-pce-pcep-extension-for-pce-controller] describes the action needed for CCIs for the static LSPs on a terminated session. Same holds true for the CCI for SRv6 SID as well.

5.5.1.5. Synchronization of SRv6 SID Allocations

[I-D.ietf-pce-pcep-extension-for-pce-controller] describes the synchronization of Central Controller's Instructions (CCI) via the LSP state synchronization as described in [RFC8231] and [RFC8232]. Same procedures are applied for the CCI for the SRv6 SIDs as well.

6. PCEP Messages

The PCEP messages are as per
[I-D.zhao-pce-pcep-extension-pce-controller-sr].

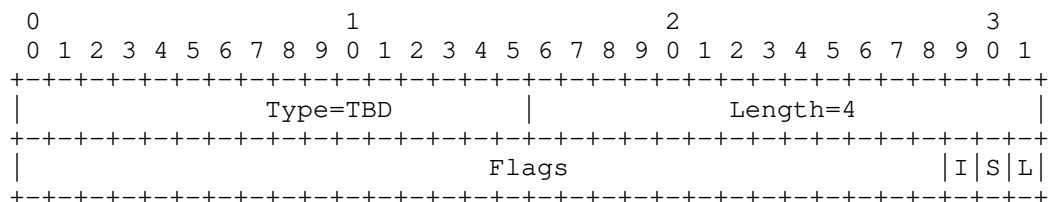
7. PCEP Objects

7.1. OPEN Object

7.1.1. PCECC Capability sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defined the PCECC-CAPABILITY sub-TLV.

A new I-bit is defined in PCECC-CAPABILITY sub-TLV for PCECC-SRv6:



[Editor's Note - The above figure is included for ease of the reader but should be removed before publication.]

I (PCECC-SRv6-CAPABILITY - 1 bit - TBD1): If set to 1 by a PCEP speaker, it indicates that the PCEP speaker is capable of PCECC-SRv6 capability and the PCE allocates the Node and Adj SRv6 SID on this session.

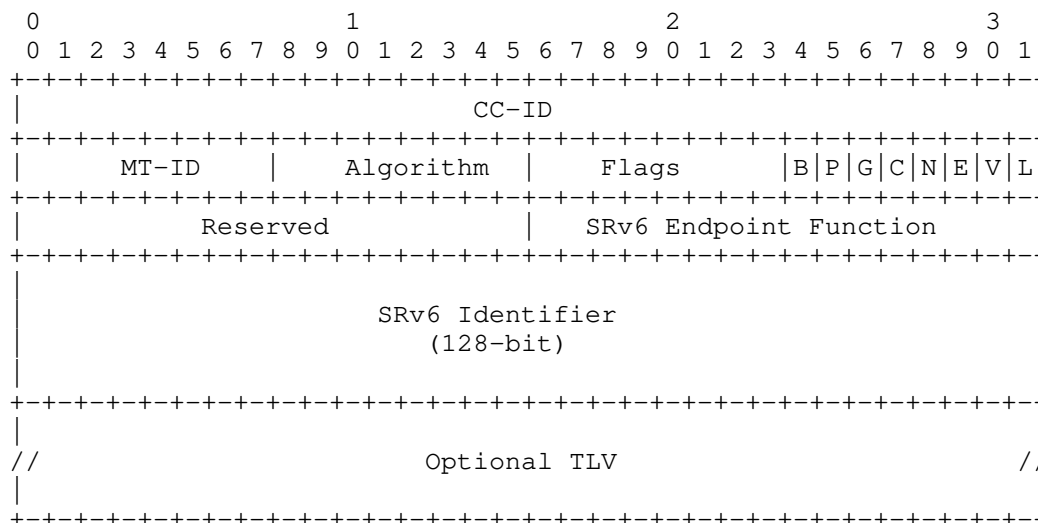
7.2. SRv6 Path Setup

The PATH-SETUP-TYPE TLV is defined in [RFC8408]. A PST value of TBD is used when Path is setup via SRv6 mode as per [I-D.ietf-pce-segment-routing-ipv6]. The procedure for SRv6 path setup as specified in [I-D.ietf-pce-segment-routing-ipv6] remains unchanged.

7.3. CCI Object

The Central Control Instructions (CCI) Object is used by the PCE to specify the controller instructions is defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]. This document defines another object-type for SRv6 purpose.

CCI Object-Type is TBD3 for SRv6 as below -



The field CC-ID is as described in [I-D.ietf-pce-pcep-extension-for-pce-controller]. The field MT-ID, Algorithm, Flags are defined in [I-D.zhao-pce-pcep-extension-pce-controller-sr].

Reserved: MUST be set to 0 while sending and ignored on receipt.

SRv6 Endpoint Function: 16-bit field representing supported functions associated with SRv6 SIDs.

SRv6 Identifier: 128-bit IPv6 addresses representing SRv6 segment.

[Editor's Note - It might be useful to separate the LOC:FUNC part in the SRv6 SID (future study)]

7.4. FEC Object

The FEC Object is used to specify the FEC information and MAY be carried within PCInitiate or PCRpt message.

FEC Object (and various Object-Types) are described in [I-D.zhao-pce-pcep-extension-pce-controller-sr]. SRv6 Node SID MUST include the FEC Object-Type 2 for IPv6 Node. SRv6 Adjacency SID MUST include the FEC Object-Type=4 for IPv6 adjacency. Further FEC object types could be added in future extensions.

8. Security Considerations

The security considerations described in [I-D.ietf-pce-pcep-extension-for-pce-controller] apply to the extensions described in this document.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

9. Manageability Considerations

9.1. Control of Function and Policy

A PCE or PCC implementation SHOULD allow to configure to enable/disable PCECC SRv6 capability as a global configuration.

9.2. Information and Data Models

[RFC7420] describes the PCEP MIB, this MIB can be extended to get the PCECC SRv6 capability status.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to enable/disable PCECC SRv6 capability.

9.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

9.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

9.5. Requirements On Other Protocols

PCEP extensions defined in this document do not put new requirements on other protocols.

9.6. Impact On Network Operations

PCEP implementation SHOULD allow a limit to be placed on the rate of PCInitiate/PCUpd messages (as per [RFC8231]) sent by PCE and processed by PCC. It SHOULD also allow sending a notification when a rate threshold is reached.

10. IANA Considerations

10.1. PCECC-CAPABILITY sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defines the PCECC-CAPABILITY sub-TLV and requests that IANA creates a registry to manage the value of the PCECC-CAPABILITY sub-TLV's Flag field. IANA is requested to allocate a new bit in the PCECC-CAPABILITY sub-TLV Flag Field registry, as follows:

Bit	Description	Reference
TBD1	SRv6	This document

10.2. PCEP Object

IANA is requested to allocate a new code-point for the new CCI object-type in "PCEP Objects" sub-registry as follows:

Object-Class Value	Name	Object-Type	Reference
TBD	CCI	TBD3: SRv6	This document

10.3. PCEP-Error Object

IANA is requested to allocate new error types and error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry for the following errors:

Error-Type	Meaning
-----	-----
19	Invalid operation.
	Error-value = TBD4 : SRv6 capability was not advertised

11. Acknowledgments

12. References

12.1. Normative References

- [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>>.
- [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>>.
- [I-D.ietf-pce-segment-routing-ipv6]
Li, C., Negi, M., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", draft-ietf-pce-segment-routing-ipv6-06 (work in progress), July 2020.
- [I-D.ietf-pce-pcep-extension-for-pce-controller]
Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-07 (work in progress), September 2020.

[I-D.zhao-pce-pcep-extension-pce-controller-sr]
Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "PCEP
Procedures and Protocol Extensions for Using PCE as a
Central Controller (PCECC) of SR-LSPs", draft-zhao-pce-
pcep-extension-pce-controller-sr-07 (work in progress),
July 2020.

12.2. Informative References

- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C.
Margarita, "Requirements for GMPLS Applications of PCE",
RFC 7025, DOI 10.17487/RFC7025, September 2013,
<<https://www.rfc-editor.org/info/rfc7025>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path
Computation Element Architecture", RFC 7399,
DOI 10.17487/RFC7399, October 2014,
<<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J.
Hardwick, "Path Computation Element Communication Protocol
(PCEP) Management Information Base (MIB) Module",
RFC 7420, DOI 10.17487/RFC7420, December 2014,
<<https://www.rfc-editor.org/info/rfc7420>>.
- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for
Application-Based Network Operations", RFC 7491,
DOI 10.17487/RFC7491, March 2015,
<<https://www.rfc-editor.org/info/rfc7491>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre,
"Recommendations for Secure Use of Transport Layer
Security (TLS) and Datagram Transport Layer Security
(DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May
2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X.,
and D. Dhody, "Optimizations of Label Switched Path State
Synchronization Procedures for a Stateful PCE", RFC 8232,
DOI 10.17487/RFC8232, September 2017,
<<https://www.rfc-editor.org/info/rfc8232>>.

- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <<https://www.rfc-editor.org/info/rfc8665>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.
- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/info/rfc8754>>.
- [I-D.ietf-teas-pcecc-use-cases]
Li, Z., Khasanov, B., Dhody, D., Zhao, Q., Ke, Z., Fang, L., Zhou, C., Communications, T., Rachitskiy, A., and A. Gulida, "The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).", draft-ietf-teas-pcecc-use-cases-06 (work in progress), September 2020.

[I-D.ietf-pce-pcep-yang]

Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-14 (work in progress), July 2020.

[I-D.litkowski-pce-state-sync]

Litkowski, S., Sivabalan, S., Li, C., and H. Zheng, "Inter Stateful Path Computation Element (PCE) Communication Procedures.", draft-litkowski-pce-state-sync-08 (work in progress), July 2020.

[I-D.dhodylee-pce-pcep-ls]

Dhody, D., Peng, S., Lee, Y., Ceccarelli, D., and A. Wang, "PCEP extensions for Distribution of Link-State and TE Information", draft-dhodylee-pce-pcep-ls-17 (work in progress), July 2020.

[I-D.ietf-spring-srv6-network-programming]

Filsfils, C., Camarillo, P., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-ietf-spring-srv6-network-programming-24 (work in progress), October 2020.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: pengshuping@huawei.com

Xuesong Geng
Huawei Technologies
China

EMail: gengxuesong@huawei.com

Mahendra Singh Negi
RtBrick Inc
N-17L, 18th Cross Rd, HSR Layout
Bangalore, Karnataka 560102
India

EMail: mahend.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 7 September 2022

Z. Li
S. Peng
X. Geng
Huawei Technologies
M. Negi
RtBrick Inc
6 March 2022

Path Computation Element Communication Protocol (PCEP) Procedures and
Extensions for Using the PCE as a Central Controller (PCECC) for SRv6
SID Allocation and Distribution.
draft-dhody-pce-pcep-extension-pce-controller-srv6-08

Abstract

The Path Computation Element (PCE) is a core component of Software-Defined Networking (SDN) systems.

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. This document specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers, in addition to computing the paths for packet flows in the for Segment Routing (SR) in IPv6 (SRv6) network and telling the edge routers what instructions to attach to packets as they enter the network. PCECC is further enhanced for SRv6 SID (Segment Identifier) allocation and distribution.

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 7 September 2022.

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	3
2. Terminology	5
2.1. Requirements Language	5
3. PCECC SRv6	5
4. PCEP Requirements	6
5. Procedures for Using the PCE as a Central Controller (PCECC) in SRv6	6
5.1. Stateful PCE Model	6
5.2. New Functions	6
5.3. PCECC Capability Advertisement	7
5.4. PCEP session IP address and TED Router ID	7
5.5. SRv6 Path Operations	7
5.5.1. PCECC Segment Routing in IPv6 (SRv6)	8
5.5.1.1. PCECC SRv6 Node/Prefix SID allocation	8
5.5.1.2. PCECC SRv6 Adjacency SID allocation	9
5.5.1.3. Redundant PCEs	9
5.5.1.4. Re-Delegation and Cleanup	9
5.5.1.5. Synchronization of SRv6 SID Allocations	9
6. PCEP Messages	9
7. PCEP Objects	10
7.1. OPEN Object	10
7.1.1. PCECC Capability sub-TLV	10
7.2. SRv6 Path Setup	10
7.3. CCI Object	10
7.4. FEC Object	11
8. Security Considerations	12
9. Manageability Considerations	12
9.1. Control of Function and Policy	12
9.2. Information and Data Models	12
9.3. Liveness Detection and Monitoring	13
9.4. Verify Correct Operations	13
9.5. Requirements On Other Protocols	13

9.6. Impact On Network Operations	13
10. IANA Considerations	13
10.1. PCECC-CAPABILITY sub-TLV	13
10.2. PCEP Object	13
10.3. PCEP-Error Object	14
11. References	14
11.1. Normative References	14
11.2. Informative References	15
Appendix A. Contributor Addresses	18
Authors' Addresses	18

1. Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload the path computation function from routers in an MPLS traffic-engineered (TE) network. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands. Since then, the role and function of the PCE have grown to cover a number of other uses (such as GMPLS [RFC7025]) and to allow delegated control [RFC8231] and PCE-initiated use of network resources [RFC8281].

According to [RFC7399], Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system, called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place traffic flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths for MPLS Label Switched Paths (LSPs), paths were requested by network elements (known as Path Computation Clients (PCCs)), and the results of the path computations were supplied to network elements using the Path Computation Element Communication Protocol (PCEP) [RFC5440]. This protocol was later extended to allow a PCE to send unsolicited requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE and PCC. [RFC8283] further examines the motivations and

applicability for PCEP as a Southbound Interface (SBI), and introduces the implications for the protocol. [I-D.ietf-teas-pcecc-use-cases] describes the use cases for the PCECC architecture.

[RFC9050] specify the procedures and PCEP extensions for using the PCE as the central controller for static LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path.

Segment Routing (SR) technology leverages the source routing and tunneling paradigms. A source node can choose a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as a set of "segments" advertised by link-state routing protocols (IS-IS or OSPF). [RFC8402] provides an introduction to SR architecture. The corresponding IS-IS and OSPF extensions are specified in [RFC8667] and [RFC8665], respectively. It relies on a series of forwarding instructions being placed in the header of a packet. The list of segments forming the path is called the Segment List and is encoded in the packet header. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [RFC8754]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment. The segment routing architecture supports operations that can be used to steer packet flows in a network, thus providing a form of traffic engineering. [RFC8664] and [I-D.ietf-pce-segment-routing-ipv6] specify the SR specific PCEP extensions.

PCECC may further use PCEP for SR SID (Segment Identifier) allocation and distribution to all the SR nodes with some benefits. The SR nodes continue to rely on IGP for distributed computation (nexthop selection, protection etc) where PCE (and PCEP) does only the allocation and distribution of SRv6 SIDs in the network. Note that the topology at PCE is still learned via existing mechanisms.

[I-D.ietf-pce-pcep-extension-pce-controller-sr] specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers (SR-MPLS SID distribution), in addition to computing the paths for packet flows in a segment routing network and telling the edge routers what instructions to attach to packets as they enter the network. This document extends this to include SRv6 SID distribution as well.

2. Terminology

Terminologies used in this document is the same as described in the document [RFC8283].

2.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. PCECC SRv6

[RFC8664] specifies extensions to PCEP that allow a stateful PCE to compute, update, or initiate SR-TE paths for MPLS dataplane. An ingress node of an SR-TE path appends all outgoing packets with a list of MPLS labels (SIDs). This is encoded in SR-ERO subobject, capable of carrying a label (SID) as well as the identity of the node/adjacency label (SID). [I-D.ietf-pce-segment-routing-ipv6] extends the procedure to include support for SRv6 paths.

As per [RFC8754], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in an illustration provided in [RFC8986]. The SR is applied to IPV6 data plane using SRH. An SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node. [I-D.ietf-pce-segment-routing-ipv6] specify the SRv6-ERO subobject capable of carrying an SRv6 SID as well as the identity of the node/adjacency represented by the SID.

[RFC8283] examines the motivations and applicability for PCECC and use of PCEP as an SBI. Section 3.1.5. of [RFC8283] highlights the use of PCECC for configuring the forwarding actions on the routers and assume responsibility for managing the identifier space. It simplifies the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. This allows the operator to introduce the advantages of SDN (such as programmability) into the network. Further Section 3.3. of [I-D.ietf-teas-pcecc-use-cases] describes some of the scenarios where the PCECC technique could be useful. Section 4 of [RFC8283] also describe the implications on the protocol when used as an SDN SBI. The operator needs to evaluate the advantages offered by PCECC against the operational and scalability needs of the PCECC.

As per [RFC8283], PCECC can allocate and provision the node/prefix/adjacency label (SID) via PCEP. As per [I-D.ietf-teas-pcecc-use-cases] this is also applicable to SRv6 SIDs.

The rest of the processing is similar to existing stateful PCE for SRv6 [I-D.ietf-pce-segment-routing-ipv6].

4. PCEP Requirements

Following key requirements for PCECC-SRv6 should be considered when designing the PCECC-based solution:

- * A PCEP speaker supporting this document needs to have the capability to advertise its PCECC-SRv6 capability to its peers.
- * PCEP procedures need to allow for PCC-based SRv6 SID allocations.
- * PCEP procedures need to provide a means to update (or clean up) the SRv6 SID to the PCC.
- * PCEP procedures need to provide a means to synchronize the SRv6 SID allocations between the PCE to the PCC in the PCEP messages.

5. Procedures for Using the PCE as a Central Controller (PCECC) in SRv6

5.1. Stateful PCE Model

Active stateful PCE is described in [RFC8231]. A PCE as a Central Controller (PCECC) reuses the existing active stateful PCE mechanism as much as possible to control the LSPs.

5.2. New Functions

This document uses the same PCEP messages and its extensions which are described in [RFC9050] and [I-D.ietf-pce-pcep-extension-pce-controller-sr] for PCECC-SRv6 as well.

The PCEP messages PCRpt, PCInitiate, PCUpd are used to send LSP Reports, LSP setup, and LSP update respectively. The extended PCInitiate message described in [RFC9050] is used to download or clean up CCIs (a new CCI Object-Type=TBD3 for SRv6 SID). The extended PCRpt message described in [RFC9050] is also used to report the CCIs (SRv6 SIDs) from PCC to PCE.

[RFC9050] specify an object called CCI for the encoding of the central controller's instructions.

[I-D.ietf-pce-pcep-extension-pce-controller-sr] defined a CCI object-type for SR-MPLS. This document further defines a new CCI object-type=TBD3 for SRv6.

5.3. PCECC Capability Advertisement

During the PCEP initialization phase, PCEP speakers (PCE or PCC) advertise their support of and willingness to use PCEP extensions for the PCECC. A PCEP speaker includes the PCECC-CAPABILITY sub-TLV in the PATH-SETUP-TYPE-CAPABILITY TLV as per [RFC9050].

A new S bit is added in the PCECC-CAPABILITY sub-TLV to indicate support for PCECC-SR-MPLS in [I-D.ietf-pce-pcep-extension-pce-controller-sr]. This document adds another I bit to indicate support for SR in IPv6. A PCC MUST set the I bit in the PCECC-CAPABILITY sub-TLV and include the SRv6-PCE-CAPABILITY sub-TLV ([I-D.ietf-pce-segment-routing-ipv6]) in the OPEN object (inside the PATH-SETUP-TYPE-CAPABILITY TLV) to support the PCECC SRv6 extensions defined in this document.

If the I bit is set in PCECC-CAPABILITY sub-TLV and the SRv6-PCE-CAPABILITY sub-TLV is not advertised, or is advertised without the I bit set, in the OPEN object, the receiver MUST:

- * send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBD4 (SRv6 capability was not advertised) and
- * terminate the session.

The rest of the processing is as per [RFC9050] and [I-D.ietf-pce-pcep-extension-pce-controller-sr].

5.4. PCEP session IP address and TED Router ID

As described in [I-D.ietf-pce-pcep-extension-pce-controller-sr], it is important to link the session IP address with the Router ID in TED for successful PCECC-SRv6 operations.

5.5. SRv6 Path Operations

[RFC8664] specify the PCEP extension to allow a stateful PCE to compute and initiate SR-TE paths, as well as a PCC to request a path subject to certain constraint(s) and optimization criteria in SR networks. [I-D.ietf-pce-segment-routing-ipv6] extends it to support SRv6.

The Path Setup Type for SRv6 (PST=TBD) is used on the PCEP session with the Ingress as per [I-D.ietf-pce-segment-routing-ipv6].

5.5.1. PCECC Segment Routing in IPv6 (SRv6)

Segment Routing (SR) as described in [RFC8402] depends on "segments" that are advertised by Interior Gateway Protocols (IGPs). The SR-node allocates and advertises the SID (node, adj, etc) and floods them via the IGP. This document proposes a new mechanism where PCE allocates the SRv6 SID centrally and uses PCEP to distribute them to all nodes. In some deployments, PCE (and PCEP) are better suited than IGP because of the centralized nature of PCE and direct TCP based PCEP sessions to the node. Note that only the SRv6 SID allocation and distribution is done by the PCEP, all other SRv6 operations (nexthop selection, protection, etc) are still done by the node (and the IGPs).

5.5.1.1. PCECC SRv6 Node/Prefix SID allocation

Each node (PCC) is allocated a node SRv6 SID by the PCECC. The PCECC sends the PCInitiate message to update the SRv6 SID table of each node. The TE router ID is determined from the TED or from "IPv4/IPv6 Router-ID" sub-TLV [I-D.dhodylee-pce-pcep-ls], in the OPEN Object.

On receiving the SRv6 node SID allocation, each node (PCC) uses the local routing information to determine the next-hop and download the forwarding instructions accordingly. The PCInitiate message uses the FEC object [I-D.ietf-pce-pcep-extension-pce-controller-sr].

On receiving the SRv6 node SID allocation:

For the local SID, the node (PCC) needs to update SID with associated function (END function in this case) in "My Local SID Table" ([RFC8986]).

For the non-local SID, the node (PCC) uses the local routing information to determine the next-hop and download the forwarding instructions accordingly.

The forwarding behavior and the end result is similar to IGP based "Node-SID" in SRv6. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node as per [RFC8402].

PCE relies on the Node/Prefix SRv6 SID clean up using the same PCInitiate message as per [RFC8281].

5.5.1.2. PCECC SRv6 Adjacency SID allocation

For PCECC-SRv6, apart from node-SID, Adj-SID is used where each adjacency is allocated an Adj-SID by the PCECC. The PCECC sends PCInitiate message to update the SRv6 SID entry for each adjacency to all nodes in the domain. Each node (PCC) download the SRv6 SID instructions accordingly. Similar to SRv6 Node/Prefix Label allocation, the PCInitiate message in this case uses the FEC object.

The forwarding behavior and the end result is similar to IGP based "Adj-SID" in SRv6 as per [RFC8402].

The handling of adjacencies on the LAN subnetworks is specified in [RFC8402]. PCECC MUST assign Adj-SID for every pair of routers in the LAN. The rest of the protocol mechanism remains the same.

PCE relies on the Adj label clean up using the same PCInitiate message as per [RFC8281].

5.5.1.3. Redundant PCEs

[I-D.ietf-pce-state-sync] describes the synchronization mechanism between the stateful PCEs. The SRv6 SIDs allocated by a PCE MUST also be synchronized among PCEs for PCECC-SRv6 state synchronization. Note that the SRv6 SIDs are independent of the SRv6 paths, and remains intact till any topology change. The redundant PCEs MUST have a common view of all SRv6 SIDs allocated in the domain.

5.5.1.4. Re-Delegation and Cleanup

[RFC9050] describes the action needed for CCIs for the static LSPs on a terminated session. Same holds true for the CCI for SRv6 SID as well.

5.5.1.5. Synchronization of SRv6 SID Allocations

[RFC9050] describes the synchronization of CCIs via the LSP state synchronization as described in [RFC8231] and [RFC8232]. Same procedures are applied for the SRv6 SID CCIs.

6. PCEP Messages

The PCEP messages are as per [I-D.ietf-pce-pcep-extension-pce-controller-sr].

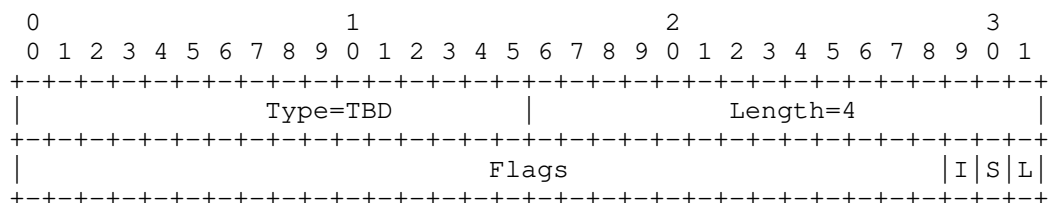
7. PCEP Objects

7.1. OPEN Object

7.1.1. PCECC Capability sub-TLV

[RFC9050] defined the PCECC-CAPABILITY sub-TLV.

A new I-bit is defined in PCECC-CAPABILITY sub-TLV for PCECC-SRv6:



[Editor's Note - The above figure is included for ease of the reader but should be removed before publication.]

I (PCECC-SRv6-CAPABILITY - 1 bit - TBD1): If set to 1 by a PCEP speaker, it indicates that the PCEP speaker is capable of PCECC-SRv6 capability and the PCE allocates the Node and Adj SRv6 SID on this session.

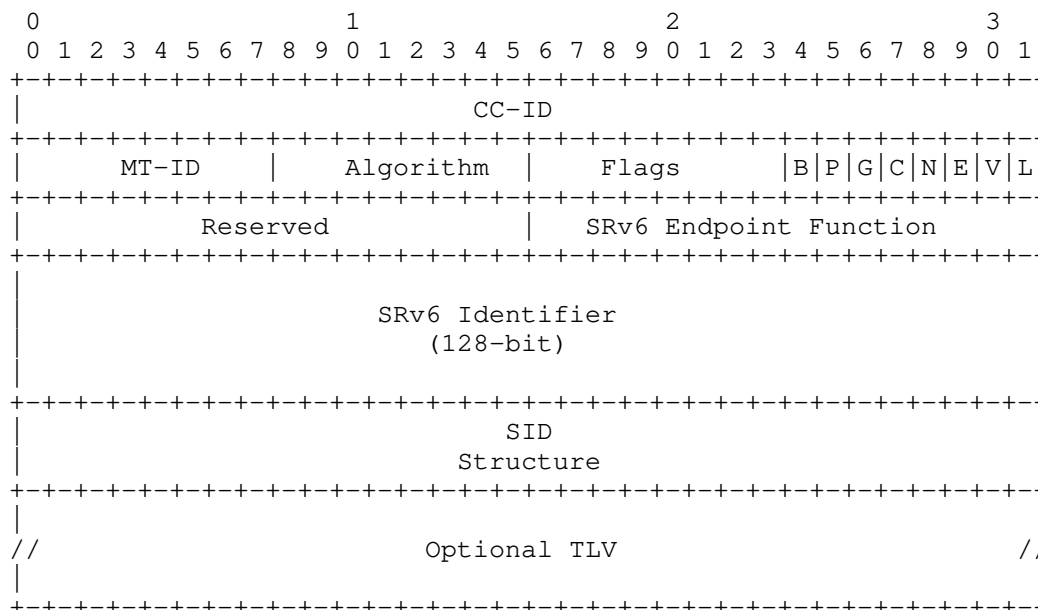
7.2. SRv6 Path Setup

The PATH-SETUP-TYPE TLV is defined in [RFC8408]. A PST value of TBD is used when Path is setup via SRv6 mode as per [I-D.ietf-pce-segment-routing-ipv6]. The procedure for SRv6 path setup as specified in [I-D.ietf-pce-segment-routing-ipv6] remains unchanged.

7.3. CCI Object

The Central Control Instructions (CCI) Object is used by the PCE to specify the controller instructions is defined in [RFC9050]. This document defines another object-type for SRv6 purpose.

CCI Object-Type is TBD3 for SRv6 as below -



The field CC-ID is as described in [RFC9050]. The field MT-ID, Algorithm, Flags are defined in [I-D.ietf-pce-pcep-extension-pce-controller-sr].

Reserved: MUST be set to 0 while sending and ignored on receipt.

SRv6 Endpoint Function: 16-bit field representing supported functions associated with SRv6 SIDs.

SRv6 Identifier: 128-bit IPv6 addresses representing SRv6 segment.

SID Structure: 64-bit field formatted as per "SID Structure" in [I-D.ietf-pce-segment-routing-ipv6]. The sum of all four sizes in the SID Structure must be lower or equal to 128 bits. If the sum of all four sizes advertised in the SID Structure is larger than 128 bits, the corresponding SRv6 SID MUST be considered invalid and a PCERR message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Invalid SRv6 SID Structure") is returned.

7.4. FEC Object

The FEC Object is used to specify the FEC information and MAY be carried within PCInitiate or PCRpt message.

FEC Object (and various Object-Types) are described in [I-D.ietf-pce-pcep-extension-pce-controller-sr]. SRv6 Node SID MUST include the FEC Object-Type 2 for IPv6 Node. SRv6 Adjacency SID MUST include the FEC Object-Type=4 for IPv6 adjacency. Further FEC object types could be added in future extensions.

8. Security Considerations

As per [RFC8283], the security considerations for a PCE-based controller are a little different from those for any other PCE system. That is, the operation relies heavily on the use and security of PCEP, so consideration should be given to the security features discussed in [RFC5440] and the additional mechanisms described in [RFC8253]. It further lists the vulnerability of a central controller architecture, such as a central point of failure, denial of service, and a focus for interception and modification of messages sent to individual Network Elements (NEs).

The PCECC extension builds on the existing PCEP messages; thus, the security considerations described in [RFC5440], [RFC8231], [RFC8281], [RFC9050], and [I-D.ietf-pce-pcep-extension-pce-controller-sr] continue to apply.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on mutually-authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

9. Manageability Considerations

9.1. Control of Function and Policy

A PCE or PCC implementation SHOULD allow to configure to enable/disable PCECC SRv6 capability as a global configuration.

9.2. Information and Data Models

[RFC7420] describes the PCEP MIB, this MIB can be extended to get the PCECC SRv6 capability status.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to enable/disable PCECC SRv6 capability.

9.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

9.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

9.5. Requirements On Other Protocols

PCEP extensions defined in this document do not put new requirements on other protocols.

9.6. Impact On Network Operations

PCEP implementation SHOULD allow a limit to be placed on the rate of PCInitiate/PCUpd messages (as per [RFC8231]) sent by PCE and processed by PCC. It SHOULD also allow sending a notification when a rate threshold is reached.

10. IANA Considerations

10.1. PCECC-CAPABILITY sub-TLV

[RFC9050] defines the PCECC-CAPABILITY sub-TLV and requests that IANA creates a registry to manage the value of the PCECC-CAPABILITY sub-TLV's Flag field. IANA is requested to allocate a new bit in the PCECC-CAPABILITY sub-TLV Flag Field registry, as follows:

Bit	Description	Reference
TBD1	SRv6	This document

Table 1

10.2. PCEP Object

IANA is requested to allocate a new code-point for the new CCI object-type in "PCEP Objects" sub-registry as follows:

Object-Class Value	Name	Object-Type	Reference
TBD	CCI		[RFC9050]
		TBD3: SRv6	This document

Table 2

10.3. PCEP-Error Object

IANA is requested to allocate new error types and error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry for the following errors:

Error-Type	Meaning
19	Invalid operation. Error-value = TBD4 :
	SRv6 capability was not advertised

The Reference is marked as "This document".

11. References

11.1. Normative References

- [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>>.
- [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>>.
- [I-D.ietf-pce-segment-routing-ipv6]
Li(Editor), C., Negi, M. S., Sivabalan, S., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", Work in Progress, Internet-Draft, draft-ietf-pce-segment-routing-ipv6-11, 10 January 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-segment-routing-ipv6-11>>.
- [RFC9050] Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) Procedures and Extensions for Using the PCE as a Central Controller (PCECC) of LSPs", RFC 9050, DOI 10.17487/RFC9050, July 2021, <<https://www.rfc-editor.org/info/rfc9050>>.
- [I-D.ietf-pce-pcep-extension-pce-controller-sr]
Li, Z., Peng, S., Negi, M. S., Zhao, Q., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) Procedures and Extensions for Using PCE as a Central Controller (PCECC) for Segment Routing (SR) MPLS Segment Identifier (SID) Allocation and Distribution.", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-extension-pce-controller-sr-04, 6 March 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-extension-pce-controller-sr-04>>.

11.2. 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>>.

- [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C. Margaria, "Requirements for GMPLS Applications of PCE", RFC 7025, DOI 10.17487/RFC7025, September 2013, <<https://www.rfc-editor.org/info/rfc7025>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <<https://www.rfc-editor.org/info/rfc7420>>.
- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.

- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <<https://www.rfc-editor.org/info/rfc8665>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.
- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/info/rfc8754>>.
- [RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <<https://www.rfc-editor.org/info/rfc8986>>.
- [I-D.ietf-teas-pcecc-use-cases]
Li, Z. (., Dhody, D., Zhao, Q., Ke, K., Khasanov, B., Fang, L., Zhou, C., Zhang, B., Rachitskiy, A., and A. Gulida, "The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).", Work in Progress, Internet-Draft, draft-ietf-teas-pcecc-use-cases-08, 25 October 2021, <<https://datatracker.ietf.org/doc/html/draft-ietf-teas-pcecc-use-cases-08>>.

[I-D.ietf-pce-pcep-yang]

Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura,
"A YANG Data Model for Path Computation Element
Communications Protocol (PCEP)", Work in Progress,
Internet-Draft, draft-ietf-pce-pcep-yang-18, 25 January
2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-yang-18>>.

[I-D.ietf-pce-state-sync]

Litkowski, S., Sivabalan, S., Li, C., and H. Zheng, "Inter
Stateful Path Computation Element (PCE) Communication
Procedures.", Work in Progress, Internet-Draft, draft-
ietf-pce-state-sync-01, 20 October 2021,
<<https://datatracker.ietf.org/doc/html/draft-ietf-pce-state-sync-01>>.

[I-D.dhodylee-pce-pcep-ls]

Dhody, D., Peng, S., Lee, Y., Ceccarelli, D., Wang, A.,
Mishra, G., and S. Sivabalan, "PCEP extensions for
Distribution of Link-State and TE Information", Work in
Progress, Internet-Draft, draft-dhodylee-pce-pcep-ls-23, 5
March 2022, <<https://datatracker.ietf.org/doc/html/draft-dhodylee-pce-pcep-ls-23>>.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing
100095
China
Email: pengshuping@huawei.com

Xuesong Geng
Huawei Technologies
China
Email: gengxuesong@huawei.com

Mahendra Singh Negi
RtBrick Inc
N-17L, 18th Cross Rd, HSR Layout
Bangalore 560102
Karnataka
India
Email: mahend.ietf@gmail.com

PCE Working Group
Internet-Draft
Updates: 8231 (if approved)
Intended status: Standards Track
Expires: May 5, 2021

C. Li
H. Zheng
Huawei Technologies
S. Litkowski
Cisco
November 1, 2020

Extension for Stateful PCE to allow Optional Processing of PCEP Objects
draft-dhody-pce-stateful-pce-optional-07

Abstract

This document introduces a mechanism to mark some of the Path Computation Element (PCE) Communication Protocol (PCEP) objects as optional during PCEP messages exchange for the Stateful PCE model to allow relaxing some constraints. This document introduces this relaxation and updates RFC 8231.

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 May 5, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	3
2. Overview	3
2.1. Usage Example	4
3. PCEP Extension	4
3.1. STATEFUL-PCE-CAPABILITY TLV	4
3.2. Handling of P flag	5
3.2.1. The PCRpt Message	5
3.2.2. The PCUpd Message and the PCInitiate Message	5
3.3. Handling of I flag	5
3.3.1. The PCUpd Message	5
3.3.2. The PCRpt Message	6
3.3.3. The PCInitiate Message	6
3.4. Delegation	6
3.5. Unknown Object Handling	6
4. Security Considerations	7
5. IANA Considerations	7
5.1. STATEFUL-PCE-CAPABILITY TLV	7
6. Manageability Considerations	7
6.1. Control of Function and Policy	7
6.2. Information and Data Models	7
6.3. Liveness Detection and Monitoring	8
6.4. Verify Correct Operations	8
6.5. Requirements On Other Protocols	8
6.6. Impact On Network Operations	8
7. Acknowledgments	8
8. References	8
8.1. Normative References	8
8.2. Informative References	9
Appendix A. Contributors	11
Authors' Addresses	11

1. Introduction

[RFC5440] describes the Path Computation Element Communication Protocol (PCEP) which 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 Stateful PCE Model [RFC8231] describes a set of extensions to PCEP to 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 control.

[RFC5440] defined the P flag (Processing-Rule) in the Common Object Header to allow a PCC to specify in a Path Computation Request (PCReq) message (sent to a PCE) whether the object must be taken into account by the PCE during path computation or is optional. The I flag (Ignore) is used by the PCE in a Path Computation Reply (PCRep) message to indicate to a PCC whether or not an optional object was considered by the PCE during path computation. Stateful PCE [RFC8231] specified that the P and I flags of the PCEP objects defined in [RFC8231] is to be set to zero on transmission and ignored on receipt, since they are exclusively related to path computation requests. The behavior for P and I flag in other messages defined in [RFC5440] and other extension was not specified. This document clarifies how the P and I flag could be used in the stateful PCE model to identify optional objects in the Path Computation State Report (PCRpt) [RFC8231], the Path Computation Update Request (PCUpd) [RFC8231], and the LSP Initiate Request (PCInitiate) [RFC8281] message.

This document updates [RFC8231] with respect to usage of the P and I flag as well as the handling of unknown objects in the stateful PCEP message exchange.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Overview

[RFC5440] describes the handling of unknown objects as per the setting of the P flag for the PCReq message. Further, [RFC8231] defined the usage of the LSP Error Code TLV in the PCRpt message in response to failed LSP Update Request via the PCUpd message (for example, due to an unsupported object/TLV).

This document clarifies the procedure of marking some objects as 'optional to be processed' by the PCEP peer in the stateful PCEP messages. Furthermore, this document updates the procedure for handling unknown objects in the stateful PCEP messages based on the P flag.

2.1. Usage Example

The PCRpt message is used to report the current state of an LSP. As part of the message both the <intended-attribute-list> and <actual-attribute-list> is encoded (see [RFC8231]). For example, the <intended-attribute-list> could include the METRIC object to indicate a limiting constraint (B flag set) for the Path Delay Variation metric [RFC8233]. In some scenarios, it would be useful to state that this limiting constraint can be relaxed by the PCE in case it cannot find a path. Similarly in the case of an association group [RFC8697] such as Disjoint Association [RFC8800], the PCE may need to completely relax the disjointness constraint in order to provide a path to all the LSPs that are part of the association. In these case it would be useful to mark the objects as 'optional' and it could be ignored by the PCEP peer. Also, it would be useful for the PCEP speaker to learn if the PCEP peer has relaxed the constraint and ignored the processing of the PCEP object.

Thus, this document simply clarifies, how the already existing P and I flag in the PCEP common object header could be used during the stateful PCEP message exchange.

3. PCEP Extension

3.1. STATEFUL-PCE-CAPABILITY TLV

A PCEP speaker indicates its ability to support the handling of the P and I flag in the stateful PCEP message exchange during the PCEP initialization phase, as follows. When the PCEP session is established, a PCC sends an Open message with an OPEN object that contains the STATEFUL-PCE-CAPABILITY TLV, as defined in [RFC8231]. A new flag, the R (RELAX) flag, is added in this TLV to indicate the support for relaxing the processing of some objects via the use of the P and I flag in the PCEP common object header.

R (RELAX bit - TBD1): If set to 1 by a PCEP Speaker, the R flag indicates that the PCEP Speaker is willing to send and receive PCEP objects with the P and I flags in the PCEP common object header for the stateful PCE messages. In case the bit is unset, it indicates that the PCEP Speaker would not handle the P and I flags in the PCEP common object header for stateful PCE messages.

The R flag MUST be set by both a PCC and a PCE to indicate support for the handling of the P and I flag in the PCEP common object header to allow relaxing some constraints by marking objects as optional to process. If the PCEP speaker that did not set the R flag but receives PCEP objects with P or I bit set, MUST behave as per the processing rule in [RFC8231] i.e., the bits are simply ignored.

3.2. Handling of P flag

3.2.1. The PCRpt Message

The P flag in the PCRpt message [RFC8231] allows a PCC to specify to a PCE whether the object must be taken into account by the PCE (during path computation, re-optimization, or state maintenance) or is optional to process. When the P flag is set in the PCRpt message received on a PCEP session on which R bit was set by both peers, the object MUST be taken into account by the PCE. Conversely, when the P flag is cleared, the object is optional and the PCE is free to ignore it. The P flag for the mandatory objects such as the LSP and the ERO object (intended path) MUST be set in the PCRpt message. If a mandatory object is received with the P flag set incorrectly according to the rules stated above, the receiving peer MUST send a PCErr message with Error-Type=10 (Reception of an invalid object) and Error-value=1 (reception of an object with P flag not set). On a PCEP session on which R bit was set by both peers, the PCC SHOULD set the P flag by default, unless a local configuration or local policy indicates that some constraints (corresponding PCEP objects) can be marked as optional and could be ignored by the PCE.

3.2.2. The PCUpd Message and the PCInitiate Message

The P flag in the PCUpd message [RFC8231] and the PCInitiate message [RFC8281] allows a PCE to specify to a PCC whether the object must be taken into account by the PCC (during path setup) or is optional to process. When the P flag is set in the PCUpd/PCInitiate message received on a PCEP session on which R bit was set by both peers, the object MUST be taken into account by the PCC. Conversely, when the P flag is cleared, the object is optional and the PCC is free to ignore it. The P flag for the mandatory objects such as the SRP, the LSP and the ERO MUST be set in the PCUpd/PCInitiate message. If a mandatory object is received with the P flag set incorrectly according to the rules stated above, the receiving peer MUST send a PCErr message with Error-Type=10 (Reception of an invalid object) and Error-value=1 (reception of an object with P flag not set). By default, the PCE SHOULD set the P flag, unless a local configuration or local policy indicates that some constraints (corresponding PCEP objects) can be marked as optional and could be ignored by the PCC.

3.3. Handling of I flag

3.3.1. The PCUpd Message

The I flag in the PCUpd message [RFC8231] allows a PCE to indicate to a PCC whether or not an optional object was processed. The PCE MAY include the ignored optional object in its update request and set the

I flag to indicate that the optional object was ignored. When the I flag is cleared, the PCE indicates that the optional object was processed.

3.3.2. The PCRpt Message

The I flag in the PCRpt message [RFC8231] allows a PCC to indicate to a PCE whether or not an optional object was processed in response to an LSP Update Request (PCUpd) or LSP Initiate Request (PCInitiate). The PCC MAY include the ignored optional object in its report and set the I flag to indicate that the optional object was ignored at PCC. When the I flag is cleared, the PCC indicates that the optional object was processed. The I flag has no meaning if the PCRpt message is not in response to a PCUpd or PCInitiate message (i.e. without the SRP object in the PCRpt message).

3.3.3. The PCInitiate Message

The I flag has no meaning in the PCinitiate message [RFC8281] and is ignored.

3.4. Delegation

Delegation is an operation to grant a PCE temporary rights to modify a subset of parameters on one or more LSPs by a PCC as described in [RFC8051]. Note that for the delegated LSPs, the PCE can update and mark some object as ignored even when the PCC had set the P flag during delegation. Similarly, the PCE can update and mark some object as a must to process even when the PCC had not set the P flag during delegation.

The PCC MUST acknowledge this by sending the PCRpt message with the P flag set as per the PCE expectation for the corresponding object. In case PCC cannot accept this, it would react as per the processing rules of unacceptable update in [RFC8231].

3.5. Unknown Object Handling

This document updates the handling of unknown objects in stateful PCEP messages as per the setting of P flag in the common object header in a similar way as [RFC5440], i.e. if a PCEP speaker does not understand an object with the P flag set or understands the object but decides to ignore the object, the entire stateful PCEP message MUST be rejected and the PCE MUST send a PCErr message with Error-Type="Unknown Object" or "Not supported Object" [RFC5440]. In case the P flag is not set, the PCEP speaker is free to ignore the object and continue with message processing as defined.

[RFC8231] defined LSP Error Code TLV to be carried in PCRpt message in the LSP object to convey error information. This document does not change that procedure.

4. Security Considerations

This document clarifies how the already existing P and I flag in PCEP common object header could be used during stateful PCEP exchanges. It updates the unknown object error handling in stateful PCEP message exchange. These changes on its own do not add any new security concerns. The security considerations identified in [RFC5440], [RFC8231], and [RFC8281] continue to apply.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

5. IANA Considerations

5.1. STATEFUL-PCE-CAPABILITY TLV

[RFC8231] defines the STATEFUL-PCE-CAPABILITY TLV; per that RFC, IANA created a "STATEFUL-PCE-CAPABILITY TLV Flag Field" subregistry to manage the value of the STATEFUL-PCE-CAPABILITY TLV's Flag field. IANA is requested to allocate a new bit in the subregistry, as follows:

Bit	Description	Reference
TBD1	RELAX bit	[This-I.D.]

6. Manageability Considerations

6.1. Control of Function and Policy

An operator MUST be allowed to configure the capability to support relaxation of constraints in the stateful PCEP message exchange. They SHOULD also allow configuration of related LSP constraints (or parameters) that are optional to process.

6.2. Information and Data Models

An implementation SHOULD allow the operator to view the capability defined in this document. To serve this purpose, the PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended in the future.

6.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

6.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440].

6.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.

6.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440].

7. Acknowledgments

Thanks to Jonathan Hardwick for discussion and suggestions around this draft.

8. References

8.1. Normative References

- [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>>.
- [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>>.

8.2. Informative References

- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-15 (work in progress), October 2020.
- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki, "Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September 2017, <<https://www.rfc-editor.org/info/rfc8233>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.

- [RFC8697] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "Path Computation Element Communication Protocol (PCEP) Extensions for Establishing Relationships between Sets of Label Switched Paths (LSPs)", RFC 8697, DOI 10.17487/RFC8697, January 2020, <<https://www.rfc-editor.org/info/rfc8697>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.

Appendix A. Contributors

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Authors' Addresses

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan, Guangdong 523808
China

Email: zhenghaomian@huawei.com

Stephane Litkowski
Cisco

Email: slitkows.ietf@gmail.com

PCE Working Group
Internet-Draft
Updates: 8231 (if approved)
Intended status: Standards Track
Expires: November 6, 2021

C. Li
H. Zheng
Huawei Technologies
S. Litkowski
Cisco
May 5, 2021

Extension for Stateful PCE to allow Optional Processing of PCEP Objects
draft-dhody-pce-stateful-pce-optional-08

Abstract

This document introduces a mechanism to mark some of the Path Computation Element (PCE) Communication Protocol (PCEP) objects as optional during PCEP messages exchange for the Stateful PCE model to allow relaxing some constraints. This document introduces this relaxation and updates RFC 8231.

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 November 6, 2021.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	3
2. Overview	3
2.1. Usage Example	4
3. PCEP Extension	4
3.1. STATEFUL-PCE-CAPABILITY TLV	4
3.2. Handling of P flag	5
3.2.1. The PCRpt Message	5
3.2.2. The PCUpd Message and the PCInitiate Message	5
3.3. Handling of I flag	5
3.3.1. The PCUpd Message	5
3.3.2. The PCRpt Message	6
3.3.3. The PCInitiate Message	6
3.4. Delegation	6
3.5. Unknown Object Handling	6
4. Security Considerations	7
5. IANA Considerations	7
5.1. STATEFUL-PCE-CAPABILITY TLV	7
6. Manageability Considerations	7
6.1. Control of Function and Policy	7
6.2. Information and Data Models	7
6.3. Liveness Detection and Monitoring	8
6.4. Verify Correct Operations	8
6.5. Requirements On Other Protocols	8
6.6. Impact On Network Operations	8
7. Acknowledgments	8
8. References	8
8.1. Normative References	8
8.2. Informative References	9
Appendix A. Contributors	11
Authors' Addresses	11

1. Introduction

[RFC5440] describes the Path Computation Element Communication Protocol (PCEP) which 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 Stateful PCE Model [RFC8231] describes a set of extensions to PCEP to 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 control.

[RFC5440] defined the P flag (Processing-Rule) in the Common Object Header to allow a PCC to specify in a Path Computation Request (PCReq) message (sent to a PCE) whether the object must be taken into account by the PCE during path computation or is optional. The I flag (Ignore) is used by the PCE in a Path Computation Reply (PCRep) message to indicate to a PCC whether or not an optional object was considered by the PCE during path computation. Stateful PCE [RFC8231] specified that the P and I flags of the PCEP objects defined in [RFC8231] is to be set to zero on transmission and ignored on receipt, since they are exclusively related to path computation requests. The behavior for P and I flag in other messages defined in [RFC5440] and other extension was not specified. This document clarifies how the P and I flag could be used in the stateful PCE model to identify optional objects in the Path Computation State Report (PCRpt) [RFC8231], the Path Computation Update Request (PCUpd) [RFC8231], and the LSP Initiate Request (PCInitiate) [RFC8281] message.

This document updates [RFC8231] with respect to usage of the P and I flag as well as the handling of unknown objects in the stateful PCEP message exchange.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Overview

[RFC5440] describes the handling of unknown objects as per the setting of the P flag for the PCReq message. Further, [RFC8231] defined the usage of the LSP Error Code TLV in the PCRpt message in response to failed LSP Update Request via the PCUpd message (for example, due to an unsupported object/TLV).

This document clarifies the procedure of marking some objects as 'optional to be processed' by the PCEP peer in the stateful PCEP messages. Furthermore, this document updates the procedure for handling unknown objects in the stateful PCEP messages based on the P flag.

2.1. Usage Example

The PCRpt message is used to report the current state of an LSP. As part of the message both the <intended-attribute-list> and <actual-attribute-list> is encoded (see [RFC8231]). For example, the <intended-attribute-list> could include the METRIC object to indicate a limiting constraint (B flag set) for the Path Delay Variation metric [RFC8233]. In some scenarios, it would be useful to state that this limiting constraint can be relaxed by the PCE in case it cannot find a path. Similarly in the case of an association group [RFC8697] such as Disjoint Association [RFC8800], the PCE may need to completely relax the disjointness constraint in order to provide a path to all the LSPs that are part of the association. In these case it would be useful to mark the objects as 'optional' and it could be ignored by the PCEP peer. Also, it would be useful for the PCEP speaker to learn if the PCEP peer has relaxed the constraint and ignored the processing of the PCEP object.

Thus, this document simply clarifies, how the already existing P and I flag in the PCEP common object header could be used during the stateful PCEP message exchange.

3. PCEP Extension

3.1. STATEFUL-PCE-CAPABILITY TLV

A PCEP speaker indicates its ability to support the handling of the P and I flag in the stateful PCEP message exchange during the PCEP initialization phase, as follows. When the PCEP session is established, a PCC sends an Open message with an OPEN object that contains the STATEFUL-PCE-CAPABILITY TLV, as defined in [RFC8231]. A new flag, the R (RELAX) flag, is added in this TLV to indicate the support for relaxing the processing of some objects via the use of the P and I flag in the PCEP common object header.

R (RELAX bit - TBD1): If set to 1 by a PCEP Speaker, the R flag indicates that the PCEP Speaker is willing to send and receive PCEP objects with the P and I flags in the PCEP common object header for the stateful PCE messages. In case the bit is unset, it indicates that the PCEP Speaker would not handle the P and I flags in the PCEP common object header for stateful PCE messages.

The R flag MUST be set by both a PCC and a PCE to indicate support for the handling of the P and I flag in the PCEP common object header to allow relaxing some constraints by marking objects as optional to process. If the PCEP speaker did not set the R flag but receives PCEP objects with P or I bit set, it MUST behave as per the processing rule in [RFC8231] i.e., the bits are simply ignored.

3.2. Handling of P flag

3.2.1. The PCRpt Message

The P flag in the PCRpt message [RFC8231] allows a PCC to specify to a PCE whether the object must be taken into account by the PCE (during path computation, re-optimization, or state maintenance) or is optional to process. When the P flag is set in the PCRpt message received on a PCEP session on which R bit was set by both peers, the object MUST be taken into account by the PCE. Conversely, when the P flag is cleared, the object is optional and the PCE is free to ignore it. The P flag for the mandatory objects such as the LSP and the ERO object (intended path) MUST be set in the PCRpt message. If a mandatory object is received with the P flag set incorrectly according to the rules stated above, the receiving peer MUST send a PCErr message with Error-Type=10 (Reception of an invalid object) and Error-value=1 (reception of an object with P flag not set). On a PCEP session on which R bit was set by both peers, the PCC SHOULD set the P flag by default, unless a local configuration or local policy indicates that some constraints (corresponding PCEP objects) can be marked as optional and could be ignored by the PCE.

3.2.2. The PCUpd Message and the PCInitiate Message

The P flag in the PCUpd message [RFC8231] and the PCInitiate message [RFC8281] allows a PCE to specify to a PCC whether the object must be taken into account by the PCC (during path setup) or is optional to process. When the P flag is set in the PCUpd/PCInitiate message received on a PCEP session on which R bit was set by both peers, the object MUST be taken into account by the PCC. Conversely, when the P flag is cleared, the object is optional and the PCC is free to ignore it. The P flag for the mandatory objects such as the SRP, the LSP and the ERO MUST be set in the PCUpd/PCInitiate message. If a mandatory object is received with the P flag set incorrectly according to the rules stated above, the receiving peer MUST send a PCErr message with Error-Type=10 (Reception of an invalid object) and Error-value=1 (reception of an object with P flag not set). By default, the PCE SHOULD set the P flag, unless a local configuration or local policy indicates that some constraints (corresponding PCEP objects) can be marked as optional and could be ignored by the PCC.

3.3. Handling of I flag

3.3.1. The PCUpd Message

The I flag in the PCUpd message [RFC8231] allows a PCE to indicate to a PCC whether or not an optional object was processed. The PCE MAY include the ignored optional object in its update request and set the

I flag to indicate that the optional object was ignored. When the I flag is cleared, the PCE indicates that the optional object was processed.

3.3.2. The PCRpt Message

The I flag in the PCRpt message [RFC8231] allows a PCC to indicate to a PCE whether or not an optional object was processed in response to an LSP Update Request (PCUpd) or LSP Initiate Request (PCInitiate). The PCC MAY include the ignored optional object in its report and set the I flag to indicate that the optional object was ignored at PCC. When the I flag is cleared, the PCC indicates that the optional object was processed. The I flag has no meaning if the PCRpt message is not in response to a PCUpd or PCInitiate message (i.e. without the SRP object in the PCRpt message).

3.3.3. The PCInitiate Message

The I flag has no meaning in the PCinitiate message [RFC8281] and is ignored.

3.4. Delegation

Delegation is an operation to grant a PCE temporary rights to modify a subset of parameters on one or more LSPs by a PCC as described in [RFC8051]. Note that for the delegated LSPs, the PCE can update and mark some objects as ignored even when the PCC had set the P flag during delegation. Similarly, the PCE can update and mark some object as a must to process even when the PCC had not set the P flag during delegation.

The PCC MUST acknowledge this by sending the PCRpt message with the P flag set as per the PCE expectation for the corresponding object. In case PCC cannot accept this, it would react as per the processing rules of unacceptable update in [RFC8231].

3.5. Unknown Object Handling

This document updates the handling of unknown objects in the stateful PCEP messages as per the setting of the P flag in the common object header in a similar way as [RFC5440], i.e. if a PCEP speaker does not understand an object with the P flag set or understands the object but decides to ignore the object, the entire stateful PCEP message MUST be rejected and the PCE MUST send a PCErr message with Error-Type="Unknown Object" or "Not supported Object" [RFC5440]. In case the P flag is not set, the PCEP speaker is free to ignore the object and continue with message processing as defined.

[RFC8231] defined LSP Error Code TLV to be carried in PCRpt message in the LSP object to convey error information. This document does not change that procedure.

4. Security Considerations

This document clarifies how the already existing P and I flag in PCEP common object header could be used during stateful PCEP exchanges. It updates the unknown object error handling in stateful PCEP message exchange. These changes on their own do not add any new security concerns. The security considerations identified in [RFC5440], [RFC8231], and [RFC8281] continue to apply.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

5. IANA Considerations

5.1. STATEFUL-PCE-CAPABILITY TLV

[RFC8231] defines the STATEFUL-PCE-CAPABILITY TLV; per that RFC, IANA created a "STATEFUL-PCE-CAPABILITY TLV Flag Field" subregistry to manage the value of the STATEFUL-PCE-CAPABILITY TLV's Flag field. IANA is requested to allocate a new bit in the subregistry, as follows:

Bit	Description	Reference
<hr/>		
TBD1	RELAX bit	[This-I.D.]

6. Manageability Considerations

6.1. Control of Function and Policy

An operator MUST be allowed to configure the capability to support relaxation of constraints in the stateful PCEP message exchange. They SHOULD also allow configuration of related LSP constraints (or parameters) that are optional to process.

6.2. Information and Data Models

An implementation SHOULD allow the operator to view the capability defined in this document. To serve this purpose, the PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended in the future.

6.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

6.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440].

6.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.

6.6. Impact On Network Operations

Mechanisms defined in this document do not have any impact on network operations in addition to those already listed in [RFC5440].

7. Acknowledgments

Thanks to Jonathan Hardwick for discussion and suggestions around this draft.

8. References

8.1. Normative References

- [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>>.
- [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>>.

8.2. Informative References

- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-16 (work in progress), February 2021.
- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki, "Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September 2017, <<https://www.rfc-editor.org/info/rfc8233>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.

- [RFC8697] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "Path Computation Element Communication Protocol (PCEP) Extensions for Establishing Relationships between Sets of Label Switched Paths (LSPs)", RFC 8697, DOI 10.17487/RFC8697, January 2020, <<https://www.rfc-editor.org/info/rfc8697>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.

Appendix A. Contributors

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Authors' Addresses

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan, Guangdong 523808
China

Email: zhenghaomian@huawei.com

Stephane Litkowski
Cisco

Email: slitkows.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 5, 2021

C. Li
H. Zheng
Huawei Technologies
S. Sivabalan
Ciena
S. Sidor
Z. Ali
Cisco Systems, Inc.
November 1, 2020

Conveying Vendor-Specific Information in the Path Computation Element
(PCE) Communication Protocol (PCEP) extensions for Stateful PCE.
draft-dhody-pce-stateful-pce-vendor-11

Abstract

A Stateful Path Computation Element (PCE) maintains information on the current network state, including: computed Label Switched Path (LSPs), reserved resources within the network, and the pending path computation requests. This information may then be considered when computing new traffic engineered LSPs, and for the associated and the dependent LSPs, received from a Path Computation Client (PCC).

RFC 7470 defines a facility to carry vendor-specific information in Path Computation Element Communication Protocol (PCEP).

This document extends this capability for the Stateful PCEP messages.

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 May 5, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements Language	3
2. Procedures for the Vendor Information Object	3
3. Procedures for the Vendor Information TLV	6
4. Vendor Information Object and TLV	6
5. Manageability Considerations	7
5.1. Control of Function and Policy	7
5.2. Information and Data Models	7
5.3. Liveness Detection and Monitoring	7
5.4. Verify Correct Operations	7
5.5. Requirements On Other Protocols	7
5.6. Impact On Network Operations	7
6. IANA Considerations	8
7. Implementation Status	8
7.1. Cisco Systems	8
8. Security Considerations	9
9. Acknowledgments	9
10. References	9
10.1. Normative References	9
10.2. Informative References	10
Appendix A. Contributor Addresses	11
Authors' Addresses	11

1. Introduction

The Path Computation Element Communication Protocol (PCEP) [RFC5440] provides mechanisms for a Path Computation Element (PCE) to perform path computation in response to a Path Computation Client (PCC) request.

A Stateful PCE is capable of considering, for the purposes of the path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB)). [RFC8051] describes general considerations for a Stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

[RFC8231] describes a set of extensions to PCEP to provide stateful control. A Stateful PCE has access to not only the information carried by the network's Interior Gateway Protocol (IGP), but also the set of active paths and their reserved resources for its computations. The additional state allows the PCE to compute constrained paths while considering individual LSPs and their interactions. [RFC8281] describes the set up, maintenance and teardown of PCE-initiated LSPs under the Stateful PCE model. These extensions added new messages in PCEP for Stateful PCE.

[RFC7470] defined Vendor Information object that can be used to carry arbitrary, proprietary information such as vendor-specific constraints. It also defined VENDOR-INFORMATION-TLV that can be used to carry arbitrary information within any existing or future PCEP object that supports TLVs.

This document extend the usage of Vendor Information Object and VENDOR-INFORMATION-TLV to Stateful PCE. The VENDOR-INFORMATION-TLV can be carried inside any of the new objects added in PCEP for Stateful PCE as per [RFC7470], this document extend the stateful PCEP messages to also include the Vendor Information Object as well.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Procedures for the Vendor Information Object

A Path Computation LSP State Report message [RFC8231] (also referred to as PCRpt message) is a PCEP message sent by a PCC to a PCE to report the current state of an LSP. A PCC that wants to convey proprietary or vendor-specific information or metrics to a PCE does so by including a Vendor Information object in the PCRpt message. The contents and format of the object are described in Section 4 of

[RFC7470]. The PCE determines how to interpret the information in the Vendor Information object by examining the Enterprise Number it contains.

The Vendor Information object is OPTIONAL in a PCRpt message. Multiple instances of the object MAY be used on a single PCRpt message. Different instances of the object can have different Enterprise Numbers.

The format of the PCRpt message (with [RFC8231] as base) is updated as follows:

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

Where:

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

```
<state-report> ::= [<SRP>]
                    <LSP>
                    <path>
                    [<vendor-info-list>]
```

Where:

```
<vendor-info-list> ::= <VENDOR-INFORMATION>
                        [<vendor-info-list>]
```

<path> is defined in [RFC8231].

A Path Computation LSP Update Request message (also referred to as PCUpd message) is a PCEP message sent by a PCE to a PCC to update attributes of an LSP. The Vendor Information object can be included in a PCUpd message to convey proprietary or vendor-specific information.

The format of the PCUpd message (with [RFC8231] as base) is updated as follows:

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

Where:

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

```
<update-request> ::= <SRP>
                        <LSP>
                        <path>
                        [<vendor-info-list>]
```

Where:

```
<vendor-info-list> ::= <VENDOR-INFORMATION>
                        [<vendor-info-list>]
```

<path> is defined in [RFC8231].

A Path Computation LSP Initiate Message (also referred to as PCInitiate message) is a PCEP message sent by a PCE to a PCC to trigger an LSP instantiation or deletion. The Vendor Information object can be included in a PCInitiate message to convey proprietary or vendor-specific information.

The format of the PCInitiate message (with [RFC8281] as base) is updated as follows:

```
<PCInitiate Message> ::= <Common Header>
                           <PCE-initiated-lsp-list>
```

Where:

```
<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-instantiation> ::= <SRP>
                                         <LSP>
                                         [<END-POINTS>]
                                         <ERO>
                                         [<attribute-list>]
                                         [<vendor-info-list>]
```

Where:

```
<vendor-info-list> ::= <VENDOR-INFORMATION>
                        [<vendor-info-list>]
```

<PCE-initiated-lsp-deletion> and <attribute-list> is as per [RFC8281].

A legacy implementation that does not recognize the Vendor Information object will act according to the procedures set out in [RFC8231] and [RFC8281]. An implementation that supports the Vendor Information object, but receives one carrying an Enterprise Number that it does not support, MUST ignore the object in the same way as described in [RFC7470].

3. Procedures for the Vendor Information TLV

The Vendor Information TLV can be used to carry vendor-specific information that applies to a specific PCEP object by including the TLV in the object. This includes objects used in Stateful PCE extension such as SRP and LSP object. All the procedures as per section 3 of [RFC7470].

4. Vendor Information Object and TLV

[RFC7470] specify the format of VENDOR-INFORMATION Object and VENDOR-INFORMATION-TLV.

5. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440], [RFC7470] and [RFC8231] apply to PCEP protocol extensions defined in this document. In addition, requirements and considerations listed in this section apply.

5.1. Control of Function and Policy

As stated in [RFC7470], this capability, the associated vendor specific information and policy SHOULD made configurable. This information can be used in Stateful PCEP messages as well.

5.2. Information and Data Models

The PCEP YANG module is specified in [I-D.ietf-pce-pcep-yang]. It is NOT RECOMMENDED that standard YANG module be augmented with details of vendor information. It MAY be extended to include the use of this information and the Enterprise Numbers that the object and TLVs contain.

5.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

5.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

5.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.

5.6. Impact On Network Operations

Mechanisms defined in [RFC5440] and [RFC8231] also apply to PCEP extensions defined in this document. Further, the mechanism described in this document can help the operator to request control of the LSPs at a particular PCE.

6. IANA Considerations

There are no IANA consideration in this document.

7. Implementation Status

[NOTE TO RFC EDITOR : This whole section and the reference to RFC 7942 is to be removed before publication as an RFC]

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".

7.1. Cisco Systems

- o Organization: Cisco Systems, Inc.
- o Implementation: Cisco IOS-XR PCE and PCC
- o Description: Vendor Information Object used in PCRpt, PCUpd and PCInitiate messages.
- o Maturity Level: Production
- o Coverage: Full
- o Contact: ssidor@cisco.com

8. Security Considerations

The protocol extensions defined in this document do not change the nature of PCEP. Therefore, the security considerations set out in [RFC5440], [RFC7470], [RFC8231] and [RFC8281] apply unchanged.

As stated in [RFC6952], PCEP implementations SHOULD support the TCP-AO [RFC5925] and not use TCP MD5 because of TCP MD5's known vulnerabilities and weakness. PCEP also support Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525].

9. Acknowledgments

Thanks to Avantika, Mahendra Singh Negi, Udayasree Palle and Swapna K for their suggestions.

10. References

10.1. Normative References

- [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>>.
- [RFC7470] Zhang, F. and A. Farrel, "Conveying Vendor-Specific Constraints in the Path Computation Element Communication Protocol", RFC 7470, DOI 10.17487/RFC7470, March 2015, <<https://www.rfc-editor.org/info/rfc7470>>.
- [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>>.

10.2. Informative References

- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-15 (work in progress), October 2020.
- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <<https://www.rfc-editor.org/info/rfc5925>>.
- [RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <<https://www.rfc-editor.org/info/rfc6952>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [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>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Mike Koldychev
Cisco Systems
Kanata, Ontario
Canada

EMail: mkoldych@cisco.com

Authors' Addresses

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan, Guangdong 523808
P.R.China

Email: zhenghaomian@huawei.com

Siva Sivabalan
Ciena
385 Terry Fox Drive
Kanata, Ontario K2K 0L1
Canada

Email: msiva282@gmail.com

Samuel Sidor
Cisco Systems, Inc.

Email: ssidor@cisco.com

Zafar Ali
Cisco Systems, Inc.

Email: zali@cisco.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 October 2022

C. Li
H. Zheng
Huawei Technologies
S. Sivabalan
Ciena
S. Sidor
Z. Ali
Cisco Systems, Inc.
19 April 2022

Conveying Vendor-Specific Information in the Path Computation Element
(PCE) Communication Protocol (PCEP) extensions for Stateful PCE.
draft-dhody-pce-stateful-pce-vendor-14

Abstract

A Stateful Path Computation Element (PCE) maintains information on the current network state, including: computed Label Switched Path (LSPs), reserved resources within the network, and the pending path computation requests. This information may then be considered when computing new traffic engineered LSPs, and for the associated and the dependent LSPs, received from a Path Computation Client (PCC).

RFC 7470 defines a facility to carry vendor-specific information in Path Computation Element Communication Protocol (PCEP).

This document extends this capability for the Stateful PCEP messages.

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 21 October 2022.

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
1.1. Requirements Language	3
2. Procedures for the Vendor Information Object	3
3. Procedures for the Vendor Information TLV	6
4. Vendor Information Object and TLV	6
5. Manageability Considerations	7
5.1. Control of Function and Policy	7
5.2. Information and Data Models	7
5.3. Liveness Detection and Monitoring	7
5.4. Verify Correct Operations	7
5.5. Requirements On Other Protocols	7
5.6. Impact On Network Operations	7
6. IANA Considerations	8
7. Implementation Status	8
7.1. Cisco Systems	8
8. Security Considerations	9
9. Acknowledgments	9
10. References	9
10.1. Normative References	9
10.2. Informative References	10
Appendix A. Contributor Addresses	11
Authors' Addresses	11

1. Introduction

The Path Computation Element Communication Protocol (PCEP) [RFC5440] provides mechanisms for a Path Computation Element (PCE) to perform path computation in response to a Path Computation Client (PCC) request.

A Stateful PCE is capable of considering, for the purposes of the path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB)). [RFC8051] describes general considerations for a Stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

[RFC8231] describes a set of extensions to PCEP to provide stateful control. A Stateful PCE has access to not only the information carried by the network's Interior Gateway Protocol (IGP), but also the set of active paths and their reserved resources for its computations. The additional state allows the PCE to compute constrained paths while considering individual LSPs and their interactions. [RFC8281] describes the set up, maintenance and teardown of PCE-initiated LSPs under the Stateful PCE model. These extensions added new messages in PCEP for Stateful PCE.

[RFC7470] defined Vendor Information object that can be used to carry arbitrary, proprietary information such as vendor-specific constraints. It also defined VENDOR-INFORMATION-TLV that can be used to carry arbitrary information within any existing or future PCEP object that supports TLVs.

This document extend the usage of Vendor Information Object and VENDOR-INFORMATION-TLV to Stateful PCE. The VENDOR-INFORMATION-TLV can be carried inside any of the new objects added in PCEP for Stateful PCE as per [RFC7470], this document extend the stateful PCEP messages to also include the Vendor Information Object as well.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Procedures for the Vendor Information Object

A Path Computation LSP State Report message (also referred to as PCRpt message) [RFC8231] is a PCEP message sent by a PCC to a PCE to report the current state of an LSP. A PCC that wants to convey proprietary or vendor-specific information or metrics to a PCE does so by including a Vendor Information object in the PCRpt message. The contents and format of the object are described in Section 4 of

[RFC7470]. The PCE determines how to interpret the information in the Vendor Information object by examining the Enterprise Number it contains.

The Vendor Information object is OPTIONAL in a PCRpt message. Multiple instances of the object MAY be used on a single PCRpt message. Different instances of the object can have different Enterprise Numbers.

The format of the PCRpt message (with [RFC8231] as base) is updated as follows:

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

Where:

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

```
<state-report> ::= [<SRP>]
                    <LSP>
                    <path>
                    [<vendor-info-list>]
```

Where:

```
<vendor-info-list> ::= <VENDOR-INFORMATION>
                        [<vendor-info-list>]
```

<path> is defined in [RFC8231].

A Path Computation LSP Update Request message (also referred to as PCUpd message) [RFC8231] is a PCEP message sent by a PCE to a PCC to update attributes of an LSP. The Vendor Information object can be included in a PCUpd message to convey proprietary or vendor-specific information.

The format of the PCUpd message (with [RFC8231] as base) is updated as follows:

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

Where:

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

```
<update-request> ::= <SRP>  
                    <LSP>  
                    <path>  
                    [<vendor-info-list>]
```

Where:

```
<vendor-info-list> ::= <VENDOR-INFORMATION>  
                    [<vendor-info-list>]
```

<path> is defined in [RFC8231].

A Path Computation LSP Initiate Message (also referred to as PCInitiate message) [RFC8281] is a PCEP message sent by a PCE to a PCC to trigger an LSP instantiation or deletion. The Vendor Information object can be included in a PCInitiate message to convey proprietary or vendor-specific information.

The format of the PCInitiate message (with [RFC8281] as base) is updated as follows:

```
<PCInitiate Message> ::= <Common Header>
                           <PCE-initiated-lsp-list>
```

Where:

```
<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-instantiation> ::= <SRP>
                                       <LSP>
                                       [<END-POINTS>]
                                       <ERO>
                                       [<attribute-list>]
                                       [<vendor-info-list>]
```

Where:

```
<vendor-info-list> ::= <VENDOR-INFORMATION>
                       [<vendor-info-list>]
```

<PCE-initiated-lsp-deletion> and <attribute-list> is as per [RFC8281].

A legacy implementation that does not recognize the Vendor Information object will act according to the procedures set out in [RFC8231] and [RFC8281]. An implementation that supports the Vendor Information object, but receives one carrying an Enterprise Number that it does not support, MUST ignore the object in the same way as described in [RFC7470].

3. Procedures for the Vendor Information TLV

The Vendor Information TLV can be used to carry vendor-specific information that applies to a specific PCEP object by including the TLV in the object. This includes objects used in Stateful PCE extension such as SRP and LSP object. All the procedures as per section 3 of [RFC7470].

4. Vendor Information Object and TLV

[RFC7470] specify the format of VENDOR-INFORMATION Object and VENDOR-INFORMATION-TLV.

5. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440], [RFC7470], [RFC8231], and [RFC8281] apply to PCEP protocol extensions defined in this document. In addition, requirements and considerations listed in this section apply.

5.1. Control of Function and Policy

As stated in [RFC7470], this capability, the associated vendor specific information and policy SHOULD made configurable. This information can be used in Stateful PCEP messages as well.

5.2. Information and Data Models

The PCEP YANG module is specified in [I-D.ietf-pce-pcep-yang]. It is RECOMMENDED that standard YANG module not be augmented with details of vendor information. It MAY be extended to include the use of this information and the Enterprise Numbers that the object and TLVs contain.

5.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

5.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

5.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.

5.6. Impact On Network Operations

Mechanisms defined in [RFC5440] and [RFC8231] also apply to PCEP extensions defined in this document. Further, the mechanism described in this document can help the operator to request control of the LSPs at a particular PCE.

6. IANA Considerations

There are no IANA consideration in this document.

7. Implementation Status

[NOTE TO RFC EDITOR : This whole section and the reference to RFC 7942 is to be removed before publication as an RFC]

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".

7.1. Cisco Systems

- * Organization: Cisco Systems, Inc.
- * Implementation: Cisco IOS-XR PCE and PCC
- * Description: Vendor Information Object used in PCRpt, PCUpd and PCInitiate messages.
- * Maturity Level: Production
- * Coverage: Full
- * Contact: ssidior@cisco.com

8. Security Considerations

The protocol extensions defined in this document do not change the nature of PCEP. Therefore, the security considerations set out in [RFC5440], [RFC7470], [RFC8231] and [RFC8281] apply unchanged.

As stated in [RFC6952], PCEP implementations SHOULD support the TCP-AO [RFC5925] and not use TCP MD5 because of TCP MD5's known vulnerabilities and weakness. PCEP also support Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525].

9. Acknowledgments

Thanks to Avantika, Mahendra Singh Negi, Udayasree Palle and Swapna K for their suggestions.

10. References

10.1. Normative References

- [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>>.
- [RFC7470] Zhang, F. and A. Farrel, "Conveying Vendor-Specific Constraints in the Path Computation Element Communication Protocol", RFC 7470, DOI 10.17487/RFC7470, March 2015, <<https://www.rfc-editor.org/info/rfc7470>>.
- [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>>.

10.2. Informative References

- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-yang-18, 25 January 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-yang-18>>.
- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <<https://www.rfc-editor.org/info/rfc5925>>.
- [RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <<https://www.rfc-editor.org/info/rfc6952>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [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>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Mike Koldychev
Cisco Systems
Kanata, Ontario
Canada

EMail: mkoldych@cisco.com

Authors' Addresses

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing
100095
China
Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan
Guangdong, 523808
China
Email: zhenghaomian@huawei.com

Siva Sivabalan
Ciena
385 Terry Fox Drive
Kanata Ontario K2K 0L1
Canada
Email: msiva282@gmail.com

Samuel Sidor
Cisco Systems, Inc.
Email: ssidor@cisco.com

Zafar Ali
Cisco Systems, Inc.
Email: zali@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 3, 2021

H. Bidgoli, Ed.
Nokia
V. Voyer
Bell Canada
S. Rajarathinam
Nokia
E. Hemmati
Cisco System
T. Saad
Juniper Networks
S. Sivabalan
Ciena
October 30, 2020

PCEP extensions for p2mp sr policy
draft-hsd-pce-sr-p2mp-policy-02

Abstract

SR P2MP policies are set of policies that enable architecture for P2MP service delivery. This document specifies extensions to the Path Computation Element Communication Protocol (PCEP) that allow a stateful PCE to compute and initiate P2MP paths from a Root to a set of Leaves.

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 May 3, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Conventions used in this document	4
3. Overview of PCEP Operation in SR P2MP Network	4
3.1. High level view of P2MP Policy Objects	5
3.1.1. Shared Tree vs Non-Shared Replication Segment	6
3.2. existing drafts used for defining a P2MP Policy	7
3.2.1. Existing drafts/rfcs used by this draft	7
3.2.2. P2MP Policy Identification	8
3.2.3. Replication Segment Identificaton	9
3.3. High Level Procedures for P2MP SR LSP Instantiation	9
3.3.1. PCE-Init Procedure	9
3.3.2. PCC-Init Procedure	10
3.3.3. Comon Procedure	10
3.3.4. Global Optimiatiom of the Candidate Path	12
3.3.5. Fast Reroute	12
3.3.6. Connecting Replication Segment via Segment List	13
3.4. SR P2MP Policy and Replication Segment TLVs and Objects	14
3.4.1. SR P2MP Policy Objects	14
3.4.2. Replication Segment Objects	14
3.4.3. P2MP Policy vs Replication Segment	15
3.4.4. P2MP Policy and Replication Segment general considerations	15
4. Object Format	15
4.1. Open Message and Capablity Exchange	16
4.2. Symbolic Name in PCInit Message from PCC	17
4.3. P2MP Policy Specific Objects and TLVs	17
4.3.1. P2MP Policy Association Group for P2MP Policy	17
4.3.1.1. P2MP SR Policy Association Group Policy Identifiers TLV	17
4.3.1.2. P2MP SR Policy Association Group Candidate Path Identifiers TLV	18
4.3.1.3. P2MP SR Policy Association Group Candidate Path Attributes TLV	19
4.3.2. P2MP-END-POINTS Object	19
4.4. P2MP Policy and Replication Segment Identifier Object and TLV	21

4.4.1. Extension of the LSP Object, SR-P2MP-LSPID-TLV . . .	21
4.5. Replication Segment	22
4.5.1. The format of the replication segment message	23
4.5.2. Label action rules in replicating segment	23
4.5.3. SR-ERO Rules	24
4.5.3.1. SR-ERO subobject changes	24
5. Examples of PCEP messages between PCE and PCEP	25
5.1. PCE Initiate	25
5.2. PCC Initiate or PCE Initiate Respond	27
5.3. PCE P2MP Path-Instance Calculation and Replication Segment download	28
5.4. PCC Rpt for PCE Update and Init Messages	36
6. Tree Deletion	37
7. Fragmentation	38
8. Example Workflows	38
9. IANA Consideration	38
10. Security Considerations	38
11. Acknowledgments	38
12. References	38
12.1. Normative References	38
12.2. Informative References	39
Authors' Addresses	40

1. Introduction

The draft [draft-ietf-pim-sr-p2mp-policy] defines a variant of the SR Policy [draft-ietf-spring-segment-routing-policy] for constructing a P2MP segment to support multicast service delivery.

A Point-to-Multipoint (P2MP) Policy connects a Root node to a set of Leaf nodes, optionally through a set of intermediate replication nodes. We also define a Replication segment [draft-ietf-spring-sr-replication-segment], which corresponds to the state of a P2MP segment on a particular node, as an example the forwarding instructions for the replication SID.

A P2MP Policy is relevant on the root of the P2MP Tree and it contains candidate paths. The candidate paths are made of path-instances and each path-instance is constructed via replication segments. These replication segments are programmed on the root, leaves and optionally intermediate replication nodes.

It should be noted that two replication segments can be connected directly, or they can be connected or steered via unicast SR segment or a segment list.

For a P2MP Tree, a controller may be used to compute paths from a Root node to a set of Leaf nodes, optionally via a set of replication

nodes. A packet is replicated at the root node and optionally on Replication nodes towards each Leaf node.

We define two types of a P2MP Tree: Spray and Replication.

A Point-to-Multipoint service delivery could be via Ingress Replication (aka Spray in some SR context), i.e., the root unicast individual copies of traffic to each leaf. The corresponding P2MP Policy consists of replication segments only for the root and the leaves and they are connected via a unicast SR Segment.

A Point-to-Multipoint service delivery could also be via Downstream Replication (aka TreeSID in some SR context), i.e., the root and some downstream replication nodes replicate the traffic along the way as it traverses closer to the leaves.

The leaves and the root can be explicitly configured on the PCE or PCC can update the PCE with the information of the discovered root and leaves. As an example Multicast protocols like mvpn procedures [RFC6513] or pim can be used to discovery the leaves and roots on the PCC and update the PCE with these relevant information. The controller can calculate the P2MP Policy based on these info.

In all of above cases a set of new PCEP object and TLVs are needed to update and instantiate the P2MP tree. This draft explains the procedure needed to instantiate a P2MP TreeSID.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Overview of PCEP Operation in SR P2MP Network

After discovering the root and the leaves on the PCE (via different mechanism mentioned in previous sections) computes the P2MP Tree and identifying the relevant Replication routers, the PCE programs the PCCs with relevant information needed to create a P2MP Tree.

As per draft [draft-ietf-pim-sr-p2mp-policy] a P2MP Policy is defined by Root-ID, Tree-ID and a set of leaves. A P2MP policy is a variant of SR policy as such it uses the same concept as draft [draft-ietf-pce-segment-routing-policy-cpl]. In short a P2MP policy uses a collection of SR P2MP Candidate paths. Candidate paths are computed by the PCE and can be used for P2MP Tree redundancy, only a single candidate path is active at each time. Each candidate paths can be globally optimized and is consists of multiple path-instances.

A path-instance can be thought of as a P2MP LSP. If a candidate path needs to be globally optimized two path-instances can be programmed on the root node and via make before procedures the candidate path can be switched from path-instance 1 to the 2nd path-instance. The forwarding states of these path-instances are build via replication segments, in short each path-instance initiated on the root has its own set of replication segments on the Root, Transit and Leaf nodes.

A replication segment is set of forwarding instructions on a specific node. As an example the push information on the Root node or swap and outgoing interface information on the transit nodes or pop information on the bud and leaves nodes.

PCE could also calculate and download additional information for the replication segments, such as protections next-hops for link protection (FRR).

3.1. High level view of P2MP Policy Objects

o SR P2MP Policy

- * Is only relevant on the Root of the P2MP path and is a policy on PCE which contains information about:
 - + Root node of the P2MP Segment
 - + Leaf nodes of the P2MP Segment
 - + Tree-ID, which is a unique identifier of the P2MP tree on the Root
 - + It also contains a set of Candidate paths and their path-instances for P2MP tree redundancy and global optimization
 - + Optional Constrains used to build these candidate paths
 - + P2MP policy information is downloaded only on the Root node and is identified via <Root-ID, Tree-ID>

o Candidate Path:

- * Is used for P2MP Tree redundancy where the candidate path with the highest preference is the active path.
- * It can contain up to two path-instance for global optimization procedures (i.e. make before break), each identified with their own path-instance ID

- * Contains information about protocol-id, originator, discriminator, preference, path-instances
- o Replication Segment:
 - * Is the forwarding information needed on each node for building the forwarding path for each path-instance of the P2MP Candidate path.
 - * As it will be explained in upcoming sections there are 2 ways to identify the replication segment, shared and non-shared
 - + It is identified via Tree-ID and Root-ID and path-instance for non-shared replication segment.
 - + It is identified via Node-ID, Replication-ID, for shared replication segment
 - + Contains forwarding instructions for the path-instances
 - + On the forwarding plane the Replication Segment is identified via the incoming Replication SID.
 - + Replication segment information is downloaded on Root, Transit and Leaf nodes respectively.

3.1.1. Shared Tree vs Non-Shared Replication Segment

A non-shared Replication Segment is used when the label field of the PMSI Tunnel Attribute (PTA) is set to zero as per [draft-parekh-bess-mvpn-sr-p2mp]. In short this is used when there is no upstream assigned label in the PTA (provider tunnel attribute) and aggregate of MVPNs into a single P-Tunnel is not desired.

On other hand shared Replication Segment is used when the label field of the PTA is not set to Zero and there is an upstream assigned label in the PTA. In this case multiple MVPNs (VRFs) can be aggregate into a single Provider Tunnel and the upstream assigned label distinguishes the MVPNs context.

It should be noted that the shared Replication Segment can also be used to build a bypass tunnel for the purpose of FRR. This might be desirable if the bypass tunnel is build via the PCE and downloaded to the PCC for link protection. In this case multiple non-shared Replication Segments can use the shared replication segment as their bypass tunnel for link protection. This replication segments used in this bypass tunnel should only create a unicast bypass tunnel to

protect the link between two replication segments on the primary path.

3.2. existing drafts used for defining a P2MP Policy

P2MP Policy reuses current drafts and PCEP objects to update the PCE with Root and Leaves information when PCC Initiated method is used. Also current drafts are used as much as possible to update the PCC with relevant information to build the P2MP Policy and its Replication Segments.

In addition this draft will introduced new TLVs and Objects specific to a programing P2MP policy and its replication segment.

3.2.1. Existing drafts/rfcs used by this draft

- o [RFC8231] The bases for a stateful PCE, and reuses the following objects or a variant of them
 - * <SRP Object>
 - * <LSP Object>
 - * A variation of the LSP identifier TLV is defined in this draft, to support P2MP LSP Identifier
- o [RFC8236] P2MP capabilities advertisement
- o [draft-ietf-pce-segment-routing-policy-cp] Candidate paths for P2MP Policy is used for Tree Redundancy. As an example a P2MP Policy can have multiple candidate paths. Each protecting the primary candidate path. The active path is chosen via the preference of the candidate path.
- o [RFC3209] Defines the instance-ID, instance-ID is used for global optimization of a candidate path with in a P2MP policy. Each Candidate path can have 2 path-instances. These path-instances are equivalent to sub-lsps (instance-IDs). There are used for MBB and global optimization procedures. instance-ID is equivalent to LSP ID
- o [draft-ietf-spring-segment-routing-policy] Segment-list, used for connecting two non-adjacent replication policy via a unicast binding SID or Segment-list.
- o [RFC8306] P2MP End Point objects, used for the PCC to update the PCE with discovered Leaves.

- o [draft-sivabalan-pce-binding-label-sid] Section 3; Path binding TLV is used to indicate the incoming replication SID
- o [draft-koldychev-pce-multipath] Forwarding instruction for a P2MP LSP is defined by a set of SR-ERO sub-objects in the ERO object, ERO-ATTRIBUTES object and MULTIPATH-BACKUP TLV as defined in this draft.
- o [RFC8664] SR-ERO Sub Object used in the multipath.

It should be noted that the [draft-dhs-spring-sr-p2mp-policy-yang] can provide further details of the high level P2MP Policy Model.

3.2.2. P2MP Policy Identification

A P2MP Policy and its candidate path can be identified on the root via the P2MP LSP Object. This Object is a variation of the LSP ID Object defined in [RFC8231] and is as follow:

- o PLSP-ID: [RFC8231], is assigned by PCC and is unique per candidate path. It is constant for the lifetime of a PCEP session. Stand-by candidate paths will be assigned a new PLSP-ID by PCC. Stand-by candidate paths can co-exist with the active candidate path.
 - * Note: Every candidate path in the SR-P2MP Policy is unique with its own unique PLSP-ID and Instance-ID. But the same Tree-ID is used for all candidate paths as they are part of the same P2MP Tree.
- o Root-ID: is equivalent to the first node on the P2MP path, as per [RFC3209], Section 4.6.2.1
- o Tree-ID: is equivalent to Tunnel Identifier color which identifies a unique P2MP Policy at a ROOT and is advertised via the PTA in the BGP AD route or can be assigned manually on the root. Tree-ID needs to be unique on the root.
- o Instance-ID: LSP ID Identifier as defined in RFC 3209, is the path-instance identifier and is assigned by the PCC. As it was mentioned the candidate path can have up to two path-instance for global optimization. Note that the Root-ID, Tree-ID and Instance-ID are part of a new SR- P2MP-LSP-IDENTIFIER TLV which will be identified in this draft.
 - * Note: each Path-instance on the Root node is assigned a unique Instance-ID

3.2.3. Replication Segment Identification

The key to identify a replication segment is also a P2MP LSP Object. That said there are different rules for coding the SR-P2MP-LSP-IDENTIFIER TLV which will be explained in later sections. In short for replication segment the P2MP LSP Object does not have the association object.

3.3. High Level Procedures for P2MP SR LSP Instantiation

A P2MP policy can be instantiated via the PCC or the PCE depending on how the root and the leaves are discovered. In theory there is two way to discover the root and the leaves:

- o They can be configured and identified on the controller, PCE initiated.
- o They can be discovered on the PCC via MVPN procedures [RFC6513] or legacy multicast protocols like PIM or IGMP etc... PCC initiated.

3.3.1. PCE-Init Procedure

- o PCE is informed of the P2MP request through it's API or configuration mechanism to instantiate a P2MP tunnel.
- o PCE will initiate the P2MP Policy for the request, by sending a PCInitiate message to the Root. This PCInitiation message will have the association object to identify the Candidate Path
 - * Optionally the EROs can be added to the PCInitiate message to construct the replication segment on the root.
- o Root in response to the PCInitiate message. It will generate PLSP-ID for the candidate paths and an Instance-ID for the Path-Instance (LSP-ID) contained within a candidate path. The tree-id for the P2MP Policy is also filled. PCC will report back the PLSP-ID, Instance-ID and tree-id via PCRpt message
 - * Optionally, the Root can add any additional leaves that were discovered by multicast procedures in this PCRpt message.
- o PCE based on any update to the configured leaves or if any new discovered leaves, can re-compute the P2MP Policy and its replication segments from the Root to the leaves.
 - * Any new EROs are sent via PCInitiate message, without the association object

- * Any update to the existing EROs are send via PCUpd message, without the association object
- o PCE will also sends a PCInitiate message to the Transit and the leaves for the Replication Segment.

3.3.2. PCC-Init Procedure

After Root (PCC) discovers the leaves (as an example via MVPN Procedures or other mechanism), the following communication happens between the PCE and PCCs

- o Root sends a PCRpt message for P2MP policy to PCE including the Root-ID, Tree-ID, PLSP-ID, Instance-ID, symbolic-path-name, and any leaves discovered until then.
- o PCE on receiving of this report, will compute the P2MP Policy and its replication segments.
 - * The PCE will send a PCUpd message to Root for P2MP policy with the Tree-ID, PLSP-ID and the Instance-ID assigned by the PCC. It should be noted the replication segment for root is also downloaded via this update message. In short a single update message that includes the association object will create the P2MP Policy and its replication segment on the Root
 - + Note: in this scenario no PCInitiate message was send from the PCE to the PCC to instantiate the P2MP Policy and its Replication segment. This is because for an PCInitiate message a brand new PLSP-ID and Instance-ID is assigned by PCC which is undesirable, since they are already assigned on the first step of this procedure.
 - * PCE will also sends a PCInitiate message to the Transit and the leaves for the Replication Segment.

3.3.3. Comon Procedure

Beyond this, the following procedures are the same for PCE or PCC Init.

- o PCE will download the replication segments for the Candidate-path's path-instances to all the leaves and transit nodes using PCInitiate message with PLSP-ID = 0, Instance-ID =0, symbolic path name, Root-address, Tree-id(assigned by the root). This PCInitiate message includes the EROs needed for the replication segments.

- o Any new candidate path for the P2MP Policy is downloaded by PCE to its connected Root by sending a PCInitiate message
 - * it should be noted, PLSP-ID, Path-Instance ID and the Tree-ID are generated by the PCC for these new candidate paths and their Path-instances
 - * The ERO objects can be included in this Initiate message
 - * The PCC will reply with a PCRpt message
 - * Any update to the Candidate Paths or Replication Segments is done via the PCUpd message. Association object need to be present for Candidate Path updates.
- o The PCE will also download the necessary replication segment for the candidate path and its path-instances to the leaves and the transit nodes via a PCInit message
- o New leaves can be discovered via Multicast procedures, and new replication segments can be instantiated or existing one updated to reach these leaves
 - * If these leaves reside on routers that are part of the P2MP LSP path, then PCUpd is sent from PCE to necessary PCCs (LEAVES, TRANSIT or ROOT) with the correct PLSP-ID, Instance-ID and Tree-ID
 - * If the new leaves are residing on routers that are not part of the P2MP Tree yet, then a PCInitiate message is sent down with PLSP-ID=0 and Instance-ID=0 on the corresponding routers.
- o The active candidate-path is indicated by the PCC through the operational bits(Up/Active) of the LSP object in the PCRpt message. If a candidate path needs to be removed, PCE sends PC Initiate message, setting the R-flag in the LSP object and R bit in the SRP-object.
- o To remove the entire P2MP-LSP, PCE needs to send PC Initiate remove messages for every candidate path of the P2MP POLICY to all the PCCs on the P2MP Tree. The R bit in the LSP Object as defined in [RFC8231], refers to the removal of the LSP as identified by the SR-P2MP-POLICY-ID-TLV (defined in this document). An all zero (SR-P2MP-LSP-ID-TLV defines to remove all the state of the corresponding PLSP-ID.
- o A candidate path is made active based on the preference of the path. If the Root is programed with multiple candidate paths from

different sources, as an example PCE and CLI, based on its tie-breaking rules, if it selects the CLI path, it will send a report to PCE for the PCE path indicating the status of label-download and sets operational bit of the LSP object to UP and Not Active . At any instance, only one path will be active

3.3.4. Global Optimiation of the Candidate Path

When a P2MP LSP needs to be optimized for any reason (i.e. it is taking a FRR tunnel or new routers are added to the network) a global optimization of the candidate path is possible.

Each Candidate Path can contain two Path-Instances. The current unoptimized Path-Instance is the active instance and its replication segments are forwarding the multicast PDUs from the root to the leaves. However the second optimized Path-Instance will be setup with its own unique replication segments throughout the network, from the Root to the leaves. These two Path-Instances can co-exist. The two Path-Instances are uniquely identified by their Instance-ID in the SR-P2MP-POLICY-ID-TLV (defined in this document). After the optimized LSP has been downloaded successfully PCC MUST send two reports, reporting UP of the new path indicating the new LSP-ID, and a second reporting the tear down of the old path with the R bit of the LSP Object SET with the old Instance-ID in the SR-P2MP-POLICY-ID-TLV. This MBB procedure will move the multicast PDUs to the optimized Path-Instance.

The leaf should be able to accept traffic from both Path-Instances to minimize the traffic outage by the Make Before Break process.

3.3.5. Fast Reroute

Currently this draft identifies the Facility FRR procedures. In addition, only LINK Protection procedures are defined. How the Facility Path is built and instantiated is beyond the scope of this document.

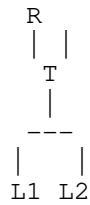


Figure 1

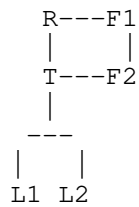


Figure 2

As an example, the bypass path (unicast bypass) between the PLR and MP can be constructed via SR or even via a shared tree (replication segment).

As an example, in figure 1 the detour path between R and T is the 2nd fiber between these nodes. As such the bypass path could be setup on the 2nd fiber. That said in figure 2 the bypass path is traversing multiple nodes and this example a unicast SR path might be ideal for setting up the detour path.

In addition, PHP procedure and implicit null label on the bypass path can be implemented to reduce the PCE programming on the MP PCC.

Optional shared replication segments can be used in networks that do not have unicast SR turned on. These shared replication segments can be programmed on the bypass nodes without a P2MP Policy. The replication segments on primary path can use these shared replication segments as a protection tunnel to protect links.

3.3.6. Connecting Replication Segment via Segment List

There could be nodes between two replication segment that do not understand P2MP Policy or Replication segment. It is possible to connect two non-adjacent Replication segment via a unicast binding SID or segment-list.

Replication segment does support the concept of a segment-list. A list of unicast SIDs (Binding SID, Adjacency SIDs or Node SIDs) can be programmed on a Replication segment via the SR-ERO sub-objects and ERO-attributes object.

How ever it should be noted that there needs to be a Replication SID as the bottom of the stack in all cases.

3.4. SR P2MP Policy and Replication Segment TLVs and Objects

3.4.1. SR P2MP Policy Objects

SR P2MP Policy can be constructed via the following objects

<Common Header>

[<SRP>]

<LSP>

[<association-list>]

optionally if the root is updating the PCE with end point list the end-point-list object can be added.

[<end-points-list>]

3.4.2. Replication Segment Objects

Replication segment can be constructed via the following objects

<Common Header>

[<SRP>]

<LSP>

[<replication-sid>]

as described in [draft-sivabalan-pce-binding-label-sid]

[<ERO-Attributes Object>]

as per [draft-koldychev-pce-multipath]

3.4.3. P2MP Policy vs Replication Segment

Note on the root the P2MP Policy and Replication Segment can be downloaded via the same message that includes the association object. That said on the transit or leaf nodes the replication segment needs to be downloaded individually as P2MP Policy is only relevant to the Root node. P2MP Policy and Replication segments objects have a very close definition, they can be told apart via the following abstracts:

- o The P2MP Policy will always have an association list object for the Candidate Paths in its PCInitiate message. While the replication segment does not have the association list object. That said they can be downloaded simultaneously by inserting the association list object and the ERO object in the same PCInitiate or PCUpd message.
- o Both P2MP Policy and Replication segment have the PLSP-ID and it is set to 0 in the PCInitiate message. For both Objects the PLSP-ID is set via the PCC.

3.4.4. P2MP Policy and Replication Segment general considerations

The above new objects and TLV's defined in this document can be included in PCRppt, PCInitiate and PCUpd messages.

It should be noted that every PCRppt, PCInitiate and PCUpd messages will contain full list of the Leaves and labels and forwarding information that is needed to build the Candidate path and its Replication segments. They will never send the delta information related to the new leaves or forwarding information that need to be added or updated. This is necessary to ensure that PCE or any new PCE is in sync with the PCC.

As such when a PCRppt, PCInitiate and PCUpd messages is send via PCEP it maintains the previous ERO Path IDs and generates new Path IDs for new instructions, as per [draft-koldychev-pce-multipath]. This means the PATH IDs are maintained for each specific forwarding instructions until these instructions are deleted. For example: When the first leaf is added the PCE will be update with PathI ID 1 to the PCC. When the second leaf is add, according to the path calculated, PCE might just append the existing instruction Path ID 1 with a new Path ID 2 to construct the new PCUpd message.

4. Object Format

4.1. Open Message and Capability Exchange

Format of the open Object:

```

      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
      +-----+-----+-----+-----+-----+-----+-----+-----+
      | Ver |   Flags |  Keepalive |  DeadTimer |           SID           |
      +-----+-----+-----+-----+-----+-----+-----+-----+
      | //                                     Optional TLVs                                     // |
      +-----+-----+-----+-----+-----+-----+-----+-----+

```

All the nodes need to establish a PCEP connection with the PCE.

During PCEP Initialization Phase, PCEP Speakers need to set flags N, M, P in the STATEFUL-PCE-CAPABILITY TLV as defined in [draft-ietf-pce-stateful-pce-p2mp] section-5.2

This draft extends the PCEP OPEN object by defining an optional TLV to indicate the PCE's capability to perform SR-P2MP path computations with a new IANA capability type.

The inclusion of this TLV in an OPEN object indicates that the sender can perform SR-P2MP path computations. This will be similar to the P2MP-CAPABILITY defined in [RFC8306] section-3.1.2 and a new value needs to be defined for SR-P2MP.

In addition a Assoc-Type-List TLV as per [RFC8697] section 3.4 should be send via PCEP open object with following association type

Association Type Value	Association Name	Reference
TBD1	P2MP SR Policy Association	This document

OP-CONF-Assoc-RANGE (Operator-configured Association Range) should not be set for this association type and must be ignored.

Finally the open message needs to include the MULTIPATH CAPABILITY TLV as defined in [draft-koldychev-pce-multipath]

4.2. Symbolic Name in PCInit Message from PCC

As per [RFC8231] section 7.3.2. a Symbolic Path Name TLV should uniquely identify the P2MP path on the PCC. This symbolic path name is a human-readable string that identifies an P2MP LSP in the network. It needs to be constant through the life time of the P2MP path.

As an example in the case of P2MP LSP the symbolic name can be p2mp policy name + candidate path name of the LSP.

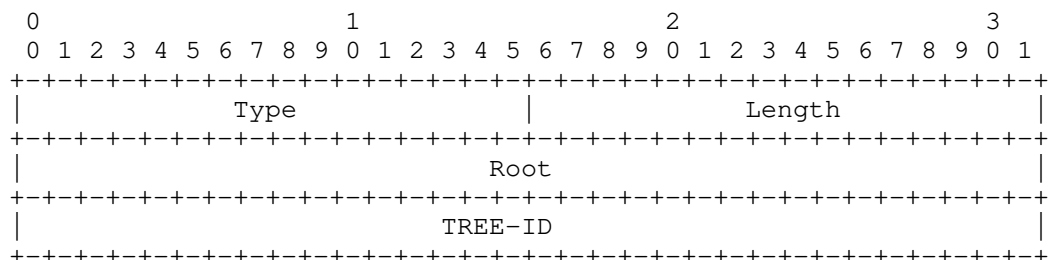
4.3. P2MP Policy Specific Objects and TLVs

4.3.1. P2MP Policy Association Group for P2MP Policy

Two ASSOCIATION object types for IPv4 and IPv6 are defined in [RFC8697]. The ASSOCIATION object includes "Association type" indicating the type of the association group. This document adds a new Association type. Association type = TBD1 "P2MP SR Policy Association Type" for SR Policy Association Group (P2MP SRPAG). As per [draft-barth-pce-segment-routing-policy-cpl] section 5, three new TLVs are identified to carry association information: P2MP-SRPAG-POL-ID-TLV, P2MP-SRPAG-CPATH-ID-TLV, P2MP-SRPAG-CPATH-ATTR-TLV

4.3.1.1. P2MP SR Policy Association Group Policy Identifiers TLV

The P2MP-SRPOLICY-POL-ID TLV is a mandatory TLV for the P2MP-SRPAG Association. Only one P2MP-SRPOLICY-POL-ID TLV can be carried and only the first occurrence is processed and any others MUST be ignored.



Type: TBD2 for "P2MP-SRPOLICY-POL-ID" TLV.

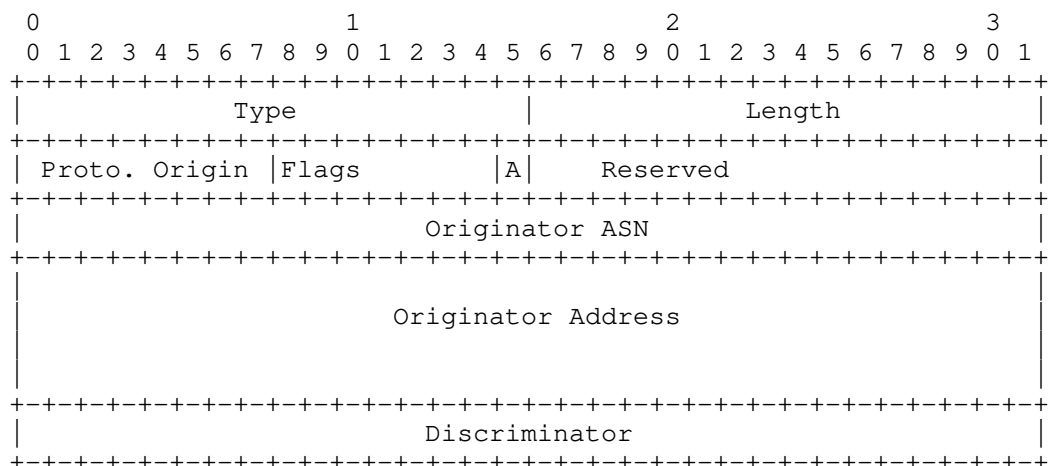
Length: 8 or 20, depending on length of End-point (IPv4 or IPv6)

Tunnel Sender Address : Can be either IPv4 or IPv6, this value is the value of the root loopback IP.

Tree-ID: Tree ID that the replication segment is part of as per draft-ietf-spring-sr-p2mp-policy

4.3.1.2. P2MP SR Policy Association Group Candidate Path Identifiers TLV

The P2MP-SRPOLICY-CPATH-ID TLV is a mandatory TLV for the P2MPSRPAG Association. Only one P2MP-SRPOLICY-CPATH-ID TLV can be carried and only the first occurrence is processed and any others MUST be ignored.



Type: TBD3 for "P2MP-SRPOLICY-CPATH-ID" TLV.

Length: 28.

Protocol Origin: 8-bit value that encodes the protocol origin, as specified in [I-D.ietf-spring-segment-routing-policy] Section 2.3.

Flags : A: This candidate path is active. At any instance only one candidate path can be active. PCC indicates the active candidate path to PCE through this bit. Reserved: MUST be set to zero on transmission and ignored on receipt.

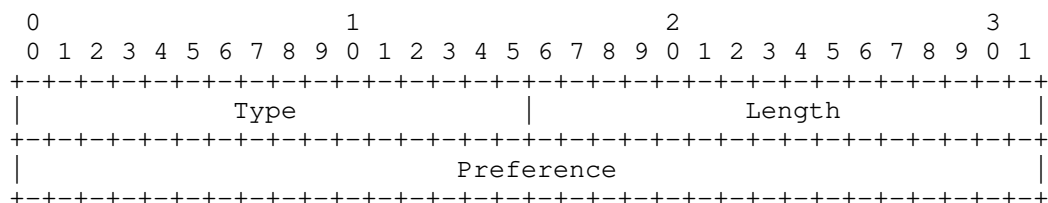
Originator ASN: Represented as 4 byte number, part of the originator identifier, as specified in [draft-ietf-spring-segment-routing-policy] Section 2.4.

Originator Address: Represented as 128 bit value where IPv4 address are encoded in lowest 32 bits, part of the originator identifier, as specified in [draft-ietf-spring-segment-routing-policy] Section 2.4.

Discriminator: 32-bit value that encodes the Discriminator of the candidate path.

4.3.1.3. P2MP SR Policy Association Group Candidate Path Attributes TLV

The P2MP-SRPOLICY-CPATH-ATTR TLV is an optional TLV for the SRPAG Association. Only one P2MP-SRPOLICY-CPATH-ATTR TLV can be carried and only the first occurrence is processed and any others MUST be ignored.



Type: TBD4 for "P2MP-SRPOLICY-CPATH-ATTR" TLV.

Length: 4. **Preference:** Numerical preference of the candidate path, as specified in [draft-ietf-spring-segment-routing-policy] Section 2.7.

If the TLV is missing, a default preference of 100 as specified in [draft-ietf-spring-segment-routing-policy] is used.

4.3.2. P2MP-END-POINTS Object

In order for the Root to indicate operations of its leaves (Add/Remove/Modify/DoNotModify), the PC Report message is extended to include P2MP End Point <P2MP End-points> Object which is defined in [RFC8306]

The format of the PC Report message is as follow:

<Common Header>

[<SRP>]

<LSP>

[<association-list>]

[<end-points-list>]

IPv4-P2MP END-POINTS:

```

 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
+-----+
|                                     Leaf type                             |
+-----+
|                                     Source IPv4 address                     |
+-----+
|                                     Destination IPv4 address               |
+-----+
~                                     ...                                   ~
+-----+
|                                     Destination IPv4 address               |
+-----+

```

IPv6-P2MP END-POINTS:

```

 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
+-----+
|                                     Leaf type                             |
+-----+
|                                     Source IPv6 address (16 bytes)          |
+-----+
|                                     Destination IPv6 address (16 bytes)     |
+-----+
~                                     ...                                   ~
+-----+
|                                     Destination IPv6 address (16 bytes)     |
+-----+

```

Leaf Types (derived from [RFC8306] section 3.3.2) :

1. New leaves to add (leaf type = 1)
2. Old leaves to remove (leaf type = 2)
3. Old leaves whose path can be modified/reoptimized (leaf type = 3), Future reserved not used for tree SID as of now.
4. Old leaves whose path must be left unchanged (leaf type = 4)

A given P2MP END-POINTS object gathers the leaves of a given type. Note that a P2MP report can mix the different types of leaves by including several P2MP END-POINTS objects. The END-POINTS object body has a variable length. These are multiples of 4 bytes for IPv4, multiples of 16 bytes, plus 4 bytes, for IPv6.

4.4. P2MP Policy and Replication Segment Identifier Object and TLV

As it was mentioned previously both P2MP Policy and Replication Segment are identified via the LSP object and more precisely via the SR-P2MP-LSPID-TLV

The P2MP Policy uses the PLSP-ID to identify the Candidate Paths and the Instance-ID to identify a Path-Instance within the Candidate path.

On the other hand the Replication Segment uses the SR-P2MP-LSPID-TLV to identify and correlate a Replication Segment to a P2MP Policy

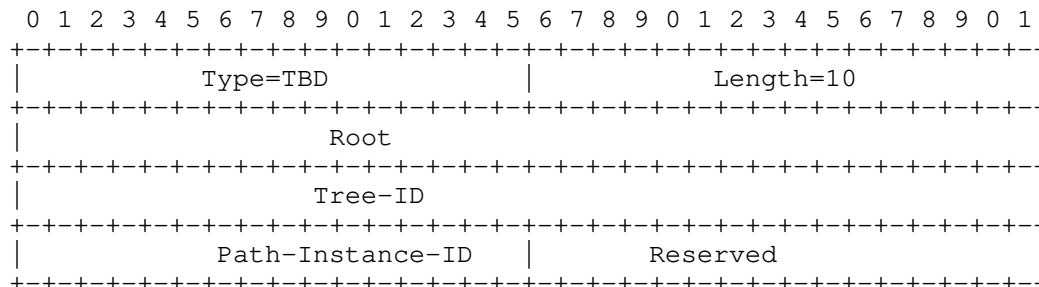
As it was noted previously on the Root, the P2MP Policy and the Replication Segment is downloaded via the same PCUpd message.

4.4.1. Extension of the LSP Object, SR-P2MP-LSPID-TLV

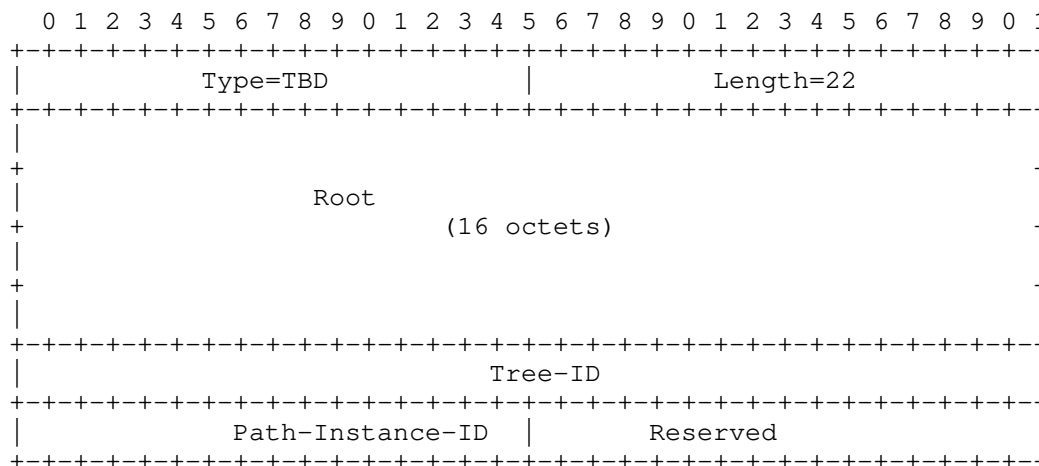
The LSP Object is defined in Section 7.3 of [RFC8231]. It specifies the PLSP-ID to uniquely identify an LSP that is constant for the life time of a PCEP session. Similarly for a P2MP tunnel, the PLSP-ID identifies a Candidate Path uniquely within the P2MP policy.

The LSP Object MUST include the new SR-P2MP-POLICY-ID-TLV (IPv4/IPv6) defined in this document below. This is a variation to the P2MP object defined in [draft-ietf-pce-stateful-pce-p2mp]

SR-IPV4-P2MP-POLICY-ID TLV:



SR-IPV6-P2MP-POLICY-ID TLV :



The type (16-bit) of the TLV is TBD (need allocation by IANA).

Root: Source Router IP Address

Tree-ID: Unique Identifier of this P2MP LSP on the Root.

Instance-ID : Contains 16 Bit instance ID.

4.5. Replication Segment

As per [draft-ietf-spring-sr-replication-segment] a replication segment has a next-hop-group which MAY contain a single outgoing replication sid OR a list of SIDs (sr-policy-sid-list) In either case there needs to be a replication SID at the bottom of the stack. This

means two replication segments can be directly connected or connected via a SR domain.

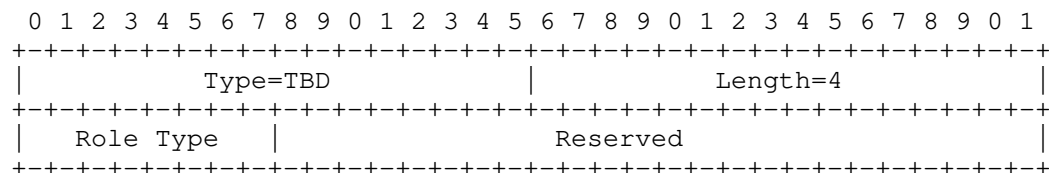
4.5.1. The format of the replication segment message

As it was mentioned in previous chapters the format of the replication segment message is close to P2MP Policy. That said the P2MP Policy contains the association object and the replication segment message does not contain the association object. The replication segment may be downloaded individually on transit and leaf nodes without the P2MP Policy. The P2MP Policy is a Root Concept. The replication segment uses SR-P2MP-LSPID-TLV as its identifier. That said this TLV is coded differently for shared and on shared case.

- o In the case of a replication segment being shared, the Tree-ID in the SR-P2MP-POLICY Identifier TLV is the replication-id of the replication segment and Root = 0, Instance-Id = 0. When downloading a shared replication segment from PCE through a PcInitiate message, the SR-P2MP-POLICY Identifier TLV is all 0, and on the report back from PCC, PCC generates PLSP-ID, Replication-id (Tree-id field will be populated with replication-id). Instance-id will be 0.

4.5.2. Label action rules in replicating segment

The node action, ingress, transit, leaf or Bud, is indicated via a new Node Role TLV. This document introduces a new SR-P2MP-NODE-ROLE TLV (Type To be assigned by IANA) that will be present in the PATH-ATTRIB object.



- o ingress, role type = 1
- o transit, role type = 2
- o leaf, role type = 3
- o bud, role type = 4

4.5.3. SR-ERO Rules

Forwarding information of a replication segment can be configured and steered via many different mechanisms.

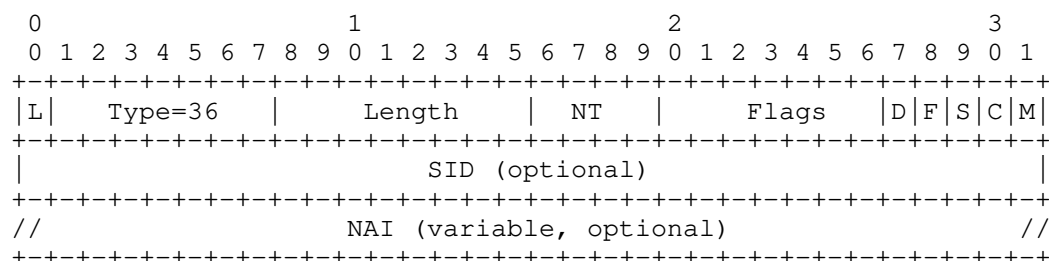
As an example a replication SID can be steered via:

1. Replication SID steered with an IPv4/IPv6 directly connected nexthop
 - * In this case there will be two SR-ERO in the ERO Object, with the Replication SID SR-ERO at the bottom and the IPv4/IPv6 SR-ERO on the top.
2. Replication SID steered with an IPv4/IPv6 loopback address that reside on the directly connected router.
 - * In this case there will be two SR-ERO in the ERO Object, with the Replication SID SR-ERO at the bottom and the IPv4/IPv6 SR-ERO on the top.
 - * In addition a new flag D is added to the SR-ERO to signal that the Loopback nexthop is connected to the directly attached router.
3. Replication SID steered with unnumbered IPv4/IPv6 directly connected Interface
4. Replication SID steered via a SR adjacency or node SID
 - * In this case even a sid-list can be used to traffic engineer the path between two Replication SID
 - * The Replication SID SR-ERO is at the bottom while all other SR-EROs are on the top in order.

4.5.3.1. SR-ERO subobject changes

SR-ERO from RFC 8664 is used to construct the forwarding information needed for Replication Segment.

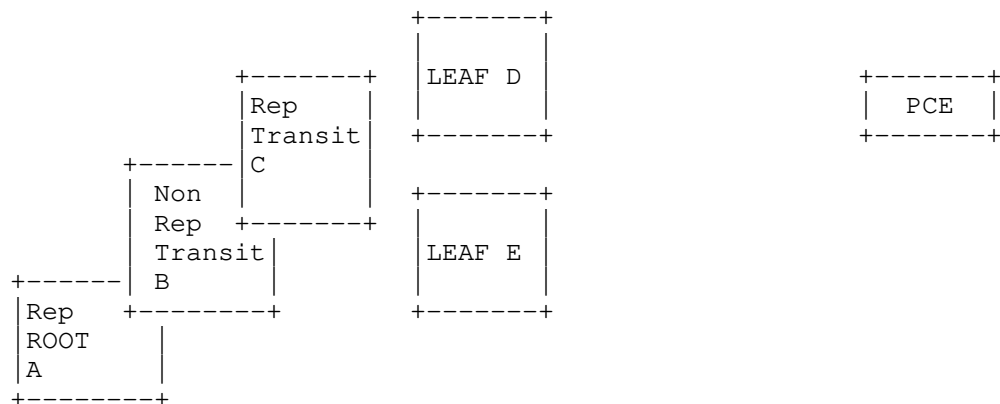
A new D flag was added to indicate a loopback nexthop that is residing on the directly attached router. It should be noted that this flag should be set only for the loopback case and not for a local interface as a nexthop.



Flags : F, S, C, M are already defined in rfc8664.

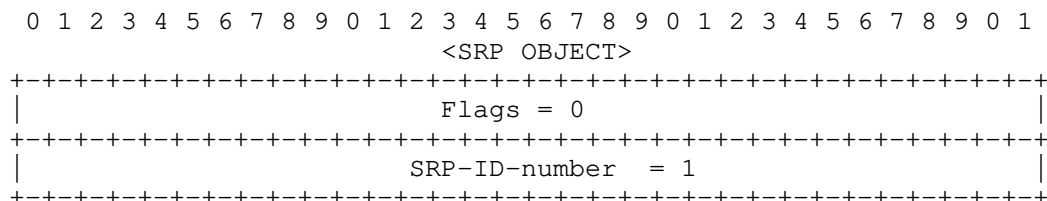
This document defines a new flag D: If the next-hop in NAI field is system IP or loopback, this bit indicates whether the system IP / loopback is directly connected router or not. If set indicates directly connected address. When this bit is set, F bit should be 0 (meaning NAI should be present)

5. Examples of PCEP messages between PCE and PCEP



5.1. PCE Initiate

For a PCE Initiate P2MP Policy a sample PC Initiate message from the PCE to the root is provided below. This is on reception of a P2MP Policy creation on the PCE:



```

| TLV Type = 28 (PathSetupType) | TLV Len = 4 |
+-----+-----+-----+-----+
|                               | PST = TBD |
+-----+-----+-----+-----+
<LSP OBJECT>

```

```

+-----+-----+-----+-----+
| PLSP-ID = 0 | A:1,D:1,N:1,C:1 |
+-----+-----+-----+-----+
| Type=17 | Length=<var> |
+-----+-----+-----+-----+
| symbolic path name |
+-----+-----+-----+-----+
| Type=TBD | Length |
+-----+-----+-----+-----+
| Root = A |
+-----+-----+-----+-----+
| Tree-ID = 0 |
+-----+-----+-----+-----+
| Instance-id = 0 | Reserved |
+-----+-----+-----+-----+

```

```

<ASSOCIATION OBJECT>
+-----+-----+-----+-----+
| Reserved | Flags | 0 |
+-----+-----+-----+-----+
| Association type= SR-P2MP-PAG | Association ID = z |
+-----+-----+-----+-----+
| IPv4 Association Source = <pce-address> |
+-----+-----+-----+-----+
| Type | Length |
+-----+-----+-----+-----+
| Root = A |
+-----+-----+-----+-----+
| TREE-ID = 0 |
+-----+-----+-----+-----+
| Type | Length |
+-----+-----+-----+-----+
| ProtOrigin 10 | Reserved |
+-----+-----+-----+-----+
| Originator ASN |
+-----+-----+-----+-----+
| Originator Address = <pce-address> |
+-----+-----+-----+-----+
| Discriminator = 1 |
+-----+-----+-----+-----+

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+	

Presence of an association object with Tree-ID = 0 in the Initiate message, is an indication to the node to create a P2MP policy and associated candidate path. An initiate message without an association object, is an indication to PCC that a Replication Segment (forwarding instructions) is being instantiated.

5.2. PCC Initiate or PCE Initiate Respond

For PCC initiated P2MP Policy, the Root will send a P2MP request to the PCE, this is achieved through Root sending a PCRpt to PCE with the Tree-ID, PLSP-ID and Instance-ID Set. Below is a sample Report generated by the Root (PCC) to the PCE

In addition for the PCE Initiated case the same PCRpt message can be send from Root (PCC) to the PCE. The Root will generate the Tree-ID, PLSP-ID, Instance-ID for the candidate path identified by the candidate path identifier TLV and sends a report back to PCE. Note, in this case the End point object is optional. The end point object (optionally) is added if the root has discovered any new leaves on the PCC.

Sample Report generated by the Root to the PCE for Leaf Add

```

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
+-----+
|                                     Flags = 0                                     |
+-----+
|                                     SRP-ID-number  = 1                             |
+-----+
| TLV Type = 28 (PathSetupType) | TLV Len = 4                                     |
+-----+
|                                     | PST = TBD                                   |
+-----+
|                                     <LSP OBJECT>                                |
|                                     PLSP-ID = 1                                | A:1,D:1,N:1 |
+-----+
|                                     Type=17                                | Length=<var> |
+-----+
|                                     symbolic path name                        |
+-----+
|                                     Type=TBD                                | Length      |
+-----+
|                                     Root = A                                |
+-----+
|                                     Tree-ID = Y                            |
+-----+
| Instance-ID =L1                                | Reserved    |
+-----+
|                                     <END POINT OBJECT>                        |
|                                     Leaf type =1                            |
+-----+
|                                     Source IPv4 address = A                    |
+-----+
|                                     Destination IPv4 address = D                |
+-----+
|                                     Destination IPv4 address = E                |
+-----+

```

5.3. PCE P2MP Path-Instance Calculation and Replication Segment download

Once the PCRpt message including the endpoints (optional in case of PCE Initiated) is sent to the PCE, PCE computes the path from root to the leaves and would send a PCInitiate to the transit and leaf nodes for instantiating the replication segment for the Path-Instance.

In addition a PCUpd is send to the Root node to construct the replication segment.

The forwarding information is downloaded via the ERO object, ERO-attribute object and SR-ERO sub-objects. For example, say PCE computed 2 candidate paths <cp1 and cp2> that needs to be downloaded on the root and their corresponding Replication Segment download to the root, transit and leaf nodes. The sample messages are explained below.

For cp1:

- o For PCC initiate case, the PCE will send a PCUpd message to download the Candidate Paths and the replication segment. Note on the root a single message with association object will achieve this.
- o For PCE initiate case, the PCE optionally sends a PCUpd message to instantiate the replication segment that were newly discovered by the PCC and send to the PCE via the PCRpt message.. Note for this case the association object might not be needed is there is no update to the P2MP Policy.

For cp2:

- o For both PCC/PCE initiate, a PCInitiate messages sent from PCE, initiating the new Candidate Path and its associated Replication Segments.

For both CP1 and CP2 on the transit and leaves, since PCE is initiating newly Replication Segments, PCE will send one PCInitiate message with two LSP objects and no association object, defining the Replication Semgnets on each candidate path. On other hand, PCE can send separate PCInitiate message for every Replication Segment. As defined in [draft-barth-pce-segment-routing-policy-cp]

A sample PCUpd message sent to the Root for cp1 is as follows, NOTE in the below example the Node B is not Replication Segment Capable as such there is a sid-list programmed on A with node SID B as steering followed by node SID of C and finally the Replication SID C at the bottom :

Note:

1. Root is connected to the next replication Segment C via non replication segment B. Hence a segment List is used.
2. The following PCUpd message send to the root is for PCC Initiated case as such it has the association object to instantiate the Candidate Path and the Replication Segment via a single message on the root.

3. For PCE Initiate message the association object can be omitted sense it is only used for instantiating or updating the Replication Segment only.

```

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
+-----+
|                                     Flags = 0                                     |
+-----+
|                                     SRP-ID-number = 2                             |
+-----+
| TLV Type = 28 (PathSetupType) | TLV Len = 4 |
+-----+
|                                     | PST = TBD |
+-----+
|                                     <LSP OBJECT>                                |
|                                     PLSP-ID = 1 | A:1,D:1,N:1,C:0 |
+-----+
|                                     Type=17 | Length=<var> |
+-----+
|                                     symbolic path name |
+-----+
|                                     Type=TBD | Length |
+-----+
|                                     Root =A |
+-----+
|                                     Tree-ID = Y |
+-----+
| Instance-ID = L1 | Reserved |
+-----+
| Type | Length |
+-----+
| BT= 0 | Reserved |
+-----+
| Binding value = incoming replication SID |
+-----+
|                                     <ASSOCIATION OBJECT>                        |
| Reserved | Flags | 0 |
+-----+
| Association type= SR-P2MP-PAG | Association ID = z |
+-----+
| IPv4 Association Source = <pce-address> |
+-----+
| Type | Length |
+-----+
| Root = A |
+-----+
| TREE-ID = 0 |
+-----+

```

Type		Length		
ProtOrigin 10		Reserved		
Originator ASN				
Originator Address = <pce-address>				
Discriminator = 1				
Type		Length		
Preference = 100 <default>				
<ERO-ATTRIBUTES OBJECT>				
Flags			Oper	
Type		Length		
1 0 0		Reserved		
ERO-path Id = 1				
Back-up ero path id = 0				
L	Type=36	Length	NT= 1 Flags	0 0 0 0
SID = node sid b				
L	Type=36	Length	NT= 1 Flags	0 0 0 0
SID = node sid c				
L	Type=36	Length	NT= 1 Flags	0 0 0 0
SID = RSID C				

A sample PC Initiate message to the Root for cp2 is as follows: Note cp2 can be either on the same path as cp1 or on a separate path, assuming that there is a 2nd path connecting A to B to C. In this example a 2nd interface is used on A and B, hence the adjacency SIDs are programmed


```

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
+-----+
|                                     Flags = 0                                     |
+-----+
|                                     SRP-ID-number  = 3                             |
+-----+
| TLV Type = 28 (PathSetupType) | TLV Len = 4                                     |
+-----+
|                                     | PST = TBD                                   |
+-----+
|                                     <LSP OBJECT>                                |
|                                     PLSP-ID = 0                                | A:1,D:1,N:1,C:1 |
+-----+
|                                     Type=17                                | Length=<var>      |
+-----+
|                                     symbolic path name                    |
|                                     Type=TBD                                | Length           |
+-----+
|                                     Root = A                                |
+-----+
|                                     Tree-ID = Y                            |
+-----+
|                                     Instance-ID = 0                        | reserved         |
+-----+
|                                     Type                                | Length           |
+-----+
|                                     BT                                | Reserved         |
+-----+
|                                     Binding Value= incoming replication sid |
+-----+
|                                     <ASSOCIATION OBJECT>                     |
+-----+
|                                     Reserved                                | Flags            | 0 |
+-----+
| Association type= SR-P2MP-PAG | Association ID = z                        |
+-----+
|                                     IPv4 Association Source = <pce-address> |
+-----+
|                                     Type                                | Length           |
+-----+
|                                     Root = A                                |
+-----+
|                                     TREE-ID = Y                            |
+-----+
|                                     Type                                | Length           |
+-----+
| ProtOrigin 10 | Reserved |

```

```

+++++
|                                     |
|                               Originator ASN                               |
|                                     |
|                               Originator Address = <pce-address>           |
|                                     |
|                               Discriminator = 2                           |
|                                     |
|                               Type                                         | Length                           |
|                                     |                                     |
|                               Preference = 50 <Lower Pref>                 |
|                                     |
|                               <ERO-ATTRIBUTES OBJECT>                     |
|                                     |
|                               Flags                                         | Oper | | |
|                                     |                                     |
|                               Type                                         | Length                           |
|                                     |                                     |
| 1|0|0|                               Reserved                           |
|                                     |
|                               ERO-path Id = 2                             |
|                                     |
|                               Back-up ero path id = 0                     |
|                                     |
|  L|  Type=36 |      Length      |  NT= 1|      Flags      |0|0|0|0|
|                                     |
|                               SID = adjacency sid a-b-int2                 |
|                                     |
|  L|  Type=36 |      Length      |  NT= 1|      Flags      |0|0|0|0|
|                                     |
|                               SID = adjacency sid b-c-int2                 |
|                                     |
|  L|  Type=36 |      Length      |  NT= 1|      Flags      |0|0|0|0|
|                                     |
|                               SID = RSID C                               |
|                                     |
+++++

```

A sample PC Initiate message to the transit Replication Segment C for cpl Lets assume C is connected to D and E via 2 fiber hence Fast Reroute is possible. Below example sets up the forwarding plane from C to Leaves D and E

```

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
+++++
|                                     |
|                               Flags = 0                                   |
|                                     |
+++++

```

```

|                               SRP-ID-number  = 4                               |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TLV Type = 28 (PathSetupType) | TLV Len = 4                                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               | PST = TBD                                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               <LSP OBJECT>                               |
|                               PLSP-ID = 0                                | A:1,D:1,N:1,C:1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Type=17                                | Length=<var>      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               symbolic path name                      |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Type=TBD                                | Length            |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Root=A                                  |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Tree-ID = Y                             |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Instance-ID =L1                        | Reserved          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Type                                    | Length            |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               BT                                     | Reserved          |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Binding Value= incoming replication sid = c1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               <With FRR over NHD2>                    |
|                               <ERO-ATTRIBUTES OBJECT>                 |
|                               <incoming label c1 swap with D1>        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Flags                                    | Oper             |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Type                                    | Length            |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1|0|1|                               Reserved                        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               ERO-path Id = 3                         |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               Back-up ero path id = 4                 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| L| Type=36 | Length | NT= 1| Flags | 0|0|0|0|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               SID = d1                                |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               ipv4-address  = NHD1                    |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                               <ERO-ATTRIBUTES OBJECT>                 |

```

```

+++++
|                                     Flags                                     | Oper|
+++++
|                                     Type                                     | Length|
+++++
|0|1|0|                               Reserved                               |
+++++
|                                     ERO-path Id = 4                         |
+++++
|                                     Back-up ero path id = 0                 |
+++++
|L|  Type=36  |  Length  |  NT= 1|  Flags  |0|0|0|0|
+++++
|                                     SID = d protect                         |
+++++
|                                     ipv4-address  = NHD2                     |
+++++
<incoming label c1 swap with E1>
<ERO-ATTRIBUTES OBJECT>
+++++
|                                     Flags                                     | Oper|
+++++
|                                     Type                                     | Length|
+++++
|1|0|1|                               Reserved                               |
+++++
|                                     ERO-path Id = 5                         |
+++++
|                                     Back-up ero path id = 6                 |
+++++
|L|  Type=36  |  Length  |  NT= 1|  Flags  |0|0|0|0|
+++++
|                                     SID = e1                               |
+++++
|                                     ipv4-address  = NHE1                     |
+++++
<ERO-ATTRIBUTES OBJECT>
+++++
|                                     Flags                                     | Oper|
+++++
|                                     Type                                     | Length|
+++++
|0|1|0|                               Reserved                               |
+++++
|                                     ERO-path Id = 6                         |
+++++
|                                     Back-up ero path id = 0                 |
+++++

```

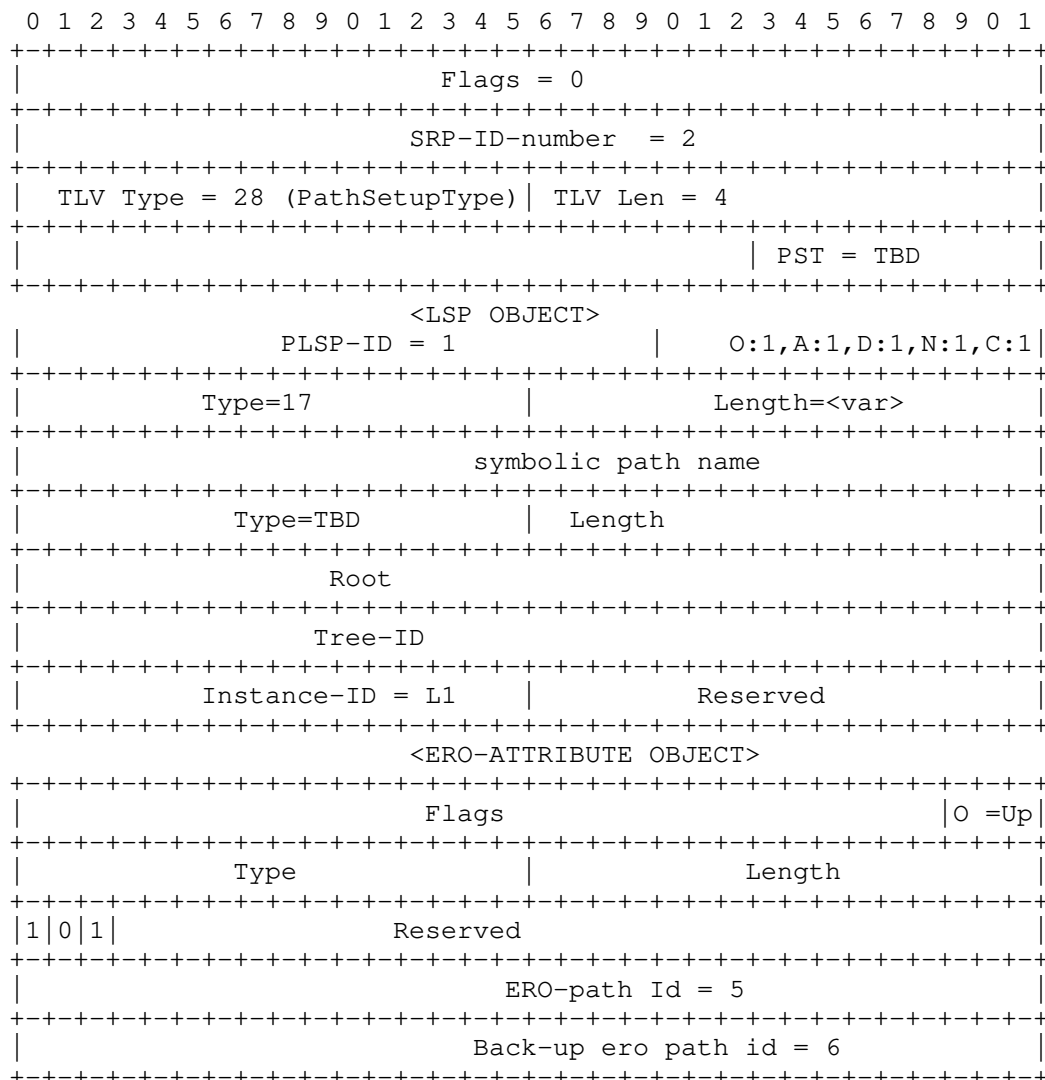
```

|L|   Type=36   |   Length   |   NT= 1|   Flags   |0|0|0|0|
+-----+-----+-----+-----+-----+-----+
|                                     SID = e protect                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     ipv4-address  = NHE2                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

5.4. PCC Rpt for PCE Update and Init Messages

In response to the PC Initiate message / PC Update message , PCC will send PC Reports to PCE indicating the state of the label download for that particular candidate path. PCC's will generate PLSP-ID for newly initiated candidate path. Here is an PC Report Message send for the root PCE Init message with cp2 on the root.



6. Tree Deletion

To delete the entire tree (P2MP LSP) , Root send a PCRpt message with the R bit of the LSP object set and all the fields of the SR-P2MP-LSP-ID TLV set to 0(indicating to remove all state associated with this P2MP tunnel). The controller in response sends a PCInitiate message with R bit in the SRP object SET to all nodes along the path to indicate deletion of a label entry.

7. Fragmentation

The Fragmentation bit in the LSP object (F bit) can be used to indicate a fragmented PCEP message

8. Example Workflows

As per slides submitted in IETF 105.

9. IANA Consideration

1. This draft extends the PCEP OPEN object by defining an optional TLV to indicate the PCE's capability to perform SR-P2MP path computations with a new IANA capability type (TBD).
2. PCEP open object with a new association type " P2MP SR Policy Association " value (TBD).
3. A new Association type. Association type = TBD1 "P2MP SR Policy Association Type" for SR Policy Association Group (P2MP SRPAG)
 1. three new TLVs are identified to carry association information: P2MP-SRPAG- POL-ID-TLV, P2MP-SRPAG-CPATH-ID-TLV, P2MP-SRPAG-CPATH-ATTR-TLV
4. Two new TLVs for Identifying the P2MP Policy and the Replication segment SR-IPV4-P2MP-POLICY-ID TLV and SR-IPV6-P2MP-POLICY-ID TLV
5. A new SR-P2MP-NODE-ROLE TLV (Type To be assigned by IANA) that will be present in the PATH-ATTRIB object

10. Security Considerations

TBD

11. Acknowledgments

The authors would like to thank Tanmoy Kundu and Stone Andrew at Nokia for their feedback and major contribution to this draft.

12. References

12.1. Normative References

- [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>>.

12.2. Informative References

- [draft-barth-pce-segment-routing-policy-cp]
.
- [draft-dhs-spring-sr-p2mp-policy-yang]
.
- [draft-ietf-pce-segment-routing-policy-cp]
.
- [draft-ietf-pce-stateful-pce-p2mp]
.
- [draft-ietf-pim-sr-p2mp-policy]
"D. Yoyer, C. Filsfils, R.Prekh, H.bidgoli, Z. Zhang,
"draft-voyer-pim-sr-p2mp-policy", October 2019.
- [draft-ietf-spring-segment-routing-policy]
.
- [draft-ietf-spring-sr-replication-segment]
"D. Yoyer, C. Filsfils, R.Prekh, H.bidgoli, Z. Zhang,
"draft-voyer-pim-sr-p2mp-policy "draft-voyer-spring-sr-
replication-segment", July 2020.
- [draft-koldychev-pce-multipath]
.
- [draft-parekh-bess-mvpn-sr-p2mp]
.
- [draft-sivabalan-pce-binding-label-sid]
.
- [RFC3209] .
- [RFC6513] .
- [RFC8231] .
- [RFC8236] .
- [RFC8306] .
- [RFC8664] .
- [RFC8697] .

Authors' Addresses

Hooman Bidgoli (editor)
Nokia
Ottawa
Canada

Email: hooman.bidgoli@nokia.com

Daniel Voyer
Bell Canada
Montreal
Canada

Email: daniel.yover@bell.ca

Saranya Rajarathinam
Nokia
Mountain View
US

Email: saranya.Rajarathinam@nokia.com

Ehsan Hemmati
Cisco System
San Jose
USA

Email: ehemmati@cisco.com

Tarek Saad
Juniper Networks
Ottawa
Canada

Email: tsaad@juniper.com

Siva Sivabalan
Ciena
Ottawa
Canada

Email: ssivabal@ciena.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: November 26, 2021

H. Bidgoli, Ed.
Nokia
V. Voyer
Bell Canada
S. Rajarathinam
Nokia
E. Hemmati
Cisco System
T. Saad
Juniper Networks
S. Sivabalan
Ciena
May 25, 2021

PCEP extensions for p2mp sr policy
draft-hsd-pce-sr-p2mp-policy-03

Abstract

SR P2MP policies are set of policies that enable architecture for P2MP service delivery. This document specifies extensions to the Path Computation Element Communication Protocol (PCEP) that allow a stateful PCE to compute and initiate P2MP paths from a Root to a set of Leaves.

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 November 26, 2021.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Conventions used in this document	4
3. Overview of PCEP Operation in SR P2MP Network	4
3.1. High level view of P2MP Policy Objects	5
3.1.1. Shared Tree vs Non-Shared Replication Segment	6
3.2. Existing drafts used for defining a P2MP Policy	7
3.2.1. Existing Documents used by this draft	7
3.2.2. P2MP Policy Identification	8
3.2.3. Replication Segment Identification	9
3.2.4. PCECC Use in Replication Segment	9
3.3. High Level Procedures for P2MP SR LSP Instantiation	9
3.3.1. PCE-Init Procedure	9
3.3.2. PCC-Init Procedure	10
3.3.3. Common Procedure	10
3.3.4. Global Optimization of the Candidate Path	11
3.3.5. Fast Reroute	12
3.3.6. Connecting Replication Segment via Segment List	13
3.4. SR P2MP Policy and Replication Segment TLVs and Objects	13
3.4.1. SR P2MP Policy Objects	13
3.4.2. Replication Segment Objects	14
3.4.3. P2MP Policy and Replication Segment general considerations	14
4. Object Format	15
4.1. Open Message and Capability Exchange	15
4.1.1. PCECC Path Setup Capability	15
4.1.2. Association Type Capability	16
4.2. Symbolic Name in PCInit Message from PCC	16
4.3. P2MP Policy Specific Objects and TLVs	16
4.3.1. P2MP Policy Association Group for P2MP Policy	16
4.3.1.1. P2MP SR Policy Association Group Policy Identifiers TLV	16
4.3.1.2. P2MP SR Policy Association Group Candidate Path Identifiers TLV	17
4.3.1.3. P2MP SR Policy Association Group Candidate Path Attributes TLV	18
4.3.2. P2MP-END-POINTS Object	18

4.4. P2MP Policy and Replication Segment Identifier Object and TLV	21
4.4.1. Extension of the LSP Object, SR-P2MP-LSPID-TLV . . .	21
4.5. Replication Segment	22
4.5.1. The format of the replication segment message	23
4.5.2. PCECC	23
4.5.3. Label action rules in replicating segment	26
4.5.4. SR-ERO Rules	27
4.5.4.1. SR-ERO subobject changes	27
5. Tree Deletion	28
6. Fragmentation	28
7. Example Workflows	28
8. IANA Consideration	33
9. Security Considerations	34
10. Acknowledgments	34
11. References	34
11.1. Normative References	34
11.2. Informative References	34
Authors' Addresses	35

1. Introduction

The draft [draft-ietf-pim-sr-p2mp-policy] defines a variant of the SR Policy [draft-ietf-spring-segment-routing-policy] for constructing a P2MP segment to support multicast service delivery.

A Point-to-Multipoint (P2MP) Policy connects a Root node to a set of Leaf nodes, optionally through a set of intermediate replication nodes. A Replication segment [draft-ietf-spring-sr-replication-segment], which corresponds to the state of a P2MP segment on a particular node which provide forwarding instructions for the segment.

A P2MP Policy is relevant on the root of the P2MP Tree and it contains candidate paths. The candidate paths are made of path-instances and each path-instance is constructed via replication segments. These replication segments are programmed on the root, leaves and optionally intermediate replication nodes.

A replication segments MAY be connected directly, or they MAY be connected or steered via unicast SR segment or a segment list.

For a P2MP Tree, a controller may be used to compute paths from a Root node to a set of Leaf nodes, optionally via a set of replication nodes. A packet is replicated at the root node and optionally on Replication nodes towards each Leaf node.

There are two types of a P2MP Tree: Spray and Replication.

A Point-to-Multipoint service delivery could be via Ingress Replication, known as Spray. The root unicasts individual copies of traffic to each leaf. The corresponding P2MP Policy consists of replication segments only for the root and the leaves and they are connected via a unicast SR Segment.

A Point-to-Multipoint service delivery could also be via Downstream Replication, known as Replication. The root and some downstream replication nodes replicate the traffic along the way as it traverses closer to the leaves.

The leaves and the root can be explicitly configured on the PCE or PCC can update the PCE with the information of the discovered root and leaves. As an example Multicast protocols like MVPN procedures [RFC6513] or PIM can be used to discovery the leaves and roots on the PCC and update the PCE with these relevant information. The controller can calculate the P2MP Policy and any of its associated replication segments with these info.

This document defines PCEP objects, TLVs and the procedures to instantiate a P2MP Policy and Replication Segments.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Overview of PCEP Operation in SR P2MP Network

After discovering the root and the leaves on the PCE (via different mechanism mentioned in previous sections), the PCE computes the P2MP Tree and identifying the relevant Replication routers, then it programs the PCCs with relevant information needed to create a P2MP Tree.

As per draft [draft-ietf-pim-sr-p2mp-policy] a P2MP Policy is defined by Root-ID, Tree-ID and a set of leaves. A P2MP policy is a variant of SR policy as such it uses the same concept as draft [draft-ietf-pce-segment-routing-policy-cp]. A P2MP policy is composed of a collection of SR P2mp Candidate Paths. Candidate paths are computed by the PCE and can be used for P2MP Tree redundancy. Only a single candidate path may be active at each time. Each candidate paths can be globally optimized, therefore it is consists of multiple path-instances. A path-instance can be considered to a P2MP LSP. If a candidate path needs to be globally optimized two path-instances can be programmed on the root node and via make before break procedures the candidate path can be switched from path-

instance 1 to the 2nd path-instance. The forwarding states of these path-instances are build via replication segments, in short each path-instance initiated on the root has its own set of replication segments on the Root, Transit and Leaf nodes.

A replication segment is set of forwarding instructions on a specific node. Each instruction may be a PUSH or SWAP operation before forwarding out of an interface, or a POP action on bud and leaf nodes.

PCE could also calculate and download additional information for the replication segments, such as protections next-hops for link protection (FRR).

3.1. High level view of P2MP Policy Objects

o SR P2MP Policy

- * Is only relevant on the Root of the P2MP path and is a policy on PCE. It is downloaded only on the rootnode and is identified via <Root-ID, Tree-ID> It contains the following information:

- + Root node of the P2MP Segment
- + Leaf nodes of the P2MP Segment
- + Tree-ID, which is a unique identifier of the P2MP tree on the Root
- + A set of Candidate paths belonging to the policy
- + Optional Constraints used to build these candidate paths

o Candidate Path:

- * Is used for P2MP Tree redundancy where the candidate path with the highest preference is the active path.
- * It can contain two path-instance for global optimization procedures (i.e. make before break)
- * Contains information regarding protocol-id, originator, discriminator, preference, path-instances

o Replication Segment:

- * Is the forwarding information needed on each node for building the forwarding path for each path-instance of the P2MP Candidate path.
- * Explained further in upcoming sections, there are 2 ways to identify the replication segment, depending if they are shared and non-shared
 - + It is identified via Tree-ID and Root-ID and path-instance for non-shared replication segment.
 - + It is identified via Node-ID, Replication-ID, for shared replication segment
 - + Contains forwarding instructions, in the form of a list of outgoing segments each of which may be a list
 - + On the forwarding plane the Replication Segment is identified via the incoming Replication SID.
 - + Replication segment information is downloaded on Root, Transit and Leaf nodes respectively.

3.1.1. Shared Tree vs Non-Shared Replication Segment

A non-shared Replication Segment is used when the label field of the PMSI Tunnel Attribute (PTA) is set to zero as per [draft-parekh-bess-mvpn-sr-p2mp]. This is used when there is no upstream assigned label in the PTA (provider tunnel attribute) and aggregate of MVPNs into a single P-Tunnel is not desired.

An alternative shared Replication Segment is used when the label field of the PTA is not set to Zero and there is an upstream assigned label in the PTA. In this case multiple MVPNs (VRFs) can be aggregate into a single Provider Tunnel and the upstream assigned label distinguishes the MVPNs context.

It should be noted that the shared Replication Segment can also be used to build a bypass tunnel for the purpose of fast re-route. This might be desirable if the bypass tunnel is build via the PCE and downloaded to the PCC for link protection. In doing so, multiple non-shared Replication Segments can use the shared replication segment as their bypass tunnel for link protection. The replication segments used in this bypass tunnel should only create a unicast bypass tunnel to protect the link between two replication segments on the primary path.

3.2. Existing drafts used for defining a P2MP Policy

This document attempts to leverage existing IETF draft and RFC documents which define PCEP objects, to update the PCE with Root and Leaves information when PCC Initiated method is used. Similarly, existing documents are utilized where feasible to update the PCC with relevant information to build the P2MP Policy and its Replication Segments. This document introduces new TLVs and Objects specific to a programming P2MP policy and its replication segment.

3.2.1. Existing Documents used by this draft

- o [RFC8231] The bases for a stateful PCE, and reuses the following objects or a variant of them
 - * <SRP Object>
 - * <LSP Object>
 - * A variation of the LSP identifier TLV is defined in this draft, to support P2MP LSP Identifier
- o [RFC8236] P2MP capabilities advertisement
- o [draft-ietf-pce-segment-routing-policy-cp] Candidate paths for P2MP Policy is used for Tree Redundancy. As an example, a P2MP Policy can have multiple candidate paths. Each protecting the primary candidate path. The active path is chosen via the preference of the candidate path.
- o [RFC3209] Defines the instance-ID, instance-ID is used for global optimization of a candidate path with in a P2MP policy. Each Candidate path can have 2 path-instances. These path-instances are equivalent to sub-lsps (instance-IDs). There are used for MBB and global optimization procedures. instance-ID is equivalent to LSP ID
- o [draft-ietf-spring-segment-routing-policy] Segment-list, used for connecting two non-adjacent replication policy via a unicast binding SID or Segment-list.
- o [RFC8306] P2MP End Point objects, used for the PCC to update the PCE with discovered Leaves.
- o [draft-ietf-pce-pcep-extension-for-pce-controller] for programming and identifying the Replication Segment. A new PCE CC Capability sub Tlv is introduced to indicated the support to handle PCE CC based label download for SR P2MP.

- o [draft-ietf-pce-multipath] Forwarding instruction for a P2MP LSP is defined by a set of SR-ERO sub-objects in the ERO object, ERO-ATTRIBUTES object and MULTIPATH-BACKUP TLV as defined in this draft.
- o [RFC8664] SR-ERO Sub Object used in the multipath.

It should be noted that the [draft-dhs-spring-sr-p2mp-policy-yang] can provide further details of the high level P2MP Policy Model.

3.2.2. P2MP Policy Identification

A P2MP Policy and its candidate path can be identified on the root via the P2MP LSP Object. This Object is a variation of the LSP ID Object defined in [RFC8231] and is as follow:

- o PLSP-ID: [RFC8231], is assigned by PCC and is unique per candidate path. It is constant for the lifetime of a PCEP session. Stand-by candidate paths will be assigned a new PLSP-ID by PCC. Stand-by candidate paths can co-exist with the active candidate path.
 - * Note: Every candidate path in the SR-P2MP Policy is unique with its own unique PLSP-ID and Instance-ID. But the same Tree-ID is used for all candidate paths as they are part of the same P2MP Tree.
- o Root-ID: is equivalent to the first node on the P2MP path, as per [RFC3209], Section 4.6.2.1
- o Tree-ID: is equivalent to Tunnel Identifier color which identifies a unique P2MP Policy at a ROOT and is advertised via the PTA in the BGP AD route or can be assigned manually on the root. Tree-ID needs to be unique on the root.
- o Instance-ID: LSP ID Identifier as defined in RFC 3209, is the path-instance identifier and is assigned by the PCC. As it was mentioned the candidate path can have up to two path-instance for global optimization. Note that the Root-ID, Tree-ID and Instance-ID are part of a new SR- P2MP-LSP-IDENTIFIER TLV which will be identified in this draft.
 - * Note: each Path-instance on the Root node is assigned a unique Instance-ID

3.2.3. Replication Segment Identification

The key to identify a replication segment is also a P2MP LSP Object. With varying encoding rules for the SR-P2MP-LSP- IDENTIFIER TLV which will be explained in later sections.

3.2.4. PCECC Use in Replication Segment

PCECC and a variant of CCI object is used in Replication Segment to identify a cross connect. A cross connect is a incoming SID and set of outgoing interfaces and their corresponding SID. The CCI objects contains the incoming SID while the outgoing interfaces are presented via the ERO objects, which each may contain a list of segments.

3.3. High Level Procedures for P2MP SR LSP Instantiation

A P2MP policy can be instantiated via the PCC or the PCE depending on how the root and the leaves are discovered. This document describes two way to discover the root and the leaves:

- o They can be configured and identified on the controller and are considered PCE initiated.
- o They can be discovered on the PCC via MVPN procedures [RFC6513] or legacy multicast protocols like PIM or IGMP etc... and are considered PCC initiated.

3.3.1. PCE-Init Procedure

- o PCE is informed of the P2MP request through its API or configuration mechanism to instantiate a P2MP tunnel.
- o PCE will initiate the P2MP Policy for the request, by sending a PCInitiate message to the Root.
- o Root in response to the PCInitiate message, will generate PLSP-ID for the candidate paths and an Instance-ID for the Path-Instance (LSP-ID) contained with in the candidate path. The tree-id for the P2MP Policy is also filled. PCC will reports back the PLSP-ID, Instance-ID and tree-id via PCRpt message
 - * Optionally, the Root can add any additional leaves that were discovered by multicast procedures in this PCRpt message.
- o PCE will send a PCInitiate message to the Root, Transit and the Leaf nodes to download the Replication Segment information. These messages will utilize the CCI object to encode the forwarding instruction information.

- o PCE will then send a PCUpdate to the root indicating the association information (Candidate path) , and implicitly indicate it to bind to the latest CCI information downloaded.

3.3.2. PCC-Init Procedure

After Root (PCC) discovers the leaves (as an example via MVPN Procedures or other mechanism), the following communication happens between the PCE and PCCs

- o Root sends a PCRpt message for P2MP policy to PCE including the Root-ID, Tree-ID, PLSP-ID, Instance-ID, symbolic-path-name, and any leaves discovered until then.
- o PCE on receiving of this report, will compute the P2MP Policy and its replication segments.
 - * PCE will send a PCInitiate message to the Root, Transit and the Leaf nodes to download the Replication Segment information. These messages will utilize the CCI object to encode the forwarding instruction information.
 - * PCE will then send a PCUpdate to the root indicating the association information (Candidate path) , and implicitly indicate it to bind to the latest CCI information downloaded.

3.3.3. Common Procedure

The following procedures are the same for PCE or PCC Init.

- o PCE will download the replication segments for the Candidate-path's path-instances to all the leaves and transit nodes using PCInitiate message with PLSP-ID = 0, Instance-ID =0, symbolic path name, Root-address, Tree-id(assigned by the root). This PCInitiate message includes the EROs needed for the replication segments. These messages will utilize the CCI object to encode the forwarding instruction information.
- o Any new candidate path for the P2MP Policy is downloaded by PCE to the Root by sending a PCInitiate message
 - * it should be noted, PLSP-ID, Path-Instance ID and the Tree-ID are generated by the PCC for these new candidate paths and their Path-instances
 - * Any update to the Candidate Paths or Replication Segments is done via the PCUpd message. Association object need to be

present for Candidate Path updates and CCI object for the replication segment updates.

- o The PCE will also download the necessary replication segment for the candidate path and its path-instances to the root, leaves and the transit nodes via a PCInit message
- o New leaves can be discovered via Multicast procedures, and new replication segments can be instantiated or existing one updated to reach these leaves
 - * If these leaves reside on routers that are part of the P2MP LSP path, then PCUpd is sent from PCE to necessary PCCs (LEAVES, TRANSIT or ROOT) with the correct PLSP-ID, Instance-ID, Tree-ID and CC-ID.
 - * If the new leaves are residing on routers that are not part of the P2MP Tree yet, then a PCInitiate message is sent down with PLSP-ID=0 and Instance-ID=0 on the corresponding routers.
- o The active candidate-path is indicated by the PCC through the operational bits(Up/Active) of the LSP object in the PCRpt message. If a candidate path needs to be removed, PCE sends PC Initiate message, setting the R-flag in the LSP object and R bit in the SRP-object.
- o To remove the entire P2MP-LSP, PCE needs to send PCInitiate remove messages for every candidate path of the P2MP POLICY to the root and send PCInitiate remove messages for every Replication Regment on all the PCCs on the P2MP Tree. The R bit in the LSP Object as defined in [RFC8231], refers to the removal of the LSP as identified by the SR-P2MP-POLICY-ID-TLV (defined in this document). An all zero (SR-P2MP-LSP-ID-TLV defines to remove all the state of the corresponding PLSP-ID.
- o A candidate path is made active based on the preference of the path. If the Root is programed with multiple candidate paths from different sources, as an example PCE and CLI, based on its tie-breaking rules, if it selects the CLI path, it will send a report to PCE for the PCE path indicating the status of label-download and sets operational bit of the LSP object to UP and Not Active . At any instance, only one path will be active

3.3.4. Global Optimization of the Candidate Path

When a P2MP LSP needs to be optimized for any reason (i.e. it is taking a FRR tunnel or new routers are added to the network) a global optimization of the candidate path is possible.

Each Candidate Path can contain two Path-Instances. The current unoptimized Path-Instance is the active instance and its replication segments are forwarding the multicast PDUs from the root to the leaves. However the second optimized Path-Instance will be setup with its own unique replication segments throughout the network, from the Root to the leaves. These two Path-Instances can co-exist. The two Path-Instances are uniquely identified by their Instance-ID in the SR-P2MP-POLICY-ID-TLV (defined in this document). After the optimized LSP has been downloaded successfully PCC MUST send two reports, reporting UP of the new path indicating the new LSP-ID, and a second reporting the tear down of the old path with the R bit of the LSP Object SET with the old Instance-ID in the SR-P2MP-POLICY-ID-TLV. This MBB procedure will move the multicast PDUs to the optimized Path-Instance.

The leaf should be able to accept traffic from both Path-Instances to minimize the traffic outage by the Make Before Break process.

3.3.5. Fast Reroute

Currently this draft identifies the Facility FRR procedures. In addition, only LINK Protection procedures are defined. How the Facility Path is built and instantiated is beyond the scope of this document.

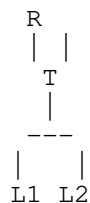


Figure 1

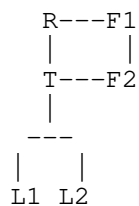


Figure 2

As an example, the bypass path (unicast bypass) between the PLR and MP can be constructed via SR or even via a shared tree (replication segment).

As an example, in figure 1 the detour path between R and T is the 2nd fiber between these nodes. As such the bypass path could be setup on the 2nd fiber. That said in figure 2 the bypass path is traversing multiple nodes and this example a unicast SR path might be ideal for setting up the detour path.

In addition, PHP procedure and implicit null label on the bypass path can be implemented to reduce the PCE programming on the MP PCC.

Optional shared replication segments can be used in networks that do not have unicast SR turned on. These shared replication segments can be programmed on the bypass nodes without a P2MP Policy. The replication segments on primary path can use these shared replication segments as a protection tunnel to protect links.

3.3.6. Connecting Replication Segment via Segment List

There could be nodes between two replication segment that do not support P2MP Policy or Replication segment. It is possible to connect two non-adjacent Replication segments via a unicast segment routing path via a SID list, comprised of any IGP supported segment types (ex: Binding, Adjacency, Node) to forward to the next replicating node. This information is encoded via the SR-ERO sub-objects and ERO-attributes objects. The last segment in an encoding SID list MUST be a replication segment

3.4. SR P2MP Policy and Replication Segment TLVs and Objects

3.4.1. SR P2MP Policy Objects

SR P2MP Policy can be constructed via the following objects

<Common Header>

<SRP>

<P2MP LSP>

[<association-list>]

optionally if the root is updating the PCE with end point list the end-point-list object can be added.

[<end-points-list>]

3.4.2. Replication Segment Objects

Replication segment can be constructed via the following objects

```

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

```

Path-attribute as per [draft-ietf-pce-multipath]

3.4.3. P2MP Policy and Replication Segment general considerations

The above new objects and TLV's defined in this document can be included in PCRpt, PCInitiate and PCUpd messages.

It should be noted that every PCRpt, PCInitiate and PCUpd messages will contain full list of the Leaves and segment and forwarding information that is needed to build the Candidate path and its Replication segments. They will never send the delta information related to the new leaves or forwarding information that need to be added or updated. This is necessary to ensure that PCE or any new PCE is in sync with the PCC.

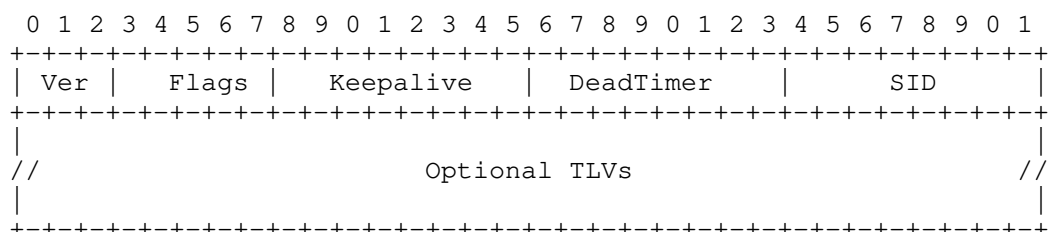
When a PCRpt, PCInitiate and PCUpd messages is sent via PCEP it maintains the previous ERO Path IDs and generates new Path IDs for new instructions, as per [draft-ietf-pce-multipath]. The PATH IDs are maintained for each specific forwarding instructions until the instructions are deleted. For example: When the first leaf is added, the PCE will update with PathID 1 to the PCC. When the second leaf is added, according to the path calculated, PCE might just append the existing instruction Path ID 1 with a new Path ID 2 to construct the new PCUpd message.

The CCI Object 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. Leaves and their corresponding Path IDs can be removed from the cross connect identified via the CCI. The CC-ID is assigned by the PCE.

4. Object Format

4.1. Open Message and Capability Exchange

Format of the open Object:



All the nodes need to establish a PCEP connection with the PCE.

During PCEP Initialization Phase, PCEP Speakers need to set flags N, M, P in the STATEFUL-PCE-CAPABILITY TLV as defined in [draft-ietf-pce-stateful-pce-p2mp] section-5.2

This draft extends the PCEP OPEN object by defining an optional TLV to indicate the PCE's capability to perform SR-P2MP path computations with a new IANA capability type.

The inclusion of this TLV in an OPEN object indicates that the sender can perform SR-P2MP path computations. This will be similar to the P2MP-CAPABILITY defined in [RFC8306] section-3.1.2 and a new value needs to be defined for SR-P2MP.

4.1.1. PCECC Path Setup Capability

A PST of PCECC is also added as per [draft-ietf-pce-pcep-extension-for-pce-controller].

This document also introduces a new bit S in the SR PCECC capacity Sub TLV indicating the support to handle PCECC based label download for Replication segment.



Also, the N,M,P bits in STATEFUL-PCE-CAPABILITY TLV should be SET.

4.1.2. Association Type Capability

A Assoc-Type-List TLV as per [RFC8697] section 3.4 should be send via PCEP open object with following association type

Association Type Value	Association Name	Reference
TBD1	P2MP SR Policy Association	This document

OP-CONF-Assoc-RANGE (Operator-configured Association Range) should not be set for this association type and must be ignored.

The open message MUST include the MULTIPATH CAPABILITY TLV as defined in [draft-ietf-pce-multipath]

4.2. Symbolic Name in PCInit Message from PCC

As per [RFC8231] section 7.3.2. a Symbolic Path Name TLV should uniquely identify the P2MP path on the PCC. This symbolic path name is a human-readable string that identifies an P2MP LSP in the network. It needs to be constant through the lifetime of the P2MP path.

As an example in the case of P2MP LSP the symbolic name can be p2mp policy name + candidate path name of the LSP.

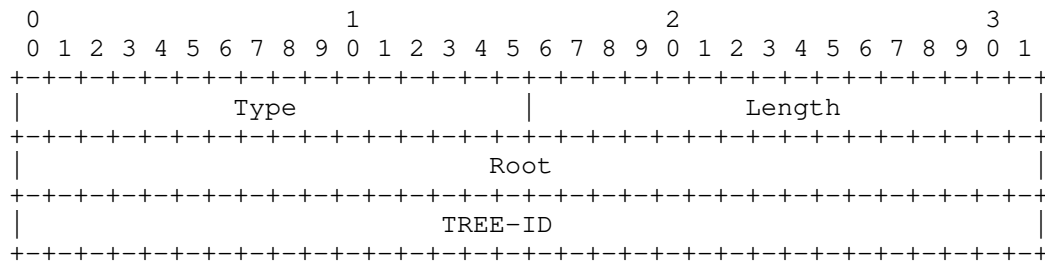
4.3. P2MP Policy Specific Objects and TLVs

4.3.1. P2MP Policy Association Group for P2MP Policy

Two ASSOCIATION object types for IPv4 and IPv6 are defined in [RFC8697]. The ASSOCIATION object includes "Association type" indicating the type of the association group. This document adds a new Association type. Association type = TBD1 "P2MP SR Policy Association Type" for SR Policy Association Group (P2MP SRPAG). As per [draft-barth-pce-segment-routing-policy-cp] section 5, three new TLVs are identified to carry association information: P2MP-SRPAG-POL-ID-TLV, P2MP-SRPAG-CPATH-ID-TLV, P2MP-SRPAG-CPATH-ATTR-TLV

4.3.1.1. P2MP SR Policy Association Group Policy Identifiers TLV

The P2MP-SRPOLICY-POL-ID TLV is a mandatory TLV for the P2MP-SRPAG Association. Only one P2MP-SRPOLICY-POL-ID TLV can be carried and only the first occurrence is processed and any others MUST be ignored.



Type: TBD2 for "P2MP-SR-POLICY-POL-ID" TLV.

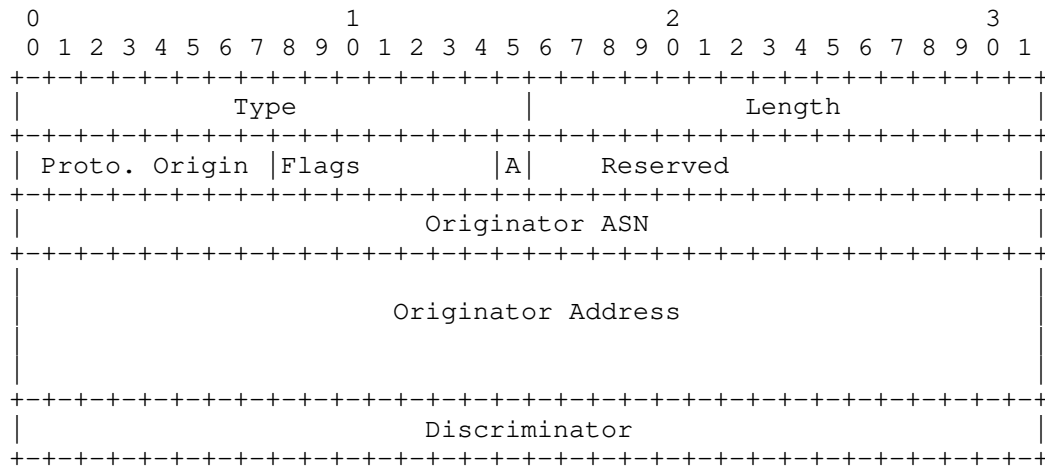
Length: 8 or 20, depending on length of End-point (IPv4 or IPv6)

Tunnel Sender Address : Can be either IPv4 or IPv6, this value is the value of the root loopback IP.

Tree-ID: Tree ID that the replication segment is part of as per draft-ietf-spring-sr-p2mp-policy

4.3.1.2. P2MP SR Policy Association Group Candidate Path Identifiers TLV

The P2MP-SRPOLICY-CPATH-ID TLV is a mandatory TLV for the P2MPSRPAG Association. Only one P2MP-SRPOLICY-CPATH-ID TLV can be carried and only the first occurrence is processed and any others MUST be ignored.



Type: TBD3 for "P2MP-SR-POLICY-CPATH-ID" TLV.

Length: 28.

Protocol Origin: 8-bit value that encodes the protocol origin, as specified in [I-D.ietf-spring-segment-routing-policy] Section 2.3.

Flags : A: This candidate path is active. At any instance only one candidate path can be active. PCC indicates the active candidate path to PCE through this bit. Reserved: MUST be set to zero on transmission and ignored on receipt.

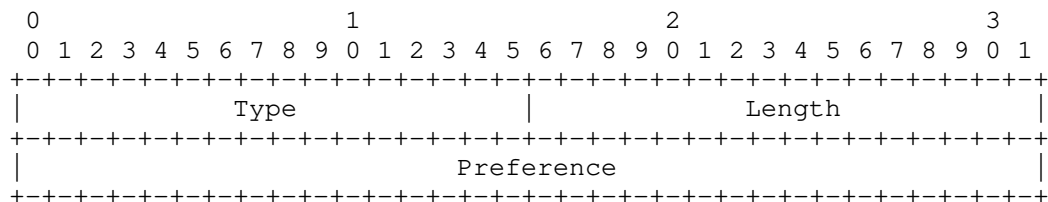
Originator ASN: Represented as 4 byte number, part of the originator identifier, as specified in [draft-ietf-spring-segment-routing-policy] Section 2.4.

Originator Address: Represented as 128 bit value where IPv4 address are encoded in lowest 32 bits, part of the originator identifier, as specified in [draft-ietf-spring-segment-routing-policy] Section 2.4.

Discriminator: 32-bit value that encodes the Discriminator of the candidate path.

4.3.1.3. P2MP SR Policy Association Group Candidate Path Attributes TLV

The P2MP-SRPOLICY-CPATH-ATTR TLV is an optional TLV for the SRPAG Association. Only one P2MP-SRPOLICY-CPATH-ATTR TLV can be carried and only the first occurrence is processed and any others MUST be ignored.



Type: TBD4 for "P2MP-SRPOLICY-CPATH-ATTR" TLV.

Length: 4. Preference: Numerical preference of the candidate path, as specified in [draft-ietf-spring-segment-routing-policy] Section 2.7.

If the TLV is missing, a default preference of 100 as specified in [draft-ietf-spring-segment-routing-policy] is used.

4.3.2. P2MP-END-POINTS Object

In order for the Root to indicate operations of its leaves (Add/Remove/Modify/DoNotModify), the PC Report message is

extended to include P2MP End Point <P2MP End-points> Object which is defined in [RFC8306]

The format of the PC Report message is as follow:

<Common Header>

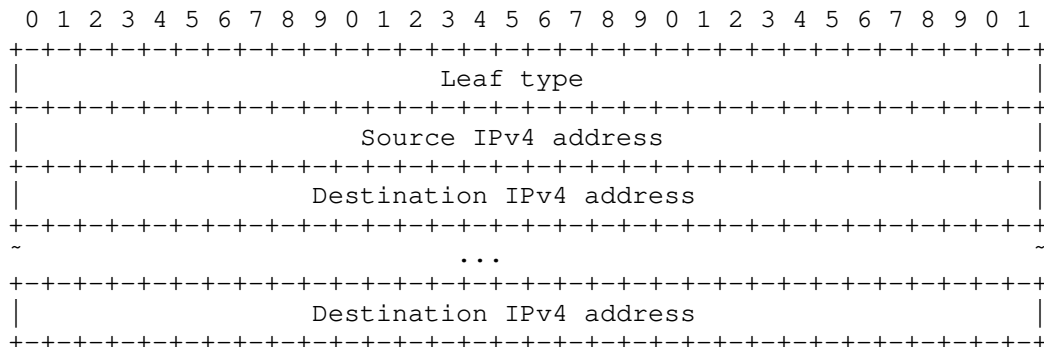
[<SRP>]

<LSP>

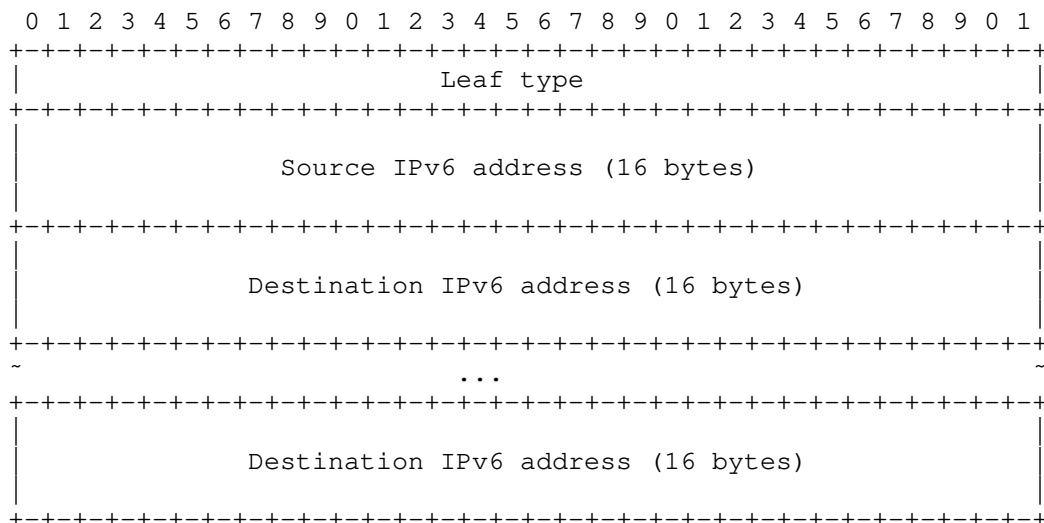
[<association-list>]

[<end-points-list>]

IPv4-P2MP END-POINTS:



IPv6-P2MP END-POINTS:



Leaf Types (derived from [RFC8306] section 3.3.2) :

1. New leaves to add (leaf type = 1)
2. Old leaves to remove (leaf type = 2)
3. Old leaves whose path can be modified/reoptimized (leaf type = 3), Future reserved not used for tree SID as of now.
4. Old leaves whose path must be left unchanged (leaf type = 4)

5. the entire pce leaf list is overwritten and replaced with the new leaf list (leaf type = 5)

A given P2MP END-POINTS object gathers the leaves of a given type. Note that a P2MP report can mix the different types of leaves by including several P2MP END-POINTS objects. The END-POINTS object body has a variable length. These are multiples of 4 bytes for IPv4, multiples of 16 bytes, plus 4 bytes, for IPv6.

4.4. P2MP Policy and Replication Segment Identifier Object and TLV

As it was mentioned previously both P2MP Policy and Replication Segment are identified via the LSP object and more precisely via the SR-P2MP-LSPID-TLV

The P2MP Policy uses the PLSP-ID to identify the Candidate Paths and the Instance-ID to identify a Path-Instance within the Candidate path.

On the other hand the Replication Segment uses the SR-P2MP-LSPID-TLV to identify and correlate a Replication Segment to a P2MP Policy

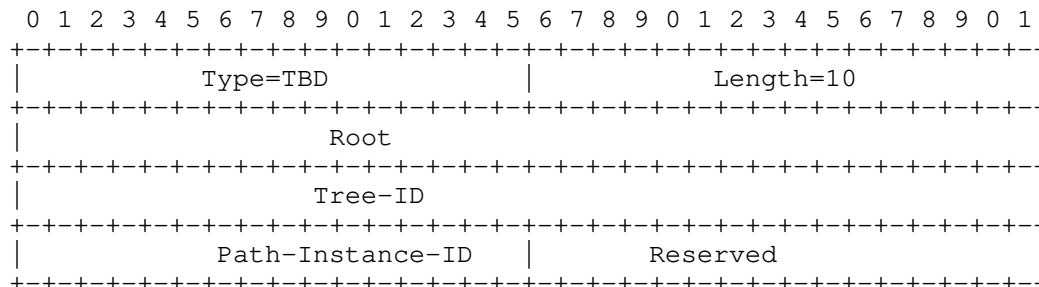
As it was noted previously on the Root, the P2MP Policy and the Replication Segment is downloaded via the same PCUpd message.

4.4.1. Extension of the LSP Object, SR-P2MP-LSPID-TLV

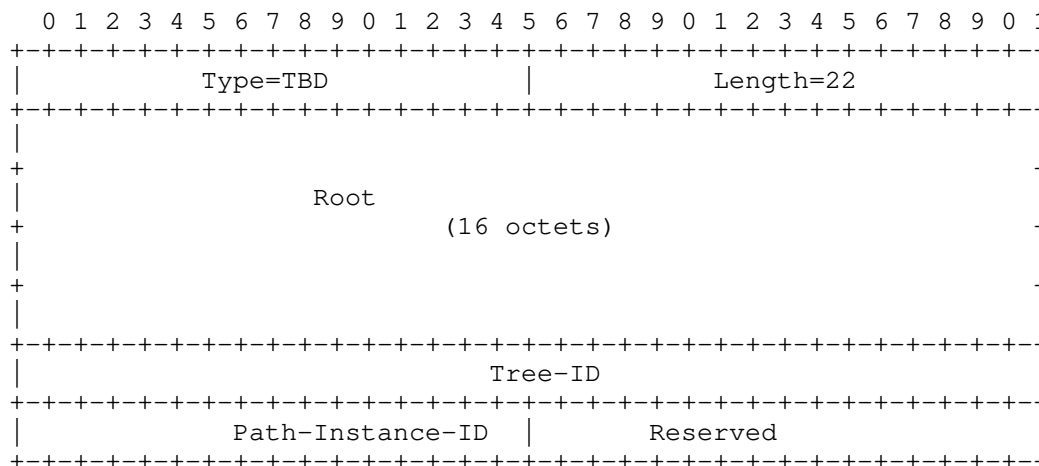
The LSP Object is defined in Section 7.3 of [RFC8231]. It specifies the PLSP-ID to uniquely identify an LSP that is constant for the life time of a PCEP session. Similarly, for a P2MP tunnel, the PLSP-ID identifies a Candidate Path uniquely within the P2MP policy.

The LSP Object MUST include the new SR-P2MP-POLICY-ID-TLV (IPv4/IPv6) defined in this document below. This is a variation to the P2MP object defined in [draft-ietf-pce-stateful-pce-p2mp]

SR-IPV4-P2MP-POLICY-ID TLV:



SR-IPV6-P2MP-POLICY-ID TLV :



The type (16-bit) of the TLV is TBD (need allocation by IANA).

Root: Source Router IP Address

Tree-ID: Unique Identifier of this P2MP LSP on the Root.

Instance-ID : Contains 16 Bit instance ID.

4.5. Replication Segment

As per [draft-ietf-spring-sr-replication-segment] a replication segment has a next-hop-group which MAY contain a single outgoing replication SID or a list of SIDs (sr-policy-sid-list) In either case there needs to be a replication SID at the bottom of the stack. This

means two replication segments can be directly connected or connected via a SR domain.

4.5.1. The format of the replication segment message

The format of a Replication Segment message encoding is similar to P2MP Policy. However, the P2MP Policy contains the association object and the replication segment message does not contain the association object. In addition the replication segment uses the CCI object to identify a P2MP cross connect. The replication segment is downloaded individually to the root, transit and leaf nodes without the P2MP Policy. The P2MP Policy is a Root Concept. The replication segment uses SR-P2MP-LSPID-TLV as its identifier. The TLV is coded differently for shared and non-shared case.

- o In the case of a replication segment being shared, the Tree-ID in the SR-P2MP-POLICY Identifier TLV is the replication-id of the Replication Segment and Root = 0, Instance-Id = 0. When downloading a shared replication segment from PCE through a PCEInitiate message, the SR-P2MP-POLICY Identifier TLV is all 0, and on the report back from PCC, PCC generates PLSP-ID, Replication-id (Tree-id field will be populated with replication-id). Instance-id will be 0.

4.5.2. PCECC

The CCI Object as defined in [draft-ietf-pce-pcep-extension-for-pce-controller] is used to identify a forwarding instruction in the Replication Segment. A forwarding instruction is incoming SID and a set of outgoing branches. The CCI Object-Type of 1 is used for the MPLS Label. The label in the CCI Object is the incoming SID. The outgoing SIDs are defined by the ERO Objects.

The CCI Object can be include in Reports, initiate and Update messages for Replication Segments.

The PCEInitiate message defined in [RFC8281] and extended in [draft-ietf-pce-pcep-extension-for-pce-controller] is further extended to support SR-P2MP replication segment based central control instructions.

The format of the extended PCInitiate message is as follows:

```
<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> |
                                         (<CCI><intended-path>))
```

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

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

Where:

```
<PCE-initiated-lsp-instantiation> and
<PCE-initiated-lsp-deletion> are as per
[RFC8281].
```

The LSP and SRP object is defined in [RFC8231]. The <intended-path> is as per [RFC8281] [draft-ietf-pce-multipath] (PATH-ATTRIB and ERO).

The format of the PCRpt message is as follows:

```
<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> |
                               (<CCI><intended-path>))
```

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

Where:

<path> is as per [RFC8231] and the LSP and SRP object are also defined in [RFC8231].

The <intended-path> is as per [draft-ietf-pce-multipath] (PATH-ATTRIB and ERO).

This document extends the use of PCUpd message with SR-P2MP CCI as follows:

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

Where:

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

```
<update-request> ::= (<lsp-update-request> |
                        <central-control-update>)
```

```
<lsp-update-request> ::= <SRP>
                        <LSP>
                        <path>
```

```
<central-control-update> ::= <SRP>
                        <LSP>
                        (<CCI><intended-path>)
```

Where:

<path> is as per [RFC8231] and the LSP and SRP object are also defined in [RFC8231].

The <intended-path> is as per [draft-ietf-pce-multipath] (PATH-ATTRIB and ERO).

4.5.3. Label action rules in replicating segment

The node action and role of ingress, transit, leaf or bud, is indicated via a new Node Role TLV. This document introduces a new SR-P2MP-NODE-ROLE TLV (Type To be assigned by IANA) that will be present in the PATH-ATTRIB object.

```

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=4                |
+-----+-----+-----+-----+-----+-----+-----+-----+
|   Role Type   |                                     | Reserved          |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

- o ingress, role type = 1
- o transit, role type = 2
- o leaf, role type = 3
- o bud, role type = 4

4.5.4. SR-ERO Rules

Forwarding information of a replication segment can be configured and steered via many different mechanisms.

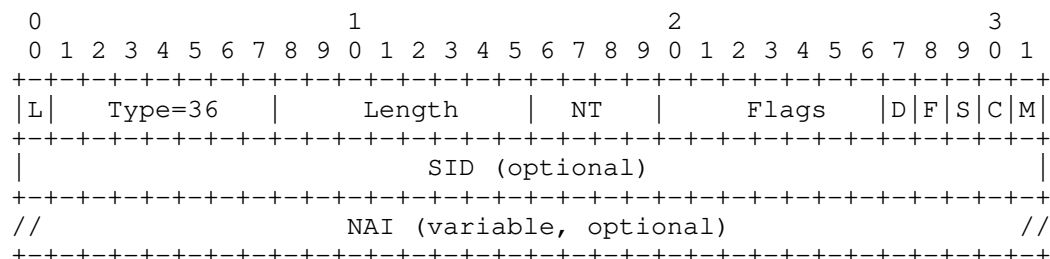
As an example a replication SID can be steered via:

1. Replication SID steered with an IPv4/IPv6 directly connected nexthop
 - * In this case there will be two SR-ERO in the ERO Object, with the Replication SID SR-ERO at the bottom and the IPv4/IPv6 SR-ERO on the top.
2. Replication SID steered with an IPv4/IPv6 loopback address that reside on the directly connected router.
 - * In this case there will be two SR-ERO in the ERO Object, with the Replication SID SR-ERO at the bottom and the IPv4/IPv6 SR-ERO on the top.
 - * In addition a new flag D is added to the SR-ERO to signal that the Loopback nexthop is connected to the directly attached router.
3. Replication SID steered with unnumbered IPv4/IPv6 directly connected Interface
4. Replication SID steered via a SR adjacency or node SID
 - * In this case even a sid-list can be used to traffic engineer the path between two Replication Segment
 - * The Replication SID SR-ERO is at the bottom while the segments describing the path are on top in order.

4.5.4.1. SR-ERO subobject changes

SR-ERO from RFC 8664 is used to construct the forwarding information needed for Replication Segment.

A new D flag was added to indicate a loopback nexthop that is residing on the directly attached router. It should be noted that this flag should be set only for the loopback case and not for a local interface as a nexthop.



Flags : F, S, C, M are already defined in rfc8664.

This document defines a new flag D: If the next-hop in NAI field is system IP or loopback, this bit indicates whether the system IP / loopback is directly connected router or not. If set indicates directly connected address. When this bit is set, F bit should be 0 (meaning NAI should be present)

5. Tree Deletion

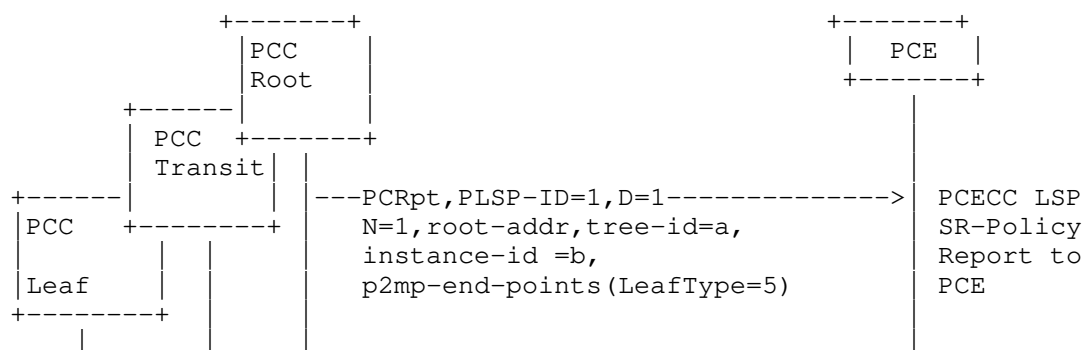
To delete the entire tree (P2MP LSP), Root send a PCRpt message with the R bit of the LSP object set and all the fields of the SR-P2MP-LSP-ID TLV set to 0(indicating to remove all state associated with this P2MP tunnel). The PCE in response sends a PCInitiate message with R bit in the SRP object SET to all nodes along the path to indicate deletion of the entries.

6. Fragmentation

The Fragmentation bit in the LSP object (F bit) can be used to indicate a fragmented PCEP message

7. Example Workflows

PCC-Initiated Workflow

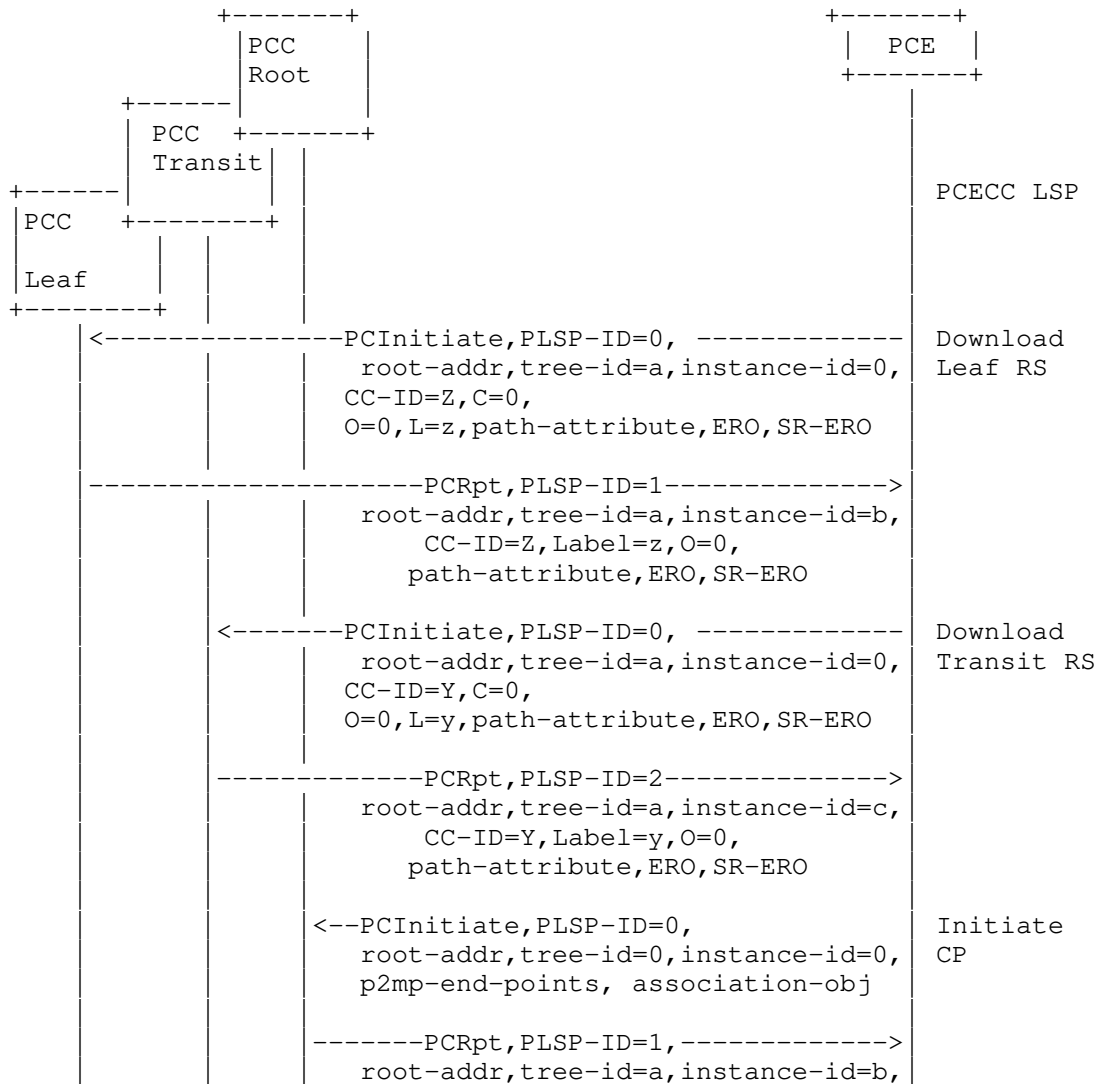


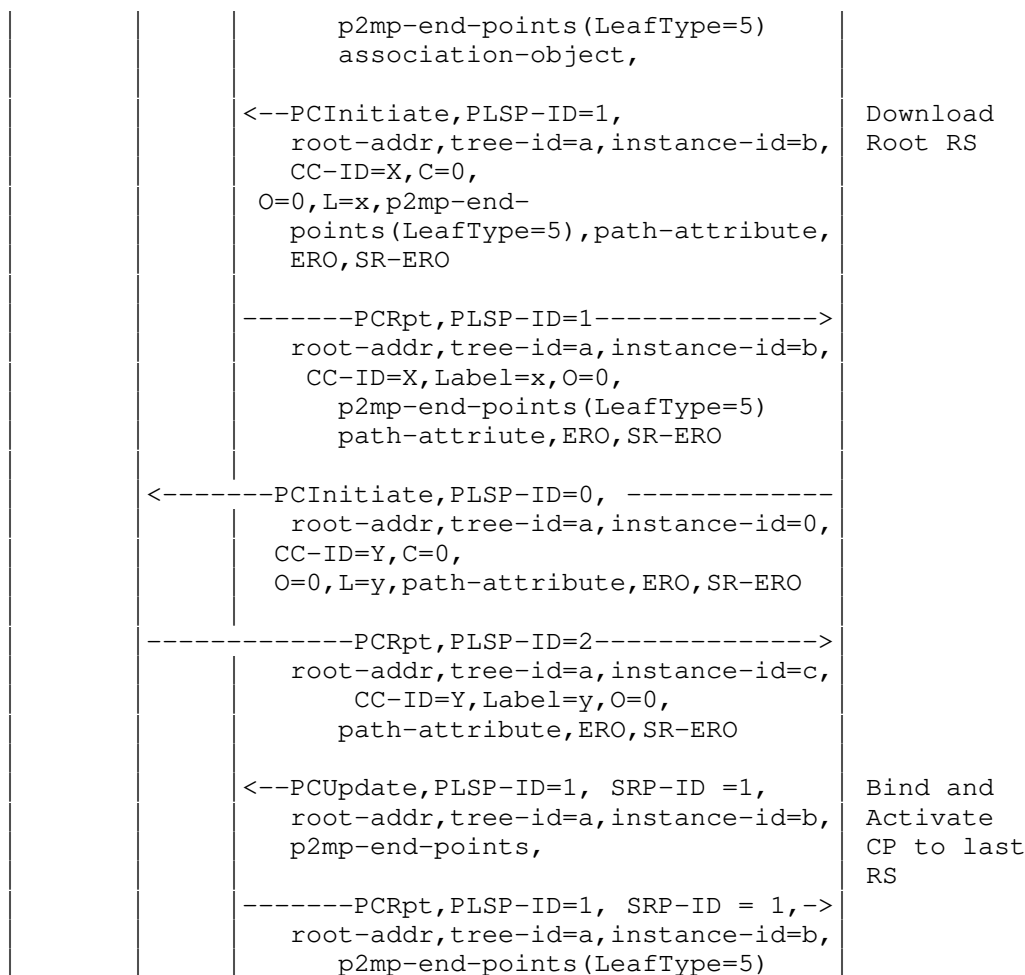
		<pre><--PCUpdate,PLSP-ID=1, SRP-ID =1, root-addr,tree-id=a,instance-id=b, p2mp-end-points, association-obj</pre>	Update CP
		<pre>-----PCRpt,PLSP-ID=1, SRP-ID = 1,-> root-addr,tree-id=a,instance-id=b, p2mp-end-points(LeafType=5) association-object,</pre>	
<-----		<pre>PCInitiate,PLSP-ID=0, ----- root-addr,tree-id=a,instance-id=0, CC-ID=Z,C=0, O=0,L=z,path-attribute,ERO,SR-ERO</pre>	Download Leaf Replication Segment (RS)
		<pre>-----PCRpt,PLSP-ID=1-----> root-addr,tree-id=a,instance-id=b, CC-ID=Z,Label=z,O=0, path-attribute,ERO,SR-ERO</pre>	
	<-----	<pre>PCInitiate,PLSP-ID=0, ----- root-addr,tree-id=a,instance-id=0, CC-ID=Y,C=0, O=0,L=y,path-attribute,ERO,SR-ERO</pre>	Download Transit RS
		<pre>-----PCRpt,PLSP-ID=2-----> root-addr,tree-id=a,instance-id=c, CC-ID=Y,Label=y,O=0, path-attribute,ERO,SR-ERO</pre>	
		<pre><--PCInitiate,PLSP-ID=1, root-addr,tree-id=a,instance-id=b, CC-ID=X,C=0, O=0,L=x,p2mp-end- points(LeafType=5),path-attribute, ERO,SR-ERO</pre>	Download Root RS
		<pre>-----PCRpt,PLSP-ID=1-----> root-addr,tree-id=a,instance-id=b, CC-ID=X,Label=x,O=0, p2mp-end-points(LeafType=5) path-attribute,ERO,SR-ERO</pre>	
		<pre><--PCUpdate,PLSP-ID=1, SRP-ID =2, root-addr,tree-id=a,instance-id=b, p2mp-end-points</pre>	Activate CP to last RS
		<pre>-----PCRpt,PLSP-ID=1, SRP-ID =2, -> root-addr,tree-id=a,instance-id=b,</pre>	

p2mp-end-points (LeafType=5)

Note that on transit / leaf Initiate is with PLSP-ID = 0. Therefore PLSP-ID is locally unique to a node. It should be noted that the CC-ID does not need to be constant across all nodes that make up the path.

PCE-Initiated workflow

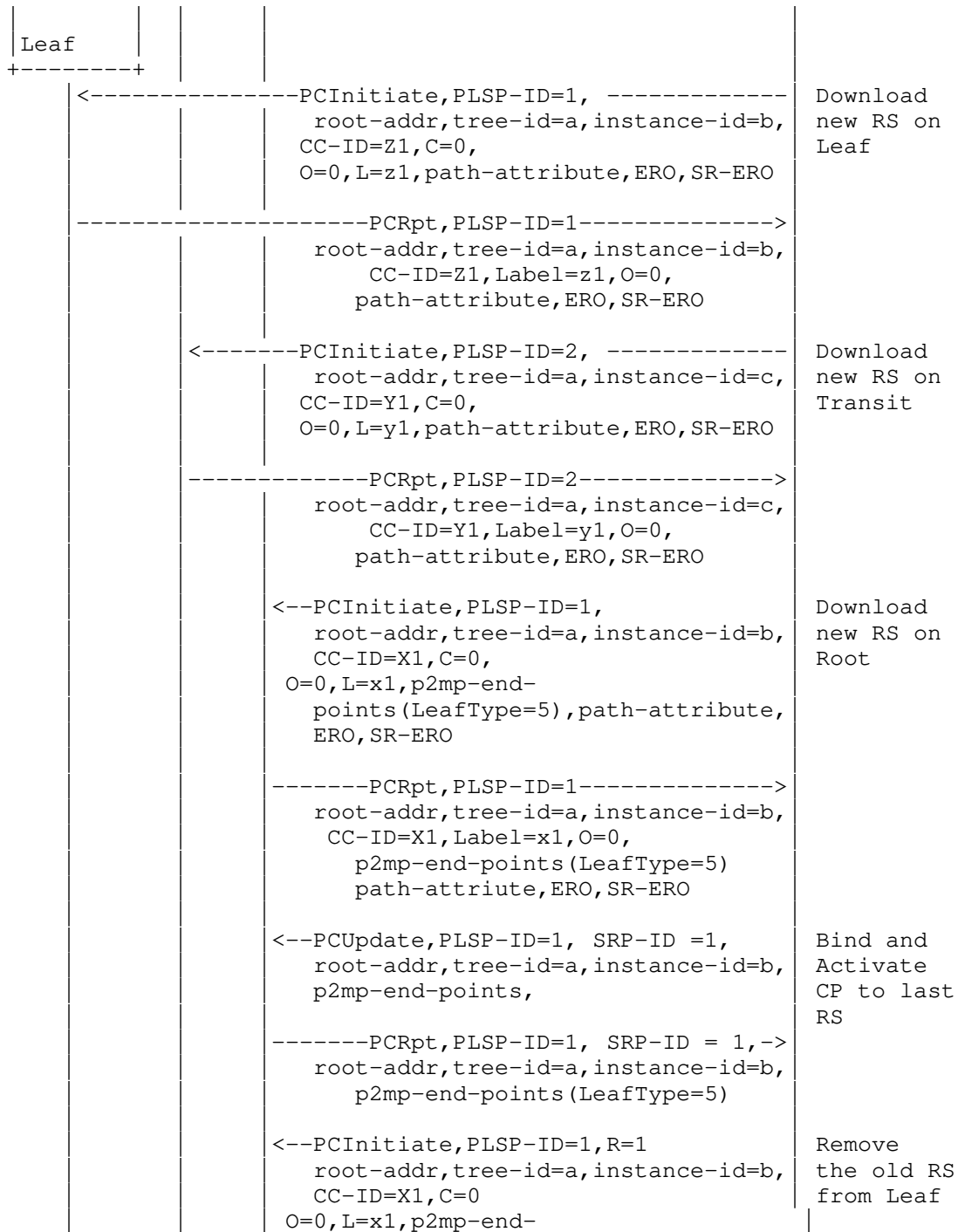


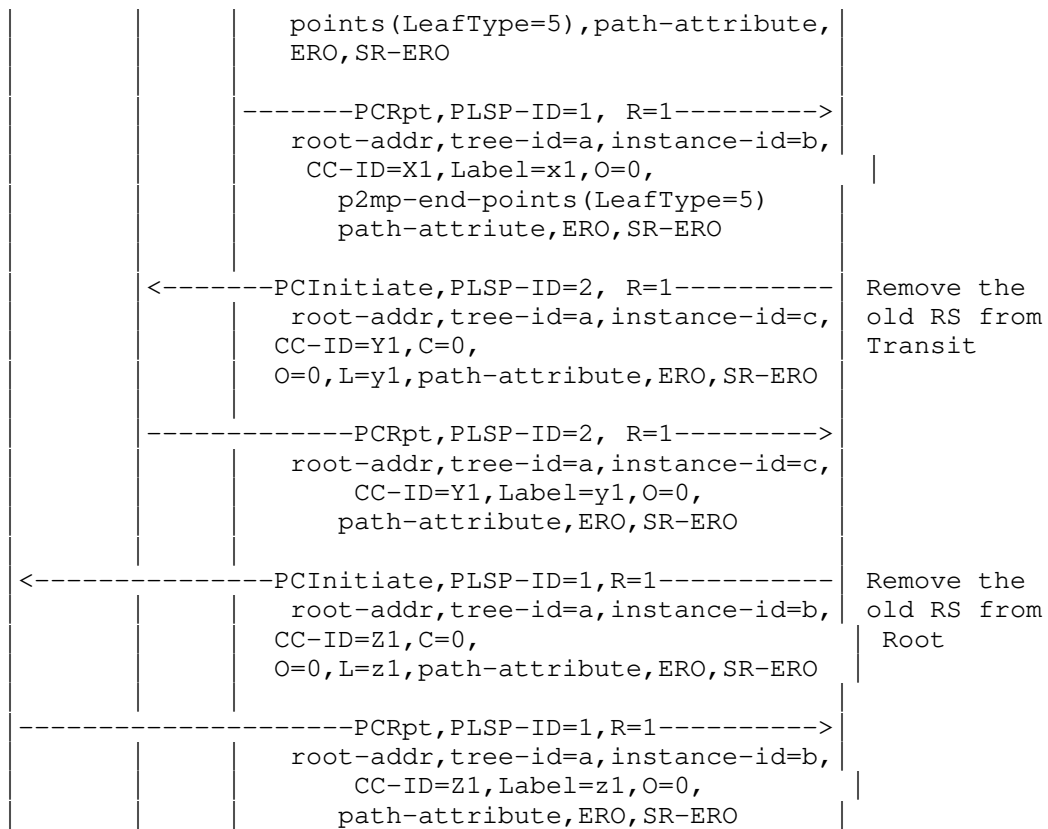


MBB Workflow:

Common (PCE-INIT, PCC-INIT) MBB







8. IANA Consideration

1. This draft extends the PCEP OPEN object by defining an optional TLV to indicate the PCE's capability to perform SR-P2MP path computations with a new IANA capability type (TBD).
2. PCEP open object with a new association type " P2MP SR Policy Association " value (TBD).
3. A new Association type. Association type = TBD1 "P2MP SR Policy Association Type" for SR Policy Association Group (P2MP SRPAG)
 1. three new TLVs are identified to carry association information: P2MP-SRPAG- POL-ID-TLV, P2MP-SRPAG-CPATH-ID-TLV, P2MP-SRPAG-CPATH-ATTR-TLV
4. Two new TLVs for Identifying the P2MP Policy and the Replication segment SR-IPV4-P2MP-POLICY-ID TLV and SR-IPV6-P2MP-POLICY-ID TLV

5. A new SR-P2MP-NODE-ROLE TLV (Type To be assigned by IANA) that will be present in the PATH-ATTRIB object

9. Security Considerations

TBD

10. Acknowledgments

The authors would like to thank Tanmoy Kundu and Stone Andrew at Nokia for their feedback and major contribution to this draft.

11. References

11.1. Normative References

[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>>.

11.2. Informative References

[draft-barth-pce-segment-routing-policy-cp]

.

[draft-dhs-spring-sr-p2mp-policy-yang]

.

[draft-ietf-pce-multipath]

.

[draft-ietf-pce-pcep-extension-for-pce-controller]

.

[draft-ietf-pce-segment-routing-policy-cp]

.

[draft-ietf-pce-stateful-pce-p2mp]

.

[draft-ietf-pim-sr-p2mp-policy]

"D. Yoyer, C. Filsfils, R.Prekh, H.bidgoli, Z. Zhang,
"draft-voyer-pim-sr-p2mp-policy"", October 2019.

[draft-ietf-spring-segment-routing-policy]

.

[draft-ietf-spring-sr-replication-segment]
"D. Yoyer, C. Filsfils, R.Prekh, H.bidgoli, Z. Zhang,
"draft-voyer-pim-sr-p2mp-policy "draft-voyer-spring-sr-
replication-segment"", July 2020.

[draft-parekh-bess-mvpn-sr-p2mp]

.

[draft-sivabalan-pce-binding-label-sid]

.

[RFC3209] .

[RFC5440] .

[RFC6513] .

[RFC8231] .

[RFC8236] .

[RFC8281] .

[RFC8306] .

[RFC8664] .

[RFC8697] .

Authors' Addresses

Hooman Bidgoli (editor)
Nokia
Ottawa
Canada

Email: hooman.bidgoli@nokia.com

Daniel Voyer
Bell Canada
Montreal
Canada

Email: daniel.yover@bell.ca

Saranya Rajarathinam
Nokia
Mountain View
US

Email: saranya.Rajarathinam@nokia.com

Ehsan Hemmati
Cisco System
San Jose
USA

Email: ehemmati@cisco.com

Tarek Saad
Juniper Networks
Ottawa
Canada

Email: tsaad@juniper.com

Siva Sivabalan
Ciena
Ottawa
Canada

Email: ssivabal@ciena.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2021

S. Sivabalan
Ciena Corporation
C. Filsfils
Cisco Systems, Inc.
J. Tantsura
Apstra, Inc.
J. Hardwick
Metaswitch Networks
S. Previdi
C. Li
Huawei Technologies
October 31, 2020

Carrying Binding Label/Segment-ID in PCE-based Networks.
draft-ietf-pce-binding-label-sid-05

Abstract

In order to provide greater scalability, network opacity, and service independence, Segment Routing (SR) utilizes a Binding Segment Identifier (BSID). It is possible to associate a BSID to RSVP-TE signaled Traffic Engineering Label Switching Path or binding Segment-ID (SID) to SR Traffic Engineering path. Such a binding label/SID can be used by an upstream node for steering traffic into the appropriate TE path to enforce SR policies. This document proposes an approach for reporting binding label/SID to Path Computation Element (PCE) for supporting PCE-based Traffic Engineering policies.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 May 4, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	5
3. Path Binding TLV	6
3.1. SRv6 Endpoint Behavior and SID Structure	7
4. Operation	8
5. Binding SID in SR-ERO	10
6. Binding SID in SRv6-ERO	10
7. Implementation Status	10
7.1. Huawei	11
7.2. Cisco	11
8. Security Considerations	11
9. Manageability Considerations	12
9.1. Control of Function and Policy	12
9.2. Information and Data Models	12
9.3. Liveness Detection and Monitoring	12
9.4. Verify Correct Operations	12
9.5. Requirements On Other Protocols	12
9.6. Impact On Network Operations	12
10. IANA Considerations	13
10.1. PCEP TLV Type Indicators	13
10.1.1. TE-PATH-BINDING TLV	13
10.1.2. Binding SID Flags	13
10.2. PCEP Error Type and Value	14
11. Acknowledgements	14

12. References	14
12.1. Normative References	14
12.2. Informative References	16
Appendix A. Contributor Addresses	17
Authors' Addresses	17

1. Introduction

A PCE can compute Traffic Engineering paths (TE paths) through a network that are subject to various constraints. Currently, TE paths are either set up using the RSVP-TE signaling protocol or Segment Routing (SR). We refer to such paths as RSVP-TE paths and SR-TE paths respectively in this document.

As per [RFC8402] SR allows a headend node to steer a packet flow along any path. The headend node is said to steer a flow into an Segment Routing Policy (SR Policy). Further, as per [I-D.ietf-spring-segment-routing-policy], an SR Policy is a framework that enables instantiation of an ordered list of segments on a node for implementing a source routing policy with a specific intent for traffic steering from that node.

As described in [RFC8402], Binding Segment Identifier (BSID) is bound to an Segment Routed (SR) Policy, instantiation of which may involve a list of SIDs. Any packets received with an active segment equal to BSID are steered onto the bound SR Policy. A BSID may be either a local (SR Local Block (SRLB)) or a global (SR Global Block (SRGB)) SID. As per Section 6.4 of [I-D.ietf-spring-segment-routing-policy] a BSID can also be associated with any type of interfaces or tunnel to enable the use of a non-SR interface or tunnels as segments in a SID-list.

[RFC5440] describes the Path Computation Element Protocol (PCEP) for communication between a Path Computation Client (PCC) and a PCE or between a pair of PCEs as per [RFC4655]. [RFC8231] specifies extension to PCEP that allows a PCC to delegate its LSPs to a stateful PCE. A stateful PCE can then update the state of LSPs delegated to it. [RFC8281] specifies a mechanism allowing a PCE to dynamically instantiate an LSP on a PCC by sending the path and characteristics. The PCEP extension to setup and maintain SR-TE paths is specified in [RFC8664].

[RFC8664] provides a mechanism for a network controller (acting as a PCE) to instantiate candidate paths for an SR Policy onto a head-end node (acting as a PCC) using PCEP. For more information on the SR Policy Architecture, see [I-D.ietf-spring-segment-routing-policy].

Binding label/SID has local significance to the ingress node of the corresponding TE path. When a stateful PCE is deployed for setting up TE paths, it may be desirable to report the binding label or SID to the stateful PCE for the purpose of enforcing end-to-end TE/SR policy. A sample Data Center (DC) use-case is illustrated in the following diagram. In the MPLS DC network, an SR LSP (without traffic engineering) is established using a prefix SID advertised by BGP (see [RFC8669]). In IP/MPLS WAN, an SR-TE LSP is setup using the PCE. The list of SIDs of the SR-TE LSP is {A, B, C, D}. The gateway node 1 (which is the PCC) allocates a binding SID X and reports it to the PCE. In order for the access node to steer the traffic over the SR-TE LSP, the PCE passes the SID stack {Y, X} where Y is the prefix SID of the gateway node-1 to the access node. In the absence of the binding SID X, the PCE should pass the SID stack {Y, A, B, C, D} to the access node. This example also illustrates the additional benefit of using the binding SID to reduce the number of SIDs imposed on the access nodes with a limited forwarding capacity.

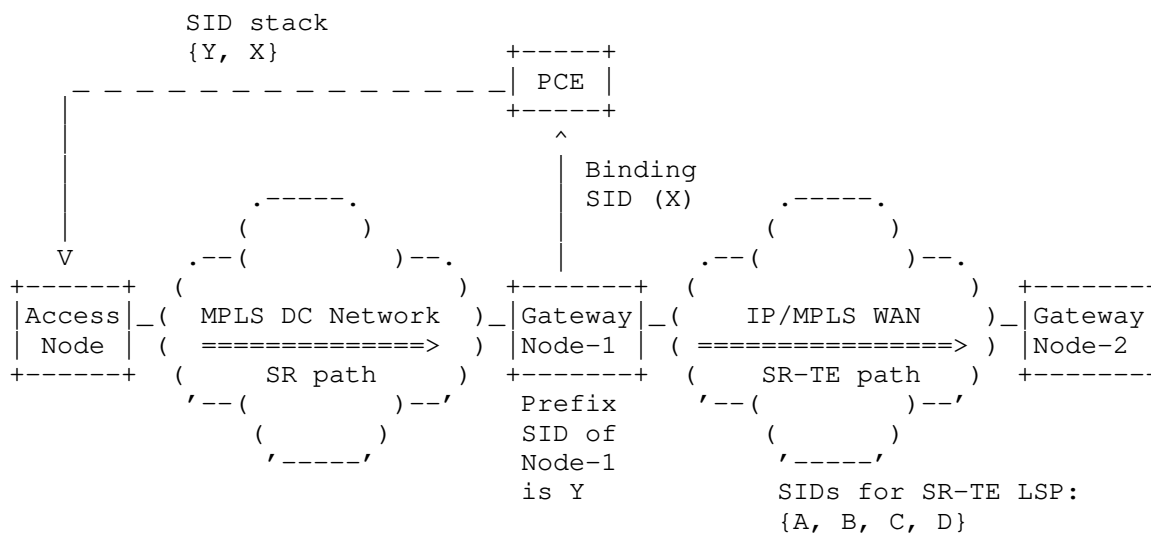


Figure 1: A sample Use-case of Binding SID

A PCC could report the binding label/SID allocated by it to the stateful PCE via Path Computation State Report (PCRpt) message. It is also possible for a stateful PCE to request a PCC to allocate a specific binding label/SID by sending an Path Computation Update Request (PCUpd) message. If the PCC can successfully allocate the specified binding value, it reports the binding value to the PCE.

Otherwise, the PCC sends an error message to the PCE indicating the cause of the failure. A local policy or configuration at the PCC SHOULD dictate if the binding label/SID needs to be assigned.

In this document, we introduce a new OPTIONAL TLV that a PCC can use in order to report the binding label/SID associated with a TE LSP, or a PCE to request a PCC to allocate a specific binding label/SID value. This TLV is intended for TE LSPs established using RSVP-TE, SR, or any other future method. Also, in the case of SR-TE LSPs, the TLV can carry a binding MPLS label (for SR-TE path with MPLS data-plane) or a binding IPv6 SID (e.g., IPv6 address for SR-TE paths with IPv6 data-plane). Binding value means either MPLS label or SID throughout this document.

Additionally, to support the PCE based central controller [RFC8283] operation where the PCE would take responsibility for managing some part of the MPLS label space for each of the routers that it controls, the PCE could directly make the binding label/SID allocation and inform the PCC. See [I-D.ietf-pce-pcep-extension-for-pce-controller] for details.

2. Terminology

The following terminologies are used in this document:

BSID: Binding Segment Identifier.

LER: Label Edge Router.

LSP: Label Switched Path.

LSR: Label Switching Router.

PCC: Path Computation Client.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

RSVP-TE: Resource ReserVation Protocol-Traffic Engineering.

SID: Segment Identifier.

SR: Segment Routing.

SRGB: Segment Routing Global Block.

SRLB: Segment Routing Local Block.

TLV: Type, Length, and Value.

3. Path Binding TLV

The new optional TLV is called "TE-PATH-BINDING TLV" (whose format is shown in the figure below) is defined to carry binding label or SID for a TE path. This TLV is associated with the LSP object specified in ([RFC8231]). The type of this TLV is to be allocated by IANA.

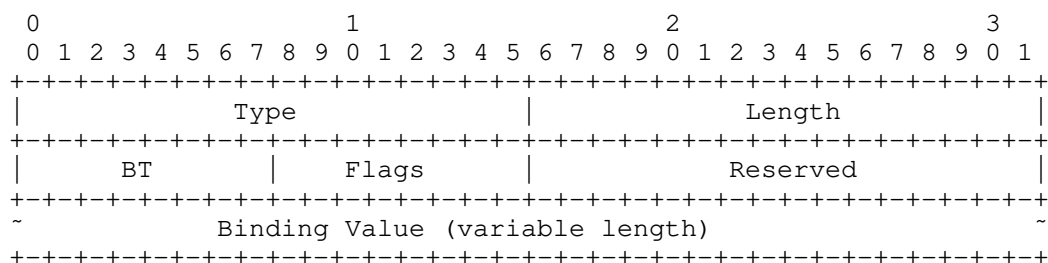


Figure 2: TE-PATH-BINDING TLV

TE-PATH-BINDING TLV is a generic TLV such that it is able to carry MPLS label binding as well as SRv6 Binding SID. It is formatted according to the rules specified in [RFC5440].

Binding Type (BT): A one octet field identifies the type of binding included in the TLV. This document specifies the following BT values:

- o BT = 0: The binding value is an MPLS label carried in the format specified in [RFC5462] where only the label value is valid, and other fields MUST be considered invalid. The Length MUST be set to 7.
- o BT = 1: Similar to the case where BT is 0 except that all the fields on the MPLS label entry are set on transmission. However, the receiver MAY choose to override TC, S, and TTL values according its local policy. The Length MUST be set to 8.
- o BT = 2: The binding value is an SRv6 SID with a format of a 16 octet IPv6 address, representing the binding SID for SRv6. The Length MUST be set to 20.
- o BT = 3: The binding value is a 24 octet field, defined in Section 3.1, that contains the SRv6 SID as well as its Behavior and Structure. The Length MUST be set to 28.

Flags: 1 octet of flags. Following flags are defined in the new registry "SR Policy Binding SID Flags" as described in Section 10.1.2:

```

  0 1 2 3 4 5 6 7
+---+---+---+---+
|           |I|S|
+---+---+---+---+

```

where:

- o S-Flag: This flag encodes the "Specified-BSID-only" behavior. It is used as described in Section 6.2.3 of [I-D.ietf-spring-segment-routing-policy].
- o I-Flag: This flag encodes the "Drop Upon Invalid" behavior. It is used as described in Section 8.2 of [I-D.ietf-spring-segment-routing-policy].

Reserved: MUST be set to 0 while sending and ignored on receipt.

Binding Value: A variable length field, padded with trailing zeros to a 4-octet boundary. For the BT as 0, the 20 bits represent the MPLS label. For the BT as 1, the 32-bits represent the label stack entry as per [RFC5462]. For the BT as 2, the 128-bits represent the SRv6 SID. For the BT as 3, the Binding Value contains SRv6 Endpoint Behavior and SID Structure, defined in Section 3.1.

3.1. SRv6 Endpoint Behavior and SID Structure

Carried as the Binding Value in the TE-PATH-BINDING TLV when the BT is set to 3. Applicable for SRv6 Binding SIDs [I-D.ietf-spring-srv6-network-programming].

```

      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
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               SRv6 Binding SID (16 octets)          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           Reserved           |           Endpoint Behavior          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   LB Length   |   LN Length   | Fun. Length   |   Arg. Length   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 4: SRv6 Endpoint Behavior and SID Structure

Reserved: 2 octets. MUST be set to 0 on transmit and ignored on receipt.

Endpoint Behavior: 2 octets. The Endpoint Behavior code point for this SRv6 SID as defined in section 9.2 of [I-D.ietf-spring-srv6-network-programming]. When set with the value 0, the choice of behavior is considered unset.

LB Length: 1 octet. SRv6 SID Locator Block length in bits.

LN Length: 1 octet. SRv6 SID Locator Node length in bits.

Function Length: 1 octet. SRv6 SID Function length in bits.

Argument Length: 1 octet. SRv6 SID Arguments length in bits.

4. Operation

The binding value is allocated by the PCC and reported to a PCE via PCRpt message. If a PCE does not recognize the TE-PATH-BINDING TLV, it would ignore the TLV in accordance with ([RFC5440]). If a PCE recognizes the TLV but does not support the TLV, it MUST send PCErr with Error-Type = 2 (Capability not supported).

If a TE-PATH-BINDING TLV is absent in PCRpt message, PCE MUST assume that the corresponding LSP does not have any binding. If a PCE recognizes an invalid binding value (e.g., label value from the reserved label space when MPLS label binding is used), it MUST send the PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error Value = 2 ("Bad label value") as specified in [RFC8664].

Multiple TE-PATH-BINDING TLVs are allowed to be present in the same LSP object. This signifies the presence of multiple binding SIDs for the given LSP.

For SRv6 BSIDs, it is RECOMMENDED to always explicitly specify the SRv6 Endpoint Behavior and SID Structure in the TE-PATH-BINDING TLV by setting the BT (Binding Type) to 3, instead of 2. The choice of interpreting SRv6 Endpoint Behavior and SID Structure when none is explicitly specified is left up to the implementation.

If a PCE requires a PCC to allocate a specific binding value, it may do so by sending a PCUpd or PCInitiate message containing a TE-PATH-BINDING TLV. If the value can be successfully allocated, the PCC reports the binding value to the PCE. If the PCC considers the binding value specified by the PCE invalid, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and

Error Value = TBD3 ("Invalid SID"). If the binding value is valid, but the PCC is unable to allocate the binding value, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error Value = TBD4 ("Unable to allocate the specified label/SID").

If a PCC receives TE-PATH-BINDING TLV in any message other than PCUpd or PCInitiate, it MUST close the corresponding PCEP session with the reason "Reception of a malformed PCEP message" (according to [RFC5440]). Similarly, if a PCE receives a TE-PATH-BINDING TLV in any message other than a PCRpt or if the TE-PATH-BINDING TLV is associated with any object other than LSP object, the PCE MUST close the corresponding PCEP session with the reason "Reception of a malformed PCEP message" (according to [RFC5440]).

If a PCC wishes to withdraw or modify a previously reported binding value, it MUST send a PCRpt message without any TE-PATH-BINDING TLV or with the TE-PATH-BINDING TLV containing the new binding value respectively.

If a PCE wishes to modify a previously requested binding value, it MUST send a PCUpd message with TE-PATH-BINDING TLV containing the new binding value. Absence of TE-PATH-BINDING TLV in PCUpd message means that the PCE does not specify a binding value in which case the binding value allocation is governed by the PCC's local policy.

If a PCC receives a valid binding value from a PCE which is different than the current binding value, it MUST try to allocate the new value. If the new binding value is successfully allocated, the PCC MUST report the new value to the PCE. Otherwise, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error Value = TBD4 ("Unable to allocate the specified label/SID").

In some cases, a stateful PCE can request the PCC to allocate a binding value. It may do so by sending a PCUpd message containing an empty TE-PATH-BINDING TLV, i.e., no binding value is specified (making the length field of the TLV as 4). A PCE can also request PCC to allocate a binding value at the time of initiation by sending a PCInitiate message with an empty TE-PATH-BINDING TLV. If the PCC is unable to allocate a binding value, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error-Value = TBD5 ("Unable to allocate label/SID").

5. Binding SID in SR-ERO

In PCEP messages, LSP route information is carried in the Explicit Route Object (ERO), which consists of a sequence of subobjects. [RFC8664] defines a new ERO subobject "SR-ERO subobject" capable of carrying a SID as well as the identity of the node/adjacency (NAI) represented by the SID. The NAI Type (NT) field indicates the type and format of the NAI contained in the SR-ERO. In case of binding SID, the NAI MUST NOT be included and NT MUST be set to zero. So as per Section 5.2.1 of [RFC8664], for NT=0, the F bit is set to 1, the S bit needs to be zero and the Length is 8. Further the M bit is set. If these conditions are not met, the entire ERO MUST be considered invalid and a PCErr message is sent with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 11 ("Malformed object").

6. Binding SID in SRv6-ERO

[RFC8664] defines a new ERO subobject "SRv6-ERO subobject" for SRv6 SID. The NAI MUST NOT be included and NT MUST be set to zero. So as per Section 5.2.1 of [RFC8664], for NT=0, the F bit is set to 1, the S bit needs to be zero and the Length is 24. If these conditions are not met, the entire ERO is considered invalid and a PCErr message is sent with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 11 ("Malformed object") (as per [RFC8664]).

7. Implementation Status

[Note to the RFC Editor - remove this section before publication, as well as remove the reference to RFC 7942.]

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".

7.1. Huawei

- o Organization: Huawei
- o Implementation: Huawei's Router and Controller
- o Description: An experimental code-point is used and plan to request early code-point allocation from IANA after WG adoption.
- o Maturity Level: Production
- o Coverage: Full
- o Contact: chenglil3@huawei.com

7.2. Cisco

- o Organization: Cisco Systems
- o Implementation: Head-end and controller.
- o Description: An experimental code-point is currently used.
- o Maturity Level: Production
- o Coverage: Full
- o Contact: mkoldych@cisco.com

8. Security Considerations

The security considerations described in [RFC5440], [RFC8231], [RFC8281] and [RFC8664] are applicable to this specification. No additional security measure is required.

As described [RFC8664], SR allows a network controller to instantiate and control paths in the network. A rouge PCE can manipulate binding SID allocations to move traffic around for some other LSPs that uses BSID in its SR-ERO.

Thus, as per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253], as per the recommendations

and best current practices in BCP195 [RFC7525] (unless explicitly set aside in [RFC8253]).

9. Manageability Considerations

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

9.1. Control of Function and Policy

A PCC implementation SHOULD allow the operator to configure the policy based on which PCC needs to allocate the binding label/SID.

9.2. Information and Data Models

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to include policy configuration for binding label/SID allocation.

9.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

9.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8664].

9.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.

9.6. Impact On Network Operations

Mechanisms defined in [RFC5440], [RFC8231], and [RFC8664] also apply to PCEP extensions defined in this document. Further, the mechanism described in this document can help the operator to request control of the LSPs at a particular PCE.

10. IANA Considerations

10.1. PCEP TLV Type Indicators

This document defines a new PCEP TLV; IANA is requested to make the following allocations from the "PCEP TLV Type Indicators" sub-registry of the PCEP Numbers registry, as follows:

Value	Name	Reference
TBD1	TE-PATH-BINDING	This document

10.1.1. TE-PATH-BINDING TLV

IANA is requested to create a sub-registry to manage the value of the Binding Type field in the TE-PATH-BINDING TLV.

Value	Description	Reference
0	MPLS Label	This document
1	MPLS Label Stack Entry	This document
2	SRv6 SID	This document
3	SRv6 SID with Behavior and Structure	This document

10.1.2. Binding SID Flags

IANA is requested to create a sub-registry to manage the value of the Binding SID Flags field in the TE-PATH-BINDING-TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (count from 0 as the most significant bit)
- o Flag Name
- o Reference

Bit	Description	Reference
7	Specified-BSID-Only Flag (S-Flag)	This document
6	Drop Upon Invalid Flag (I-Flag)	This document

10.2. PCEP Error Type and Value

This document defines a new Error-type and Error-Values for the PCErr message. IANA is requested to allocate new error-type and error-values within the "PCEP-ERROR Object Error Types and Values" subregistry of the PCEP Numbers registry, as follows:

Error-Type	Meaning
-----	-----
TBD2	Binding label/SID failure:
	Error-value = TBD3: Invalid SID
	Error-value = TBD4: Unable to allocate the specified label/SID
	Error-value = TBD5: Unable to allocate label/SID

11. Acknowledgements

We like to thank Milos Fabian and Mrinmoy Das for thier valuable comments.

12. References

12.1. Normative References

- [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>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<https://www.rfc-editor.org/info/rfc5462>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.

- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [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>>.
- [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>>.
- [I-D.ietf-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-ietf-spring-srv6-network-programming-24 (work in progress), October 2020.

12.2. Informative References

- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.
- [RFC8669] Previdi, S., Filssils, C., Lindem, A., Ed., Sreekantiah, A., and H. Gredler, "Segment Routing Prefix Segment Identifier Extensions for BGP", RFC 8669, DOI 10.17487/RFC8669, December 2019, <<https://www.rfc-editor.org/info/rfc8669>>.
- [I-D.ietf-spring-segment-routing-policy] Filssils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-08 (work in progress), July 2020.
- [I-D.ietf-pce-pcep-extension-for-pce-controller] Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-07 (work in progress), September 2020.
- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-14 (work in progress), July 2020.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Mahendra Singh Negi
RtBrick India
N-17L, Floor-1, 18th Cross Rd, HSR Layout Sector-3
Bangalore, Karnataka 560102
India

EMail: mahend.ietf@gmail.com

Mike Koldychev
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada

Email: mkoldych@cisco.com

Zafar Ali
Cisco Systems, Inc.

Email: zali@cisco.com

Authors' Addresses

Siva Sivabalan
Ciena Corporation

EMail: msiva282@gmail.com

Clarence Filsfils
Cisco Systems, Inc.
Pegasus Parc
De kleetlaan 6a, DIEGEM BRABANT 1831
BELGIUM

EMail: cfilsfil@cisco.com

Jeff Tantsura
Apstra, Inc.

Email: jefftant.ietf@gmail.com

Jonathan Hardwick
Metaswitch Networks
100 Church Street
Enfield, Middlesex
UK

Email: Jonathan.Hardwick@metaswitch.com

Stefano Previdi
Huawei Technologies

Email: stefano@previdi.net

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: chengli13@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 September 2022

S. Sivabalan
Ciena Corporation
C. Filsfils
Cisco Systems, Inc.
J. Tantsura
Microsoft Corporation
S. Previdi
C. Li, Ed.
Huawei Technologies
20 March 2022

Carrying Binding Label/Segment Identifier (SID) in PCE-based Networks.
draft-ietf-pce-binding-label-sid-15

Abstract

In order to provide greater scalability, network confidentiality, and service independence, Segment Routing (SR) utilizes a Binding Segment Identifier (SID) (called BSID) as described in RFC 8402. It is possible to associate a BSID to an RSVP-TE-signaled Traffic Engineering Label Switched Path or an SR Traffic Engineering path. The BSID can be used by an upstream node for steering traffic into the appropriate TE path to enforce SR policies. This document specifies the concept of binding value, which can be either an MPLS label or Segment Identifier. It further specifies an extension to Path Computation Element (PCE) communication Protocol (PCEP) for reporting the binding value by a Path Computation Client (PCC) to the PCE to support PCE-based Traffic Engineering policies.

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 21 September 2022.

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	3
1.1. Motivation and Example	4
1.2. Summary of the Extension	5
2. Requirements Language	5
3. Terminology	5
4. Path Binding TLV	6
4.1. SRv6 Endpoint Behavior and SID Structure	8
5. Operation	10
6. Binding SID in SR-ERO	12
7. Binding SID in SRv6-ERO	12
8. PCE Allocation of Binding label/SID	12
9. Implementation Status	14
9.1. Huawei	15
9.2. Cisco	15
10. Security Considerations	16
11. Manageability Considerations	16
11.1. Control of Function and Policy	17
11.2. Information and Data Models	17
11.3. Liveness Detection and Monitoring	17
11.4. Verify Correct Operations	17
11.5. Requirements On Other Protocols	17
11.6. Impact On Network Operations	17
12. IANA Considerations	17
12.1. PCEP TLV Type Indicators	17
12.1.1. TE-PATH-BINDING TLV	18
12.2. LSP Object	19
12.3. PCEP Error Type and Value	19
13. Acknowledgements	20
14. References	20
14.1. Normative References	20
14.2. Informative References	22
Appendix A. Contributor Addresses	24

Authors' Addresses	24
--------------------	----

1. Introduction

A Path Computation Element (PCE) can compute Traffic Engineering paths (TE paths) through a network where those paths are subject to various constraints. Currently, TE paths are set up using either the RSVP-TE signaling protocol or Segment Routing (SR). We refer to such paths as RSVP-TE paths and SR-TE paths respectively in this document.

As per [RFC8402] SR allows a head-end node to steer a packet flow along a given path via a Segment Routing Policy (SR Policy). As per [I-D.ietf-spring-segment-routing-policy], an SR Policy is a framework that enables the instantiation of an ordered list of segments on a node for implementing a source routing policy with a specific intent for traffic steering from that node.

As described in [RFC8402], a Binding Segment Identifier (BSID) is bound to a Segment Routing (SR) Policy, instantiation of which may involve a list of Segment Identifiers (SIDs). Any packets received with an active segment equal to a BSID are steered onto the bound SR Policy. A BSID may be either a local (SR Local Block (SRLB)) or a global (SR Global Block (SRGB)) SID. As per Section 6.4 of [I-D.ietf-spring-segment-routing-policy] a BSID can also be associated with any type of interface or tunnel to enable the use of a non-SR interface or tunnel as a segment in a SID list. In this document, the term 'binding label/SID' is used to generalize the allocation of binding value for both SR and non-SR paths.

[RFC5440] describes the PCEP for communication between a Path Computation Client (PCC) and a PCE or between a pair of PCEs as per [RFC4655]. [RFC8231] specifies extensions to PCEP that allow a PCC to delegate its Label Switched Paths (LSPs) to a stateful PCE. A stateful PCE can then update the state of LSPs delegated to it. [RFC8281] specifies a mechanism allowing a PCE to dynamically instantiate an LSP on a PCC by sending the path and characteristics. This document specifies an extension to PCEP to manage the binding of label/SID that can be applied to SR, RSVP-TE, and other path setup types.

[RFC8664] provides a mechanism for a PCE (acting as a network controller) to instantiate SR-TE paths (candidate paths) for an SR Policy onto a head-end node (acting as a PCC) using PCEP. For more information on the SR Policy Architecture, see [I-D.ietf-spring-segment-routing-policy].

1.1. Motivation and Example

A binding label/SID has local significance to the ingress node of the corresponding TE path. When a stateful PCE is deployed for setting up TE paths, a binding label/SID reported from the PCC to the stateful PCE is useful for the purpose of enforcing end-to-end TE/SR policy. A sample Data Center (DC) and IP/MPLS WAN use-case is illustrated in Figure 1 with a multi-domain PCE. In the IP/MPLS WAN, an SR-TE LSP is set up using the PCE. The list of SIDs of the SR-TE LSP is {A, B, C, D}. The gateway node 1 (which is the PCC) allocates a binding SID X and reports it to the PCE. In the MPLS DC network, an end-to-end SR-TE LSP is established. In order for the access node to steer the traffic towards Node-1 and over the SR-TE path in WAN, the PCE passes the SID stack {Y, X} where Y is the node SID of the gateway node-1 to the access node and X is the BSID. In the absence of the BSID X, the PCE would need to pass the SID stack {Y, A, B, C, D} to the access node. This example also illustrates the additional benefit of using the binding label/SID to reduce the number of SIDs imposed by the access nodes with a limited forwarding capacity.

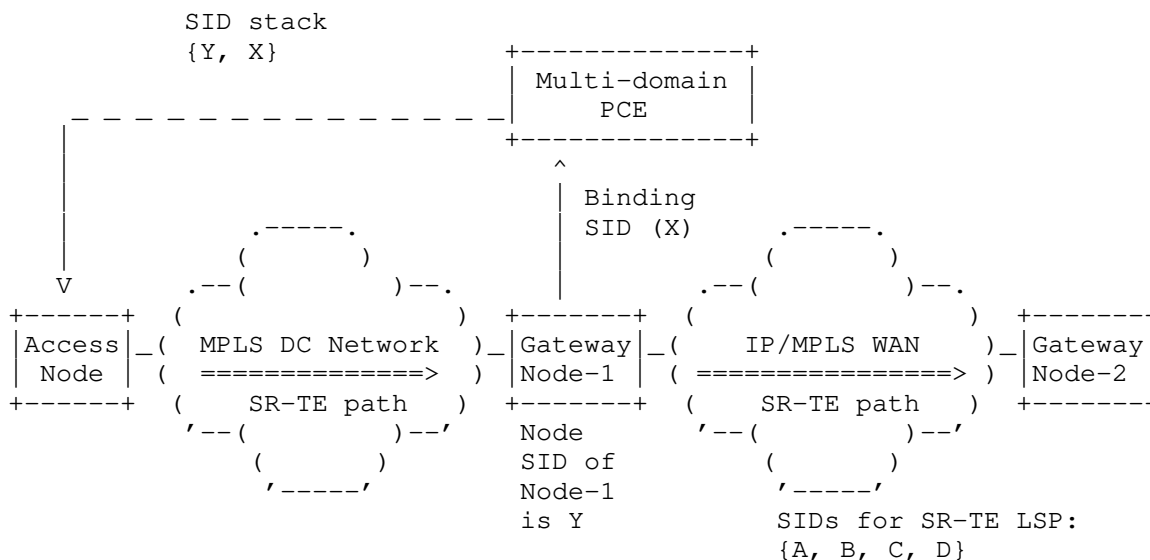


Figure 1: A Sample Use-case of Binding SID

Using the extension defined in this document, a PCC could report to the stateful PCE the binding label/SID it allocated via a Path Computation LSP State Report (PCRpt) message. It is also possible for a stateful PCE to request a PCC to allocate a specific binding label/SID by sending a Path Computation LSP Update Request (PCUpd) message. If the PCC can successfully allocate the specified binding

value, it reports the binding value to the PCE. Otherwise, the PCC sends an error message to the PCE indicating the cause of the failure. A local policy or configuration at the PCC SHOULD dictate if the binding label/SID needs to be assigned.

1.2. Summary of the Extension

To implement the needed changes to PCEP, in this document, we introduce a new OPTIONAL TLV that a PCC can use in order to report the binding label/SID associated with a TE LSP, or a PCE to request a PCC to allocate any or a specific binding label/SID value. This TLV is intended for TE LSPs established using RSVP-TE, SR-TE, or any other future method. In the case of SR-TE LSPs, the TLV can carry a binding label (for SR-TE path with MPLS data-plane) or a binding IPv6 SID (e.g., IPv6 address for SR-TE paths with IPv6 data-plane). Throughout this document, the term "binding value" means either an MPLS label or a SID.

As another way to use the extension specified in this document, to support the PCE-based central controller [RFC8283] operation where the PCE would take responsibility for managing some part of the MPLS label space for each of the routers that it controls, the PCE could directly make the binding label/SID allocation and inform the PCC. See Section 8 for details.

In addition to specifying a new TLV, this document specifies how and when a PCC and PCE can use this TLV, how they can allocate a binding label/SID, and associated error handling.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

The following terminologies are used in this document:

BSID: Binding Segment Identifier.

binding label/SID: a generic term used for the binding segment for both SR and non-SR paths.

binding value: a generic term used for the binding segment as it can be encoded in various formats (as per the binding type(BT)).

LSP: Label Switched Path.

PCC: Path Computation Client.

PCEP: Path Computation Element communication Protocol.

RSVP-TE: Resource ReserVation Protocol-Traffic Engineering.

SID: Segment Identifier.

SR: Segment Routing.

4. Path Binding TLV

The new optional TLV called "TE-PATH-BINDING TLV" (whose format is shown in Figure 2) is defined to carry the binding label/SID for a TE path. This TLV is associated with the LSP object specified in [RFC8231]. This TLV can also be carried in the PCEP-ERROR object [RFC5440] in case of error. Multiple instances of TE-PATH-BINDING TLVs MAY be present in the LSP and PCEP-ERROR object. The type of this TLV is 55 (early allocated by IANA). The length is variable.

[Note to RFC Editor: Please remove "(early allocated by IANA)" before publication]

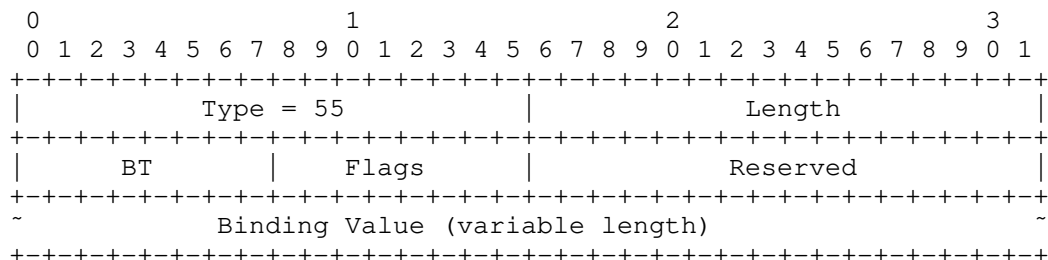


Figure 2: TE-PATH-BINDING TLV

TE-PATH-BINDING TLV is a generic TLV such that it is able to carry binding label/SID (i.e. MPLS label or SRv6 SID). It is formatted according to the rules specified in [RFC5440]. The value portion of the TLV comprises:

Binding Type (BT): A one-octet field that identifies the type of binding included in the TLV. This document specifies the following BT values:

- * BT = 0: The binding value is a 20-bit MPLS label value. The TLV is padded to 4-bytes alignment. The Length MUST be set to 7 (the padding is not included in the length, as per [RFC5440] Section 7.1) and the first 20 bits are used to encode the MPLS label value.
- * BT = 1: The binding value is a 32-bit MPLS label stack entry as per [RFC3032] with Label, TC [RFC5462], S, and TTL values encoded. Note that the receiver MAY choose to override TC, S, and TTL values according to its local policy. The Length MUST be set to 8.
- * BT = 2: The binding value is an SRv6 SID with the format of a 16-octet IPv6 address, representing the binding SID for SRv6. The Length MUST be set to 20.
- * BT = 3: The binding value is a 24 octet field, defined in Section 4.1, that contains the SRv6 SID as well as its Behavior and Structure. The Length MUST be set to 28.

Section 12.1.1 defines the IANA registry used to maintain all these binding types as well as any future ones. Note that multiple TE-PATH-BINDING TLVs with same or different Binding Types MAY be present for the same LSP. A PCEP speaker could allocate multiple TE-PATH-BINDING TLVs (of the same BT), and use different binding values in different domains or use-cases based on a local policy.

Flags: 1 octet of flags. The following flag is defined in the new registry "TE-PATH-BINDING TLV Flag field" as described in Section 12.1.1:

```

  0 1 2 3 4 5 6 7
+---+---+---+---+---+---+
|R|                                     |
+---+---+---+---+---+---+

```

Figure 3: Flags

where:

- * R (Removal - 1 bit): When set, the requesting PCEP peer requires the removal of the binding value for the LSP. When unset, the PCEP peer indicates that the binding value is added or retained for the LSP. This flag is used in the PCRpt and PCUpd messages. It is ignored in other PCEP messages.

- * The unassigned flags MUST be set to 0 while sending and ignored on receipt.

Reserved: MUST be set to 0 while sending and ignored on receipt.

Binding Value: A variable-length field, padded with trailing zeros to a 4-octet boundary. When the BT is 0, the 20 bits represent the MPLS label. When the BT is 1, the 32 bits represent the MPLS label stack entry as per [RFC3032]. When the BT is 2, the 128 bits represent the SRv6 SID. When the BT is 3, the Binding Value also contains the SRv6 Endpoint Behavior and SID Structure, defined in Section 4.1. In this document, the TE-PATH-BINDING TLV is considered to be empty if no Binding Value is present. Note that the length of the TLV would be 4 in such a case.

4.1. SRv6 Endpoint Behavior and SID Structure

This section specifies the format of the Binding Value in the TE-PATH-BINDING TLV when the BT is set to 3 for the SRv6 Binding SIDs [RFC8986]. The format is shown in Figure 4.

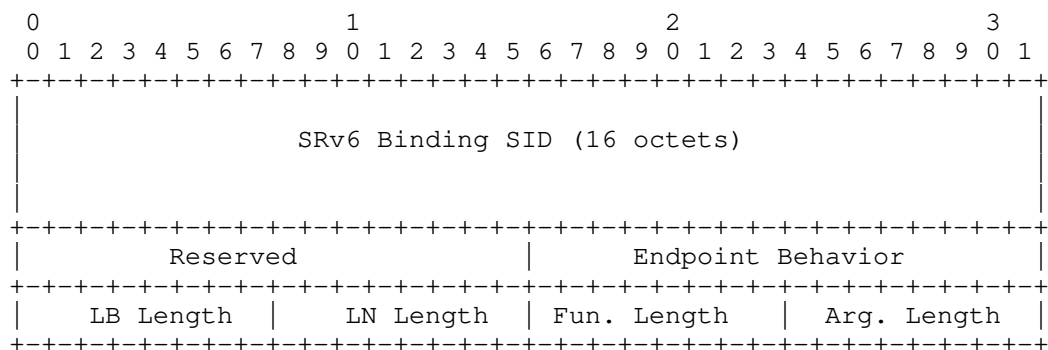


Figure 4: SRv6 Endpoint Behavior and SID Structure

The Binding Value consists of:

- * SRv6 Binding SID: 16 octets. The 128-bit IPv6 address, representing the binding SID for SRv6.
- * Reserved: 2 octets. It MUST be set to 0 on transmit and ignored on receipt.

- * Endpoint Behavior: 2 octets. The Endpoint Behavior code point for this SRv6 SID as per the IANA subregistry called "SRv6 Endpoint Behaviors", created by [RFC8986]. When the field is set with the value 0, the endpoint behavior is considered unknown.
- * [RFC8986] defines an SRv6 SID as consisting of LOC:FUNCT:ARG, where a locator (LOC) is encoded in the L most significant bits of the SID, followed by F bits of function (FUNCT) and A bits of arguments (ARG). A locator may be represented as B:N where B is the SRv6 SID locator block (IPv6 prefix allocated for SRv6 SIDs by the operator) and N is the identifier of the parent node instantiating the SID called locator node. The following fields are used to advertise the length of each individual part of the SRv6 SID as defined in :
 - LB Length: 1 octet. SRv6 SID Locator Block length in bits.
 - LN Length: 1 octet. SRv6 SID Locator Node length in bits.
 - Function Length: 1 octet. SRv6 SID Function length in bits.
 - Argument Length: 1 octet. SRv6 SID Arguments length in bits.

The total of the locator block, locator node, function, and argument lengths MUST be lower or equal to 128 bits. If this condition is not met, the corresponding TE-PATH-BINDING TLV is considered invalid. Also, if the Endpoint Behavior is found to be unknown or inconsistent, it is considered invalid. A PCERR message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 37 ("Invalid SRv6 SID Structure") MUST be sent in such cases.

The SRv6 SID Structure could be used by the PCE for ease of operations and monitoring. For example, this information could be used for validation of SRv6 SIDs being instantiated in the network and checked for conformance to the SRv6 SID allocation scheme chosen by the operator as described in Section 3.2 of [RFC8986]. In the future, PCE could also be used for verification and the automation for securing the SRv6 domain by provisioning filtering rules at SR domain boundaries as described in Section 5 of [RFC8754]. The details of these potential applications are outside the scope of this document.

5. Operation

The binding value is usually allocated by the PCC and reported to a PCE via a PCRpt message (see Section 8 where PCE does the allocation). If a PCE does not recognize the TE-PATH-BINDING TLV, it would ignore the TLV in accordance with [RFC5440]. If a PCE recognizes the TLV but does not support the TLV, it MUST send a PCErr with Error-Type = 2 (Capability not supported).

Multiple TE-PATH-BINDING TLVs are allowed to be present in the same LSP object. This signifies the presence of multiple binding SIDs for the given LSP. In the case of multiple TE-PATH-BINDING TLVs, the existing instances of TE-PATH-BINDING TLVs MAY be included in the LSP object. In case of an error condition, the whole message is rejected and the resulting PCErr message MAY include the offending TE-PATH-BINDING TLV in the PCEP-ERROR object.

If a PCE recognizes an invalid binding value (e.g., label value from the reserved MPLS label space), it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error Value = 2 ("Bad label value") as specified in [RFC8664].

For SRv6 BSIDs, it is RECOMMENDED to always explicitly specify the SRv6 Endpoint Behavior and SID Structure in the TE-PATH-BINDING TLV by setting the BT (Binding Type) to 3. This can enable the sender to have control of the SRv6 Endpoint Behavior and SID Structure. A sender MAY choose to set the BT to 2, in which case the receiving implementation chooses how to interpret the SRv6 Endpoint Behavior and SID Structure according to local policy.

If a PCC wishes to withdraw a previously reported binding value, it MUST send a PCRpt message with the specific TE-PATH-BINDING TLV with R flag set to 1. If a PCC wishes to modify a previously reported binding, it MUST withdraw the former binding value (with R flag set in the former TE-PATH-BINDING TLV) and include a new TE-PATH-BINDING TLV containing the new binding value. Note that other instances of TE-PATH-BINDING TLVs that are unchanged MAY also be included. If the unchanged instances are not included, they will remain associated with the LSP.

If a PCE requires a PCC to allocate a (or several) specific binding value(s), it may do so by sending a PCUpd or PCInitiate message containing a TE-PATH-BINDING TLV(s). If the value(s) can be successfully allocated, the PCC reports the binding value(s) to the PCE. If the PCC considers the binding value specified by the PCE invalid, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error Value = TBD3 ("Invalid SID"). If the binding value is valid, but the PCC is unable to allocate the

binding value, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error Value = TBD4 ("Unable to allocate the specified binding value"). Note that, in case of an error, the PCC rejects the PCUpd or PCInitiate message in its entirety and can include the offending TE-PATH-BINDING TLV in the PCEP-ERROR object.

If a PCE wishes to request the withdrawal of a previously reported binding value, it MUST send a PCUpd message with the specific TE-PATH-BINDING TLV with R flag set to 1. If a PCE wishes to modify a previously requested binding value, it MUST request the withdrawal of the former binding value (with R flag set in the former TE-PATH-BINDING TLV) and include a new TE-PATH-BINDING TLV containing the new binding value. If a PCC receives a PCUpd message with TE-PATH-BINDING TLV where the R flag is set to 1, but either the binding value is missing (empty TE-PATH-BINDING TLV) or the binding value is incorrect, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error Value = TBD6 ("Unable to remove the binding value").

In some cases, a stateful PCE may want to request that the PCC allocate a binding value of the PCC's own choosing. It instructs the PCC by sending a PCUpd message containing an empty TE-PATH-BINDING TLV, i.e., no binding value is specified (bringing the Length field of the TLV to 4). A PCE can also request a PCC to allocate a binding value at the time of initiation by sending a PCInitiate message with an empty TE-PATH-BINDING TLV. Only one such instance of empty TE-PATH-BINDING TLV, per BT, SHOULD be included in the LSP object and others ignored on receipt. If the PCC is unable to allocate a new binding value as per the specified BT, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error-Value = TBD5 ("Unable to allocate a new binding label/SID").

As previously noted, if a message contains an invalid TE-PATH-BINDING TLV that leads to an error condition, the whole message is rejected including any other valid instances of TE-PATH-BINDING TLVs, if any. The resulting error message MAY include the offending TE-PATH-BINDING TLV in the PCEP-ERROR object.

If a PCC receives a TE-PATH-BINDING TLV in any message other than PCUpd or PCInitiate, it MUST close the corresponding PCEP session with the reason "Reception of a malformed PCEP message" (according to [RFC5440]). Similarly, if a PCE receives a TE-PATH-BINDING TLV in any message other than a PCRpt or if the TE-PATH-BINDING TLV is associated with any object other than an LSP or PCEP-ERROR object, the PCE MUST close the corresponding PCEP session with the reason "Reception of a malformed PCEP message" (according to [RFC5440]).

If a TE-PATH-BINDING TLV is absent in the PCRpt message and no binding values were reported before, the PCE MUST assume that the corresponding LSP does not have any binding. Similarly, if TE-PATH-BINDING TLV is absent in the PCUpd message and no binding values were reported before, the PCC's local policy dictates how the binding allocations are made for a given LSP.

Note that some binding types have similar information but different binding value formats. For example, BT=(2 or 3) is used for the SRv6 SID and BT=(0 or 1) is used for the MPLS Label. In case a PCEP speaker receives multiple TE-PATH-BINDING TLVs with the same SRv6 SID or MPLS Label but different BT values, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error-Value = TBD7 ("Inconsistent binding types").

6. Binding SID in SR-ERO

In PCEP messages, LSP route information is carried in the Explicit Route Object (ERO), which consists of a sequence of subobjects. [RFC8664] defines the "SR-ERO subobject" capable of carrying a SID as well as the identity of the node/adjacency (NAI) represented by the SID. The NAI Type (NT) field indicates the type and format of the NAI contained in the SR-ERO. In case of binding SID, the NAI MUST NOT be included and NT MUST be set to zero. [RFC8664] Section 5.2.1 specifies bit settings and error handling in the case when NT=0.

7. Binding SID in SRv6-ERO

[I-D.ietf-pce-segment-routing-ipv6] defines the "SRv6-ERO subobject" for an SRv6 SID. Similarly to SR-ERO (Section 6), the NAI MUST NOT be included and the NT MUST be set to zero. [RFC8664] Section 5.2.1 specifies bit settings and error handling in the case when NT=0.

8. PCE Allocation of Binding label/SID

Section 5 already includes the scenario where a PCE requires a PCC to allocate a specified binding value by sending a PCUpd or PCInitiate message containing a TE-PATH-BINDING TLV. This section specifies an OPTIONAL feature for the PCE to allocate the binding label/SID of its own accord in the case where the PCE also controls the label space of the PCC and can make the label allocation on its own as described in [RFC8283]. Note that the act of requesting a specific binding value (Section 5) is different from the act of allocating a binding label/SID as described in this section.

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE

and PCC. [RFC9050] specifies the procedures and PCEP extensions for using the PCE as the central controller. It assumes that the exclusive label range to be used by a PCE is known and set on both PCEP peers. A future extension could add the capability to advertise this range via a possible PCEP extension as well (see [I-D.li-pce-controlled-id-space]).

When PCECC operations are supported as per [RFC9050], the binding label/SID MAY also be allocated by the PCE itself. Both peers need to exchange the PCECC capability as described in [RFC9050] before the PCE can allocate the binding label/SID on its own.

A new P flag in the LSP object [RFC8231] is introduced to indicate that the allocation needs to be made by the PCE. Note that the P flag could be used for other types of allocations (such as path segments [I-D.ietf-pce-sr-path-segment]) in future.

- * P (PCE-allocation): If the bit is set to 1, it indicates that the PCC requests PCE to make allocations for this LSP. The TE-PATH-BINDING TLV in the LSP object identifies that the allocation is for a binding label/SID. A PCC MUST set this bit to 1 and include a TE-PATH-BINDING TLV in the LSP object if it wishes to request for allocation of binding label/SID by the PCE in the PCEP message. A PCE MUST also set this bit to 1 and include a TE-PATH-BINDING TLV to indicate that the binding label/SID is allocated by PCE and encoded in the PCEP message towards the PCC. Further, if the binding label/SID is allocated by the PCC, the PCE MUST set this bit to 0 and follow the procedure described in Section 5.

Note that -

- * A PCE could allocate the binding label/SID of its own accord for a PCE-initiated or delegated LSP, and inform the PCC in the PCInitiate message or PCUpd message by setting P=1 and including TE-PATH-BINDING TLV in the LSP object.
- * To let the PCC allocate the binding label/SID, a PCE MUST set P=0 and include an empty TE-PATH-BINDING TLV (i.e., no binding value is specified) in the LSP object in PCInitiate/PCUpd message.
- * To request that the PCE allocate the binding label/SID, a PCC MUST set P=1, D=1, and include an empty TE-PATH-BINDING TLV in PCRpt message. The PCE will attempt to allocate it and respond to the PCC with PCUpd message including the allocated binding label/SID in the TE-PATH-BINDING TLV and P=1, D=1 in the LSP object. If the PCE is unable to allocate, it MUST send a PCErr message with Error-Type = TBD2 ("Binding label/SID failure") and Error-Value = TBD5 ("Unable to allocate a new binding label/SID").

- * If one or both speakers (PCE and PCC) have not indicated support and willingness to use the PCEP extensions for the PCECC as per [RFC9050] and a PCEP peer receives P=1 in the LSP object, it MUST:
 - send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=16 (Attempted PCECC operations when PCECC capability was not advertised) and
 - terminate the PCEP session.
- * A legacy PCEP speaker that does not recognize the P flag in the LSP object would ignore it in accordance with [RFC8231].

It is assumed that the label range to be used by a PCE is known and set on both PCEP peers. The exact mechanism is out of the scope of [RFC9050] or this document. Note that the specific BSID could be from the PCE-controlled or the PCC-controlled label space. The PCE can directly allocate the label from the PCE-controlled label space using P=1 as described above, whereas the PCE can request the allocation of a specific BSID from the PCC-controlled label space with P=0 as described in Section 5.

Note that, the P-Flag in the LSP object SHOULD NOT be set to 1 without the presence of TE-PATH-BINDING TLV or any other future TLV defined for PCE allocation. On receipt of such an LSP object, the P-Flag is ignored. The presence of TE-PATH-BINDING TLV with P=1 indicates the allocation is for the binding label/SID. In the future, some other TLV (such as one defined in [I-D.ietf-pce-sr-path-segment]) could also be used alongside P=1 to indicate allocation of a different attribute. A future document should not attempt to assign semantics to P=1 without limiting its scope that both PCEP peers could agree on.

9. Implementation Status

[Note to the RFC Editor - remove this section before publication, as well as remove the reference to RFC 7942.]

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".

9.1. Huawei

- * Organization: Huawei
- * Implementation: Huawei's Router and Controller
- * Description: An experimental code-point is used and will be modified to the value allocated in this document.
- * Maturity Level: Production
- * Coverage: Full
- * Contact: c.l@huawei.com

9.2. Cisco

- * Organization: Cisco Systems
- * Implementation: Head-end and controller.
- * Description: An experimental code-point is used and will be modified to the value allocated in this document.
- * Maturity Level: Production
- * Coverage: Full

* Contact: mkoldych@cisco.com

10. Security Considerations

The security considerations described in [RFC5440], [RFC8231], [RFC8281], [RFC8664], and [RFC9050] are applicable to this specification. No additional security measure is required.

As described in [RFC8402] and [RFC8664], SR intrinsically involves an entity (whether head-end or a central network controller) controlling and instantiating paths in the network without the involvement of (other) nodes along those paths. Binding SIDs are in effect shorthand aliases for longer path representations, and the alias expansion is in principle known only by the node that acts on it. In this document, the expansion of the alias is shared between PCC and PCE, and rogue actions by either PCC or PCE could result in shifting or misdirecting traffic in ways that are hard for other nodes to detect. In particular, when a PCE propagates paths of the form {A, B, BSID} to other entities, the BSID values are opaque, and a rogue PCE can substitute a BSID from a different LSP in such paths to move traffic without the recipient of the path knowing the ultimate destination.

The case of BT=3 provides additional opportunities for malfeasance, as it purports to convey information about internal SRv6 SID structure. There is no mechanism defined to validate this internal structure information, and mischaracterizing the division of bits into locator block, locator node, function, and argument can result in different interpretation of the bits by PCC and PCE. Most notably, shifting bits into or out of the "argument" is a direct vector for affecting processing, but other attacks are also possible.

Thus, as per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in BCP195 [RFC7525] (unless explicitly set aside in [RFC8253]).

11. Manageability Considerations

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

11.1. Control of Function and Policy

A PCC implementation SHOULD allow the operator to configure the policy the PCC needs to apply when allocating the binding label/SID.

If BT is set to 2, the operator needs to have local policy set to decide the SID structure and the SRv6 Endpoint Behavior of the BSID.

11.2. Information and Data Models

The PCEP YANG module [I-D.ietf-pce-pcep-yang] will be extended to include policy configuration for binding label/SID allocation.

11.3. Liveness Detection and Monitoring

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

11.4. Verify Correct Operations

The mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8664].

11.5. Requirements On Other Protocols

The mechanisms defined in this document do not imply any new requirements on other protocols.

11.6. Impact On Network Operations

The mechanisms defined in [RFC5440], [RFC8231], and [RFC8664] also apply to the PCEP extensions defined in this document.

12. IANA Considerations

IANA maintains the "Path Computation Element Protocol (PCEP) Numbers" registry. This document requests IANA actions to allocate code points for the protocol elements defined in this document.

12.1. PCEP TLV Type Indicators

This document defines a new PCEP TLV; IANA is requested to confirm the following early allocations from the "PCEP TLV Type Indicators" subregistry of the PCEP Numbers registry, as follows:

Value	Description	Reference
55	TE-PATH-BINDING	This document

Table 1

12.1.1. TE-PATH-BINDING TLV

IANA is requested to create a new subregistry "TE-PATH-BINDING TLV BT field" to manage the value of the Binding Type field in the TE-PATH-BINDING TLV. Initial values for the subregistry are given below. New values are assigned by Standards Action [RFC8126].

Value	Description	Reference
0	MPLS Label	This document
1	MPLS Label Stack Entry	This document
2	SRv6 SID	This document
3	SRv6 SID with Behavior and Structure	This document
4-255	Unassigned	This document

Table 2

IANA is requested to create a new subregistry "TE-PATH-BINDING TLV Flag field" to manage the Flag field in the TE-PATH-BINDING TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- * Bit number (count from 0 as the most significant bit)
- * Description
- * Reference

Bit	Description	Reference
0	R (Removal)	This document
1-7	Unassigned	This document

Table 3

12.2. LSP Object

IANA is requested to confirm the early allocation for a new code-point in the "LSP Object Flag Field" sub-registry for the new P flag as follows:

Bit	Description	Reference
0	PCE-allocation	This document

Table 4

12.3. PCEP Error Type and Value

This document defines a new Error-type and associated Error-Values for the PCErr message. IANA is requested to allocate new error-type and error-values within the "PCEP-ERROR Object Error Types and Values" subregistry of the PCEP Numbers registry, as follows:

Error-Type	Meaning	Error-value	Reference
TBD2	Binding label/ SID failure	0: Unassigned	This document
		TBD3: Invalid SID	This document
		TBD4: Unable to allocate the specified binding value	This document
		TBD5: Unable to allocate a new binding label/SID	This document
		TBD6: Unable to remove the binding value	This document
		TBD7: Inconsistent binding types	This document

Table 5

13. Acknowledgements

We would like to thank Milos Fabian, Mrinmoy Das, Andrew Stone, Tom Petch, Aijun Wang, Olivier Dugeon, and Adrian Farrel for their valuable comments.

Thanks to Julien Meuric for shepherding. Thanks to John Scudder for the AD review.

Thanks to Theresa Enghardt for the GENART review.

Thanks to Martin Vigoureux, Benjamin Kaduk, Eric Vyncke, Lars Eggert, Murray Kucherawy, and Erik Kline for the IESG reviews.

14. References

14.1. Normative References

- [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>>.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <<https://www.rfc-editor.org/info/rfc3032>>.
- [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>>.
- [RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <<https://www.rfc-editor.org/info/rfc5462>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.

- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8402] Filtsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8664] Sivabalan, S., Filtsfils, 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>>.
- [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>>.
- [RFC8986] Filtsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <<https://www.rfc-editor.org/info/rfc8986>>.
- [RFC9050] Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) Procedures and Extensions for Using the PCE as a Central Controller (PCECC) of LSPs", RFC 9050, DOI 10.17487/RFC9050, July 2021, <<https://www.rfc-editor.org/info/rfc9050>>.
- [I-D.ietf-pce-segment-routing-ipv6]
Li, C., Negi, M., Sivabalan, S., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", Work in Progress, Internet-Draft, draft-ietf-pce-segment-routing-ipv6-12, 6 March 2022, <<https://www.ietf.org/internet-drafts/draft-ietf-pce-segment-routing-ipv6-12.txt>>.

14.2. 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>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.
- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/info/rfc8754>>.
- [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-21, 19 March 2022, <<https://www.ietf.org/archive/id/draft-ietf-spring-segment-routing-policy-21.txt>>.
- [I-D.ietf-pce-pcep-yang]
Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-yang-18, 25 January 2022, <<https://www.ietf.org/archive/id/draft-ietf-pce-pcep-yang-18.txt>>.
- [I-D.li-pce-controlled-id-space]
Li, C., Chen, M., Wang, A., Cheng, W., and C. Zhou, "PCE Controlled ID Space", Work in Progress, Internet-Draft, draft-li-pce-controlled-id-space-10, 24 February 2022, <<https://www.ietf.org/archive/id/draft-li-pce-controlled-id-space-10.txt>>.
- [I-D.ietf-pce-sr-path-segment]
Li, C., Chen, M., Cheng, W., Gandhi, R., and Q. Xiong, "Path Computation Element Communication Protocol (PCEP) Extension for Path Segment in Segment Routing (SR)", Work in Progress, Internet-Draft, draft-ietf-pce-sr-path-segment-05, 13 February 2022, <<https://www.ietf.org/archive/id/draft-ietf-pce-sr-path-segment-05.txt>>.

Appendix A. Contributor Addresses

Jonathan Hardwick
Microsoft
United Kingdom

Email: jonhardwick@microsoft.com

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Mahendra Singh Negi
RtBrick India
N-17L, Floor-1, 18th Cross Rd, HSR Layout Sector-3
Bangalore, Karnataka 560102
India

Email: mahend.ietf@gmail.com

Mike Koldychev
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada

Email: mkoldych@cisco.com

Zafar Ali
Cisco Systems, Inc.

Email: zali@cisco.com

Authors' Addresses

Siva Sivabalan
Ciena Corporation
Email: msiva282@gmail.com

Clarence Filsfils
Cisco Systems, Inc.
Pegasus Parc
BRABANT 1831 De kleetlaan 6a
Belgium
Email: cfilsfil@cisco.com

Jeff Tantsura
Microsoft Corporation
Email: jefftant.ietf@gmail.com

Stefano Previdi
Huawei Technologies
Email: stefano@previdi.net

Cheng Li (editor)
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing
100095
China
Email: c.l@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 6, 2021

C. Li
Huawei Technologies
M. Negi
RtBrick Inc
M. Koldychev
Cisco Systems, Inc.
P. Kaladharan
RtBrick Inc
Y. Zhu
China Telecom
November 2, 2020

PCEP Extensions for Segment Routing leveraging the IPv6 data plane
draft-ietf-pce-segment-routing-ipv6-07

Abstract

The Source Packet Routing in Networking (SPRING) architecture describes how Segment Routing (SR) can be used to steer packets through an IPv6 or MPLS network using the source routing paradigm. SR enables any head-end node to select any path without relying on a hop-by-hop signaling technique (e.g., LDP or RSVP-TE).

It depends only on "segments" that are advertised by Link- State IGPs. A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE).

Since SR can be applied to both MPLS and IPv6 forwarding plane, a PCE should be able to compute SR-Path for both MPLS and IPv6 forwarding plane. This document describes the extensions required for SR support for IPv6 data plane in Path Computation Element communication Protocol (PCEP). The PCEP extension and mechanism to support SR-MPLS is described in RFC 8664. This document extends it to support SRv6 (SR over IPv6).

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 May 6, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	5
3. Overview of PCEP Operation in SRv6 Networks	5
3.1. Operation Overview	6
3.2. SRv6-Specific PCEP Message Extensions	6
4. Object Formats	7
4.1. The OPEN Object	7
4.1.1. The SRv6 PCE Capability sub-TLV	7
4.2. The RP/SRP Object	8
4.3. ERO	8
4.3.1. SRv6-ERO Subobject	9
4.3.1.1. SID Structure	11
4.4. RRO	12
4.4.1. SRv6-RRO Subobject	12

5.	Procedures	13
5.1.	Exchanging the SRv6 Capability	13
5.2.	ERO Processing	15
5.2.1.	SRv6 ERO Validation	15
5.2.2.	Interpreting the SRv6-ERO	16
5.3.	RRO Processing	16
6.	Security Considerations	16
7.	IANA Considerations	17
7.1.	PCEP ERO and RRO subobjects	17
7.2.	New SRv6-ERO Flag Registry	17
7.3.	PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators	18
7.4.	SRv6 PCE Capability Flags	18
7.5.	New Path Setup Type	18
7.6.	ERROR Objects	19
8.	Acknowledgements	19
9.	References	19
9.1.	Normative References	19
9.2.	Informative References	21
	Appendix A. Contributor	23
	Authors' Addresses	23

1. Introduction

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based. A segment can have a semantic local to an SR node or global within an SR domain. SR allows to enforce a flow through any path and service chain while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locator, etc. The list of segment forming the path is called the Segment List and is encoded in the packet header. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [RFC8754]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment.

Segment Routing use cases are described in [RFC7855] and [RFC8354]. Segment Routing protocol extensions are defined in [RFC8667], and [RFC8666].

As per [RFC8754], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in an illustration provided in [I-D.ietf-spring-srv6-network-programming].

The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). The SR is applied to IPV6 forwarding plane using SRH. A SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node.

[RFC5440] describes Path Computation Element communication Protocol (PCEP) for communication between a Path Computation Client (PCC) and a Path Computation Element (PCE) or between a pair of PCEs. A PCE or a PCC operating as a PCE (in hierarchical PCE environment) computes paths for MPLS Traffic Engineering LSPs (MPLS-TE LSPs) based on various constraints and optimization criteria. [RFC8231] specifies extensions to PCEP that allow a stateful PCE to compute and recommend network paths in compliance with [RFC4657] and defines objects and TLVs for MPLS-TE LSPs. Stateful PCEP extensions provide synchronization of LSP state between a PCC and a PCE or between a pair of PCEs, delegation of LSP control, reporting of LSP state from a PCC to a PCE, controlling the setup and path routing of an LSP from a PCE to a PCC. Stateful PCEP extensions are intended for an operational model in which LSPs are configured on the PCC, and control over them is delegated to the PCE.

A mechanism to dynamically initiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE is specified in [RFC8281]. As per [RFC8664], it is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [RFC8281] using the SR specific PCEP extensions specified in [RFC8664]. [RFC8664] specifies PCEP extensions for supporting a SR-TE LSP for MPLS data plane. This document extends [RFC8664] to support SR for IPv6 data plane. Additionally, using procedures described in this document, a PCC can request an SRv6 path from either stateful or a stateless PCE. This specification relies on the PATH-SETUP-TYPE TLV and procedures specified in [RFC8408].

This specification provides a mechanism for a network controller (acting as a PCE) to instantiate candidate paths for an SR Policy onto a head-end node (acting as a PCC) using PCEP. For more information on the SR Policy Architecture, see [I-D.ietf-spring-segment-routing-policy].

2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [RFC8051]: Stateful PCE, Delegation.

The message formats in this document are specified using Routing Backus-Naur Format (RBNF) encoding as specified in [RFC5511].

NAI: Node or Adjacency Identifier.

PCC: Path Computation Client.

PCE: Path Computation Element.

PCEP: Path Computation Element Protocol.

SR: Segment Routing.

SID: Segment Identifier.

SRv6: Segment Routing for IPv6 forwarding plane.

SRH: IPv6 Segment Routing Header.

SR Path: IPv6 Segment List (List of IPv6 SIDs representing a path in IPv6 SR domain)

Further, note that the term LSP used in the PCEP specifications, would be equivalent to a SRv6 Path (represented as a list of SRv6 segments) in the context of supporting SRv6 in PCEP.

3. Overview of PCEP Operation in SRv6 Networks

Basic operations for PCEP speakers is as per [RFC8664]. SRv6 Paths computed by a PCE can be represented as an ordered list of SRv6 segments of 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment" in this document.

[RFC8664] defined a new Explicit Route Object (ERO) subobject denoted by "SR-ERO subobject" capable of carrying a SID as well as the identity of the node/adjacency represented by the SID. SR-capable PCEP speakers should be able to generate and/or process such ERO subobject. An ERO containing SR-ERO subobjects can be included in the PCEP Path Computation Reply (PCRep) message defined in [RFC5440], the PCEP LSP Initiate Request message (PCInitiate) defined in

[RFC8281], as well as in the PCEP LSP Update Request (PCUpd) and PCEP LSP State Report (PCRpt) messages defined in defined in [RFC8231].

This document define new subobjects "SRv6-ERO" and "SRv6-RRO" in ERO and RRO respectively to carry SRv6 SID (IPv6 Address). SRv6-capable PCEP speakers MUST be able to generate and/or process this.

When a PCEP session between a PCC and a PCE is established, both PCEP speakers exchange their capabilities to indicate their ability to support SRv6 specific functionality.

In summary, this document:

- o Defines a new PCEP capability for SRv6.
- o Defines a new subobject SRv6-ERO in ERO.
- o Defines a new subobject SRv6-RRO in RRO.
- o Defines a new path setup type carried in the PATH-SETUP-TYPE TLV and the PATH-SETUP-TYPE-CAPABILITY TLV.

3.1. Operation Overview

In SR networks, an ingress node of an SR path appends all outgoing packets with an SR header consisting of a list of SIDs (IPv6 Prefix in case of SRv6). The header has all necessary information to guide the packets from the ingress node to the egress node of the path, and hence there is no need for any signaling protocol.

For IPv6 in control plane with MPLS data-plane, mechanism remains same as [RFC8664]

This document describes extensions to SR path for IPv6 data plane. SRv6 Path (i.e. ERO) consists of an ordered set of SRv6 SIDs (see details in Figure 2).

A PCC or PCE indicates its ability to support SRv6 during the PCEP session Initialization Phase via a new SRv6-PCE-CAPABILITY sub-TLV (see details in Section 4.1.1).

3.2. SRv6-Specific PCEP Message Extensions

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable length body made up of mandatory and/or optional objects. This document does not require any changes in the format of PCReq and PCRep messages specified in [RFC5440], PCInitiate message specified in [RFC8281], and PCRpt and PCUpd messages

specified in [RFC8231]. However, PCEP messages pertaining to SRv6 MUST include PATH-SETUP-TYPE TLV in the RP or SRP object to clearly identify that SRv6 is intended.

4. Object Formats

4.1. The OPEN Object

4.1.1. The SRv6 PCE Capability sub-TLV

This document defines a new Path Setup Type (PST) [RFC8408] for SRv6, as follows:

- o PST = TBD2: Path is setup using SRv6.

A PCEP speaker MUST indicate its support of the function described in this document by sending a PATH-SETUP-TYPE-CAPABILITY TLV in the OPEN object with this new PST included in the PST list.

This document also defines the SRv6-PCE-CAPABILITY sub-TLV. PCEP speakers use this sub-TLV to exchange information about their SRv6 capability. If a PCEP speaker includes PST=TBD2 in the PST List of the PATH-SETUP-TYPE-CAPABILITY TLV then it MUST also include the SRv6-PCE-CAPABILITY sub-TLV inside the PATH-SETUP-TYPE-CAPABILITY TLV.

The format of the SRv6-PCE-CAPABILITY sub-TLV is shown in the following figure:

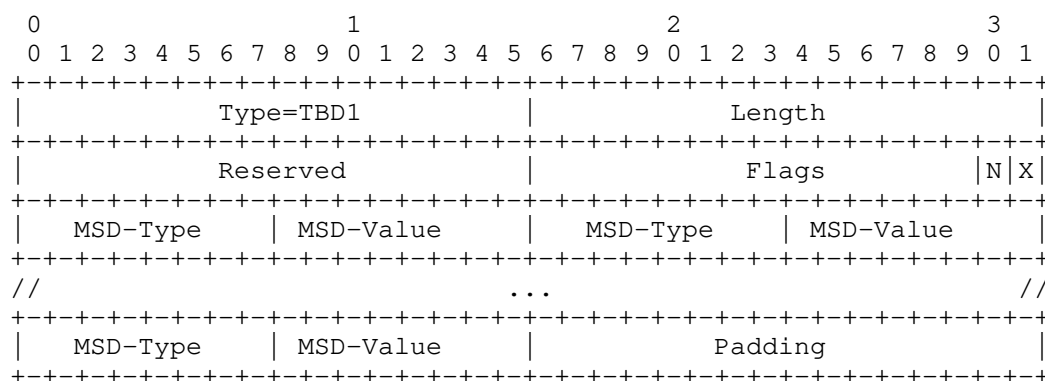


Figure 1: SRv6-PCE-CAPABILITY sub-TLV format

The code point for the TLV type (TBD1) is to be defined by IANA. The TLV length is variable.

The value comprises of -

Reserved: 2 octet, this field MUST be set to 0 on transmission, and ignored on receipt.

Flags: 2 octet, two bits are currently assigned in this document.

N bit: A PCC sets this flag bit to 1 to indicate that it is capable of resolving a Node or Adjacency Identifier (NAI) to a SRv6-SID.

X bit: A PCC sets this bit to 1 to indicate that it does not impose any limit on MSD (irrespective of the MSD-Type).

Unassigned bits MUST be set to 0 and ignored on receipt.

A pair of (MSD-Type, MSD-Value): Where MSD-Type (1 octet) is as per the IGP MSD Type registry created by [RFC8491] and populated with SRv6 MSD types as per [I-D.ietf-lsr-isis-srv6-extensions]; MSD-Value (1 octet) is as per [RFC8491].

This sub-TLV format is compliant with the PCEP TLV format defined in [RFC5440]. That is, the sub-TLV is composed of 2 octets for the type, 2 octets specifying the length, and a Value field. The Type field when set to TBD1 identifies the SRv6-PCE-CAPABILITY sub-TLV and the presence of the sub-TLV indicates the support for the SRv6 paths in PCEP. The Length field defines the length of the value portion in octets. The TLV is padded to 4-octet alignment, and padding is not included in the Length field. The number of (MSD-Type,MSD-Value) pairs can be determined from the Length field of the TLV.

4.2. The RP/SRP Object

In order to indicate the SRv6 path, RP or SRP object MUST include the PATH-SETUP-TYPE TLV specified in [RFC8408]. This document defines a new Path Setup Type (PST=TBD2) for SRv6.

The LSP-IDENTIFIERS TLV MAY be present for the above PST type.

4.3. ERO

In order to support SRv6, new subobject "SRv6-ERO" is defined in ERO.

4.3.1. SRv6-ERO Subobject

An SRv6-ERO subobject is formatted as shown in the following figure.

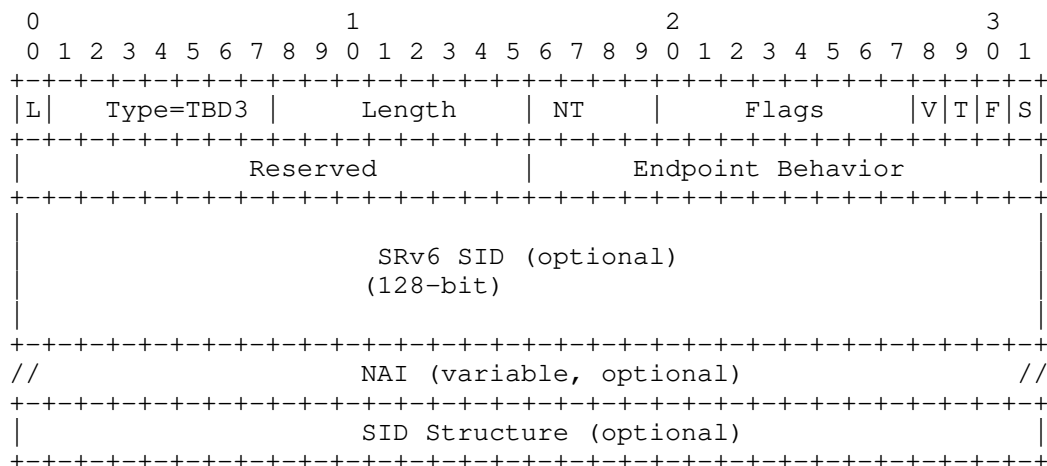


Figure 2: SRv6-ERO Subobject Format

The fields in the SRv6-ERO Subobject are as follows:

The 'L' Flag: Indicates whether the subobject represents a loose-hop (see [RFC3209]). If this flag is set to zero, a PCC MUST NOT overwrite the SID value present in the SRv6-ERO subobject. Otherwise, a PCC MAY expand or replace one or more SID values in the received SRv6-ERO based on its local policy.

Type: indicates the content of the subobject, i.e. when the field is set to TBD3, the subobject is a SRv6-ERO subobject representing a SRv6 SID.

Length: Contains the total length of the subobject in octets. The Length MUST be at least 24, and MUST be a multiple of 4. An SRv6-ERO subobject MUST contain at least one of a SRv6-SID or an NAI. The S and F bit in the Flags field indicates whether the SRv6-SID or NAI fields are absent.

NAI Type (NT): Indicates the type and format of the NAI contained in the object body, if any is present. If the F bit is set to zero (see below) then the NT field has no meaning and MUST be ignored by the receiver. This document reuses NT types defined in [RFC8664]:

If NT value is 0, the NAI MUST NOT be included.

When NT value is 2, the NAI is as per the 'IPv6 Node ID' format defined in [RFC8664], which specifies an IPv6 address. This is used to identify the owner of the SRv6 Identifier. This is optional, as the LOC (the locator portion) of the SRv6 SID serves a similar purpose (when present).

When NT value is 4, the NAI is as per the 'IPv6 Adjacency' format defined in [RFC8664], which specify a pair of IPv6 addresses. This is used to identify the IPv6 Adjacency and used with the SRv6 Adj-SID.

When NT value is 6, the NAI is as per the 'link-local IPv6 addresses' format defined in [RFC8664], which specify a pair of (global IPv6 address, interface ID) tuples. It is used to identify the IPv6 Adjacency and used with the SRv6 Adj-SID.

SR-MPLS specific NT types are not valid in SRv6-ERO.

Flags: Used to carry additional information pertaining to the SRv6-SID. This document defines the following flag bits. The other bits MUST be set to zero by the sender and MUST be ignored by the receiver.

- o S: When this bit is set to 1, the SRv6-SID value in the subobject body is absent. In this case, the PCC is responsible for choosing the SRv6-SID value, e.g., by looking up in the SR-DB using the NAI which, in this case, MUST be present in the subobject. If the S bit is set to 1 then F bit MUST be set to zero.
- o F: When this bit is set to 1, the NAI value in the subobject body is absent. The F bit MUST be set to 1 if NT=0, and otherwise MUST be set to zero. The S and F bits MUST NOT both be set to 1.
- o T: When this bit is set to 1, the SID Structure value in the subobject body is present. The T bit MUST be set to 0 when S bit is set to 1. If the T bit is set when the S bit is set, the T bit MUST be ignored. Thus, the T bit indicates the presence of an optional 8-byte SID Structure when SRv6 SID is included. The SID Structure is defined in Section 4.3.1.1.
- o V: The "SID verification" bit usage is as per Section 5.1 of [I-D.ietf-spring-segment-routing-policy].

[Editor's Note: Need to check if another flag - A flag for the SR Algorithm is required.]

Reserved: MUST be set to zero while sending and ignored on receipt.

Endpoint Behavior: A 16 bit field representing the behavior associated with the SRv6 SIDs. This information is optional and plays no role in the fields in SRH imposed on the packet. It could be used for maintainability and diagnostic purpose. If behavior is not known, 0 is used. The list of Endpoint behavior are defined in [I-D.ietf-spring-srv6-network-programming].

SRv6 SID: SRv6 Identifier is the 128 bit IPv6 addresses representing the SRv6 segment.

NAI: The NAI associated with the SRv6-SID. The NAI's format depends on the value in the NT field, and is described in [RFC8664].

At least one of the SRv6-SID or the NAI MUST be included in the SRv6-ERO subobject, and both MAY be included.

4.3.1.1. SID Structure

The SID Structure is an optional part of the SR-ERO subobject, as described in Section 4.3.1. It is formatted as shown in the following figure.

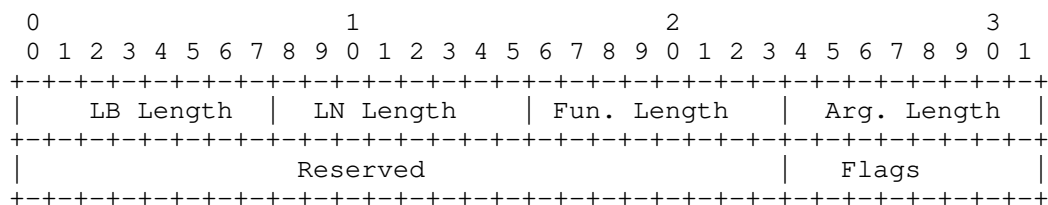


Figure 3: SID Structure Format

where:

LB Length: 1 octet. SRv6 SID Locator Block length in bits.

LN Length: 1 octet. SRv6 SID Locator Node length in bits.

Fun. Length: 1 octet. SRv6 SID Function length in bits.

Arg. Length: 1 octet. SRv6 SID Arguments length in bits.

The sum of all four sizes in the SID Structure must be lower or equal to 128 bits. If the sum of all four sizes advertised in the SID Structure is larger than 128 bits, the corresponding SRv6 SID MUST be considered as an Error. A PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Invalid SRv6 SID Structure").

Reserved: MUST be set to zero while sending and ignored on receipt.

Flags: Currently no flags are defined. Unassigned bits must be set to zero while sending and ignored on receipt.

The SRv6 SID Structure TLV provides the detailed encoding information of an SRv6 SID, which is useful in the use cases that require to know the SRv6 SID structure. When a PCEP speaker receives the SRv6 SID and its structure information, the SRv6 SID can be parsed based on the SRv6 SID Structure TLV and/or possible local policies. The consumers of SRv6 SID structure MAY be other use cases, and the processing and usage of SRv6 SID structure TLV are different based on use cases. This is out of the scope of this document, and will be described in other documents.

4.4. RRO

In order to support SRv6, new subobject "SRv6-RRO" is defined in RRO.

4.4.1. SRv6-RRO Subobject

A PCC reports an SRv6 path to a PCE by sending a PCRpt message, per [RFC8231]. The RRO on this message represents the SID list that was applied by the PCC, that is, the actual path taken. The procedures of [RFC8664] with respect to the RRO apply equally to this specification without change.

An RRO contains one or more subobjects called "SRv6-RRO subobjects" whose format is shown below:

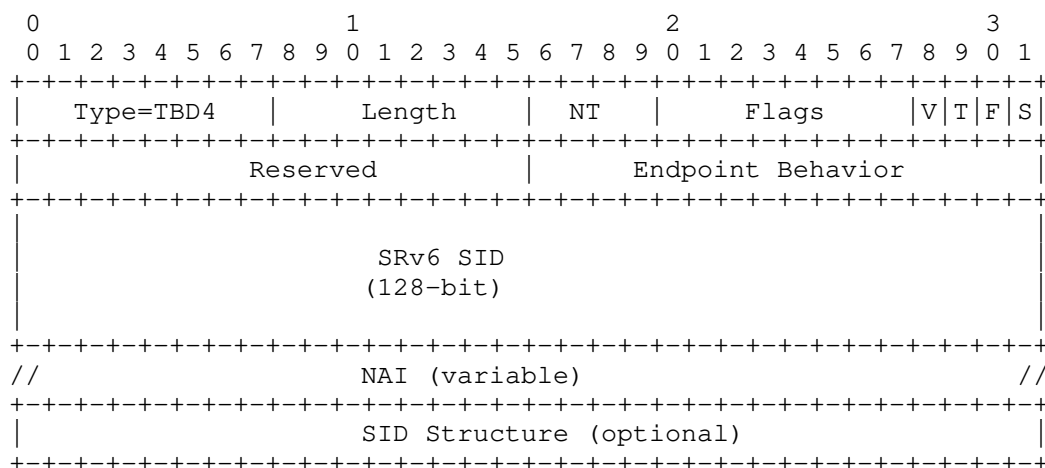


Figure 4: SRv6-RRO Subobject Format

The format of the SRv6-RRO subobject is the same as that of the SRv6-ERO subobject, but without the L flag.

The V flag has no meaning in the SRv6-RRO and is ignored on receipt at the PCE.

Ordering of SRv6-RRO subobjects by PCC in PCRpt message remains as per [RFC8664].

5. Procedures

5.1. Exchanging the SRv6 Capability

A PCC indicates that it is capable of supporting the head-end functions for SRv6 by including the SRv6-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCE. A PCE indicates that it is capable of computing SRv6 paths by including the SRv6-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCC.

If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a PST list containing PST=TBD2, but the SRv6-PCE-CAPABILITY sub-TLV is absent, then the PCEP speaker MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value TBD5 (to be assigned by IANA) (Missing PCE-SRv6-CAPABILITY sub-TLV) and MUST then close the PCEP session. If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a SRv6-PCE-CAPABILITY sub-TLV, but the PST list does not contain PST=TBD2, then the PCEP speaker MUST ignore the SRv6-PCE-CAPABILITY sub-TLV.

The number of SRv6 SIDs that can be imposed on a packet depends on the PCC's IPv6 data plane's capability. If a PCC sets the X flag to 1 then the MSD is not used and MUST NOT be included. If a PCE receives an SRv6-PCE-CAPABILITY sub-TLV with the X flag set to 1 then it MUST ignore any MSD-Type, MSD-Value fields and MUST assume that the sender can impose any length of SRH. If a PCC sets the X flag to zero, then it sets the SRv6 MSD-Type, MSD-Value fields that it can impose on a packet. If a PCE receives an SRv6-PCE-CAPABILITY sub-TLV with the X flag and SRv6 MSD-Type, MSD-Value fields both set to zero then it is considered as an error and the PCE MUST respond with a PCErr message (Error-Type=1 "PCEP session establishment failure" and Error-Value=1 "reception of an invalid Open message or a non Open message."). In case the MSD-Type in SRv6-PCE-CAPABILITY sub-TLV received by the PCE does not correspond to one of the SRv6 MSD types, the PCE MUST respond with a PCErr message (Error-Type=1 "PCEP session establishment failure" and Error-Value=1 "reception of an invalid Open message or a non Open message.").

Note that the MSD-Type, MSD-Value exchanged via the SRv6-PCE-CAPABILITY sub-TLV indicates the SRv6 SID imposition limit for the PCC node. However, if a PCE learns these via different means, e.g routing protocols, as specified in:
[I-D.li-ospf-ospfv3-srv6-extensions];
[I-D.ietf-lsr-isis-srv6-extensions]; [I-D.ietf-idr-bgppls-srv6-ext],
then it ignores the values in the SRv6-PCE-CAPABILITY sub-TLV. Furthermore, whenever a PCE learns the other advanced SRv6 MSD via different means, it MUST use that value regardless of the values exchanged in the SRv6-PCE-CAPABILITY sub-TLV.

Once an SRv6-capable PCEP session is established with a non-zero SRv6 MSD value, the corresponding PCE MUST NOT send SRv6 paths with a number of SIDs exceeding that SRv6 MSD value (based on the SRv6 MSD Type). If a PCC needs to modify the SRv6 MSD value, it MUST close the PCEP session and re-establish it with the new value. If a PCEP session is established with a non-zero SRv6 MSD value, and the PCC receives an SRv6 path containing more SIDs than specified in the SRv6 MSD value (based on the SRv6 MSD type), the PCC MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value 3 (Unsupported number of Segment ERO subobjects). If a PCEP session is established with an SRv6 MSD value of zero, then the PCC MAY specify an SRv6 MSD for each path computation request that it sends to the PCE, by including a "maximum SID depth" metric object on the request similar to [RFC8664].

The N flag, X flag and (MSD-Type,MSD-Value) pair inside the SRv6-PCE-CAPABILITY sub-TLV are meaningful only in the Open message sent from a PCC to a PCE. As such, a PCE MUST set the flags to zero and not include any (MSD-Type,MSD-Value) pair in the SRv6-PCE-CAPABILITY sub-

TLV in an outbound message to a PCC. Similarly, a PCC MUST ignore N,X flag and any (MSD-Type,MSD-Value) pair received from a PCE. If a PCE receives multiple SRv6-PCE-CAPABILITY sub-TLVs in an Open message, it processes only the first sub-TLV received.

5.2. ERO Processing

The ERO processing remains as per [RFC5440] and [RFC8664].

5.2.1. SRv6 ERO Validation

If a PCC does not support the SRv6 PCE Capability and thus cannot recognize the SRv6-ERO or SRv6-RRO subobjects, it will respond according to the rules for a malformed object per [RFC5440].

On receiving an SRv6-ERO, a PCC MUST validate that the Length field, the S bit, the F bit, the T bit, and the NT field are consistent, as follows.

- o If NT=0, the F bit MUST be 1, the S bit MUST be zero and the Length MUST be 24.
- o If NT=2, the F bit MUST be zero. If the S bit is 1, the Length MUST be 24, otherwise the Length MUST be 40.
- o If NT=4, the F bit MUST be zero. If the S bit is 1, the Length MUST be 40, otherwise the Length MUST be 56.
- o If NT=6, the F bit MUST be zero. If the S bit is 1, the Length MUST be 48, otherwise the Length MUST be 64.
- o NT types (1,3, and 5) are not valid for SRv6.
- o If T bit is 1, then S bit MUST be zero.

If a PCC finds that the NT field, Length field, S bit, F bit, and T bit are not consistent, it MUST consider the entire ERO invalid and MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 11 ("Malformed object").

If a PCEP speaker that does not recognize the NT value received in SRv6-ERO subobject, it would behave as per [RFC8664].

In case a PCEP speaker receives the SRv6-ERO subobject, when the PST is not set to TBD2 or SRv6-PCE-CAPABILITY sub-TLV was not exchanged, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = TBD5 ("Attempted SRv6 when the capability was not advertised").

If a PCC receives a list of SRv6 segments, and the number of SRv6 segments exceeds the SRv6 MSD that the PCC can impose on the packet (SRH), it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Unsupported number of Segment ERO subobjects") as per [RFC8664].

When a PCEP speaker detects that all subobjects of ERO are not of type TBD3, and if it does not handle such ERO, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD ("Non-identical ERO subobjects") as per [RFC8664].

5.2.2. Interpreting the SRv6-ERO

The SRv6-ERO contains a sequence of subobjects. According to [I-D.ietf-spring-segment-routing-policy], each SRv6-ERO subobject in the sequence identifies a segment that the traffic will be directed to, in the order given. That is, the first subobject identifies the first segment the traffic will be directed to, the second SRv6-ERO subobject represents the second segment, and so on.

The PCC interprets the SRv6-ERO by converting it to an SRv6 SRH plus a next hop. The PCC sends packets along the segment routed path by prepending the SRH onto the packets and sending the resulting, modified packet to the next hop.

5.3. RRO Processing

The syntax checking rules that apply to the SRv6-RRO subobject are identical to those of the SRv6-ERO subobject, except as noted below.

If a PCEP speaker receives an SRv6-RRO subobject in which both SRv6 SID and NAI are absent, it MUST consider the entire RRO invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD6 ("Both SID and NAI are absent in SRv6-RRO subobject").

If a PCE detects that the subobjects of an RRO are a mixture of SRv6-RRO subobjects and subobjects of other types, then it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = TBD7 ("RRO mixes SRv6-RRO subobjects with other subobject types").

6. Security Considerations

The security considerations described in [RFC5440], [RFC8231] and [RFC8281], [RFC8664], are applicable to this specification. No additional security measure is required.

7. IANA Considerations

7.1. PCEP ERO and RRO subobjects

This document defines a new subobject type for the PCEP explicit route object (ERO), and a new subobject type for the PCEP record route object (RRO). The code points for subobject types of these objects is maintained in the RSVP parameters registry, under the EXPLICIT_ROUTE and ROUTE_RECORD objects. IANA is requested to allocate code-points in the RSVP Parameters registry for each of the new subobject types defined in this document.

Object	Subobject	Subobject Type
EXPLICIT_ROUTE	SRv6-ERO (PCEP-specific)	TBD3
ROUTE_RECORD	SRv6-RRO (PCEP-specific)	TBD4

7.2. New SRv6-ERO Flag Registry

IANA is requested to create a new sub-registry, named "SRv6-ERO Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the SRv6-ERO subobject. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (counting from bit 0 as the most significant bit)
- o Capability description
- o Defining RFC

The following values are defined in this document:

Bit	Description	Reference
0-7	Unassigned	
8	SID Verification (V)	This document
9	SID Structure is present (T)	This document
10	NAI is absent (F)	This document
11	SID is absent (S)	This document

7.3. PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators

IANA maintains a sub-registry, named "PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the type indicator space for sub-TLVs of the PATH-SETUP-TYPE-CAPABILITY TLV. IANA is requested to make the following assignment:

Value -----	Meaning -----	Reference -----
TBD1	SRv6-PCE-CAPABILITY	This Document

7.4. SRv6 PCE Capability Flags

IANA is requested to create a new sub-registry, named "SRv6 Capability Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the SRv6-PCE-CAPABILITY sub-TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (counting from bit 0 as the most significant bit)
- o Capability description
- o Defining RFC

The following values are defined in this document:

Bit	Description	Reference
0-13	Unassigned	
14	Node or Adjacency Identifier (NAI) is supported (N)	This document
15	Unlimited Maximum SID Depth (X)	This document

7.5. New Path Setup Type

[RFC8408] created a sub-registry within the "Path Computation Element Protocol (PCEP) Numbers" registry called "PCEP Path Setup Types". IANA is requested to allocate a new code point within this registry, as follows:

Value -----	Description -----	Reference -----
TBD2	Traffic engineering path is setup using SRv6.	This Document

7.6. ERROR Objects

IANA is requested to allocate code-points in the PCEP-ERROR Object Error Types and Values registry for the following new error-values:

Error-Type -----	Meaning -----
10	Reception of an invalid object Error-value = TBD5 (Missing PCE-SRv6-CAPABILITY sub-TLV) Error-value = TBD6 (Both SID and NAI are absent in SRv6-RRO subobject) Error-value = TBD7 (RRO mixes SRv6-RRO subobjects with other subobject types) Error-value = TBD8 (Invalid SRv6 SID Structure)
19	Invalid Operation Error-value = TBD5 (Attempted SRv6 when the capability was not advertised)

8. Acknowledgements

The authors would like to thank Jeff Tentsura, Adrian Farrel, Aijun Wang and Khasanov Boris for valuable suggestions.

9. References

9.1. Normative References

- [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>>.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.

- [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>>.
- [RFC5511] Farrel, A., "Routing Backus-Naur Form (RBNF): A Syntax Used to Form Encoding Rules in Various Routing Protocol Specifications", RFC 5511, DOI 10.17487/RFC5511, April 2009, <<https://www.rfc-editor.org/info/rfc5511>>.
- [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>>.
- [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>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491, DOI 10.17487/RFC8491, November 2018, <<https://www.rfc-editor.org/info/rfc8491>>.

[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>>.

[I-D.ietf-lsr-isis-srv6-extensions]
Psenak, P., Filsfils, C., Bashandy, A., Decraene, B., and Z. Hu, "IS-IS Extension to Support Segment Routing over IPv6 Dataplane", draft-ietf-lsr-isis-srv6-extensions-11 (work in progress), October 2020.

[I-D.ietf-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-ietf-spring-srv6-network-programming-24 (work in progress), October 2020.

9.2. Informative References

[RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <<https://www.rfc-editor.org/info/rfc4657>>.

[RFC7855] Previdi, S., Ed., Filsfils, C., Ed., Decraene, B., Litkowski, S., Horneffer, M., and R. Shakir, "Source Packet Routing in Networking (SPRING) Problem Statement and Requirements", RFC 7855, DOI 10.17487/RFC7855, May 2016, <<https://www.rfc-editor.org/info/rfc7855>>.

[RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.

[RFC8354] Brzozowski, J., Leddy, J., Filsfils, C., Maglione, R., Ed., and M. Townsley, "Use Cases for IPv6 Source Packet Routing in Networking (SPRING)", RFC 8354, DOI 10.17487/RFC8354, March 2018, <<https://www.rfc-editor.org/info/rfc8354>>.

[RFC8666] Psenak, P., Ed. and S. Previdi, Ed., "OSPFv3 Extensions for Segment Routing", RFC 8666, DOI 10.17487/RFC8666, December 2019, <<https://www.rfc-editor.org/info/rfc8666>>.

- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.
- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/info/rfc8754>>.
- [I-D.ietf-spring-segment-routing-policy] Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-09 (work in progress), November 2020.
- [I-D.li-ospf-ospfv3-srv6-extensions] Li, Z., Hu, Z., Cheng, D., Talaulikar, K., and P. Psenak, "OSPFv3 Extensions for SRv6", draft-li-ospf-ospfv3-srv6-extensions-07 (work in progress), November 2019.
- [I-D.ietf-idr-bgppls-srv6-ext] Dawra, G., Filsfils, C., Talaulikar, K., Chen, M., daniel.bernier@bell.ca, d., and B. Decraene, "BGP Link State Extensions for SRv6", draft-ietf-idr-bgppls-srv6-ext-04 (work in progress), November 2020.

Appendix A. Contributor

The following persons contributed to this document:

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Huang Wumin
Huawei Technologies
Huawei Building, No. 156 Beiqing Rd.
Beijing 100095
China

Email: huangwumin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Building, No. 156 Beiqing Rd.
Beijing 100095
China

Email: pengshuping@huawei.com

Authors' Addresses

Cheng Li(Editor)
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

EMail: c.l@huawei.com

Mahendra Singh Negi
RtBrick Inc
Bangalore, Karnataka
India

EMail: mahend.ietf@gmail.com

Mike Koldychev
Cisco Systems, Inc.
Canada

EMail: mkoldych@cisco.com

Prejeeth Kaladharan
RtBrick Inc
Bangalore, Karnataka
India

EMail: prejeeth@rtbrick.com

Yongqing Zhu
China Telecom
109 West Zhongshan Ave, Tianhe District
Bangalore, Guangzhou
P.R. China

EMail: zhuyq8@chinatelecom.cn

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 3 October 2022

C. Li
Huawei Technologies
M. Negi
RtBrick Inc
S. Sivabalan
Ciena Corporation
M. Koldychev
Cisco Systems, Inc.
P. Kaladharan
RtBrick Inc
Y. Zhu
China Telecom
1 April 2022

PCEP Extensions for Segment Routing leveraging the IPv6 data plane
draft-ietf-pce-segment-routing-ipv6-13

Abstract

The Source Packet Routing in Networking (SPRING) architecture describes how Segment Routing (SR) can be used to steer packets through an IPv6 or MPLS network using the source routing paradigm. SR enables any head-end node to select any path without relying on a hop-by-hop signaling technique (e.g., LDP or RSVP-TE).

It depends only on "segments" that are advertised by Link-State IGPs. A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE).

Since SR can be applied to both MPLS and IPv6 forwarding plane, a PCE should be able to compute SR-Path for both MPLS and IPv6 forwarding plane. This document describes the extensions required for SR support for IPv6 data plane in Path Computation Element communication Protocol (PCEP). The PCEP extension and mechanism to support SR-MPLS is described in RFC 8664. This document extends it to support SRv6 (SR over IPv6).

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 3 October 2022.

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	3
2. Terminology	5
3. Overview of PCEP Operation in SRv6 Networks	5
3.1. Operation Overview	6
3.2. SRv6-Specific PCEP Message Extensions	7
4. Object Formats	7
4.1. The OPEN Object	7
4.1.1. The SRv6 PCE Capability sub-TLV	7
4.2. The RP/SRP Object	9
4.3. ERO	9
4.3.1. SRv6-ERO Subobject	9
4.3.1.1. SID Structure	11
4.3.1.2. Order of the Optional fields	13
4.4. RRO	13
4.4.1. SRv6-RRO Subobject	13

5.	Procedures	14
5.1.	Exchanging the SRv6 Capability	14
5.2.	ERO Processing	16
5.2.1.	SRv6 ERO Validation	16
5.2.2.	Interpreting the SRv6-ERO	17
5.3.	RRO Processing	17
6.	Security Considerations	17
7.	Manageability Considerations	18
7.1.	Control of Function and Policy	18
7.2.	Information and Data Models	18
7.3.	Liveness Detection and Monitoring	18
7.4.	Verify Correct Operations	18
7.5.	Requirements On Other Protocols	18
7.6.	Impact On Network Operations	18
8.	Implementation Status	18
8.1.	Cisco's Commercial Delivery	19
9.	IANA Considerations	19
9.1.	PCEP ERO and RRO subobjects	19
9.2.	New SRv6-ERO Flag Registry	20
9.3.	PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators	20
9.4.	SRv6 PCE Capability Flags	21
9.5.	New Path Setup Type	21
9.6.	ERROR Objects	21
10.	Acknowledgements	22
11.	References	22
11.1.	Normative References	22
11.2.	Informative References	24
	Appendix A. Contributor	25
	Authors' Addresses	26

1. Introduction

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based. A segment can have a semantic local to an SR node or global within an SR domain. SR allows to enforce a flow through any path and service chain while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locator, etc. The list of segment forming the path is called the Segment List and is encoded in the packet header. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [RFC8754]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment.

Segment Routing use cases are described in [RFC7855] and [RFC8354]. Segment Routing protocol extensions are defined in [RFC8667], and [RFC8666].

As per [RFC8754], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in an illustration provided in [RFC8986].

The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). The SR is applied to IPv6 forwarding plane using SRH. A SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node.

[RFC5440] describes Path Computation Element communication Protocol (PCEP) for communication between a Path Computation Client (PCC) and a Path Computation Element (PCE) or between a pair of PCEs. A PCE or a PCC operating as a PCE (in hierarchical PCE environment) computes paths for MPLS Traffic Engineering LSPs (MPLS-TE LSPs) based on various constraints and optimization criteria. [RFC8231] specifies extensions to PCEP that allow a stateful PCE to compute and recommend network paths in compliance with [RFC4657] and defines objects and TLVs for MPLS-TE LSPs. Stateful PCEP extensions provide synchronization of LSP state between a PCC and a PCE or between a pair of PCEs, delegation of LSP control, reporting of LSP state from a PCC to a PCE, controlling the setup and path routing of an LSP from a PCE to a PCC. Stateful PCEP extensions are intended for an operational model in which LSPs are configured on the PCC, and control over them is delegated to the PCE.

A mechanism to dynamically initiate LSPs on a PCC based on the requests from a stateful PCE or a controller using stateful PCE is specified in [RFC8281]. As per [RFC8664], it is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [RFC8281] using the SR specific PCEP extensions specified in [RFC8664]. [RFC8664] specifies PCEP extensions for supporting a SR-TE LSP for MPLS data plane. This document extends [RFC8664] to support SR for IPv6 data plane. Additionally, using procedures described in this document, a PCC can request an SRv6 path from either stateful or a stateless PCE. This specification relies on the PATH-SETUP-TYPE TLV and procedures specified in [RFC8408].

This specification provides a mechanism for a network controller (acting as a PCE) to instantiate candidate paths for an SR Policy onto a head-end node (acting as a PCC) using PCEP. For more information on the SR Policy Architecture, see [I-D.ietf-spring-segment-routing-policy].

2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [RFC8051]: Stateful PCE, Delegation.

The message formats in this document are specified using Routing Backus-Naur Format (RBNF) encoding as specified in [RFC5511].

NAI: Node or Adjacency Identifier.

PCC: Path Computation Client.

PCE: Path Computation Element.

PCEP: Path Computation Element Protocol.

SR: Segment Routing.

SID: Segment Identifier.

SRv6: Segment Routing for IPv6 forwarding plane.

SRH: IPv6 Segment Routing Header.

SR Path: IPv6 Segment List (List of IPv6 SIDs representing a path in IPv6 SR domain)

Further, note that the term LSP used in the PCEP specifications, would be equivalent to a SRv6 Path (represented as a list of SRv6 segments) in the context of supporting SRv6 in PCEP.

3. Overview of PCEP Operation in SRv6 Networks

Basic operations for PCEP speakers is as per [RFC8664]. SRv6 Paths computed by a PCE can be represented as an ordered list of SRv6 segments of 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment" in this document.

[RFC8664] defined a new Explicit Route Object (ERO) subobject denoted by "SR-ERO subobject" capable of carrying a SID as well as the identity of the node/adjacency represented by the SID. SR-capable PCEP speakers should be able to generate and/or process such ERO subobject. An ERO containing SR-ERO subobjects can be included in the PCEP Path Computation Reply (PCRep) message defined in [RFC5440], the PCEP LSP Initiate Request message (PCInitiate) defined in [RFC8281], as well as in the PCEP LSP Update Request (PCUpd) and PCEP LSP State Report (PCRpt) messages defined in [RFC8231].

This document define new subobjects "SRv6-ERO" and "SRv6-RRO" in ERO and RRO respectively to carry SRv6 SID (IPv6 Address). SRv6-capable PCEP speakers MUST be able to generate and/or process this.

When a PCEP session between a PCC and a PCE is established, both PCEP speakers exchange their capabilities to indicate their ability to support SRv6 specific functionality.

In summary, this document:

- * Defines a new PCEP capability for SRv6.
- * Defines a new subobject SRv6-ERO in ERO.
- * Defines a new subobject SRv6-RRO in RRO.
- * Defines a new path setup type carried in the PATH-SETUP-TYPE TLV and the PATH-SETUP-TYPE-CAPABILITY TLV.

3.1. Operation Overview

In SR networks, an ingress node of an SR path appends all outgoing packets with an SR header consisting of a list of SIDs (IPv6 Prefix in case of SRv6). The header has all necessary information to guide the packets from the ingress node to the egress node of the path, and hence there is no need for any signaling protocol.

For IPv6 in control plane with MPLS data-plane, mechanism remains same as [RFC8664]

This document describes extensions to SR path for IPv6 data plane. SRv6 Path (i.e. ERO) consists of an ordered set of SRv6 SIDs (see details in Figure 2).

A PCC or PCE indicates its ability to support SRv6 during the PCEP session Initialization Phase via a new SRv6-PCE-CAPABILITY sub-TLV (see details in Section 4.1.1).

3.2. SRv6-Specific PCEP Message Extensions

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable length body made up of mandatory and/or optional objects. This document does not require any changes in the format of PCReq and PCRep messages specified in [RFC5440], PCInitiate message specified in [RFC8281], and PCRpt and PCUpd messages specified in [RFC8231]. However, PCEP messages pertaining to SRv6 MUST include PATH-SETUP-TYPE TLV in the RP or SRP object to clearly identify that SRv6 is intended.

4. Object Formats

4.1. The OPEN Object

4.1.1. The SRv6 PCE Capability sub-TLV

This document defines a new Path Setup Type (PST) [RFC8408] for SRv6, as follows:

* PST = 3 (early allocated by IANA): Path is setup using SRv6.

A PCEP speaker MUST indicate its support of the function described in this document by sending a PATH-SETUP-TYPE-CAPABILITY TLV in the OPEN object with this new PST included in the PST list.

This document also defines the SRv6-PCE-CAPABILITY sub-TLV. PCEP speakers use this sub-TLV to exchange information about their SRv6 capability. If a PCEP speaker includes PST=3 (early allocated by IANA) in the PST List of the PATH-SETUP-TYPE-CAPABILITY TLV then it MUST also include the SRv6-PCE-CAPABILITY sub-TLV inside the PATH-SETUP-TYPE-CAPABILITY TLV.

The format of the SRv6-PCE-CAPABILITY sub-TLV is shown in the following figure:

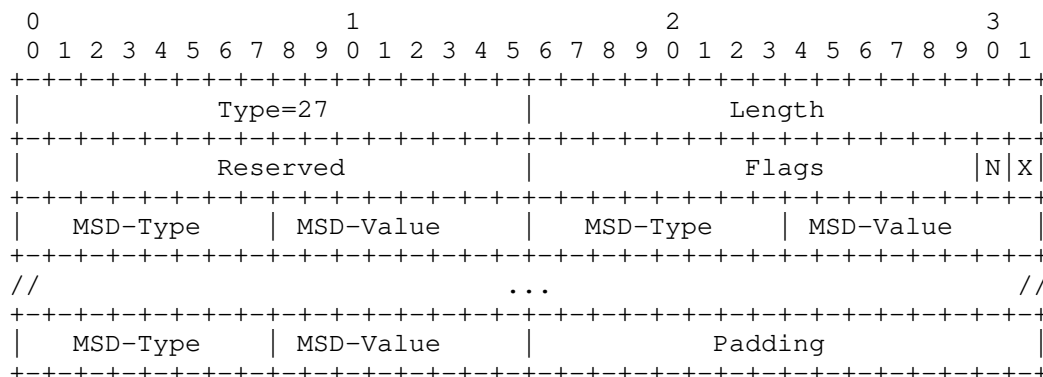


Figure 1: SRv6-PCE-CAPABILITY sub-TLV format

The code point for the TLV type (27) (early allocated by IANA) is to be defined by IANA. The TLV length is variable.

The value comprises of -

Reserved: 2 octet, this field MUST be set to 0 on transmission, and ignored on receipt.

Flags: 2 octet, two bits are currently assigned in this document.

- N bit: A PCC sets this flag bit to 1 to indicate that it is capable of resolving a Node or Adjacency Identifier (NAI) to a SRv6-SID.
- X bit: A PCC sets this bit to 1 to indicate that it does not impose any limit on MSD (irrespective of the MSD-Type).
- Unassigned bits MUST be set to 0 and ignored on receipt.

A pair of (MSD-Type, MSD-Value): Where MSD-Type (1 octet) is as per the IGP MSD Type registry created by [RFC8491] and populated with SRv6 MSD types as per [I-D.ietf-lsr-isis-srv6-extensions]; MSD-Value (1 octet) is as per [RFC8491].

This sub-TLV format is compliant with the PCEP TLV format defined in [RFC5440]. That is, the sub-TLV is composed of 2 octets for the type, 2 octets specifying the length, and a Value field. The Type field when set to 27 (early allocated by IANA) identifies the SRv6-PCE-CAPABILITY sub-TLV and the presence of the sub-TLV indicates the support for the SRv6 paths in PCEP. The Length field defines the length of the value portion in octets. The TLV is padded to 4-octet

alignment, and padding is not included in the Length field. The number of (MSD-Type,MSD-Value) pairs can be determined from the Length field of the TLV.

4.2. The RP/SRP Object

In order to indicate the SRv6 path, RP or SRP object MUST include the PATH-SETUP-TYPE TLV specified in [RFC8408]. This document defines a new Path Setup Type (PST=3(early allocated by IANA)) for SRv6.

The LSP-IDENTIFIERS TLV MAY be present for the above PST type.

4.3. ERO

In order to support SRv6, new subobject "SRv6-ERO" is defined in ERO.

4.3.1. SRv6-ERO Subobject

An SRv6-ERO subobject is formatted as shown in the following figure.

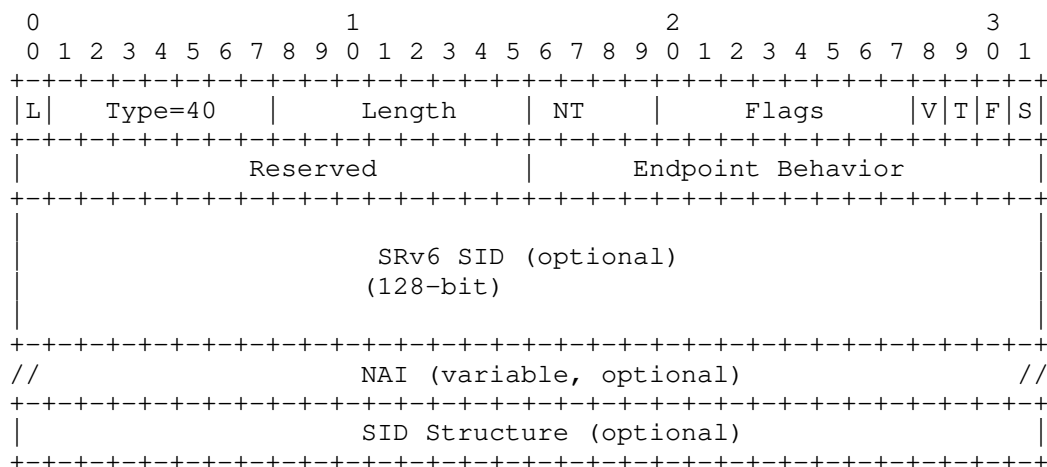


Figure 2: SRv6-ERO Subobject Format

The fields in the SRv6-ERO Subobject are as follows:

The 'L' Flag: Indicates whether the subobject represents a loose-hop (see [RFC3209]). If this flag is set to zero, a PCC MUST NOT overwrite the SID value present in the SRv6-ERO subobject. Otherwise, a PCC MAY expand or replace one or more SID values in the received SRv6-ERO based on its local policy.

Type: indicates the content of the subobject, i.e. when the field is set to 40 (early allocated by IANA), the subobject is a SRv6-ERO subobject representing a SRv6 SID.

Length: Contains the total length of the subobject in octets. The Length MUST be at least 24, and MUST be a multiple of 4. An SRv6-ERO subobject MUST contain at least one of a SRv6-SID or an NAI. The S and F bit in the Flags field indicates whether the SRv6-SID or NAI fields are absent.

NAI Type (NT): Indicates the type and format of the NAI contained in the object body, if any is present. If the F bit is set to one (see below) then the NT field has no meaning and MUST be ignored by the receiver. This document reuses NT types defined in [RFC8664]:

If NT value is 0, the NAI MUST NOT be included.

When NT value is 2, the NAI is as per the 'IPv6 Node ID' format defined in [RFC8664], which specifies an IPv6 address. This is used to identify the owner of the SRv6 Identifier. This is optional, as the LOC (the locator portion) of the SRv6 SID serves a similar purpose (when present).

When NT value is 4, the NAI is as per the 'IPv6 Adjacency' format defined in [RFC8664], which specify a pair of IPv6 addresses. This is used to identify the IPv6 Adjacency and used with the SRv6 Adj-SID.

When NT value is 6, the NAI is as per the 'link-local IPv6 addresses' format defined in [RFC8664], which specify a pair of (global IPv6 address, interface ID) tuples. It is used to identify the IPv6 Adjacency and used with the SRv6 Adj-SID.

SR-MPLS specific NT types are not valid in SRv6-ERO.

Flags: Used to carry additional information pertaining to the SRv6-SID. This document defines the following flag bits. The other bits MUST be set to zero by the sender and MUST be ignored by the receiver.

- * S: When this bit is set to 1, the SRv6-SID value in the subobject body is absent. In this case, the PCC is responsible for choosing the SRv6-SID value, e.g., by looking up in the SR-DB using the NAI which, in this case, MUST be present in the subobject. If the S bit is set to 1 then F bit MUST be set to zero.

- * F: When this bit is set to 1, the NAI value in the subobject body is absent. The F bit MUST be set to 1 if NT=0, and otherwise MUST be set to zero. The S and F bits MUST NOT both be set to 1.
- * T: When this bit is set to 1, the SID Structure value in the subobject body is present. The T bit MUST be set to 0 when S bit is set to 1. If the T bit is set when the S bit is set, the T bit MUST be ignored. Thus, the T bit indicates the presence of an optional 8-byte SID Structure when SRv6 SID is included. The SID Structure is defined in Section 4.3.1.1.
- * V: The "SID verification" bit usage is as per Section 5.1 of [I-D.ietf-spring-segment-routing-policy].

Reserved: MUST be set to zero while sending and ignored on receipt.

Endpoint Behavior: A 16 bit field representing the behavior associated with the SRv6 SIDs. This information is optional and plays no role in the fields in SRH imposed on the packet. It could be used for maintainability and diagnostic purpose. If behavior is not known, 0 is used. The list of Endpoint behavior are defined in [RFC8986].

SRv6 SID: SRv6 Identifier is the 128 bit IPv6 addresses representing the SRv6 segment.

NAI: The NAI associated with the SRv6-SID. The NAI's format depends on the value in the NT field, and is described in [RFC8664].

At least one of the SRv6-SID or the NAI MUST be included in the SRv6-ERO subobject, and both MAY be included.

4.3.1.1. SID Structure

The SID Structure is an optional part of the SR-ERO subobject, as described in Section 4.3.1.

[RFC8986] defines an SRv6 SID as consisting of LOC:FUNCT:ARG, where a locator (LOC) is encoded in the L most significant bits of the SID, followed by F bits of function (FUNCT) and A bits of arguments (ARG). A locator may be represented as B:N where B is the SRv6 SID locator block (IPv6 prefix allocated for SRv6 SIDs by the operator) and N is the identifier of the parent node instantiating the SID called locator node.

It is formatted as shown in the following figure.

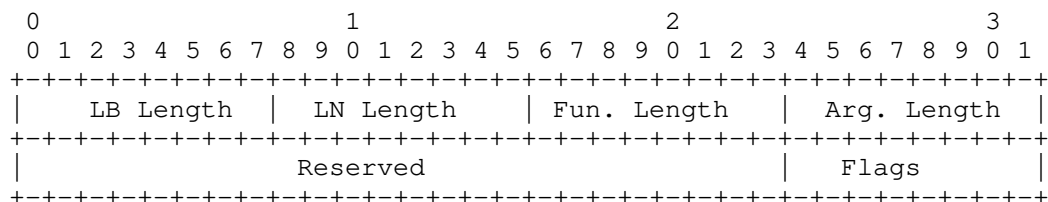


Figure 3: SID Structure Format

where:

LB Length: 1 octet. SRv6 SID Locator Block length in bits.

LN Length: 1 octet. SRv6 SID Locator Node length in bits.

Fun. Length: 1 octet. SRv6 SID Function length in bits.

Arg. Length: 1 octet. SRv6 SID Arguments length in bits.

The sum of all four sizes in the SID Structure must be lower or equal to 128 bits. If the sum of all four sizes advertised in the SID Structure is larger than 128 bits, the corresponding SRv6 SID MUST be considered invalid and a PCERR message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 37 (early allocated by IANA) ("Invalid SRv6 SID Structure") is returned.

Reserved: MUST be set to zero while sending and ignored on receipt.

Flags: Currently no flags are defined. Unassigned bits must be set to zero while sending and ignored on receipt.

The SRv6 SID Structure provides the detailed encoding information of an SRv6 SID, which is useful in the use cases that require to know the SRv6 SID structure. When a PCEP speaker receives the SRv6 SID and its structure information, the SRv6 SID can be parsed based on the SRv6 SID Structure and/or possible local policies. The SRv6 SID Structure could be used by the PCE for ease of operations and monitoring. For example, this information could be used for validation of SRv6 SIDs being instantiated in the network and checked for conformance to the SRv6 SID allocation scheme chosen by the operator as described in Section 3.2 of [RFC8986]. In the future, PCE could also be used for verification and the automation for securing the SRv6 domain by provisioning filtering rules at SR domain boundaries as described in Section 5 of [RFC8754]. The details of these potential applications are outside the scope of this document.

4.3.1.2. Order of the Optional fields

The optional elements in the SRv6-ERO subobject i.e. SRv6 SID, NAI and the SID Structure MUST be encoded in the order as depicted in Figure 2. The presence of each of them is indicated by the respective flags i.e. S flag, F flag and T flag.

To allow for future compatibility, any optional element added to the SRv6-ERO subobject in future MUST specify the order of the optional element and request IANA to allocate a flag to indicate its presence from the subregistry created in Section 9.2.

4.4. RRO

In order to support SRv6, new subobject "SRv6-RRO" is defined in RRO.

4.4.1. SRv6-RRO Subobject

A PCC reports an SRv6 path to a PCE by sending a PCRpt message, per [RFC8231]. The RRO on this message represents the SID list that was applied by the PCC, that is, the actual path taken. The procedures of [RFC8664] with respect to the RRO apply equally to this specification without change.

An RRO contains one or more subobjects called "SRv6-RRO subobjects" whose format is shown below:

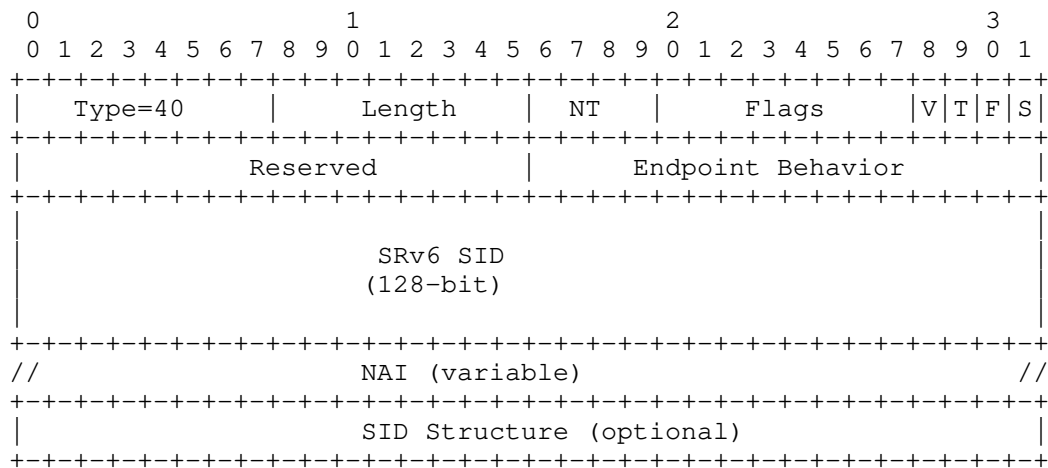


Figure 4: SRv6-RRO Subobject Format

The format of the SRv6-RRO subobject is the same as that of the SRv6-ERO subobject, but without the L flag.

The V flag has no meaning in the SRv6-RRO and is ignored on receipt at the PCE.

Ordering of SRv6-RRO subobjects by PCC in PCRpt message remains as per [RFC8664].

The ordering of optional elements in the SRv6-RRO subobject as same as described in Section 4.3.1.2.

5. Procedures

5.1. Exchanging the SRv6 Capability

A PCC indicates that it is capable of supporting the head-end functions for SRv6 by including the SRv6-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCE. A PCE indicates that it is capable of computing SRv6 paths by including the SRv6-PCE-CAPABILITY sub-TLV in the Open message that it sends to a PCC.

If a PCEP speaker receives a PATH-SETUP-TYPE-CAPABILITY TLV with a PST list containing PST=3 (early allocated by IANA), but the SRv6-PCE-CAPABILITY sub-TLV is absent, then the PCEP speaker MUST send a PCErr message with Error- Type 10 (Reception of an invalid object) and Error-Value 34 (early allocated by IANA) (Missing PCE-SRv6-CAPABILITY sub-TLV) and MUST then close the PCEP session. If a PCEP speaker receives a PATH-SETUP- TYPE-CAPABILITY TLV with a SRv6-PCE-CAPABILITY sub-TLV, but the PST list does not contain PST=3 (early allocated by IANA), then the PCEP speaker MUST ignore the SRv6-PCE-CAPABILITY sub-TLV.

The number of SRv6 SIDs that can be imposed on a packet depends on the PCC's IPv6 data plane's capability. If a PCC sets the X flag to 1 then the MSD is not used and MUST NOT be included. If a PCE receives an SRv6-PCE-CAPABILITY sub-TLV with the X flag set to 1 then it MUST ignore any MSD-Type, MSD-Value fields and MUST assume that the sender can impose any length of SRH. If a PCC sets the X flag to zero, then it sets the SRv6 MSD-Type, MSD-Value fields that it can impose on a packet. If a PCE receives an SRv6-PCE-CAPABILITY sub-TLV with the X flag and SRv6 MSD-Type, MSD-Value fields both set to zero then it is considered as an error and the PCE MUST respond with a PCErr message (Error-Type=1 "PCEP session establishment failure" and Error-Value=1 "reception of an invalid Open message or a non Open message."). In case the MSD-Type in SRv6-PCE-CAPABILITY sub-TLV received by the PCE does not correspond to one of the SRv6 MSD types, the PCE MUST respond with a PCErr message (Error-Type=1 "PCEP session establishment failure" and Error-Value=1 "reception of an invalid Open message or a non Open message.").

Note that the MSD-Type, MSD-Value exchanged via the SRv6-PCE-CAPABILITY sub-TLV indicates the SRv6 SID imposition limit for the PCC node. However, if a PCE learns these via different means, e.g routing protocols, as specified in:
[I-D.ietf-lsr-ospfv3-srv6-extensions];
[I-D.ietf-lsr-isis-srv6-extensions]; [I-D.ietf-idr-bgppls-srv6-ext],
then it ignores the values in the SRv6-PCE-CAPABILITY sub-TLV. Furthermore, whenever a PCE learns the other advanced SRv6 MSD via different means, it MUST use that value regardless of the values exchanged in the SRv6-PCE-CAPABILITY sub-TLV.

Once an SRv6-capable PCEP session is established with a non-zero SRv6 MSD value, the corresponding PCE MUST NOT send SRv6 paths with a number of SIDs exceeding that SRv6 MSD value (based on the SRv6 MSD Type). If a PCC needs to modify the SRv6 MSD value, it MUST close the PCEP session and re-establish it with the new value. If a PCEP session is established with a non-zero SRv6 MSD value, and the PCC receives an SRv6 path containing more SIDs than specified in the SRv6 MSD value (based on the SRv6 MSD type), the PCC MUST send a PCErr message with Error-Type 10 (Reception of an invalid object) and Error-Value 3 (Unsupported number of Segment ERO subobjects). If a PCEP session is established with an SRv6 MSD value of zero, then the PCC MAY specify an SRv6 MSD for each path computation request that it sends to the PCE, by including a "maximum SID depth" metric object on the request similar to [RFC8664].

The N flag, X flag and (MSD-Type,MSD-Value) pair inside the SRv6-PCE-CAPABILITY sub-TLV are meaningful only in the Open message sent from a PCC to a PCE. As such, a PCE MUST set the flags to zero and not include any (MSD-Type,MSD-Value) pair in the SRv6-PCE-CAPABILITY sub-

TLV in an outbound message to a PCC. Similarly, a PCC MUST ignore N,X flag and any (MSD-Type,MSD-Value) pair received from a PCE. If a PCE receives multiple SRv6-PCE-CAPABILITY sub-TLVs in an Open message, it processes only the first sub-TLV received.

5.2. ERO Processing

The ERO processing remains as per [RFC5440] and [RFC8664].

5.2.1. SRv6 ERO Validation

If a PCC does not support the SRv6 PCE Capability and thus cannot recognize the SRv6-ERO or SRv6-RRO subobjects, it will respond according to the rules for a malformed object per [RFC5440].

On receiving an SRv6-ERO, a PCC MUST validate that the Length field, the S bit, the F bit, the T bit, and the NT field are consistent, as follows.

- * If NT=0, the F bit MUST be 1, the S bit MUST be zero and the Length MUST be 24.
- * If NT=2, the F bit MUST be zero. If the S bit is 1, the Length MUST be 24, otherwise the Length MUST be 40.
- * If NT=4, the F bit MUST be zero. If the S bit is 1, the Length MUST be 40, otherwise the Length MUST be 56.
- * If NT=6, the F bit MUST be zero. If the S bit is 1, the Length MUST be 48, otherwise the Length MUST be 64.
- * NT types (1,3, and 5) are not valid for SRv6.
- * If T bit is 1, then S bit MUST be zero.

If a PCC finds that the NT field, Length field, S bit, F bit, and T bit are not consistent, it MUST consider the entire ERO invalid and MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 11 ("Malformed object").

If a PCEP speaker that does not recognize the NT value received in SRv6-ERO subobject, it would behave as per [RFC8664].

In case a PCEP speaker receives the SRv6-ERO subobject, when the PST is not set to 3 (early allocated by IANA) or SRv6-PCE-CAPABILITY sub-TLV was not exchanged, it MUST send a PCErr message with Error-Type = 19 ("Invalid Operation") and Error-Value = 19 (early allocated by IANA) ("Attempted SRv6 when the capability was not advertised").

If a PCC receives a list of SRv6 segments, and the number of SRv6 segments exceeds the SRv6 MSD that the PCC can impose on the packet (SRH), it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 3 ("Unsupported number of SR-ERO subobjects") as per [RFC8664].

When a PCEP speaker detects that all subobjects of ERO are not of type 40 (early allocated by IANA), and if it does not handle such ERO, it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 20 ("Inconsistent SIDs in SR-ERO / SR-RRO subobjects") as per [RFC8664].

5.2.2. Interpreting the SRv6-ERO

The SRv6-ERO contains a sequence of subobjects. According to [I-D.ietf-spring-segment-routing-policy], each SRv6-ERO subobject in the sequence identifies a segment that the traffic will be directed to, in the order given. That is, the first subobject identifies the first segment the traffic will be directed to, the second SRv6-ERO subobject represents the second segment, and so on.

The PCC interprets the SRv6-ERO by converting it to an SRv6 SRH plus a next hop. The PCC sends packets along the segment routed path by prepending the SRH onto the packets and sending the resulting, modified packet to the next hop.

5.3. RRO Processing

The syntax checking rules that apply to the SRv6-RRO subobject are identical to those of the SRv6-ERO subobject, except as noted below.

If a PCEP speaker receives an SRv6-RRO subobject in which both SRv6 SID and NAI are absent, it MUST consider the entire RRO invalid and send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 35 (early allocated by IANA) ("Both SID and NAI are absent in SRv6-RRO subobject").

If a PCE detects that the subobjects of an RRO are a mixture of SRv6-RRO subobjects and subobjects of other types, then it MUST send a PCErr message with Error-Type = 10 ("Reception of an invalid object") and Error-Value = 36 (early allocated by IANA) ("RRO mixes SRv6-RRO subobjects with other subobject types").

6. Security Considerations

The security considerations described in [RFC5440], [RFC8231] and [RFC8281], [RFC8664], are applicable to this specification. No additional security measure is required.

7. Manageability Considerations

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

7.1. Control of Function and Policy

A PCEP implementation SHOULD allow the operator to configure the policy based on which it allocates the SIDs.

7.2. Information and Data Models

The PCEP YANG module is defined in [I-D.ietf-pce-pcep-yang]. An implementation SHOULD allow the operator to view the SIDs allocated to the LSP for the SR path.

7.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

7.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440], [RFC8231], and [RFC8664] .

7.5. Requirements On Other Protocols

Mechanisms defined in this document do not imply any new requirements on other protocols.

7.6. Impact On Network Operations

Mechanisms defined in [RFC5440], [RFC8231], and [RFC8664] also apply to PCEP extensions defined in this document.

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's Commercial Delivery

- * Organization: Cisco Systems, Inc.
- * Implementation: IOS-XR PCE and PCC.
- * Description: Implementation with experimental codepoints.
- * Maturity Level: Production
- * Coverage: Partial
- * Contact: ssidor@cisco.com

9. IANA Considerations

9.1. PCEP ERO and RRO subobjects

This document defines a new subobject type for the PCEP explicit route object (ERO), and a new subobject type for the PCEP record route object (RRO). The code points for subobject types of these objects is maintained in the RSVP parameters registry, under the EXPLICIT_ROUTE and ROUTE_RECORD objects. IANA is requested to confirm the following early allocations in the RSVP Parameters registry for each of the new subobject types defined in this document.

Object	Subobject	Subobject Type
EXPLICIT_ROUTE	SRv6-ERO (PCEP-specific)	40
ROUTE_RECORD	SRv6-RRO (PCEP-specific)	40

9.2. New SRv6-ERO Flag Registry

IANA is requested to create a new sub-registry, named "SRv6-ERO Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the SRv6-ERO subobject. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- * Bit number (counting from bit 0 as the most significant bit)
- * Capability description
- * Defining RFC

The following values are defined in this document:

Bit	Description	Reference
0-7	Unassigned	
8	SID Verification (V)	This document
9	SID Structure is present (T)	This document
10	NAI is absent (F)	This document
11	SID is absent (S)	This document

9.3. PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators

IANA maintains a sub-registry, named "PATH-SETUP-TYPE-CAPABILITY Sub-TLV Type Indicators", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the type indicator space for sub-TLVs of the PATH-SETUP-TYPE-CAPABILITY TLV. IANA is requested to confirm the following early allocations in the sub-registry:

Value	Meaning	Reference
27	SRv6-PCE-CAPABILITY	This Document

9.4. SRv6 PCE Capability Flags

IANA is requested to create a new sub-registry, named "SRv6 Capability Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the SRv6-PCE-CAPABILITY sub-TLV. New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- * Bit number (counting from bit 0 as the most significant bit)
- * Capability description
- * Defining RFC

The following values are defined in this document:

Bit	Description	Reference
0-13	Unassigned	
14	Node or Adjacency Identifier (NAI) is supported (N)	This document
15	Unlimited Maximum SID Depth (X)	This document

9.5. New Path Setup Type

[RFC8408] created a sub-registry within the "Path Computation Element Protocol (PCEP) Numbers" registry called "PCEP Path Setup Types". IANA is requested to confirm the following early allocations in the sub-registry:

Value	Description	Reference
-----	-----	-----
3	Traffic engineering path is setup using SRv6.	This Document

9.6. ERROR Objects

IANA is requested to confirm the following early allocations in the PCEP-ERROR Object Error Types and Values registry for the following new error-values:

Error-Type	Meaning
-----	-----
10	Reception of an invalid object Error-value = 34 (Missing PCE-SRv6-CAPABILITY sub-TLV) Error-value = 35 (Both SID and NAI are absent in SRv6-RRO subobject) Error-value = 36 (RRO mixes SRv6-RRO subobjects with other subobject types) Error-value = 37 (Invalid SRv6 SID Structure)
19	Invalid Operation Error-value = 19 (Attempted SRv6 when the capability was not advertised)

10. Acknowledgements

The authors would like to thank Jeff Tantsura, Adrian Farrel, Aijun Wang, Khasanov Boris, and Robert Varga for valuable suggestions.

11. References

11.1. Normative References

- [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>>.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.
- [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>>.
- [RFC5511] Farrel, A., "Routing Backus-Naur Form (RBNF): A Syntax Used to Form Encoding Rules in Various Routing Protocol Specifications", RFC 5511, DOI 10.17487/RFC5511, April 2009, <<https://www.rfc-editor.org/info/rfc5511>>.
- [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>>.

- [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>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491, DOI 10.17487/RFC8491, November 2018, <<https://www.rfc-editor.org/info/rfc8491>>.
- [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>>.
- [RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <<https://www.rfc-editor.org/info/rfc8986>>.

[I-D.ietf-lsr-isis-srv6-extensions]

Psenak, P., Filsfils, C., Bashandy, A., Decraene, B., and Z. Hu, "IS-IS Extensions to Support Segment Routing over IPv6 Dataplane", Work in Progress, Internet-Draft, draft-ietf-lsr-isis-srv6-extensions-18, 20 October 2021, <<https://www.ietf.org/archive/id/draft-ietf-lsr-isis-srv6-extensions-18.txt>>.

11.2. Informative References

- [RFC4657] Ash, J., Ed. and J.L. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <<https://www.rfc-editor.org/info/rfc4657>>.
- [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>>.
- [RFC7855] Previdi, S., Ed., Filsfils, C., Ed., Decraene, B., Litkowski, S., Horneffer, M., and R. Shakir, "Source Packet Routing in Networking (SPRING) Problem Statement and Requirements", RFC 7855, DOI 10.17487/RFC7855, May 2016, <<https://www.rfc-editor.org/info/rfc7855>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8354] Brzozowski, J., Leddy, J., Filsfils, C., Maglione, R., Ed., and M. Townsley, "Use Cases for IPv6 Source Packet Routing in Networking (SPRING)", RFC 8354, DOI 10.17487/RFC8354, March 2018, <<https://www.rfc-editor.org/info/rfc8354>>.
- [RFC8666] Psenak, P., Ed. and S. Previdi, Ed., "OSPFv3 Extensions for Segment Routing", RFC 8666, DOI 10.17487/RFC8666, December 2019, <<https://www.rfc-editor.org/info/rfc8666>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.

- [RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <<https://www.rfc-editor.org/info/rfc8754>>.
- [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.ietf-lsr-ospfv3-srv6-extensions] Li, Z., Hu, Z., Cheng, D., Talaulikar, K., and P. Psenak, "OSPFv3 Extensions for SRv6", Work in Progress, Internet-Draft, draft-ietf-lsr-ospfv3-srv6-extensions-03, 19 November 2021, <<https://www.ietf.org/archive/id/draft-ietf-lsr-ospfv3-srv6-extensions-03.txt>>.
- [I-D.ietf-idr-bgppls-srv6-ext] Dawra, G., Filsfils, C., Talaulikar, K., Chen, M., Bernier, D., and B. Decraene, "BGP Link State Extensions for SRv6", Work in Progress, Internet-Draft, draft-ietf-idr-bgppls-srv6-ext-09, 10 November 2021, <<https://www.ietf.org/archive/id/draft-ietf-idr-bgppls-srv6-ext-09.txt>>.
- [I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V. P., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-yang-18, 25 January 2022, <<https://www.ietf.org/archive/id/draft-ietf-pce-pcep-yang-18.txt>>.

Appendix A. Contributor

The following persons contributed to this document:

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Huang Wumin
Huawei Technologies
Huawei Building, No. 156 Beiqing Rd.
Beijing 100095
China

Email: huangwumin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Building, No. 156 Beiqing Rd.
Beijing 100095
China

Email: pengshuping@huawei.com

Authors' Addresses

Cheng Li(Editor)
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing
100095
China
Email: c.l@huawei.com

Mahendra Singh Negi
RtBrick Inc
Bangalore
Karnataka
India
Email: mahend.ietf@gmail.com

Siva Sivabalan
Ciena Corporation
Email: msiva282@gmail.com

Mike Koldychev
Cisco Systems, Inc.
Canada
Email: mkoldych@cisco.com

Prejeeth Kaladharan
RtBrick Inc
Bangalore
Karnataka
India
Email: prejeeth@rtbrick.com

Yongqing Zhu
China Telecom
109 West Zhongshan Ave, Tianhe District
Bangalore
Guangzhou,
P.R. China
Email: zhuyq8@chinatelecom.cn

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2021

S. Peng
C. Li
Huawei Technologies
L. Han
China Mobile
L. Ndifor
MTN Cameroon
October 31, 2020

Support for Path MTU (PMTU) in the Path Computation Element (PCE)
communication Protocol (PCEP).
draft-li-pce-pcep-pmtu-03

Abstract

The Path Computation Element (PCE) provides path computation functions in support of traffic engineering in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks.

The Source Packet Routing in Networking (SPRING) architecture describes how Segment Routing (SR) can be used to steer packets through an IPv6 or MPLS network using the source routing paradigm. A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE).

Since the SR does not require signaling, the path maximum transmission unit (MTU) information for SR path is not available. This document specifies the extension to PCE communication protocol (PCEP) to carry path (MTU) in the PCEP messages.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 May 4, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	4
3. PCEP Extention	5
3.1. Extensions to METRIC Object	5
3.2. Stateful PCE and PCE Initiated LSPs	6
3.3. Segment Routing	7
3.4. Path MTU Adjustment	7
4. Security Considerations	7
5. IANA Considerations	8
5.1. METRIC Type	8
6. Acknowledgement	8
7. References	8
7.1. Normative References	8
7.2. Informative References	9
Authors' Addresses	10

1. Introduction

[RFC5440] describes the Path Computation Element (PCE) Communication Protocol (PCEP). PCEP enables the communication between a Path Computation Client (PCC) and a PCE, or between PCE and PCE, for the

purpose of computation of Multiprotocol Label Switching (MPLS) as well as Generalized MPLS (GMPLS) Traffic Engineering Label Switched Path (TE LSP) characteristics.

[RFC8231] specifies a set of extensions to PCEP to enable stateful control of TE LSPs within and across PCEP sessions in compliance with [RFC4657]. It includes mechanisms to effect LSP State Synchronization between PCCs and PCEs, delegation of control over LSPs to PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions. The model of operation where LSPs are initiated from the PCE is described in [RFC8281].

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based. A segment can have a semantic local to an SR node or global within an SR domain. SR allows to enforce a flow through any path and service chain while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locators, etc. The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). The SR is applied to IPV6 forwarding plane using SRH. A SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node.

As per [RFC8664], it is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [RFC8281] using the SR specific PCEP extensions specified in [RFC8664]. [RFC8664] specifies PCEP extensions for supporting a SR-TE LSP for MPLS data plane. [I-D.ietf-pce-segment-routing-ipv6] extend PCEP to support SR for IPv6 data plane.

The maximum transmission unit (MTU) is the largest size packet or frame, in bytes, that can be sent in a network. An MTU that is too large might cause retransmissions. Too small an MTU might cause the router to send and handle relatively more header overhead and acknowledgments. When an LSP is created across a set of links with different MTU sizes, the ingress router need to know what the smallest MTU is on the LSP path. If this MTU is larger than the MTU of one of the intermediate links, traffic might be dropped, because MPLS packets cannot be fragmented. Also, the ingress router may not be aware of this type of traffic loss, because the control plane for the LSP would still function normally. [RFC3209] specify the mechanism of MTU signaling in RSVP.

Since the SR does not require signaling, the path MTU information for SR path is not available. This document specifies the extension to PCEP to carry path MTU in the PCEP messages. It is assumed that the PCE is aware of the link MTU as part of the Traffic Engineering Database (TED) population. This could be done via IGP, BGP-LS or some other means. Thus the PCE can find the path MTU at the time of path computation and include this information as part of the PCEP messages.

Though the key use case for path MTU is SR, the PCEP extension (as specified in this document) creates a new metric type for path MTU, making this a generic extension that can be used independent of SR.

2. Terminology

This draft refers to the terms defined in [RFC8201], [RFC4821] and [RFC3988].

MTU: Maximum Transmission Unit, the size in bytes of the largest IP packet, including the IP header and payload, that can be transmitted on a link or path. Note that this could more properly be called the IP MTU, to be consistent with how other standards organizations use the acronym MTU.

Link MTU: The Maximum Transmission Unit, i.e., maximum IP packet size in bytes, that can be conveyed in one piece over a link. Be aware that this definition is different from the definition used by other standards organizations.

For IETF documents, link MTU is uniformly defined as the IP MTU over the link. This includes the IP header, but excludes link layer headers and other framing that is not part of IP or the IP payload.

Be aware that other standards organizations generally define link MTU to include the link layer headers.

For the MPLS data plane, this size includes the IP header and data (or other payload) and the label stack but does not include any lower-layer headers. A link may be an interface (such as Ethernet or Packet-over-SONET), a tunnel (such as GRE or IPsec), or an LSP.

Path: The set of links traversed by a packet between a source node and a destination node.

Path MTU, or PMTU: The minimum link MTU of all the links in a path between a source node and a destination node.

For the MPLS data plane, it is the MTU of an LSP from a given LSR to the egress(es), over each valid (forwarding) path. This size includes the IP header and data (or other payload) and any part of the label stack that was received by the ingress LSR before it placed the packet into the LSP (this part of the label stack is considered part of the payload for this LSP). The size does not include any lower-level headers.

3. PCEP Extension

3.1. Extensions to METRIC Object

The METRIC object is defined in Section 7.8 of [RFC5440], comprising metric-value and metric-type (T field), and a flags field, comprising a number of bit flags (B bit and C bit). This document defines a new type for the METRIC object for Path MTU.

- o T = TBD: Path MTU.
- o A network comprises of a set of N links $\{L_i, (i=1\dots N)\}$.
- o A path P of a LSP is a list of K links $\{L_{pi}, (i=1\dots K)\}$.
- o A Link MTU of link L is denoted $M(L)$.
- o A Path MTU metric for the path P = $\text{Min } \{M(L_{pi}), (i=1\dots K)\}$.

The Path MTU metric type of the METRIC object in PCEP represents the minimum of the Link MTU of all links along the path.

When PCE computes the path, it can also find the Path MTU (based on the above criteria) and include this information in the METRIC object with the above metric type in the PCEP message when replying to the PCC. In a Path Computation Reply (PCRep) message, the PCE MAY insert the METRIC object with an Explicit Route Object (ERO) so as to provide the METRIC (path MTU) for the computed path. The PCE MAY also insert the METRIC object with a NO-PATH object to indicate that the metric constraint could not be satisfied.

Further, a PCC MAY use the Path MTU metric in a Path Computation Request (PCReq) message to request a path meeting the MTU requirement of the path. In this case, the B bit MUST be set to suggest a bound (a maximum) for the Path MTU metric that must not be exceeded for the PCC to consider the computed path as acceptable. The Path MTU metric must be less than or equal to the value specified in the metric-value field.

A PCC can also use this metric to ask PCE to optimize the path MTU during path computation. In this case, the B bit MUST be cleared.

The error handling and processing of the METRIC object is as specified in [RFC5440].

3.2. Stateful PCE and PCE Initiated LSPs

[RFC8231] specifies a set of extensions to PCEP to enable stateful control of MPLS-TE and GMPLS LSPs via PCEP and the maintaining of these LSPs at the stateful PCE. It further distinguishes between an active and a passive stateful PCE. A passive stateful PCE uses LSP state information learned from PCCs to optimize path computations but does not actively update LSP state. In contrast, an active stateful PCE utilizes the LSP delegation mechanism to update LSP parameters in those PCCs that delegated control over their LSPs to the PCE. [RFC8281] describes the setup, maintenance, and teardown of PCE-initiated LSPs under the stateful PCE model. The document defines

the PCInitiate message that is used by a PCE to request a PCC to set up a new LSP.

The new metric type defined in this document can also be used with the stateful PCE extensions. The format of PCEP messages described in [RFC8231] and [RFC8281] uses <intended-attribute-list> and <attribute-list>, respectively, (where the <intended-attribute-list> is the attribute-list defined in Section 6.5 of [RFC5440]).

A PCE MAY include the path MTU metric in PCInitiate or PCUpd message to inform the PCC of the path MTU calculated for the path. A PCC MAY include the path MTU metric as a bound constraint or to indicate optimization criteria (similar to PCReq).

3.3. Segment Routing

A Segment Routed path (SR path) can be derived from an IGP Shortest Path Tree (SPT). Segment Routed Traffic Engineering paths (SR-TE paths) may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool and provisioned on the source node of the SR-TE path.

It is possible to use a PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the PCE can inform an SR-TE path on a PCC using PCEP extensions specified in [RFC8664]. Further, [I-D.ietf-pce-segment-routing-ipv6] adds the support for IPv6 data plane in SR.

The new metric type for path MTU is applicable for the SR-TE path and require no additional extensions.

3.4. Path MTU Adjustment

The path MTU metric can be used for both primary and protection path.

The minimal value of the link MTU along the path is collected, based on which minor adjustment is made to cater for overhead introduced by the protection mechanisms such as TI-LFA. The path MTU is the value of the minimum link MTU minus the overhead. In this way, the ingress node can use the path MTU directly.

4. Security Considerations

This document defines a new METRIC type that do not add any new security concerns beyond those discussed in [RFC5440] in itself. Some deployments may find the path MTU information to be extra sensitive and could be used to influence path computation and setup

with adverse effect. Additionally, snooping of PCEP messages with such data or using PCEP messages for network reconnaissance may give an attacker sensitive information about the operations of the network. Thus, such deployment should employ suitable PCEP security mechanisms like TCP Authentication Option (TCP-AO) [RFC5925] or Transport Layer Security (TLS) [RFC8253]. The procedure based on TLS is considered a security enhancement and thus is much better suited for the sensitive information.

5. IANA Considerations

This document makes following requests to IANA for action.

5.1. METRIC Type

IANA maintains the "Path Computation Element Protocol (PCEP) Numbers" registry. Within this registry, IANA maintains a subregistry for "METRIC Object T Field". IANA is requested to make the following allocation:

Value	Description	Reference
TBD	Path MTU	This document

6. Acknowledgement

We would like to thank Dhruv Dhody for his contributions for this document.

7. References

7.1. Normative References

- [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>>.
- [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>>.

7.2. Informative References

- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.
- [RFC4657] Ash, J., Ed. and J. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <<https://www.rfc-editor.org/info/rfc4657>>.
- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <<https://www.rfc-editor.org/info/rfc5925>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [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>>.

[I-D.ietf-pce-segment-routing-ipv6]

Li, C., Negi, M., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", draft-ietf-pce-segment-routing-ipv6-06 (work in progress), July 2020.

Authors' Addresses

Shuping Peng
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: pengshuping@huawei.com

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: c.l@huawei.com

Liuyan Han
China Mobile
Beijing 100053
China

Email: hanliuyan@chinamobile.com

Luc-Fabrice Ndifor
MTN Cameroon
Cameroon

Email: Luc-Fabrice.Ndifor@mtn.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 24 April 2022

S. Peng
C. Li
Huawei Technologies
L. Han
China Mobile
L. Ndifor
MTN Cameroon
21 October 2021

Support for Path MTU (PMTU) in the Path Computation Element (PCE)
communication Protocol (PCEP).
draft-li-pce-pcep-pmtu-05

Abstract

The Path Computation Element (PCE) provides path computation functions in support of traffic engineering in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks.

The Source Packet Routing in Networking (SPRING) architecture describes how Segment Routing (SR) can be used to steer packets through an IPv6 or MPLS network using the source routing paradigm. A Segment Routed Path can be derived from a variety of mechanisms, including an IGP Shortest Path Tree (SPT), explicit configuration, or a Path Computation Element (PCE).

Since the SR does not require signaling, the path maximum transmission unit (MTU) information for SR path is not available. This document specifies the extension to PCE communication protocol (PCEP) to carry path (MTU) in the PCEP messages.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 24 April 2022.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	4
3. PCEP Extension	5
3.1. Extensions to METRIC Object	5
3.2. Multi-Path Handling	6
3.3. Stateful PCE and PCE Initiated LSPs	7
3.4. Segment Routing	7
3.5. Path MTU Adjustment	7
4. Security Considerations	8
5. IANA Considerations	8
5.1. METRIC Type	8
6. Acknowledgement	8
7. References	8
7.1. Normative References	8
7.2. Informative References	9
Authors' Addresses	11

1. Introduction

[RFC5440] describes the Path Computation Element (PCE) Communication Protocol (PCEP). PCEP enables the communication between a Path Computation Client (PCC) and a PCE, or between PCE and PCE, for the purpose of computation of Multiprotocol Label Switching (MPLS) as well as Generalized MPLS (GMPLS) Traffic Engineering Label Switched Path (TE LSP) characteristics.

[RFC8231] specifies a set of extensions to PCEP to enable stateful control of TE LSPs within and across PCEP sessions in compliance with [RFC4657]. It includes mechanisms to effect LSP State Synchronization between PCCs and PCEs, delegation of control over LSPs to PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions. The model of operation where LSPs are initiated from the PCE is described in [RFC8281].

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. A segment can represent any instruction, topological or service-based. A segment can have a semantic local to an SR node or global within an SR domain. SR allows to enforce a flow through any path and service chain while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locators, etc. The SR architecture can be applied to the MPLS forwarding plane without any change, in which case an SR path corresponds to an MPLS Label Switching Path (LSP). The SR is applied to IPv6 forwarding plane using SRH. A SR path can be derived from an IGP Shortest Path Tree (SPT), but SR-TE paths may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool, or a PCE and provisioned on the ingress node.

As per [RFC8664], it is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can initiate an SR-TE path on a PCC using PCEP extensions specified in [RFC8281] using the SR specific PCEP extensions specified in [RFC8664]. [RFC8664] specifies PCEP extensions for supporting a SR-TE LSP for MPLS data plane. [I-D.ietf-pce-segment-routing-ipv6] extend PCEP to support SR for IPv6 data plane.

The maximum transmission unit (MTU) is the largest size packet or frame, in bytes, that can be sent in a network. An MTU that is too large might cause retransmissions. Too small an MTU might cause the router to send and handle relatively more header overhead and acknowledgments. When an LSP is created across a set of links with different MTU sizes, the ingress router needs to know what the smallest MTU is on the LSP path. If this MTU is larger than the MTU

of one of the intermediate links, traffic might be dropped, because MPLS packets cannot be fragmented. Also, the ingress router may not be aware of this type of traffic loss, because the control plane for the LSP would still function normally. [RFC3209] specify the mechanism of MTU signaling in RSVP.

Since the SR does not require signaling, the path MTU information for SR path is not available. This document specify the extension to PCEP to carry path MTU in the PCEP messages. It is assumed that the PCE is aware of the link MTU as part of the Traffic Engineering Database (TED) population. This could be done via IGP, BGP-LS [I-D.ietf-idr-bgp-ls-link-mtu] or some other means. Thus the PCE can find the path MTU at the time of path computation and include this information as part of the PCEP messages.

Though the key use case for path MTU is SR, the PCEP extension (as specified in this document) creates a new metric type for path MTU, making this a generic extension that can be used independent of SR.

Note that in SR, the term Maximum SID Depth (MSD) [RFC8491] refers to the maximum number of SIDs that an ingress is capable of imposing on a packet. The PMTU on the other hand determines if the IP fragmentation could be avoided.

2. Terminology

This draft refers to the terms defined in [RFC8201], [RFC4821] and [RFC3988].

MTU: Maximum Transmission Unit, the size in bytes of the largest IP packet, including the IP header and payload, that can be transmitted on a link or path. Note that this could more properly be called the IP MTU, to be consistent with how other standards organizations use the acronym MTU.

Link MTU: The Maximum Transmission Unit, i.e., maximum IP packet size in bytes, that can be conveyed in one piece over a link. Be aware that this definition is different from the definition used by other standards organizations.

For IETF documents, link MTU is uniformly defined as the IP MTU over the link. This includes the IP header, but excludes link layer headers and other framing that is not part of IP or the IP payload.

Be aware that other standards organizations generally define link MTU to include the link layer headers.

For the MPLS data plane, this size includes the IP header and data (or other payload) and the label stack but does not include any lower-layer headers. A link may be an interface (such as Ethernet or Packet-over-SONET), a tunnel (such as GRE or IPsec), or an LSP.

Path: The set of links traversed by a packet between a source node and a destination node.

Path MTU, or PMTU: The minimum link MTU of all the links in a path between a source node and a destination node.

For the MPLS data plane, it is the MTU of an LSP from a given LSR to the egress(es), over each valid (forwarding) path. This size includes the IP header and data (or other payload) and any part of the label stack that was received by the ingress LSR before it placed the packet into the LSP (this part of the label stack is considered part of the payload for this LSP). The size does not include any lower-level headers.

3. PCEP Extention

3.1. Extensions to METRIC Object

The METRIC object is defined in Section 7.8 of [RFC5440], comprising metric-value and metric-type (T field), and a flags field, comprising a number of bit flags (B bit and C bit). This document defines a new type for the METRIC object for Path MTU.

* T = TBD: Path MTU.

- * A network comprises of a set of N links $\{L_i, (i=1...N)\}$.
- * A path P of a LSP is a list of K links $\{L_{pi}, (i=1...K)\}$.
- * A Link MTU of link L is denoted $M(L)$.
- * A Path MTU metric for the path $P = \text{Min } \{M(L_{pi}), (i=1...K)\}$.

The Path MTU metric type of the METRIC object in PCEP represents the minimum of the Link MTU of all links along the path.

When PCE computes the path, it can also find the Path MTU (based on the above criteria) and include this information in the METRIC object with the above metric type in the PCEP message when replying to the PCC. In a Path Computation Reply (PCRep) message, the PCE MAY insert the METRIC object with an Explicit Route Object (ERO) so as to provide the METRIC (path MTU) for the computed path. The PCE MAY also insert the METRIC object with a NO-PATH object to indicate that the metric constraint could not be satisfied.

Further, a PCC MAY use the Path MTU metric in a Path Computation Request (PCReq) message to request a path meeting the MTU requirement of the path. In this case, the B bit MUST be set to suggest a bound (a maximum) for the Path MTU metric that must not be exceeded for the PCC to consider the computed path as acceptable. The Path MTU metric must be less than or equal to the value specified in the metric-value field.

A PCC can also use this metric to ask PCE to optimize the path MTU during path computation. In this case, the B bit MUST be cleared.

The error handling and processing of the METRIC object is as specified in [RFC5440].

3.2. Multi-Path Handling

[I-D.ietf-pce-multipath] extends PCEP to support signaling of multipath information i.e. to allow each Candidate-Path to contain multiple Segment-Lists.

The PMTU could be supported per segment list as well. The exact mechanism to support this is left for further revision of this document.

3.3. Stateful PCE and PCE Initiated LSPs

[RFC8231] specifies a set of extensions to PCEP to enable stateful control of MPLS-TE LSPs via PCEP and the maintaining of these LSPs at the stateful PCE. It further distinguishes between an active and a passive stateful PCE. A passive stateful PCE uses LSP state information learned from PCCs to optimize path computations but does not actively update LSP state. In contrast, an active stateful PCE utilizes the LSP delegation mechanism to update LSP parameters in those PCCs that delegated control over their LSPs to the PCE. [RFC8281] describes the setup, maintenance, and teardown of PCE-initiated LSPs under the stateful PCE model. The document defines the PCInitiate message that is used by a PCE to request a PCC to set up a new LSP.

The new metric type defined in this document can also be used with the stateful PCE extensions. The format of PCEP messages described in [RFC8231] and [RFC8281] uses <intended-attribute-list> and <attribute-list>, respectively, (where the <intended-attribute-list> is the attribute-list defined in Section 6.5 of [RFC5440]).

A PCE MAY include the path MTU metric in PCInitiate or PCUpd message to inform the PCC of the path MTU calculated for the path. A PCC MAY include the path MTU metric as a bound constraint or to indicate optimization criteria (similar to PCReq).

3.4. Segment Routing

A Segment Routed path (SR path) can be derived from an IGP Shortest Path Tree (SPT). Segment Routed Traffic Engineering paths (SR-TE paths) may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool and provisioned on the source node of the SR-TE path.

It is possible to use a PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the PCE can inform an SR-TE path on a PCC using PCEP extensions specified in [RFC8664]. Further, [I-D.ietf-pce-segment-routing-ipv6] adds the support for IPv6 data plane in SR.

The new metric type for path MTU is applicable for the SR-TE path and require no additional extensions.

3.5. Path MTU Adjustment

The path MTU metric can be used for both primary and protection path.

The minimal value of the link MTU along the path is collected, based on which minor adjustment is made to cater for overhead introduced by the protection mechanisms such as TI-LFA. The path MTU is the value of the minimum link MTU minus the overhead. In this way, the ingress node can use the path MTU directly.

4. Security Considerations

This document defines a new METRIC type that do not add any new security concerns beyond those discussed in [RFC5440] in itself. Some deployments may find the path MTU information to be extra sensitive and could be used to influence path computation and setup with adverse effect. Additionally, snooping of PCEP messages with such data or using PCEP messages for network reconnaissance may give an attacker sensitive information about the operations of the network. Thus, such deployment should employ suitable PCEP security mechanisms like TCP Authentication Option (TCP-AO) [RFC5925] or Transport Layer Security (TLS) [RFC8253]. The procedure based on TLS is considered a security enhancement and thus is much better suited for the sensitive information.

5. IANA Considerations

This document makes following requests to IANA for action.

5.1. METRIC Type

IANA maintains the "Path Computation Element Protocol (PCEP) Numbers" registry. Within this registry, IANA maintains a subregistry for "METRIC Object T Field". IANA is requested to make the following allocation:

Value	Description	Reference
TBD	Path MTU	This document

6. Acknowledgement

We would like to thank Dhruv Dhody for his contributions for this document.

7. References

7.1. Normative References

- [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>>.
- [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>>.

7.2. Informative References

- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <<https://www.rfc-editor.org/info/rfc3209>>.
- [RFC3988] Black, B. and K. Kompella, "Maximum Transmission Unit Signalling Extensions for the Label Distribution Protocol", RFC 3988, DOI 10.17487/RFC3988, January 2005, <<https://www.rfc-editor.org/info/rfc3988>>.
- [RFC4657] Ash, J., Ed. and J.L. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, DOI 10.17487/RFC4657, September 2006, <<https://www.rfc-editor.org/info/rfc4657>>.
- [RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007, <<https://www.rfc-editor.org/info/rfc4821>>.

- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <<https://www.rfc-editor.org/info/rfc5925>>.
- [RFC8201] McCann, J., Deering, S., Mogul, J., and R. Hinden, Ed., "Path MTU Discovery for IP version 6", STD 87, RFC 8201, DOI 10.17487/RFC8201, July 2017, <<https://www.rfc-editor.org/info/rfc8201>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8402] Filts, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8491] Tantsura, J., Chunduri, U., Aldrin, S., and L. Ginsberg, "Signaling Maximum SID Depth (MSD) Using IS-IS", RFC 8491, DOI 10.17487/RFC8491, November 2018, <<https://www.rfc-editor.org/info/rfc8491>>.
- [RFC8664] Sivabalan, S., Filts, 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>>.
- [I-D.ietf-pce-multipath]
Koldychev, M., Sivabalan, S., Saad, T., Beeram, V. P., Bidgoli, H., Yadav, B., and S. Peng, "PCEP Extensions for Signaling Multipath Information", Work in Progress, Internet-Draft, draft-ietf-pce-multipath-02, 17 October 2021, <<https://www.ietf.org/archive/id/draft-ietf-pce-multipath-02.txt>>.
- [I-D.ietf-pce-segment-routing-ipv6]
Li, C., Negi, M., Sivabalan, S., Koldychev, M., Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment Routing leveraging the IPv6 data plane", Work in Progress, Internet-Draft, draft-ietf-pce-segment-routing-ipv6-09, 27 May 2021, <<https://www.ietf.org/internet-drafts/draft-ietf-pce-segment-routing-ipv6-09.txt>>.

[I-D.ietf-idr-bgp-ls-link-mtu]

Zhu, Y., Hu, Z., Peng, S., and R. Mwehaire, "Signaling Maximum Transmission Unit (MTU) using BGP-LS", Work in Progress, Internet-Draft, draft-ietf-idr-bgp-ls-link-mtu-01, 25 May 2021, <<https://www.ietf.org/archive/id/draft-ietf-idr-bgp-ls-link-mtu-01.txt>>.

Authors' Addresses

Shuping Peng
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing
100095
China

Email: pengshuping@huawei.com

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing
100095
China

Email: c.l@huawei.com

Liuyan Han
China Mobile
Beijing
100053
China

Email: hanliuyan@chinamobile.com

Luc-Fabrice Ndifor
MTN Cameroon
Cameroon

Email: Luc-Fabrice.Ndifor@mtn.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 5, 2021

S. Litkowski
Cisco
S. Sivabalan
Ciena Corporation
C. Li
H. Zheng
Huawei Technologies
November 1, 2020

Inter Stateful Path Computation Element (PCE) Communication Procedures.
draft-litkowski-pce-state-sync-09

Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computation in response to a Path Computation Client (PCC) request. The Stateful PCE extensions allow stateful control of Multi-Protocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs) using PCEP.

A Path Computation Client (PCC) can synchronize an LSP state information to a Stateful Path Computation Element (PCE). The stateful PCE extension allows a redundancy scenario where a PCC can have redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control of a LSP to only a single PCE, and only one PCE is responsible for path computation for this delegated LSP.

There are some use cases, where an inter-PCE stateful communication can bring additional resiliency in the design, for instance when some PCC-PCE session fails. The inter-PCE stateful communication may also provide a faster update of the LSP states when such an event occurs. Finally, when, in a redundant PCE scenario, there is a need to compute a set of paths that are part of a group (so there is a dependency between the paths), there may be some cases where the computation of all paths in the group is not handled by the same PCE: this situation is called a split-brain. This split-brain scenario may lead to computation loops between PCEs or suboptimal path computation.

This document describes the procedures to allow a stateful communication between PCEs for various use-cases and also the procedures to prevent computations loops.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 May 5, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction and Problem Statement	3
1.1. Reporting LSP Changes	4
1.2. Split-Brain	5
1.3. Applicability to H-PCE	12
2. Proposed solution	12
2.1. State-sync session	12

2.2. Primary/Secondary relationship between PCE	14
3. Procedures and Protocol Extensions	14
3.1. Opening a state-sync session	14
3.1.1. Capability Advertisement	14
3.2. State synchronization	15
3.3. Incremental updates and report forwarding rules	16
3.4. Maintaining LSP states from different sources	17
3.5. Computation priority between PCEs and sub-delegation	18
3.6. Passive stateful procedures	19
3.7. PCE initiation procedures	20
4. Examples	20
4.1. Example 1	20
4.2. Example 2	22
4.3. Example 3	24
5. Using Primary/Secondary Computation and State-sync Sessions to increase Scaling	25
6. PCEP-PATH-VECTOR TLV	27
7. Security Considerations	28
8. Acknowledgements	28
9. IANA Considerations	28
9.1. PCEP-Error Object	28
9.2. PCEP TLV Type Indicators	29
9.3. STATEFUL-PCE-CAPABILITY TLV	29
10. References	29
10.1. Normative References	29
10.2. Informative References	30
Appendix A. Contributors	31
Authors' Addresses	31

1. Introduction and Problem Statement

The Path Computation Element communication Protocol (PCEP) [RFC5440] provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients' (PCCs) requests.

A stateful PCE [RFC8231] is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

The examples in this section are for illustrative purpose to showcase the need for inter-PCE stateful PCEP sessions.

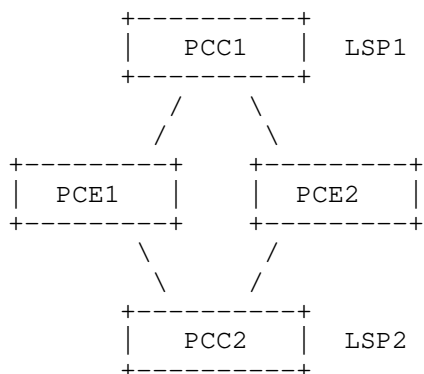
1.1. Reporting LSP Changes

When using a stateful PCE ([RFC8231]), a PCC can synchronize an LSP state information to the stateful PCE. If the PCC grants the control of the LSP to the PCE (called delegation [RFC8231]), the PCE can update the LSP parameters at any time.

In a multi PCE deployment (redundancy, loadbalancing...), with the current specification defined in [RFC8231], when a PCE makes an update, it is the PCC that is in charge of reporting the LSP status to all PCEs with LSP parameter change which brings additional hops and delays in notifying the overall network of the LSP parameter change.

This delay may affect the reaction time of the other PCEs if they need to take action after being notified of the LSP parameter change.

Apart from the synchronization from the PCC, it is also useful if there is a synchronization mechanism between the stateful PCEs. As stateful PCE make changes to its delegated LSPs, these changes (pending LSPs and the sticky resources [RFC7399]) can be synchronized immediately to the other PCEs.



In the figure above, we consider a load-balanced PCE architecture, so PCE1 is responsible to compute paths for PCC1 and PCE2 is responsible to compute paths for PCC2. When PCE1 triggers an LSP update for LSP1, it sends a PCUpd message to PCC1 containing the new parameters for LSP1. PCC1 will take the parameters into account and will send a PCRppt message to PCE1 and PCE2 reflecting the changes. PCE2 will so

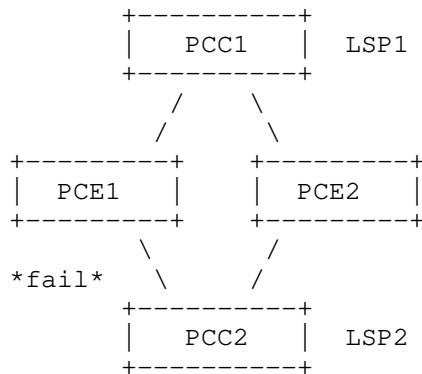
be notified of the change only after receiving the PCRpt message from PCC1.

Let's consider that the LSP1 parameters changed in such a way that LSP1 will take over resources from LSP2 with a higher priority. After receiving the report from PCC1, PCE2 will therefore try to find a new path for LSP2. If we consider that there is a round trip delay of about 150 milliseconds (ms) between the PCEs and PCC1 and a round trip delay of 10 ms between the two PCEs it will take more than 150 ms for PCE2 to be notified of the change.

Adding a PCEP session between PCE1 and PCE2 may allow to reduce the synchronization time, so PCE2 can react more quickly by taking the pending LSPs and attached resources into account during path computation and re-optimization.

1.2. Split-Brain

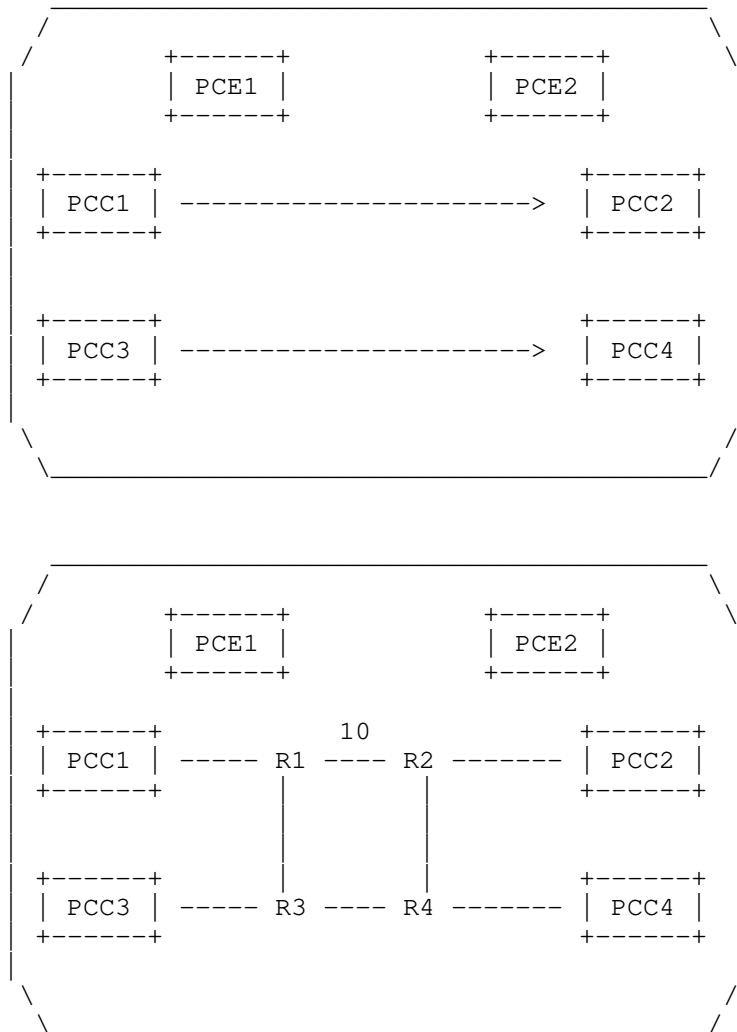
In a resiliency case, a PCC has redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control on an LSP to a single PCE only, and only this PCE is responsible for the path computation for the delegated LSP: the PCC achieves this by setting the D flag only towards the active PCE [RFC8231] selected for delegation. The election of the active PCE to delegate an LSP is controlled by each PCC. The PCC usually elects the active PCE by a local configured policy (by setting a priority). Upon PCEP session failure, or active PCE failure, PCC may decide to elect a new active PCE by sending new PCRpt message with D flag set to this new active PCE. When the failed PCE or PCEP session comes back online, it will be up to the implementation to do preemption. Doing preemption may lead to some disruption on the existing path if path results from both PCEs are not exactly the same. By considering a network with multiple PCCs and implementing multiple stateful PCEs for redundancy purpose, there is no guarantee that at any time all the PCCs delegate their LSPs to the same PCE.



In the example above, we consider that by configuration, both PCCs will firstly delegate their LSPs to PCE1. So, PCE1 is responsible for computing a path for both LSP1 and LSP2. If the PCEP session between PCC2 and PCE1 fails, PCC2 will delegate LSP2 to PCE2. So PCE1 becomes responsible only for LSP1 path computation while PCE2 is responsible for the path computation of LSP2. When the PCC2-PCE1 session is back online, PCC2 will keep using PCE2 as active PCE (consider no preemption in this example). So the result is a permanent situation where each PCE is responsible for a subset of path computation.

This situation is called a split-brain scenario, as there are multiple computation brains running at the same time while a central computation unit was required in some deployments/use cases.

Further, there are use cases where a particular LSP path computation is linked to another LSP path computation: the most common use case is path disjointness (see [RFC8800]). The set of LSPs that are dependent to each other may start from a different head-end.



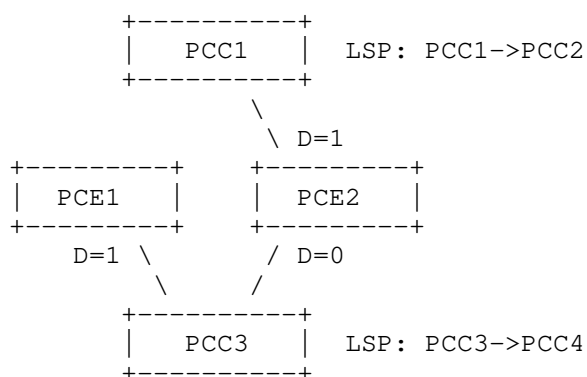
In the figure above, the requirement is to create two link-disjoint LSPs: PCC1->PCC2 and PCC3->PCC4. In the topology, all links cost metric is set to 1 except for the link 'R1-R2' which has a metric of 10. The PCEs are responsible for the path computation and PCE1 is the active primary PCE for all PCCs in the nominal case.

Scenario 1:

In the normal case (PCE1 as active primary PCE), consider that PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE1 sends a PCUpd message to PCC1 with the ERO: R1->R3->R4->R2->PCC2 (shortest path). PCC1 signals and installs the path. When PCC3->PCC4 is configured, the PCEs already knows the path of PCC1->PCC2 and can compute a link-disjoint path: the solution requires to move PCC1->PCC2 onto a new path to let room for the new LSP. PCE1 sends a PCUpd message to PCC1 with the new ERO: R1->R2->PCC2 and a PCUpd to PCC3 with the following ERO: R3->R4->PCC4. In the normal case, there is no issue for PCE1 to compute a link-disjoint path.

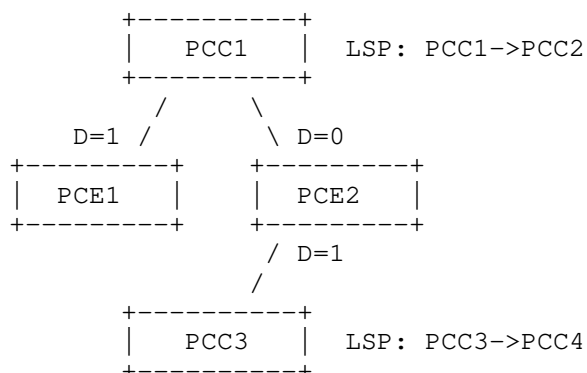
Scenario 2:

Consider that PCC1 lost its PCEP session with PCE1 (all other PCEP sessions are UP). PCC1 delegates its LSP to PCE2.



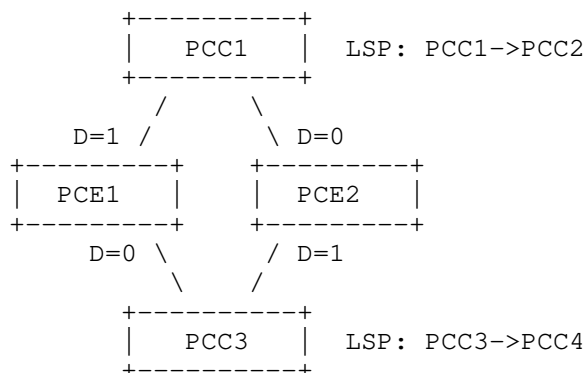
Consider that the PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE2 (which is the new active primary PCE for PCC1) sends a PCUpd message to PCC1 with the ERO: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE1 is not aware of LSPs from PCC1 any more, so it cannot compute a disjoint path for PCC3->PCC4 and will send a PCUpd message to PCC3 with the shortest path ERO: R3->R4->PCC4. When PCC3->PCC4 LSP will be reported to PCE2 by PCC3, PCE2 will ensure disjointness computation and will correctly move PCC1->PCC2 (as it owns delegation for this LSP) on the following path: R1->R2->PCC2. With this sequence of event and these PCEP sessions, disjointness is ensured.

Scenario 3:



Consider the above PCEP sessions and the PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE1 computes the shortest path as it is the only LSP in the disjoint association group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this LSP. The only solution found is to move PCC1->PCC2 LSP on another path, but PCE2 cannot do it as it does not have delegation for this LSP. In this set-up, PCEs are not able to find a disjoint path.

Scenario 4:

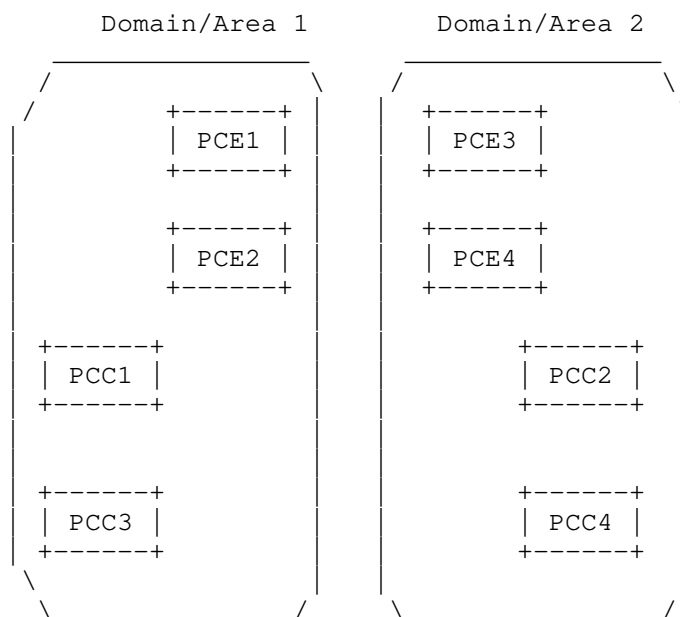


Consider the above PCEP sessions and that PCEs are configured to fall-back to the shortest path if disjointness cannot be found as described in [RFC8800]. The PCC1->PCC2 LSP is configured first, PCE1 computes the shortest path as it is the only LSP in the disjoint association group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this LSP. The only solution found is to move PCC1->PCC2 LSP on another path, but PCE2 cannot do it as it does not have delegation

initial state. When those paths will be reported to both PCEs, this will trigger CSPF again. An infinite loop of CSPF computation is then happening with a permanent flap of paths because of the split-brain situation.

This permanent computation loop comes from the inconsistency between the state of the LSPs as seen by each PCE due to the split-brain: each PCE is trying to modify at the same time its delegated path based on the last received path information which de facto invalidates this received path information.

Scenario 6: multi-domain



In the example above, suppose that the disjoint LSPs from PCC1 to PCC2 and from PCC4 to PCC3 are created. All the PCEs have the knowledge of both domain topologies (e.g. using BGP-LS [RFC7752]). For operation/management reasons, each domain uses its own group of redundant PCEs. PCE1/PCE2 in domain 1 have PCEP sessions with PCC1 and PCC3 while PCE3/PCE4 in domain 2 have PCEP sessions with PCC2 and PCC4. As PCE1/2 does not know about LSPs from PCC2/4 and PCE3/4 do not know about LSPs from PCC1/3, there is no possibility to compute the disjointness constraint. This scenario can also be seen as a split-brain scenario. This multi-domain architecture (with multiple groups of PCEs) can also be used in a single domain, where an

operator wants to limit the failure domain by creating multiple groups of PCEs maintaining a subset of PCCs. As for the multi-domain example, there will be no possibility to compute the disjoint path starting from head-ends managed by different PCE groups.

In this document, we propose a solution that addresses the possibility to compute LSP association based constraints (like disjointness) in split-brain scenarios while preventing computation loops.

1.3. Applicability to H-PCE

[RFC8751] describes general considerations and use cases for the deployment of Stateful PCE(s) using the Hierarchical PCE [RFC6805] architecture. In this architecture, there is a clear need to communicate between a child stateful PCE and a parent stateful PCE. The procedures and extensions as described in Section 3 are equally applicable to the H-PCE scenario.

2. Proposed solution

Our solution is based on :

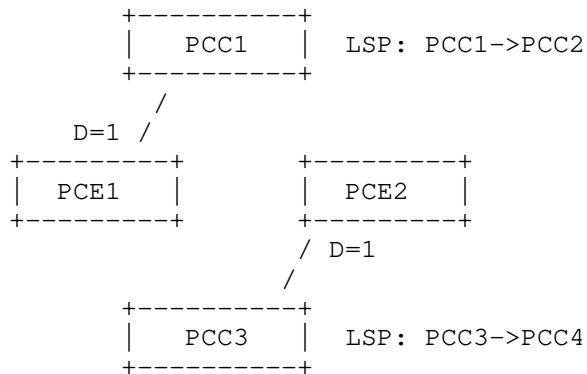
- o The creation of the inter-PCE stateful PCEP session with specific procedures.
- o A Primary/Secondary relationship between PCEs.

2.1. State-sync session

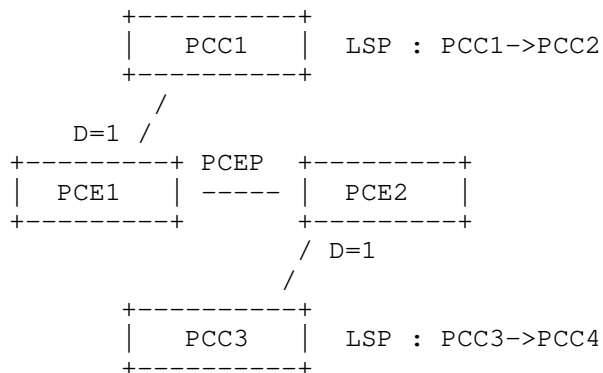
This document proposes to set-up a PCEP session between the stateful PCEs. Creating such a session is already authorized by multiple scenarios like the one described in [RFC4655] (multiple PCEs that are handling part of the path computation) and [RFC6805] (hierarchical PCE) but was only focused on the stateless PCEP sessions. As stateful PCE brings additional features (LSP state synchronization, path update, delegation, ...), thus some new behaviors need to be defined.

This inter-PCE PCEP session will allow the exchange of LSP states between PCEs that would help some scenarios where PCEP sessions are lost between PCC and PCE. This inter-PCE PCEP session is henceforth called a state-sync session.

For example, in the scenario below, there is no possibility to compute disjointness as there is no PCE that is aware of both LSPs.



If we add a state-sync session, PCE1 will be able to do state synchronization via PCRpt messages for its LSP to PCE2 and PCE2 will do the same. All the PCEs will be aware of all LSPs even if a PCC->PCE session is down. PCEs will then be able to compute disjoint paths.



The procedures associated with this state-sync session are defined in Section 3.

By just adding this state-sync session, it does not ensure that a path with LSP association based constraints can always be computed and does not prevent the computation loop, but it increases resiliency and ensures that PCEs will have the state information for all LSPs. Also, this session will allow for a PCE to update the other PCEs providing a faster synchronization mechanism than relying on PCCs only.

2.2. Primary/Secondary relationship between PCE

As seen in Section 1, performing a path computation in a split-brain scenario (multiple PCEs responsible for computation) may provide a non-optimal LSP placement, no path, or computation loops. To provide the best efficiency, an LSP association constraint-based computation requires that a single PCE performs the path computation for all LSPs in the association group. Note that, it could be all LSPs belonging to a particular association group, or all LSPs from a particular PCC, or all LSPs in the network that need to be delegated to a single PCE based on the deployment scenarios.

This document proposes to add a priority mechanism between PCEs to elect a single computing PCE. Using this priority mechanism, PCEs can agree on the PCE that will be responsible for the computation for a particular association group, or set of LSPs. The priority could be set per association, per PCC, or for all LSPs. How this priority is set or advertised is out of the scope of this document. The rest of the text considers the association group as an example.

When a single PCE is performing the computation for a particular association group, no computation loop can happen and an optimal placement will be provided. The other PCEs will only act as state collectors and forwarders.

In the scenario described in Section 2.1, PCE1 and PCE2 will decide that PCE1 will be responsible for the path computation of both LSPs. If we first configure PCC1->PCC2, PCE1 computes the shortest path at it is the only LSP in the disjoint-group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 will not perform computation even if it has delegation but forwards the delegation via PCRpt message to PCE1 through the state-sync session. PCE1 will then perform disjointness computation and will move PCC1->PCC2 onto R1->R2->PCC2 and provides an ERO to PCE2 for PCC3->PCC4: R3->R4->PCC4. The PCE2 will further update the PCC3 with the new path.

3. Procedures and Protocol Extensions

3.1. Opening a state-sync session

3.1.1. Capability Advertisement

A PCE indicates its support of state-sync procedures during the PCEP Initialization phase [RFC5440]. The OPEN object in the Open message MUST contains the "Stateful PCE Capability" TLV defined in [RFC8231]. A new P (INTER-PCE-CAPABILITY) flag is introduced to indicate the support of state-sync.

This document adds a new bit in the Flags field with :

P (INTER-PCE-CAPABILITY - 1 bit): If set to 1 by a PCEP Speaker, the PCEP speaker indicates that the session MUST follow the state-sync procedures as described in this document. The P bit MUST be set by both speakers: if a PCEP Speaker receives a STATEFUL-PCE-CAPABILITY TLV with P=0 while it advertised P=1 or if both set P flag to 0, the session SHOULD be set-up but the state-sync procedures MUST NOT be applied on this session.

The U flag [RFC8231] MUST be set when sending the STATEFUL-PCE-CAPABILITY TLV with the P flag set. In case the U flag is not set along with the P flag, the state sync capability is not enabled and it is considered as if the P flag is not set. The S flag MAY be set if optimized synchronization is required as per [RFC8232].

3.2. State synchronization

When the state sync capability has been negotiated between stateful PCEs, each PCEP speaker will behave as a PCE and as a PCC at the same time regarding the state synchronization as defined in [RFC8231]. This means that each PCEP Speaker:

- o MUST send a PCRpt message towards its neighbor with S flag set for each LSP in its LSP database learned from a PCC. (PCC role)
- o MUST send the End Of Synchronization Marker towards its neighbor when all LSPs have been reported. (PCC role)
- o MUST wait for the LSP synchronization from its neighbor to end (receiving an End Of Synchronization Marker). (PCE role)

The process of synchronization runs in parallel on each PCE (with no defined order).

The optimized state synchronization procedures MAY be used, as defined in [RFC8232].

When a PCEP Speaker sends a PCRpt on a state-sync session, it MUST add the SPEAKER-IDENTITY-TLV (defined in [RFC8232]) in the LSP Object, the value used will refer to the 'owner' PCC of the LSP. If a PCEP Speaker receives a PCRpt on a state-sync session without this TLV, it MUST discard the PCRpt message and it MUST reply with a PCErr message using error-type=6 (Mandatory Object missing) and error-value=TBD1 (SPEAKER-IDENTITY-TLV missing).

3.3. Incremental updates and report forwarding rules

During the life of an LSP, its state may change (path, constraints, operational state...) and a PCC will advertise a new PCRpt to the PCE for each such change.

When propagating LSP state changes from a PCE to other PCEs, it is mandatory to ensure that a PCE always uses the freshest state coming from the PCC.

When a PCE receives a new PCRpt from a PCC with the LSP-DB-VERSION, the PCE MUST forward the PCRpt to all its state-sync sessions and MUST add the appropriate SPEAKER-IDENTITY-TLV in the PCRpt. In addition, it MUST add a new ORIGINAL-LSP-DB-VERSION TLV (described below). The ORIGINAL-LSP-DB-VERSION contains the LSP-DB-VERSION coming from the PCC.

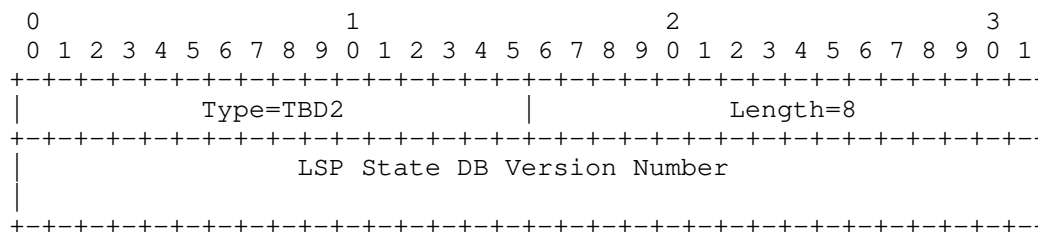
When a PCE receives a new PCRpt from a PCC without the LSP-DB-VERSION, it SHOULD NOT forward the PCRpt on any state-sync sessions and log such an event on the first occurrence.

When a PCE receives a new PCRpt from a PCC with the R flag (Remove) set and an LSP-DB-VERSION TLV, the PCE MUST forward the PCRpt to all its state-sync sessions keeping the R flag set (Remove) and MUST add the appropriate SPEAKER-IDENTITY-TLV and ORIGINAL-LSP-DB-VERSION TLV in the PCRpt message.

When a PCE receives a PCRpt from a state-sync session, it MUST NOT forward the PCRpt to other state-sync sessions. This helps to prevent message loops between PCEs. As a consequence, a full mesh of PCEP sessions between PCEs are REQUIRED.

When a PCRpt is forwarded, all the original objects and values are kept. As an example, the PLSP-ID used in the forwarded PCRpt will be the same as the original one used by the PCC. Thus an implementation supporting this document MUST consider SPEAKER-IDENTITY-TLV and PLSP-ID together to uniquely identify an LSP on the state-sync session.

The ORIGINAL-LSP-DB-VERSION TLV is encoded as follows and MUST always contain the LSP-DB-VERSION received from the owner PCC of the LSP:



Using the ORIGINAL-LSP-DB-VERSION TLV allows a PCE to keep using optimized synchronization ([RFC8232]) with another PCE. In such a case, the PCE will send a PCRpt to another PCE with both ORIGINAL-LSP-DB-VERSION TLV and LSP-DB-VERSION TLV. The ORIGINAL-LSP-DB-VERSION TLV will contain the version number as allocated by the PCC while the LSP-DB-VERSION will contain the version number allocated by the local PCE.

3.4. Maintaining LSP states from different sources

When a PCE receives a PCRpt on a state-sync session, it stores the LSP information into the original PCC address context (as the LSP belongs to the PCC). A PCE SHOULD maintain a single state for a particular LSP and SHOULD maintain the list of sources it learned a particular state from.

A PCEP speaker may receive state information for a particular LSP from different sources: the PCC that owns the LSP (through a regular PCEP session) and some PCEs (through PCEP state-sync sessions). A PCEP speaker MUST always keep the freshest state in its LSP database, overriding the previously received information.

A PCE, receiving a PCRpt from a PCC, updates the state of the LSP in its LSP-DB with the newly received information. When receiving a PCRpt from another PCE, a PCE SHOULD update the LSP state only if the ORIGINAL-LSP-DB-VERSION present in the PCRpt is greater than the current ORIGINAL-LSP-DB-VERSION of the stored LSP state. This ensures that a PCE never tries to update its stored LSP state with an old information. Each time a PCE updates an LSP state in its LSP-DB, it SHOULD reset the source list associated with the LSP state and SHOULD add the source speaker address in the source list. When a PCE receives a PCRpt which has an ORIGINAL-LSP-DB-VERSION (if coming from a PCE) or an LSP-DB-VERSION (if coming from the PCC) equals to the current ORIGINAL-LSP-DB-VERSION of the stored LSP state, it SHOULD add the source speaker address in the source list.

When a PCE receives a PCRpt requesting an LSP deletion from a particular source, it SHOULD remove this particular source from the list of sources associated with this LSP.

When the list of sources becomes empty for a particular LSP, the LSP state MUST be removed. This means that all the sources must send a PCRpt with R=1 for an LSP to make the PCE remove the LSP state.

3.5. Computation priority between PCEs and sub-delegation

A computation priority is necessary to ensure that a single PCE will perform the computation for all the LSPs in an association group: this will allow for a more optimized LSP placement and will prevent computation loops.

All PCEs in the network that are handling LSPs in a common LSP association group SHOULD be aware of each other including the computation priority of each PCE. Note that there is no need for PCC to be aware of this. The computation priority is a number and the PCE having the highest priority SHOULD be responsible for the computation. If several PCEs have the same priority value, their IP address SHOULD be used as a tie-breaker to provide a rank: the highest IP address has more priority. How PCEs are aware of the priority of each other is out of the scope of this document, but as example learning priorities could be done through PCE discovery or local configuration.

The definition of the priority could be global so the highest priority PCE will handle all path computations or more granular, so a PCE may have the highest priority for only a subset of LSPs or association-groups.

A PCEP Speaker receiving a PCRpt from a PCC with the D flag set that does not have the highest computation priority, SHOULD forward the PCRpt on all state-sync sessions (as per Section 3.3) and SHOULD set D flag on the state-sync session towards the highest priority PCE, D flag will be unset to all other state-sync sessions. This behavior is similar to the delegation behavior handled at the PCC side and is called a sub-delegation (the PCE sub-delegates the control of the LSP to another PCE). When a PCEP Speaker sub-delegates an LSP to another PCE, it loose control of the LSP and cannot update it anymore by its own decision. When a PCE receives a PCRpt with D flag set on a state-sync session, as a regular PCE, it is granted control over the LSP.

If the highest priority PCE is failing or if the state-sync session between the local PCE and the highest priority PCE failed, the local PCE MAY decide to delegate the LSP to the next highest priority PCE or to take back control of the LSP. It is a local policy decision.

When a PCE has the delegation for an LSP and needs to update this LSP, it MUST send a PCUpd message to all state-sync sessions and to

the PCC session on which it received the delegation. The D-Flag would be unset in the PCUpd for state-sync sessions whereas the D-Flag would be set for the PCC. In the case of sub-delegation, the computing PCE will send the PCUpd only to all state-sync sessions (as it has no direct delegation from a PCC). The D-Flag would be set for the state-sync session to the PCE that sub-delegated this LSP and the D-Flag would be unset for other state-sync sessions.

The PCUpd sent over a state-sync session MUST contain the SPEAKER-IDENTITY-TLV in the LSP Object (the value used must identify the target PCC). The PLSP-ID used is the original PLSP-ID generated by the PCC and learned from the forwarded PCRpt. If a PCE receives a PCUpd on a state-sync session without the SPEAKER-IDENTITY-TLV, it MUST discard the PCUpd and MUST reply with a PCErr message using error-type=6 (Mandatory Object missing) and error-value=TBD1 (SPEAKER-IDENTITY-TLV missing).

When a PCE receives a valid PCUpd on a state-sync session, it SHOULD forward the PCUpd to the appropriate PCC (identified based on the SPEAKER-IDENTITY-TLV value) that delegated the LSP originally and SHOULD remove the SPEAKER-IDENTITY-TLV from the LSP Object. The acknowledgment of the PCUpd is done through a cascaded mechanism, and the PCC is the only responsible for triggering the acknowledgment: when the PCC receives the PCUpd from the local PCE, it acknowledges it with a PCRpt as per [RFC8231]. When receiving the new PCRpt from the PCC, the local PCE uses the defined forwarding rules on the state-sync session so the acknowledgment is relayed to the computing PCE.

A PCE SHOULD NOT compute a path using an association-group constraint if it has delegation for only a subset of LSPs in the group. In this case, an implementation MAY use a local policy on PCE to decide if PCE does not compute path at all for this set of LSP or if it can compute a path by relaxing the association-group constraint.

3.6. Passive stateful procedures

In the passive stateful PCE architecture, the PCC is responsible for triggering a path computation request using a PCReq message to its PCE. Similarly to PCRpt Message, which remains unchanged for passive mode, if a PCE receives a PCReq for an LSP and if this PCE finds that it does not have the highest computation priority of this LSP, or groups..., it MUST forward the PCReq message to the highest priority PCE over the state-sync session. When the highest priority PCE receives the PCReq, it computes the path and generates a PCRep message towards the PCE that made the request. This PCE will then forward the PCRep to the requesting PCC. The handling of LSP object

and the SPEAKER-IDENTITY-TLV in PCReq and PCRep is similar to PCRpt/PCUpd messages.

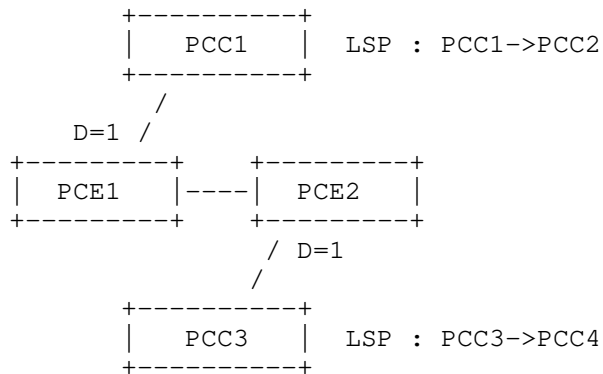
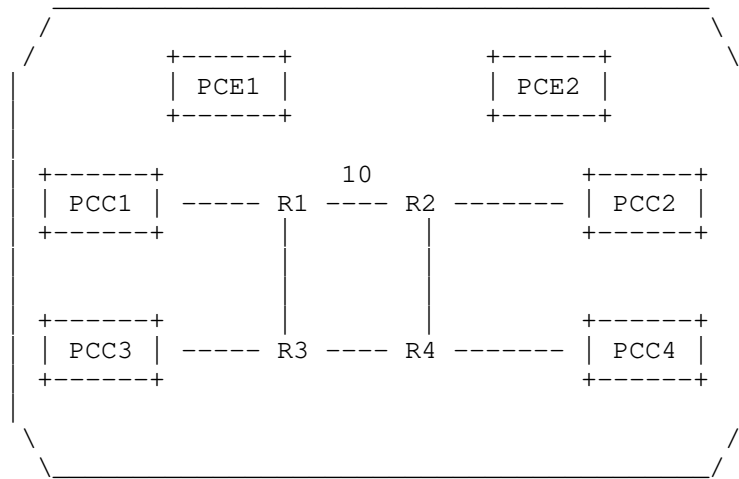
3.7. PCE initiation procedures

TBD

4. Examples

The examples in this section are for illustrative purpose to show how the behavior of the state sync inter-PCE sessions.

4.1. Example 1



PCE1 computation priority 100
PCE2 computation priority 200

Consider the PCEP sessions as shown above, where computation priority is global for all the LSPs and link disjoint between LSPs PCC1->PCC2 and PCC3->PCC4 is required.

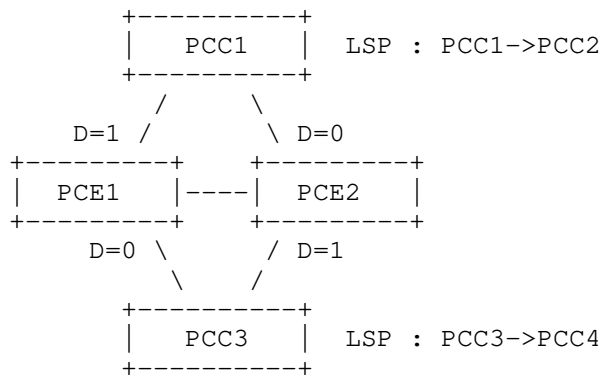
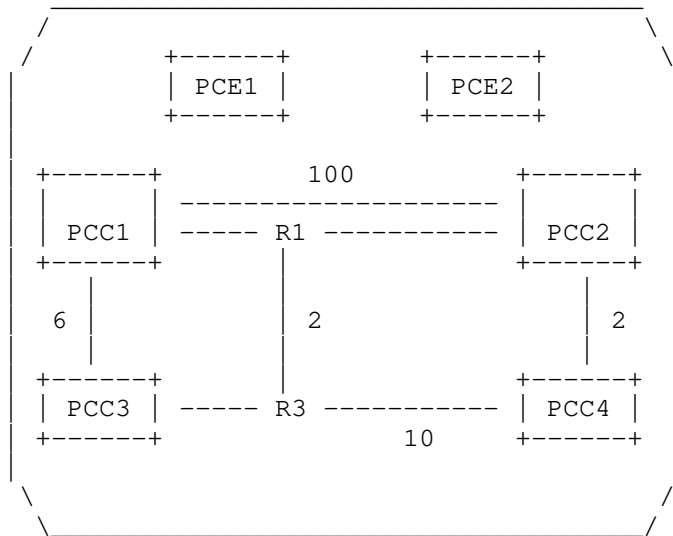
Consider the PCC1->PCC2 is configured first and PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it sub-delegates the LSP to PCE2 by sending a PCRpt with D=1 and including the SPEAKER-IDENTITY-TLV over the state-sync session. PCE2 receives the PCRpt and as it has delegation for this LSP, it computes the shortest path: R1->R3->R4->R2->PCC2. It then sends a PCUpd to PCE1 (including the SPEAKER-IDENTITY-TLV) with the computed ERO. PCE1 forwards the PCUpd to PCC1 (removing the SPEAKER-

IDENTITY-TLV). PCC1 acknowledges the PCUpd by a PCRpt to PCE1. PCE1 forwards the PCRpt to PCE2.

When PCC3->PCC4 is configured, PCC3 delegates the LSP to PCE2, PCE2 can compute a disjoint path as it has knowledge of both LSPs and has delegation also for both. The only solution found is to move PCC1->PCC2 LSP on another path, PCE2 can move PCC1->PCC2 as it has sub-delegation for it. It creates a new PCUpd with a new ERO: R1->R2-PCC2 towards PCE1 which forwards to PCC1. PCE2 sends a PCUpd to PCC3 with the path: R3->R4->PCC4.

In this set-up, PCEs are able to find a disjoint path while without state-sync and computation priority they could not.

4.2. Example 2

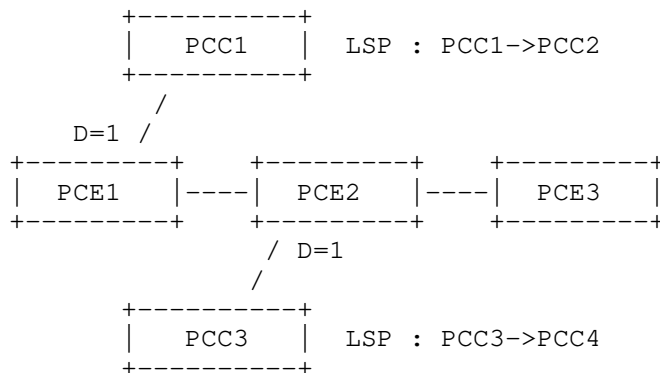
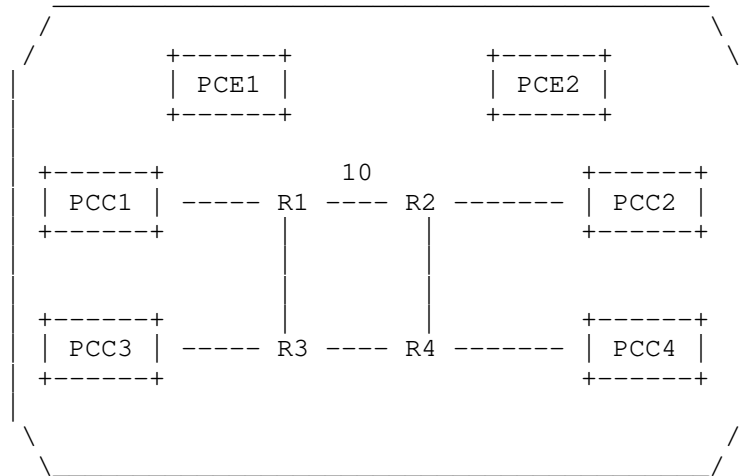


PCE1 computation priority 200

PCE2 computation priority 100

In this example, suppose both LSPs are configured almost at the same time. PCE1 sub-delegates PCC1->PCC2 to PCE2 while PCE2 keeps delegation for PCC3->PCC4, PCE2 computes a path for PCC1->PCC2 and PCC3->PCC4 and can achieve disjointness computation easily. No computation loop happens in this case.

4.3. Example 3



PCE1 computation priority 100
 PCE2 computation priority 200
 PCE3 computation priority 300

With the PCEP sessions as shown above, consider the need to have link disjoint LSPs PCC1->PCC2 and PCC3->PCC4.

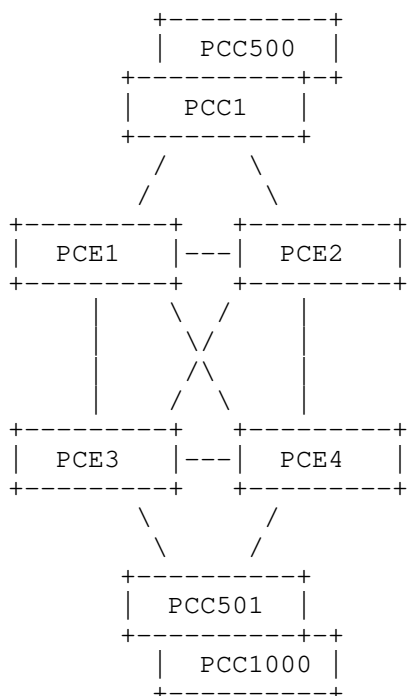
Suppose PCC1->PCC2 is configured first, PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it will sub-delegate the LSP to PCE2 (as it not aware of PCE3 and has no way to reach it). PCE2 cannot compute a path for PCC1->PCC2 as it does not have the highest priority and is not allowed to sub-delegate the LSP again towards PCE3 as per Section 3.

When PCC3->PCC4 is configured, PCC3 delegates the LSP to PCE2 that performs sub-delegation to PCE3. As PCE3 will have knowledge of only one LSP in the group, it cannot compute disjointness and can decide to fall-back to a less constrained computation to provide a path for PCC3->PCC4. In this case, it will send a PCUpd to PCE2 that will be forwarded to PCC3.

Disjointness cannot be achieved in this scenario because of lack of state-sync session between PCE1 and PCE3, but no computation loop happens. Thus it is advised for all PCEs that support state-sync to have a full mesh sessions between each other.

5. Using Primary/Secondary Computation and State-sync Sessions to increase Scaling

The Primary/Secondary computation and state-sync sessions architecture can be used to increase the scaling of the PCE architecture. If the number of PCCs is really high, it may be too resource consuming for a single PCE to maintain all the PCEP sessions while at the same time performing all path computations. Using primary/secondary computation and state-sync sessions may allow to create groups of PCEs that manage a subset of the PCCs and perform some or no path computations. Decoupling PCEP session maintenance and computation will allow increasing scaling of the PCE architecture.



In the figure above, two groups of PCEs are created: PCE1/2 maintain PCEP sessions with PCC1 up to PCC500, while PCE3/4 maintain PCEP sessions with PCC501 up to PCC1000. A granular primary/secondary policy is set-up as follows to load-share computation between PCEs:

- o PCE1 has priority 200 for association ID 1 up to 300, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.
- o PCE3 has priority 200 for association ID 301 up to 500, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.

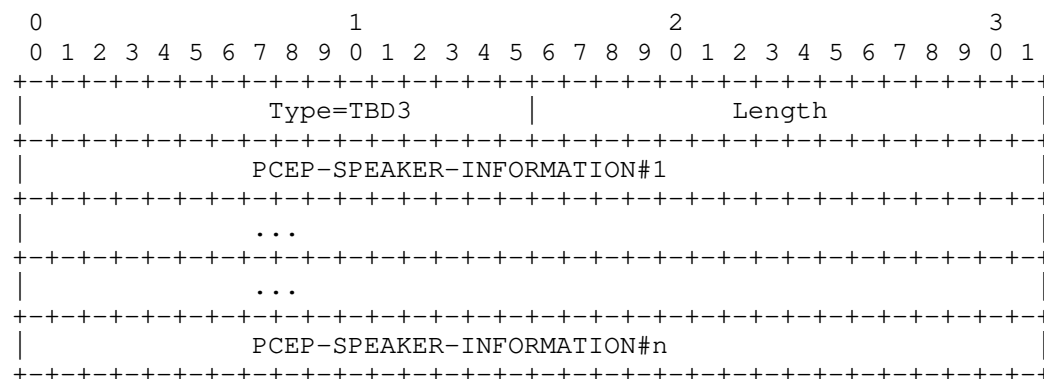
If some PCCs delegate LSPs with association ID 1 up to 300 and association source 0.0.0.0, the receiving PCE (if not PCE1) will sub-delegate the LSPs to PCE1. PCE1 becomes responsible for the computation of these LSP associations while PCE3 is responsible for the computation of another set of associations.

The procedures described in this document could help greatly in load-sharing between a group of stateful PCEs.

6. PCEP-PATH-VECTOR TLV

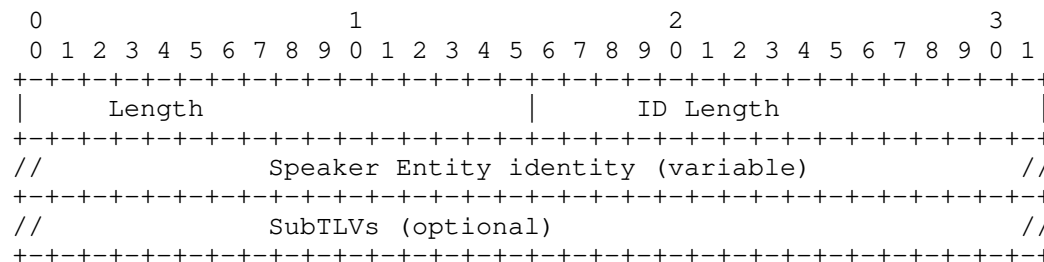
This document allows PCEP messages to be propagated among PCEP speaker. It may be useful to track information about the propagation of the messages. One of the use cases is a message loop detection mechanism, but other use cases like hop by hop information recording may also be implemented.

This document introduces the PCEP-PATH-VECTOR TLV (type TBD3) with the following format:



The TLV format and padding rules are as per [RFC5440].

The PCEP-SPEAKER-INFORMATION field has the following format:



Length: defines the total length of the PCEP-SPEAKER-INFORMATION field.

ID Length: defines the length of the Speaker identity actual field (non-padded).

Speaker Entity identity: same possible values as the SPEAKER-IDENTIFIER-TLV. Padded with trailing zeros to a 4-byte boundary.

The PCEP-SPEAKER-INFORMATION may also carry some optional subTLVs so each PCEP speaker can add local information that could be recorded. This document does not define any sub-TLV.

The PCEP-PATH-VECTOR TLV MAY be carried in the LSP Object. Its usage is purely optional.

The list of speakers within the PCEP-PATH-VECTOR TLV MUST be ordered. When sending a PCEP message (PCRpt, PCUpd, or PCInitiate), a PCEP Speaker MAY add the PCEP-PATH-VECTOR TLV with a PCEP-SPEAKER-INFORMATION containing its own information. If the PCEP message sent is the result of a previously received PCEP message, and if the PCEP-PATH-VECTOR TLV was already present in the initial message, the PCEP speaker MAY append a new PCEP-SPEAKER-INFORMATION containing its own information.

7. Security Considerations

The security considerations described in [RFC8231] and [RFC5440] apply to the extensions described in this document as well. Additional considerations related to state synchronization and sub-delegation between stateful PCEs are introduced, as it could be spoofed and could be used as an attack vector. An attacker could attempt to create too much state in an attempt to load the PCEP peer. The PCEP peer responds with a PCErr message as described in [RFC8231]. An attacker could impact LSP operations by creating bogus state. Further, state synchronization between stateful PCEs could provide an adversary with the opportunity to eavesdrop on the network. Thus, securing the PCEP session using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in [RFC7525], is RECOMMENDED.

8. Acknowledgements

Thanks to [I-D.knodel-terminology] urging for better use of terms.

9. IANA Considerations

This document requests IANA actions to allocate code points for the protocol elements defined in this document.

9.1. PCEP-Error Object

IANA is requested to allocate a new Error Value for the Error Type 9.

Error-Type	Meaning	Reference
6	Mandatory Object Missing	[RFC5440]
	Error-value=TBD1: SPEAKER-IDENTITY-TLV missing	This document

9.2. PCEP TLV Type Indicators

IANA is requested to allocate new TLV Type Indicator values within the "PCEP TLV Type Indicators" sub-registry of the PCEP Numbers registry, as follows:

Value	Meaning	Reference
TBD2	ORIGINAL-LSP-DB-VERSION TLV	This document
TBD3	PCEP-PATH-VECTOR TLV	This document

9.3. STATEFUL-PCE-CAPABILITY TLV

IANA is requested to allocate a new bit value in the STATEFUL-PCE-CAPABILITY TLV Flag Field sub-registry.

Bit	Description	Reference
TBD	INTER-PCE-CAPABILITY	This document

10. References

10.1. Normative References

- [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>>.
- [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>>.

- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.

10.2. Informative References

- [I-D.knodel-terminology]
Knodel, M. and N. Oever, "Terminology, Power, and Inclusive Language in Internet-Drafts and RFCs", draft-knodel-terminology-04 (work in progress), August 2020.
- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <<https://www.rfc-editor.org/info/rfc6805>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.

- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [RFC8751] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., and D. King, "Hierarchical Stateful Path Computation Element (PCE)", RFC 8751, DOI 10.17487/RFC8751, March 2020, <<https://www.rfc-editor.org/info/rfc8751>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.

Appendix A. Contributors

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Authors' Addresses

Stephane Litkowski
Cisco

Email: slitkows.ietf@gmail.com

Siva Sivabalan
Ciena Corporation

Email: msiva282@gmail.com

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan, Guangdong 523808
China

Email: zhenghaomian@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 26, 2021

S. Litkowski
Cisco
S. Sivabalan
Ciena Corporation
C. Li
H. Zheng
Huawei Technologies
February 22, 2021

Inter Stateful Path Computation Element (PCE) Communication Procedures.
draft-litkowski-pce-state-sync-10

Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computation in response to a Path Computation Client (PCC) request. The Stateful PCE extensions allow stateful control of Multi-Protocol Label Switching (MPLS) Traffic Engineering (TE) Label Switched Paths (LSPs) using PCEP.

A Path Computation Client (PCC) can synchronize an LSP state information to a Stateful Path Computation Element (PCE). The stateful PCE extension allows a redundancy scenario where a PCC can have redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control of a LSP to only a single PCE, and only one PCE is responsible for path computation for this delegated LSP.

There are some use cases, where an inter-PCE stateful communication can bring additional resiliency in the design, for instance when some PCC-PCE session fails. The inter-PCE stateful communication may also provide a faster update of the LSP states when such an event occurs. Finally, when, in a redundant PCE scenario, there is a need to compute a set of paths that are part of a group (so there is a dependency between the paths), there may be some cases where the computation of all paths in the group is not handled by the same PCE: this situation is called a split-brain. This split-brain scenario may lead to computation loops between PCEs or suboptimal path computation.

This document describes the procedures to allow a stateful communication between PCEs for various use-cases and also the procedures to prevent computations loops.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 August 26, 2021.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction and Problem Statement	3
1.1. Reporting LSP Changes	4
1.2. Split-Brain	5
1.3. Applicability to H-PCE	12
2. Proposed solution	12
2.1. State-sync session	12

2.2. Primary/Secondary relationship between PCE	14
3. Procedures and Protocol Extensions	14
3.1. Opening a state-sync session	14
3.1.1. Capability Advertisement	14
3.2. State synchronization	15
3.3. Incremental updates and report forwarding rules	16
3.4. Maintaining LSP states from different sources	17
3.5. Computation priority between PCEs and sub-delegation	18
3.6. Passive stateful procedures	19
3.7. PCE initiation procedures	20
4. Examples	20
4.1. Example 1	20
4.2. Example 2	22
4.3. Example 3	24
5. Using Primary/Secondary Computation and State-sync Sessions to increase Scaling	25
6. PCEP-PATH-VECTOR TLV	27
7. Security Considerations	28
8. Acknowledgements	28
9. IANA Considerations	28
9.1. PCEP-Error Object	28
9.2. PCEP TLV Type Indicators	29
9.3. STATEFUL-PCE-CAPABILITY TLV	29
10. References	29
10.1. Normative References	29
10.2. Informative References	30
Appendix A. Contributors	31
Authors' Addresses	31

1. Introduction and Problem Statement

The Path Computation Element communication Protocol (PCEP) [RFC5440] provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients' (PCCs) requests.

A stateful PCE [RFC8231] is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

The examples in this section are for illustrative purpose to showcase the need for inter-PCE stateful PCEP sessions.

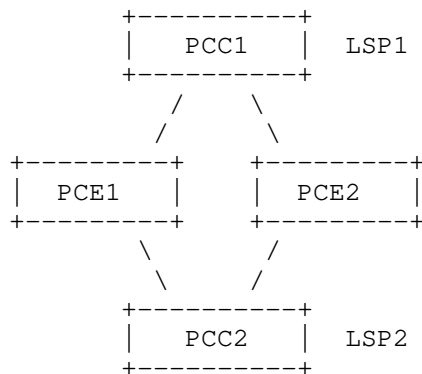
1.1. Reporting LSP Changes

When using a stateful PCE ([RFC8231]), a PCC can synchronize an LSP state information to the stateful PCE. If the PCC grants the control of the LSP to the PCE (called delegation [RFC8231]), the PCE can update the LSP parameters at any time.

In a multi PCE deployment (redundancy, loadbalancing...), with the current specification defined in [RFC8231], when a PCE makes an update, it is the PCC that is in charge of reporting the LSP status to all PCEs with LSP parameter change which brings additional hops and delays in notifying the overall network of the LSP parameter change.

This delay may affect the reaction time of the other PCEs if they need to take action after being notified of the LSP parameter change.

Apart from the synchronization from the PCC, it is also useful if there is a synchronization mechanism between the stateful PCEs. As stateful PCE make changes to its delegated LSPs, these changes (pending LSPs and the sticky resources [RFC7399]) can be synchronized immediately to the other PCEs.



In the figure above, we consider a load-balanced PCE architecture, so PCE1 is responsible to compute paths for PCC1 and PCE2 is responsible to compute paths for PCC2. When PCE1 triggers an LSP update for LSP1, it sends a PCUpd message to PCC1 containing the new parameters for LSP1. PCC1 will take the parameters into account and will send a PCRppt message to PCE1 and PCE2 reflecting the changes. PCE2 will so

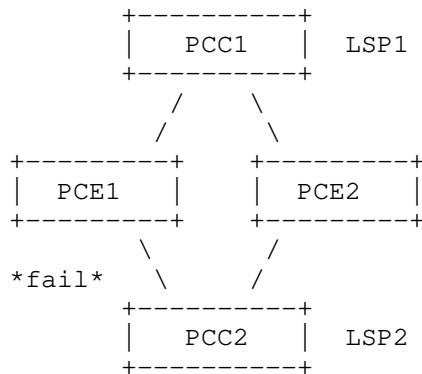
be notified of the change only after receiving the PCRpt message from PCC1.

Let's consider that the LSP1 parameters changed in such a way that LSP1 will take over resources from LSP2 with a higher priority. After receiving the report from PCC1, PCE2 will therefore try to find a new path for LSP2. If we consider that there is a round trip delay of about 150 milliseconds (ms) between the PCEs and PCC1 and a round trip delay of 10 ms between the two PCEs it will take more than 150 ms for PCE2 to be notified of the change.

Adding a PCEP session between PCE1 and PCE2 may allow to reduce the synchronization time, so PCE2 can react more quickly by taking the pending LSPs and attached resources into account during path computation and re-optimization.

1.2. Split-Brain

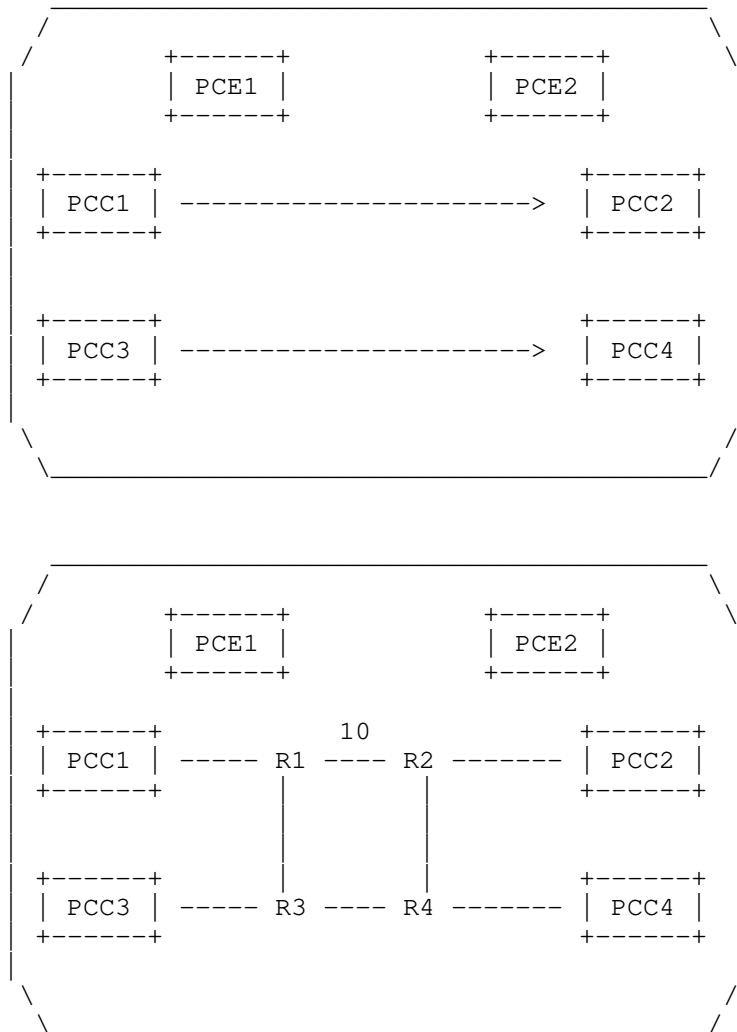
In a resiliency case, a PCC has redundant PCEP sessions towards multiple PCEs. In such a case, a PCC gives control on an LSP to a single PCE only, and only this PCE is responsible for the path computation for the delegated LSP: the PCC achieves this by setting the D flag only towards the active PCE [RFC8231] selected for delegation. The election of the active PCE to delegate an LSP is controlled by each PCC. The PCC usually elects the active PCE by a local configured policy (by setting a priority). Upon PCEP session failure, or active PCE failure, PCC may decide to elect a new active PCE by sending new PCRpt message with D flag set to this new active PCE. When the failed PCE or PCEP session comes back online, it will be up to the implementation to do preemption. Doing preemption may lead to some disruption on the existing path if path results from both PCEs are not exactly the same. By considering a network with multiple PCCs and implementing multiple stateful PCEs for redundancy purpose, there is no guarantee that at any time all the PCCs delegate their LSPs to the same PCE.



In the example above, we consider that by configuration, both PCCs will firstly delegate their LSPs to PCE1. So, PCE1 is responsible for computing a path for both LSP1 and LSP2. If the PCEP session between PCC2 and PCE1 fails, PCC2 will delegate LSP2 to PCE2. So PCE1 becomes responsible only for LSP1 path computation while PCE2 is responsible for the path computation of LSP2. When the PCC2-PCE1 session is back online, PCC2 will keep using PCE2 as active PCE (consider no preemption in this example). So the result is a permanent situation where each PCE is responsible for a subset of path computation.

This situation is called a split-brain scenario, as there are multiple computation brains running at the same time while a central computation unit was required in some deployments/use cases.

Further, there are use cases where a particular LSP path computation is linked to another LSP path computation: the most common use case is path disjointness (see [RFC8800]). The set of LSPs that are dependent to each other may start from a different head-end.



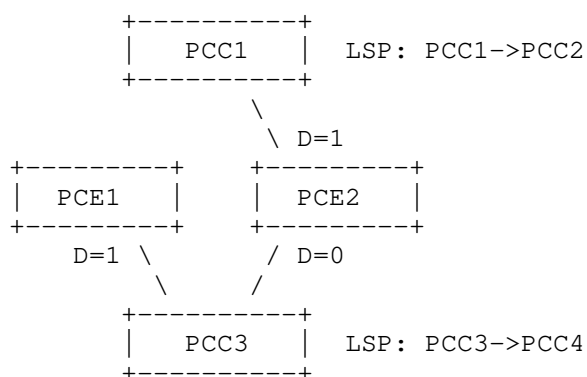
In the figure above, the requirement is to create two link-disjoint LSPs: PCC1->PCC2 and PCC3->PCC4. In the topology, all links cost metric is set to 1 except for the link 'R1-R2' which has a metric of 10. The PCEs are responsible for the path computation and PCE1 is the active primary PCE for all PCCs in the nominal case.

Scenario 1:

In the normal case (PCE1 as active primary PCE), consider that PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE1 sends a PCUpd message to PCC1 with the ERO: R1->R3->R4->R2->PCC2 (shortest path). PCC1 signals and installs the path. When PCC3->PCC4 is configured, the PCEs already knows the path of PCC1->PCC2 and can compute a link-disjoint path: the solution requires to move PCC1->PCC2 onto a new path to let room for the new LSP. PCE1 sends a PCUpd message to PCC1 with the new ERO: R1->R2->PCC2 and a PCUpd to PCC3 with the following ERO: R3->R4->PCC4. In the normal case, there is no issue for PCE1 to compute a link-disjoint path.

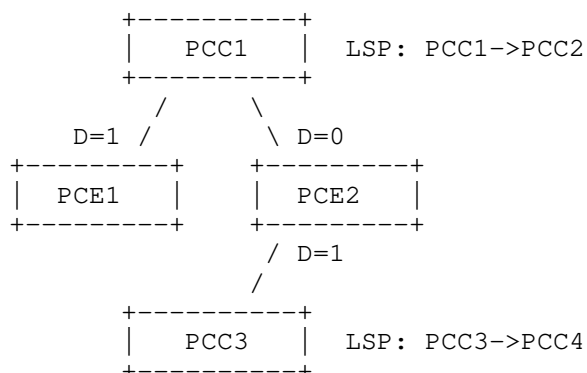
Scenario 2:

Consider that PCC1 lost its PCEP session with PCE1 (all other PCEP sessions are UP). PCC1 delegates its LSP to PCE2.



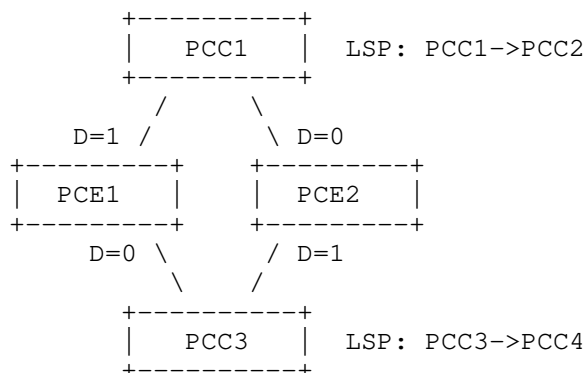
Consider that the PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE2 (which is the new active primary PCE for PCC1) sends a PCUpd message to PCC1 with the ERO: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE1 is not aware of LSPs from PCC1 any more, so it cannot compute a disjoint path for PCC3->PCC4 and will send a PCUpd message to PCC3 with the shortest path ERO: R3->R4->PCC4. When PCC3->PCC4 LSP will be reported to PCE2 by PCC3, PCE2 will ensure disjointness computation and will correctly move PCC1->PCC2 (as it owns delegation for this LSP) on the following path: R1->R2->PCC2. With this sequence of event and these PCEP sessions, disjointness is ensured.

Scenario 3:



Consider the above PCEP sessions and the PCC1->PCC2 LSP is configured first with the link disjointness constraint, PCE1 computes the shortest path as it is the only LSP in the disjoint association group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this LSP. The only solution found is to move PCC1->PCC2 LSP on another path, but PCE2 cannot do it as it does not have delegation for this LSP. In this set-up, PCEs are not able to find a disjoint path.

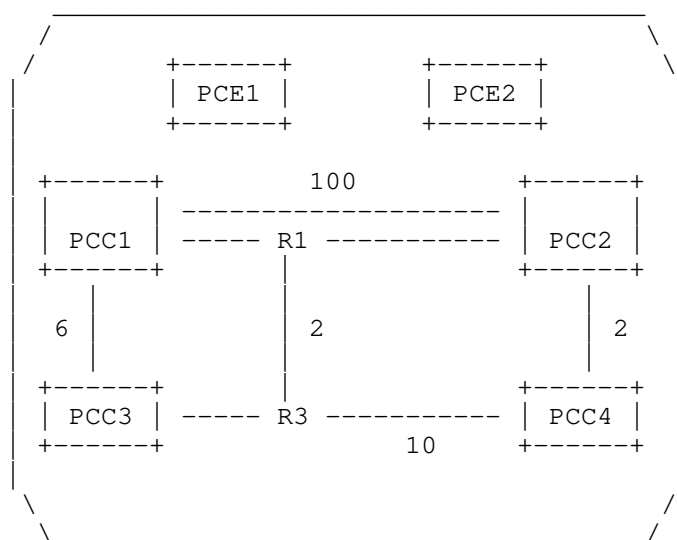
Scenario 4:



Consider the above PCEP sessions and that PCEs are configured to fall-back to the shortest path if disjointness cannot be found as described in [RFC8800]. The PCC1->PCC2 LSP is configured first, PCE1 computes the shortest path as it is the only LSP in the disjoint association group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 must compute a disjoint path for this LSP. The only solution found is to move PCC1->PCC2 LSP on another path, but PCE2 cannot do it as it does not have delegation

for this LSP. PCE2 then provides the shortest path for PCC3->PCC4: R3->R4->PCC4. When PCC3 receives the ERO, it reports it back to both PCEs. When PCE1 becomes aware of the PCC3->PCC4 path, it recomputes the constrained shortest path first (CSPF) algorithm and provides a new path for PCC1->PCC2: R1->R2->PCC2. The new path is reported back to all PCEs by PCC1. PCE2 recomputes also CSPF to take into account the new reported path. The new computation does not lead to any path update.

Scenario 5:

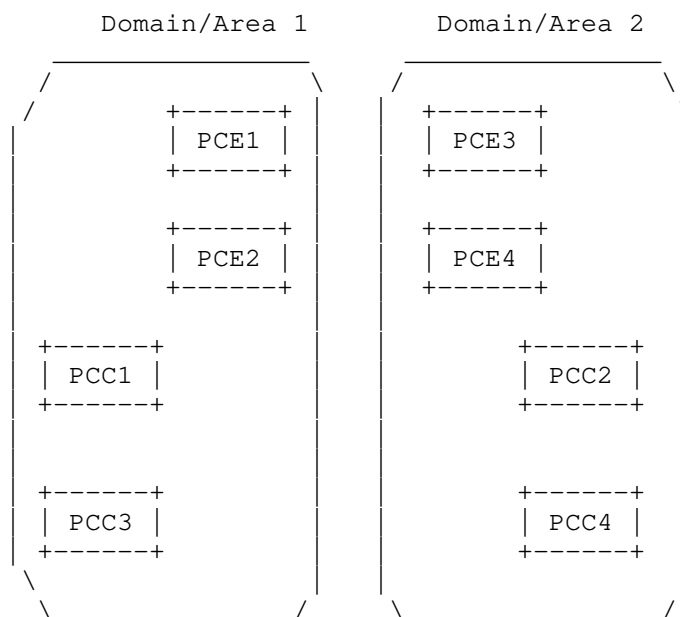


Now, consider a new network topology with the same PCEP sessions as the previous example. Suppose that both LSPs are configured almost at the same time. PCE1 will compute a path for PCC1->PCC2 while PCE2 will compute a path for PCC3->PCC4. As each PCE is not aware of the path of the second LSP in the association group (not reported yet), each PCE is computing the shortest path for the LSP. PCE1 computes ERO: R1->PCC2 for PCC1->PCC2 and PCE2 computes ERO: R3->R1->PCC2->PCC4 for PCC3->PCC4. When these shortest paths will be reported to each PCE. Each PCE will recompute disjointness. PCE1 will provide a new path for PCC1->PCC2 with ERO: PCC1->PCC2. PCE2 will provide also a new path for PCC3->PCC4 with ERO: R3->PCC4. When those new paths will be reported to both PCEs, this will trigger CSFP again. PCE1 will provide a new more optimal path for PCC1->PCC2 with ERO: R1->PCC2 and PCE2 will also provide a more optimal path for PCC3->PCC4 with ERO: R3->R1->PCC2->PCC4. So we come back to the

initial state. When those paths will be reported to both PCEs, this will trigger CSPF again. An infinite loop of CSPF computation is then happening with a permanent flap of paths because of the split-brain situation.

This permanent computation loop comes from the inconsistency between the state of the LSPs as seen by each PCE due to the split-brain: each PCE is trying to modify at the same time its delegated path based on the last received path information which de facto invalidates this received path information.

Scenario 6: multi-domain



In the example above, suppose that the disjoint LSPs from PCC1 to PCC2 and from PCC4 to PCC3 are created. All the PCEs have the knowledge of both domain topologies (e.g. using BGP-LS [RFC7752]). For operation/management reasons, each domain uses its own group of redundant PCEs. PCE1/PCE2 in domain 1 have PCEP sessions with PCC1 and PCC3 while PCE3/PCE4 in domain 2 have PCEP sessions with PCC2 and PCC4. As PCE1/2 does not know about LSPs from PCC2/4 and PCE3/4 do not know about LSPs from PCC1/3, there is no possibility to compute the disjointness constraint. This scenario can also be seen as a split-brain scenario. This multi-domain architecture (with multiple groups of PCEs) can also be used in a single domain, where an

operator wants to limit the failure domain by creating multiple groups of PCEs maintaining a subset of PCCs. As for the multi-domain example, there will be no possibility to compute the disjoint path starting from head-ends managed by different PCE groups.

In this document, we propose a solution that addresses the possibility to compute LSP association based constraints (like disjointness) in split-brain scenarios while preventing computation loops.

1.3. Applicability to H-PCE

[RFC8751] describes general considerations and use cases for the deployment of Stateful PCE(s) using the Hierarchical PCE [RFC6805] architecture. In this architecture, there is a clear need to communicate between a child stateful PCE and a parent stateful PCE. The procedures and extensions as described in Section 3 are equally applicable to the H-PCE scenario.

2. Proposed solution

Our solution is based on :

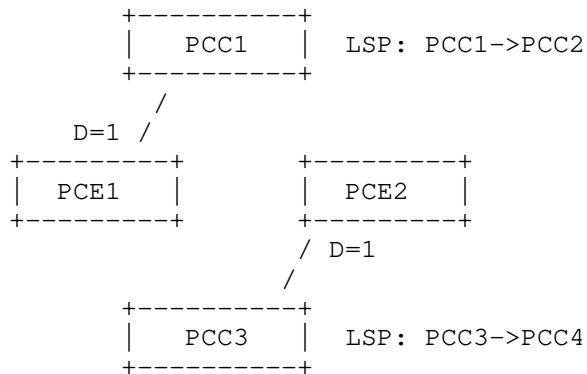
- o The creation of the inter-PCE stateful PCEP session with specific procedures.
- o A Primary/Secondary relationship between PCEs.

2.1. State-sync session

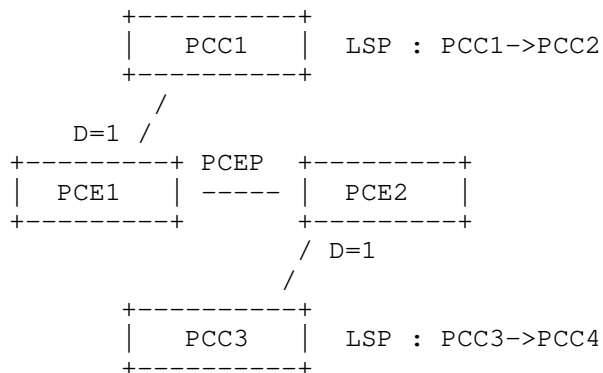
This document proposes to set-up a PCEP session between the stateful PCEs. Creating such a session is already authorized by multiple scenarios like the one described in [RFC4655] (multiple PCEs that are handling part of the path computation) and [RFC6805] (hierarchical PCE) but was only focused on the stateless PCEP sessions. As stateful PCE brings additional features (LSP state synchronization, path update, delegation, ...), thus some new behaviors need to be defined.

This inter-PCE PCEP session will allow the exchange of LSP states between PCEs that would help some scenarios where PCEP sessions are lost between PCC and PCE. This inter-PCE PCEP session is henceforth called a state-sync session.

For example, in the scenario below, there is no possibility to compute disjointness as there is no PCE that is aware of both LSPs.



If we add a state-sync session, PCE1 will be able to do state synchronization via PCRpt messages for its LSP to PCE2 and PCE2 will do the same. All the PCEs will be aware of all LSPs even if a PCC->PCE session is down. PCEs will then be able to compute disjoint paths.



The procedures associated with this state-sync session are defined in Section 3.

By just adding this state-sync session, it does not ensure that a path with LSP association based constraints can always be computed and does not prevent the computation loop, but it increases resiliency and ensures that PCEs will have the state information for all LSPs. Also, this session will allow for a PCE to update the other PCEs providing a faster synchronization mechanism than relying on PCCs only.

2.2. Primary/Secondary relationship between PCE

As seen in Section 1, performing a path computation in a split-brain scenario (multiple PCEs responsible for computation) may provide a non-optimal LSP placement, no path, or computation loops. To provide the best efficiency, an LSP association constraint-based computation requires that a single PCE performs the path computation for all LSPs in the association group. Note that, it could be all LSPs belonging to a particular association group, or all LSPs from a particular PCC, or all LSPs in the network that need to be delegated to a single PCE based on the deployment scenarios.

This document proposes to add a priority mechanism between PCEs to elect a single computing PCE. Using this priority mechanism, PCEs can agree on the PCE that will be responsible for the computation for a particular association group, or set of LSPs. The priority could be set per association, per PCC, or for all LSPs. How this priority is set or advertised is out of the scope of this document. The rest of the text considers the association group as an example.

When a single PCE is performing the computation for a particular association group, no computation loop can happen and an optimal placement will be provided. The other PCEs will only act as state collectors and forwarders.

In the scenario described in Section 2.1, PCE1 and PCE2 will decide that PCE1 will be responsible for the path computation of both LSPs. If we first configure PCC1->PCC2, PCE1 computes the shortest path at it is the only LSP in the disjoint-group that it is aware of: R1->R3->R4->R2->PCC2 (shortest path). When PCC3->PCC4 is configured, PCE2 will not perform computation even if it has delegation but forwards the delegation via PCRpt message to PCE1 through the state-sync session. PCE1 will then perform disjointness computation and will move PCC1->PCC2 onto R1->R2->PCC2 and provides an ERO to PCE2 for PCC3->PCC4: R3->R4->PCC4. The PCE2 will further update the PCC3 with the new path.

3. Procedures and Protocol Extensions

3.1. Opening a state-sync session

3.1.1. Capability Advertisement

A PCE indicates its support of state-sync procedures during the PCEP Initialization phase [RFC5440]. The OPEN object in the Open message MUST contain the "Stateful PCE Capability" TLV defined in [RFC8231]. A new P (INTER-PCE-CAPABILITY) flag is introduced to indicate the support of state-sync.

This document adds a new bit in the Flags field with :

P (INTER-PCE-CAPABILITY - 1 bit - TBD4): If set to 1 by a PCEP Speaker, the PCEP speaker indicates that the session MUST follow the state-sync procedures as described in this document. The P bit MUST be set by both speakers: if a PCEP Speaker receives a STATEFUL-PCE-CAPABILITY TLV with P=0 while it advertised P=1 or if both set P flag to 0, the session SHOULD be set-up but the state-sync procedures MUST NOT be applied on this session.

The U flag [RFC8231] MUST be set when sending the STATEFUL-PCE-CAPABILITY TLV with the P flag set. In case the U flag is not set along with the P flag, the state sync capability is not enabled and it is considered as if the P flag is not set. The S flag MAY be set if optimized synchronization is required as per [RFC8232].

3.2. State synchronization

When the state sync capability has been negotiated between stateful PCEs, each PCEP speaker will behave as a PCE and as a PCC at the same time regarding the state synchronization as defined in [RFC8231]. This means that each PCEP Speaker:

- o MUST send a PCRpt message towards its neighbor with S flag set for each LSP in its LSP database learned from a PCC. (PCC role)
- o MUST send the End Of Synchronization Marker towards its neighbor when all LSPs have been reported. (PCC role)
- o MUST wait for the LSP synchronization from its neighbor to end (receiving an End Of Synchronization Marker). (PCE role)

The process of synchronization runs in parallel on each PCE (with no defined order).

The optimized state synchronization procedures MAY be used, as defined in [RFC8232].

When a PCEP Speaker sends a PCRpt on a state-sync session, it MUST add the SPEAKER-IDENTITY-TLV (defined in [RFC8232]) in the LSP Object, the value used will refer to the 'owner' PCC of the LSP. If a PCEP Speaker receives a PCRpt on a state-sync session without this TLV, it MUST discard the PCRpt message and it MUST reply with a PCErr message using error-type=6 (Mandatory Object missing) and error-value=TBD1 (SPEAKER-IDENTITY-TLV missing).

3.3. Incremental updates and report forwarding rules

During the life of an LSP, its state may change (path, constraints, operational state...) and a PCC will advertise a new PCRpt to the PCE for each such change.

When propagating LSP state changes from a PCE to other PCEs, it is mandatory to ensure that a PCE always uses the freshest state coming from the PCC.

When a PCE receives a new PCRpt from a PCC with the LSP-DB-VERSION, the PCE MUST forward the PCRpt to all its state-sync sessions and MUST add the appropriate SPEAKER-IDENTITY-TLV in the PCRpt. In addition, it MUST add a new ORIGINAL-LSP-DB-VERSION TLV (described below). The ORIGINAL-LSP-DB-VERSION contains the LSP-DB-VERSION coming from the PCC.

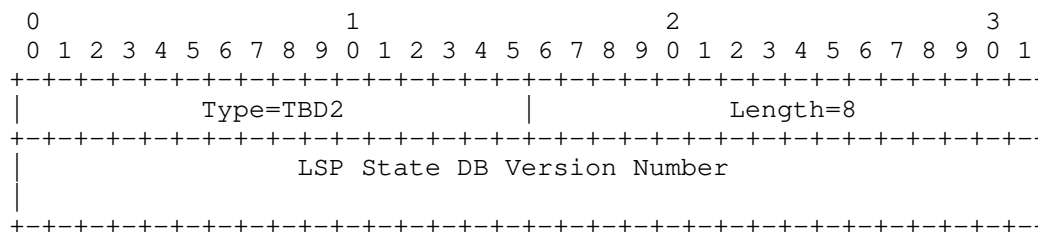
When a PCE receives a new PCRpt from a PCC without the LSP-DB-VERSION, it SHOULD NOT forward the PCRpt on any state-sync sessions and log such an event on the first occurrence.

When a PCE receives a new PCRpt from a PCC with the R flag (Remove) set and an LSP-DB-VERSION TLV, the PCE MUST forward the PCRpt to all its state-sync sessions keeping the R flag set (Remove) and MUST add the appropriate SPEAKER-IDENTITY-TLV and ORIGINAL-LSP-DB-VERSION TLV in the PCRpt message.

When a PCE receives a PCRpt from a state-sync session, it MUST NOT forward the PCRpt to other state-sync sessions. This helps to prevent message loops between PCEs. As a consequence, a full mesh of PCEP sessions between PCEs are REQUIRED.

When a PCRpt is forwarded, all the original objects and values are kept. As an example, the PLSP-ID used in the forwarded PCRpt will be the same as the original one used by the PCC. Thus an implementation supporting this document MUST consider SPEAKER-IDENTITY-TLV and PLSP-ID together to uniquely identify an LSP on the state-sync session.

The ORIGINAL-LSP-DB-VERSION TLV is encoded as follows and MUST always contain the LSP-DB-VERSION received from the owner PCC of the LSP:



Using the ORIGINAL-LSP-DB-VERSION TLV allows a PCE to keep using optimized synchronization ([RFC8232]) with another PCE. In such a case, the PCE will send a PCRpt to another PCE with both ORIGINAL-LSP-DB-VERSION TLV and LSP-DB-VERSION TLV. The ORIGINAL-LSP-DB-VERSION TLV will contain the version number as allocated by the PCC while the LSP-DB-VERSION will contain the version number allocated by the local PCE.

3.4. Maintaining LSP states from different sources

When a PCE receives a PCRpt on a state-sync session, it stores the LSP information into the original PCC address context (as the LSP belongs to the PCC). A PCE SHOULD maintain a single state for a particular LSP and SHOULD maintain the list of sources it learned a particular state from.

A PCEP speaker may receive state information for a particular LSP from different sources: the PCC that owns the LSP (through a regular PCEP session) and some PCEs (through PCEP state-sync sessions). A PCEP speaker MUST always keep the freshest state in its LSP database, overriding the previously received information.

A PCE, receiving a PCRpt from a PCC, updates the state of the LSP in its LSP-DB with the newly received information. When receiving a PCRpt from another PCE, a PCE SHOULD update the LSP state only if the ORIGINAL-LSP-DB-VERSION present in the PCRpt is greater than the current ORIGINAL-LSP-DB-VERSION of the stored LSP state. This ensures that a PCE never tries to update its stored LSP state with an old information. Each time a PCE updates an LSP state in its LSP-DB, it SHOULD reset the source list associated with the LSP state and SHOULD add the source speaker address in the source list. When a PCE receives a PCRpt which has an ORIGINAL-LSP-DB-VERSION (if coming from a PCE) or an LSP-DB-VERSION (if coming from the PCC) equals to the current ORIGINAL-LSP-DB-VERSION of the stored LSP state, it SHOULD add the source speaker address in the source list.

When a PCE receives a PCRpt requesting an LSP deletion from a particular source, it SHOULD remove this particular source from the list of sources associated with this LSP.

When the list of sources becomes empty for a particular LSP, the LSP state MUST be removed. This means that all the sources must send a PCRpt with R=1 for an LSP to make the PCE remove the LSP state.

3.5. Computation priority between PCEs and sub-delegation

A computation priority is necessary to ensure that a single PCE will perform the computation for all the LSPs in an association group: this will allow for a more optimized LSP placement and will prevent computation loops.

All PCEs in the network that are handling LSPs in a common LSP association group SHOULD be aware of each other including the computation priority of each PCE. Note that there is no need for PCC to be aware of this. The computation priority is a number and the PCE having the highest priority SHOULD be responsible for the computation. If several PCEs have the same priority value, their IP address SHOULD be used as a tie-breaker to provide a rank: the highest IP address has more priority. How PCEs are aware of the priority of each other is out of the scope of this document, but as example learning priorities could be done through PCE discovery or local configuration.

The definition of the priority could be global so the highest priority PCE will handle all path computations or more granular, so a PCE may have the highest priority for only a subset of LSPs or association-groups.

A PCEP Speaker receiving a PCRpt from a PCC with the D flag set that does not have the highest computation priority, SHOULD forward the PCRpt on all state-sync sessions (as per Section 3.3) and SHOULD set D flag on the state-sync session towards the highest priority PCE, D flag will be unset to all other state-sync sessions. This behavior is similar to the delegation behavior handled at the PCC side and is called a sub-delegation (the PCE sub-delegates the control of the LSP to another PCE). When a PCEP Speaker sub-delegates an LSP to another PCE, it loose control of the LSP and cannot update it anymore by its own decision. When a PCE receives a PCRpt with D flag set on a state-sync session, as a regular PCE, it is granted control over the LSP.

If the highest priority PCE is failing or if the state-sync session between the local PCE and the highest priority PCE failed, the local PCE MAY decide to delegate the LSP to the next highest priority PCE or to take back control of the LSP. It is a local policy decision.

When a PCE has the delegation for an LSP and needs to update this LSP, it MUST send a PCUpd message to all state-sync sessions and to

the PCC session on which it received the delegation. The D-Flag would be unset in the PCUpd for state-sync sessions whereas the D-Flag would be set for the PCC. In the case of sub-delegation, the computing PCE will send the PCUpd only to all state-sync sessions (as it has no direct delegation from a PCC). The D-Flag would be set for the state-sync session to the PCE that sub-delegated this LSP and the D-Flag would be unset for other state-sync sessions.

The PCUpd sent over a state-sync session MUST contain the SPEAKER-IDENTITY-TLV in the LSP Object (the value used must identify the target PCC). The PLSP-ID used is the original PLSP-ID generated by the PCC and learned from the forwarded PCRpt. If a PCE receives a PCUpd on a state-sync session without the SPEAKER-IDENTITY-TLV, it MUST discard the PCUpd and MUST reply with a PCErr message using error-type=6 (Mandatory Object missing) and error-value=TBD1 (SPEAKER-IDENTITY-TLV missing).

When a PCE receives a valid PCUpd on a state-sync session, it SHOULD forward the PCUpd to the appropriate PCC (identified based on the SPEAKER-IDENTITY-TLV value) that delegated the LSP originally and SHOULD remove the SPEAKER-IDENTITY-TLV from the LSP Object. The acknowledgment of the PCUpd is done through a cascaded mechanism, and the PCC is the only responsible for triggering the acknowledgment: when the PCC receives the PCUpd from the local PCE, it acknowledges it with a PCRpt as per [RFC8231]. When receiving the new PCRpt from the PCC, the local PCE uses the defined forwarding rules on the state-sync session so the acknowledgment is relayed to the computing PCE.

A PCE SHOULD NOT compute a path using an association-group constraint if it has delegation for only a subset of LSPs in the group. In this case, an implementation MAY use a local policy on PCE to decide if PCE does not compute path at all for this set of LSP or if it can compute a path by relaxing the association-group constraint.

3.6. Passive stateful procedures

In the passive stateful PCE architecture, the PCC is responsible for triggering a path computation request using a PCReq message to its PCE. Similarly to PCRpt Message, which remains unchanged for passive mode, if a PCE receives a PCReq for an LSP and if this PCE finds that it does not have the highest computation priority of this LSP, or groups..., it MUST forward the PCReq message to the highest priority PCE over the state-sync session. When the highest priority PCE receives the PCReq, it computes the path and generates a PCRep message towards the PCE that made the request. This PCE will then forward the PCRep to the requesting PCC. The handling of LSP object

and the SPEAKER-IDENTITY-TLV in PCReq and PCRep is similar to PCRpt/PCUpd messages.

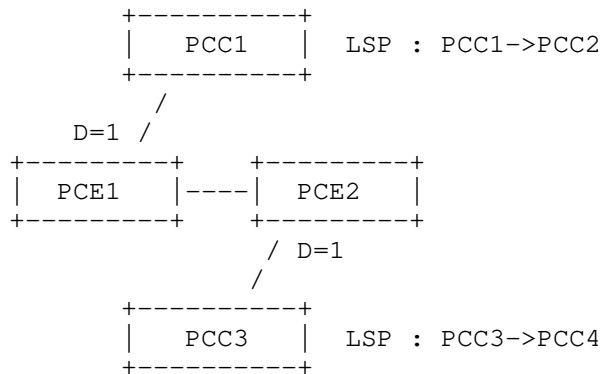
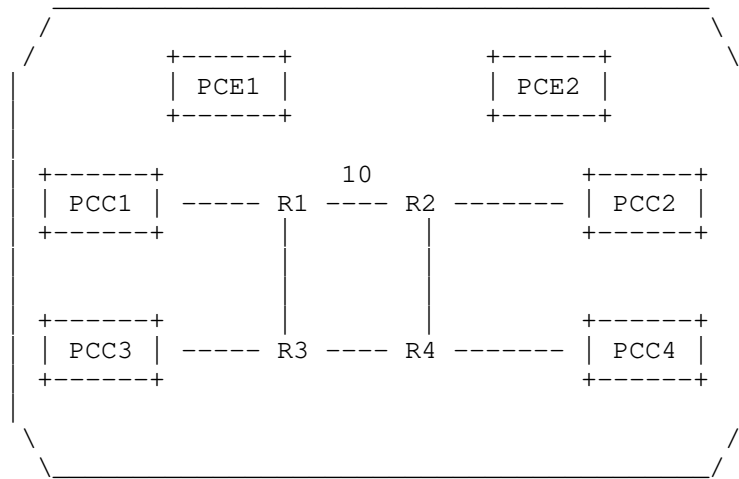
3.7. PCE initiation procedures

It is possible that a PCE does not have a PCEP session with the headend to initiate a LSP as per [RFC8281]. A PCE could send the PCInitiate message on the state-sync sessions to other PCE to request it to create a PCE-Initiated LSP on its behalf. If the PCE is able to initiate the LSP it would report it on the state-sync session via PCRpt message. If the PCE does not have a session to the headend, it MUST send a PCErr message with Error-type=24 (PCE instantiation error) and Error-value=TBD5 (No PCEP session with the headend). PCE could try to initiate via another state-sync PCE if available.

4. Examples

The examples in this section are for illustrative purpose to show how the behavior of the state sync inter-PCE sessions.

4.1. Example 1



PCE1 computation priority 100
PCE2 computation priority 200

Consider the PCEP sessions as shown above, where computation priority is global for all the LSPs and link disjoint between LSPs PCC1->PCC2 and PCC3->PCC4 is required.

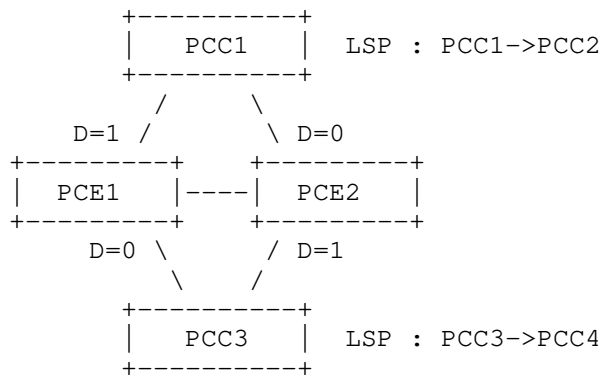
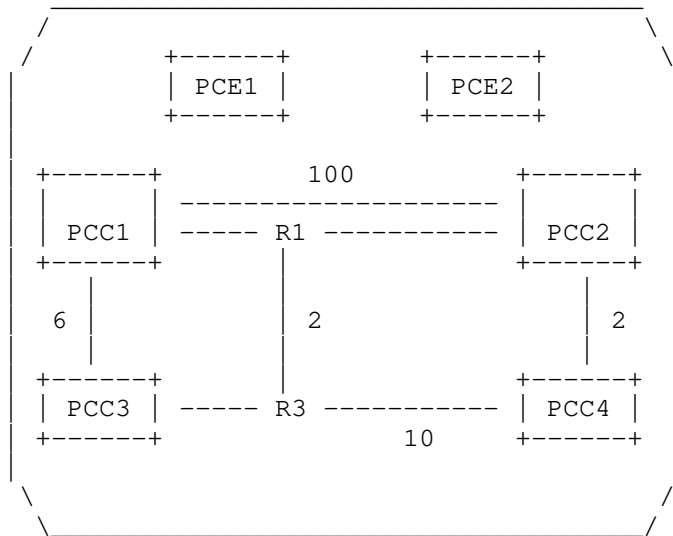
Consider the PCC1->PCC2 is configured first and PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it sub-delegates the LSP to PCE2 by sending a PCRpt with D=1 and including the SPEAKER-IDENTITY-TLV over the state-sync session. PCE2 receives the PCRpt and as it has delegation for this LSP, it computes the shortest path: R1->R3->R4->R2->PCC2. It then sends a PCUpd to PCE1 (including the SPEAKER-IDENTITY-TLV) with the computed ERO. PCE1 forwards the PCUpd to PCC1 (removing the SPEAKER-

IDENTITY-TLV). PCC1 acknowledges the PCUpd by a PCRpt to PCE1. PCE1 forwards the PCRpt to PCE2.

When PCC3->PCC4 is configured, PCC3 delegates the LSP to PCE2, PCE2 can compute a disjoint path as it has knowledge of both LSPs and has delegation also for both. The only solution found is to move PCC1->PCC2 LSP on another path, PCE2 can move PCC1->PCC2 as it has sub-delegation for it. It creates a new PCUpd with a new ERO: R1->R2-PCC2 towards PCE1 which forwards to PCC1. PCE2 sends a PCUpd to PCC3 with the path: R3->R4->PCC4.

In this set-up, PCEs are able to find a disjoint path while without state-sync and computation priority they could not.

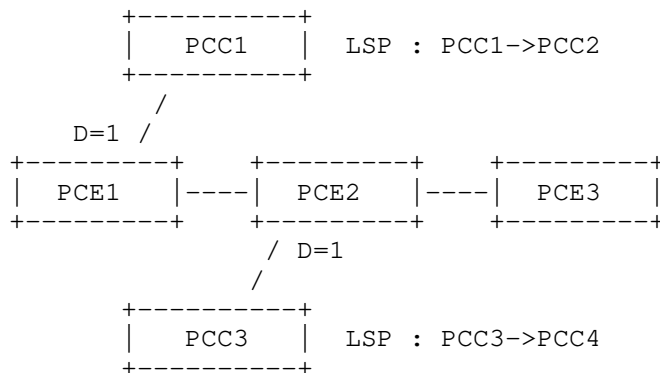
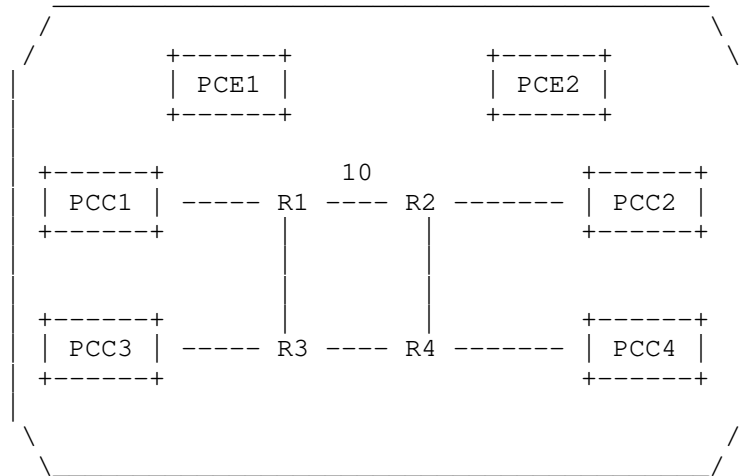
4.2. Example 2



PCE1 computation priority 200
 PCE2 computation priority 100

In this example, suppose both LSPs are configured almost at the same time. PCE1 sub-delegates PCC1->PCC2 to PCE2 while PCE2 keeps delegation for PCC3->PCC4, PCE2 computes a path for PCC1->PCC2 and PCC3->PCC4 and can achieve disjointness computation easily. No computation loop happens in this case.

4.3. Example 3



PCE1 computation priority 100
PCE2 computation priority 200
PCE3 computation priority 300

With the PCEP sessions as shown above, consider the need to have link disjoint LSPs PCC1->PCC2 and PCC3->PCC4.

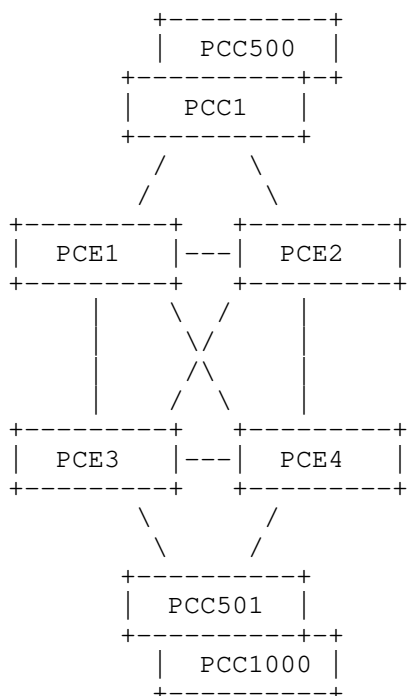
Suppose PCC1->PCC2 is configured first, PCC1 delegates the LSP to PCE1, but as PCE1 does not have the highest computation priority, it will sub-delegate the LSP to PCE2 (as it not aware of PCE3 and has no way to reach it). PCE2 cannot compute a path for PCC1->PCC2 as it does not have the highest priority and is not allowed to sub-delegate the LSP again towards PCE3 as per Section 3.

When PCC3->PCC4 is configured, PCC3 delegates the LSP to PCE2 that performs sub-delegation to PCE3. As PCE3 will have knowledge of only one LSP in the group, it cannot compute disjointness and can decide to fall-back to a less constrained computation to provide a path for PCC3->PCC4. In this case, it will send a PCUpd to PCE2 that will be forwarded to PCC3.

Disjointness cannot be achieved in this scenario because of lack of state-sync session between PCE1 and PCE3, but no computation loop happens. Thus it is advised for all PCEs that support state-sync to have a full mesh sessions between each other.

5. Using Primary/Secondary Computation and State-sync Sessions to increase Scaling

The Primary/Secondary computation and state-sync sessions architecture can be used to increase the scaling of the PCE architecture. If the number of PCCs is really high, it may be too resource consuming for a single PCE to maintain all the PCEP sessions while at the same time performing all path computations. Using primary/secondary computation and state-sync sessions may allow to create groups of PCEs that manage a subset of the PCCs and perform some or no path computations. Decoupling PCEP session maintenance and computation will allow increasing scaling of the PCE architecture.



In the figure above, two groups of PCEs are created: PCE1/2 maintain PCEP sessions with PCC1 up to PCC500, while PCE3/4 maintain PCEP sessions with PCC501 up to PCC1000. A granular primary/secondary policy is set-up as follows to load-share computation between PCEs:

- o PCE1 has priority 200 for association ID 1 up to 300, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.
- o PCE3 has priority 200 for association ID 301 up to 500, association source 0.0.0.0. All other PCEs have a decreasing priority for those associations.

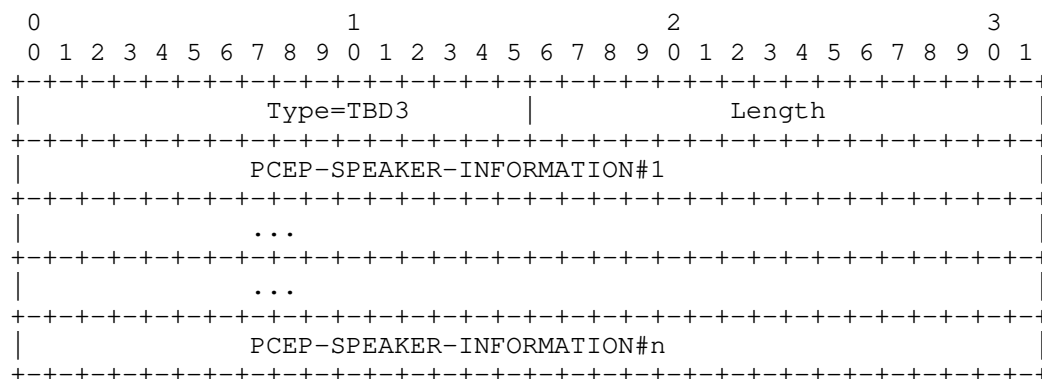
If some PCCs delegate LSPs with association ID 1 up to 300 and association source 0.0.0.0, the receiving PCE (if not PCE1) will sub-delegate the LSPs to PCE1. PCE1 becomes responsible for the computation of these LSP associations while PCE3 is responsible for the computation of another set of associations.

The procedures described in this document could help greatly in load-sharing between a group of stateful PCEs.

6. PCEP-PATH-VECTOR TLV

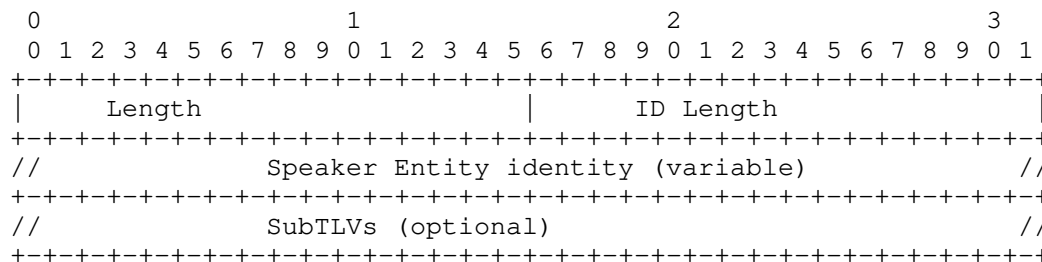
This document allows PCEP messages to be propagated among PCEP speaker. It may be useful to track information about the propagation of the messages. One of the use cases is a message loop detection mechanism, but other use cases like hop by hop information recording may also be implemented.

This document introduces the PCEP-PATH-VECTOR TLV (type TBD3) with the following format:



The TLV format and padding rules are as per [RFC5440].

The PCEP-SPEAKER-INFORMATION field has the following format:



Length: defines the total length of the PCEP-SPEAKER-INFORMATION field.

ID Length: defines the length of the Speaker identity actual field (non-padded).

Speaker Entity identity: same possible values as the SPEAKER-IDENTIFIER-TLV. Padded with trailing zeros to a 4-byte boundary.

The PCEP-SPEAKER-INFORMATION may also carry some optional subTLVs so each PCEP speaker can add local information that could be recorded. This document does not define any sub-TLV.

The PCEP-PATH-VECTOR TLV MAY be carried in the LSP Object. Its usage is purely optional.

The list of speakers within the PCEP-PATH-VECTOR TLV MUST be ordered. When sending a PCEP message (PCRpt, PCUpd, or PCInitiate), a PCEP Speaker MAY add the PCEP-PATH-VECTOR TLV with a PCEP-SPEAKER-INFORMATION containing its own information. If the PCEP message sent is the result of a previously received PCEP message, and if the PCEP-PATH-VECTOR TLV was already present in the initial message, the PCEP speaker MAY append a new PCEP-SPEAKER-INFORMATION containing its own information.

7. Security Considerations

The security considerations described in [RFC8231] and [RFC5440] apply to the extensions described in this document as well. Additional considerations related to state synchronization and sub-delegation between stateful PCEs are introduced, as it could be spoofed and could be used as an attack vector. An attacker could attempt to create too much state in an attempt to load the PCEP peer. The PCEP peer responds with a PCErr message as described in [RFC8231]. An attacker could impact LSP operations by creating bogus state. Further, state synchronization between stateful PCEs could provide an adversary with the opportunity to eavesdrop on the network. Thus, securing the PCEP session using Transport Layer Security (TLS) [RFC8253], as per the recommendations and best current practices in [RFC7525], is RECOMMENDED.

8. Acknowledgements

Thanks to [I-D.knodel-terminology] urging for better use of terms.

9. IANA Considerations

This document requests IANA actions to allocate code points for the protocol elements defined in this document.

9.1. PCEP-Error Object

IANA is requested to allocate a new Error Value for the Error Type 9.

Error-Type	Meaning	Reference
6	Mandatory Object Missing Error-value=TBD1: SPEAKER-IDENTITY-TLV missing	[RFC5440] This document
24	LSP instantiation error Error-value=TBD5: No PCEP session with the headend	[RFC8281] This document

9.2. PCEP TLV Type Indicators

IANA is requested to allocate new TLV Type Indicator values within the "PCEP TLV Type Indicators" sub-registry of the PCEP Numbers registry, as follows:

Value	Meaning	Reference
TBD2	ORIGINAL-LSP-DB-VERSION TLV	This document
TBD3	PCEP-PATH-VECTOR TLV	This document

9.3. STATEFUL-PCE-CAPABILITY TLV

IANA is requested to allocate a new bit value in the STATEFUL-PCE-CAPABILITY TLV Flag Field sub-registry.

Bit	Description	Reference
TBD4	INTER-PCE-CAPABILITY	This document

10. References

10.1. Normative References

- [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>>.
- [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>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.

10.2. Informative References

- [I-D.knodel-terminology] Knodel, M. and N. Oever, "Terminology, Power, and Inclusive Language in Internet-Drafts and RFCs", draft-knodel-terminology-04 (work in progress), August 2020.
- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <<https://www.rfc-editor.org/info/rfc6805>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.

- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <<https://www.rfc-editor.org/info/rfc8051>>.
- [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>>.
- [RFC8751] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., and D. King, "Hierarchical Stateful Path Computation Element (PCE)", RFC 8751, DOI 10.17487/RFC8751, March 2020, <<https://www.rfc-editor.org/info/rfc8751>>.
- [RFC8800] Litkowski, S., Sivabalan, S., Barth, C., and M. Negi, "Path Computation Element Communication Protocol (PCEP) Extension for Label Switched Path (LSP) Diversity Constraint Signaling", RFC 8800, DOI 10.17487/RFC8800, July 2020, <<https://www.rfc-editor.org/info/rfc8800>>.

Appendix A. Contributors

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

Email: dhruv.ietf@gmail.com

Authors' Addresses

Stephane Litkowski
Cisco

Email: slitkows.ietf@gmail.com

Siva Sivabalan
Ciena Corporation

Email: msiva282@gmail.com

Cheng Li
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: c.l@huawei.com

Haomian Zheng
Huawei Technologies
H1, Huawei Xiliu Beipo Village, Songshan Lake
Dongguan, Guangdong 523808
China

Email: zhenghaomian@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 6, 2021

A. Tokar
S. Sidor
Cisco Systems, Inc.
S. Sivabalan
Ciena
S. Peng
Huawei Technologies
M. Negi
RtBrick Inc
September 2, 2020

Carrying SID Algorithm information in PCE-based Networks.
draft-tokar-pce-sid-algo-02

Abstract

The Algorithm associated with a prefix Segment-ID (SID) defines the path computation Algorithm used by Interior Gateway Protocols (IGPs). This information is available to controllers such as the Path Computation Element (PCE) via topology learning. This document proposes an approach for informing headend routers regarding the Algorithm associated with each prefix SID used in PCE-computed paths, as well as signalling a specific SID algorithm as a constraint to the PCE.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 March 6, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	3
3. Object Formats	3
3.1. SR ERO Subobject	4
3.2. LSPA Object	4
4. Operation	5
4.1. SR-ERO NAI Encoding	5
4.2. SID Algorithm Constraint	5
5. Security Considerations	6
6. IANA Considerations	6
6.1. PCEP SR-ERO NAI Types	6
6.2. PCEP TLV Types	6
7. Normative References	6
Appendix A. Contributors	7
Authors' Addresses	7

1. Introduction

A PCE can compute SR-TE paths using prefix 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.

Similarly, 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. In addition, in the case of multiple (redundant) PCEs, when the headend receives a path from the primary PCE, it needs to be able to report the complete path information - including the Algorithm - to the backup PCE so that in

HA scenarios, the backup PCE can verify the prefix SIDs appropriately.

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. A new TLV is introduced for this purpose.

Refer to [RFC8665] and [RFC8667] for details about the prefix SID Algorithm.

This document introduces two new NAI types for the SR-ERO subobject, which is defined in [RFC8664]. A new TLV for signalling SID Algorithm constraint to the PCE is also introduced, to be carried inside the LSPA object, which is defined in [RFC5440].

The mechanisms described in this document are equally applicable to both SR-MPLS and SRv6.

2. Terminology

The following terminologies are used in this document:

ERO: Explicit Route Object

IGP: Interior Gateway Protocol

NAI: Node or Adjacency Identifier.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

SID: Segment Identifier.

SR: Segment Routing.

SR-TE: Segment Routing Traffic Engineering.

LSP: Label Switched Path.

LSPA: Label Switched Path Attributes.

3. Object Formats

3.1. SR ERO Subobject

The SR-ERO subobject encoding is extended with additional NAI types.

The following new NAI types (NT) are defined:

- o NT=TBD1: The NAI is an IPv4 node ID with Algorithm.
- o NT=TBD2: The NAI is an IPv6 node ID with Algorithm.

This document defines the following NAIs:

'IPv4 Node ID with Algorithm' is specified as an IPv4 address and Algorithm identifier. In this case, the NT value is TBD1 and the NAI field length is 8 octets.

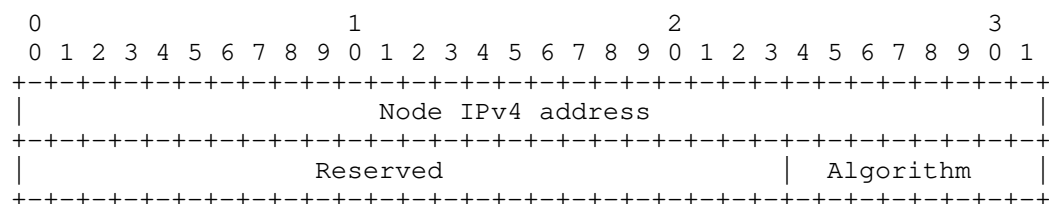


Figure 1: NAI for IPv4 Node SID with Algorithm

'IPv6 Node ID with Algorithm' is specified as an IPv6 address and Algorithm identifier. In this case, the NT value is TBD2 and the NAI field length is 20 octets.

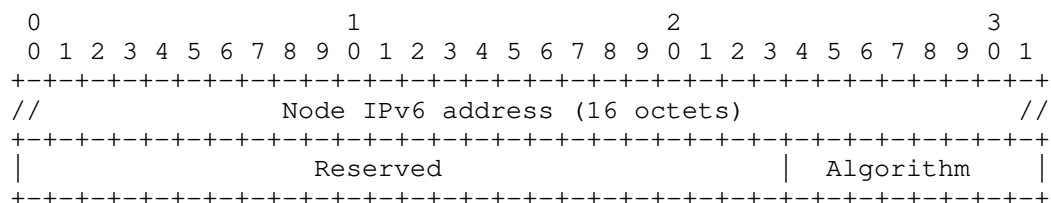


Figure 2: NAI for IPv6 Node SID with Algorithm

3.2. LSPA Object

A new TLV for the LSPA Object with TLV type=TBD3 is introduced to carry the SID Algorithm constraint. This TLV SHOULD only be used when PST (Path Setup type) = SR or SRv6.

The format of the SID Algorithm TLV is as follows:

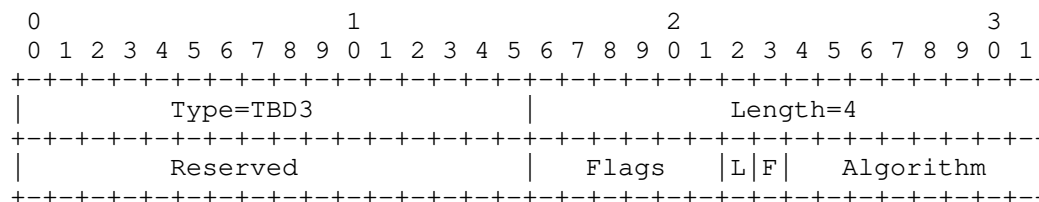


Figure 3: SID Algorithm TLV Format

The code point for the TLV type is TBD3. The TLV length is 4 octets.

The 32-bit value is formatted as follows.

Reserved: MUST be set to zero by the sender and MUST be ignored by the receiver.

Flags: This document defines the following flag bits. The other bits MUST be set to zero by the sender and MUST be ignored by the receiver.

- * **F (Fallback):** If set to 1 and the PCE is unable to compute a path using only prefix SIDs with the specified Algorithm, the PCE MAY compute an alternate fallback path without constraining to the specified Algorithm.
- * **L (Loose):** If set to 1, the PCE MAY insert prefix SIDs with a different Algorithm, but it MUST prefer the specified Algorithm whenever possible.

Algorithm: SID Algorithm the PCE MUST take into account while computing a path for the LSP.

4. Operation

4.1. SR-ERO NAI Encoding

IPv4 prefix SIDs used by SR-TE paths with an associated Algorithm SHOULD be encoded with 'IPv4 Node ID with Algorithm' NAI.

IPv6 prefix SIDs used by SR-TE paths with an associated Algorithm SHOULD be encoded with 'IPv6 Node ID with Algorithm' NAI.

4.2. SID Algorithm Constraint

In order to signal a specific SID Algorithm constraint to the PCE, the headend MUST encode the SID ALGORITHM TLV inside the LSPA object.

When the PCE receives a SID Algorithm constraint, it MUST only take prefix SIDs with the specified Algorithm into account during path computation. However, if the L flag is set in the SID Algorithm TLV, the PCE MAY insert prefix SIDs with a different Algorithm in order to successfully compute a path.

If the PCE is unable to find a path with the given SID Algorithm constraint, it MUST bring the LSP down. Alternatively, if the F flag is set in the SID Algorithm TLV, the PCE MAY attempt to compute a path without taking the Algorithm constraint into account at all.

5. Security Considerations

No additional security measure is required.

6. IANA Considerations

6.1. PCEP SR-ERO NAI Types

IANA is requested to allocate new SR-ERO NAI types for the new NAI types specified in this document.

Value	Description	Reference
TBD1	IPv4 Node ID with Algorithm	This document
TBD2	IPv6 Node ID with Algorithm	This document

6.2. PCEP TLV Types

IANA is requested to allocate a new TLV type for the new LSPA TLV specified in this document.

Value	Description	Reference
TBD3	SID Algorithm	This document

7. Normative References

- [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>>.
- [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>>.
- [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>>.
- [RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <<https://www.rfc-editor.org/info/rfc8665>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.

Appendix A. Contributors

Mike Koldychev
Cisco Systems
Kanata, Ontario
Canada

Email: mkoldych@cisco.com

Authors' Addresses

Alex Tokar
Cisco Systems, Inc.
Eurovea Central 3.
Pribinova 10
Bratislava 811 09
Slovakia

Email: atokar@cisco.com

Samuel Sidor
Cisco Systems, Inc.
Eurovea Central 3.
Pribinova 10
Bratislava 811 09
Slovakia

Email: ssidor@cisco.com

Siva Sivabalan
Ciena
385 Terry Fox Drive
Kanata, Ontario K2K 0L1
Canada

Email: msiva282@gmail.com

Shuping Peng
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing 100095
China

Email: pengshuping@huawei.com

Mahendra Singh Negi
RtBrick Inc
Bangalore, Karnataka
India

Email: mahend.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: 7 April 2022

A. Tokar
S. Sidor
Cisco Systems, Inc.
S. Peng
ZTE Corporation
S. Sivabalan
Ciena
T. Saad
Juniper Networks
S. Peng
Huawei Technologies
M. Negi
RtBrick Inc
4 October 2021

Carrying SID Algorithm information in PCE-based Networks.
draft-tokar-pce-sid-algo-05

Abstract

The Algorithm associated with a prefix Segment-ID (SID) defines the path computation Algorithm used by Interior Gateway Protocols (IGPs). This information is available to controllers such as the Path Computation Element (PCE) via topology learning. This document proposes an approach for informing headend routers regarding the Algorithm associated with each prefix SID used in PCE-computed paths, as well as signalling a specific SID algorithm as a constraint to the PCE.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

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 7 April 2022.

Copyright Notice

Copyright (c) 2021 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Terminology	3
3. Object Formats	4
3.1. OPEN Object	4
3.1.1. SR PCE Capability Sub-TLV	4
3.1.2. SRv6 PCE Capability sub-TLV	4
3.2. SR-ERO Subobject	5
3.3. SRv6-ERO Subobject	5
3.4. LSPA Object	6
4. Operation	7
4.1. SR-ERO and SRv6-ERO Encoding	7
4.2. SID Algorithm Constraint	7
5. Security Considerations	7
6. IANA Considerations	8
6.1. SR Capability Flag	8
6.2. SRv6 PCE Capability Flag	8
6.3. SR-ERO Flag	8
6.4. SRv6-ERO Flag	9
6.5. PCEP TLV Types	9
7. Normative References	9
Appendix A. Contributors	10
Authors' Addresses	11

1. Introduction

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.

Similarly, 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. In addition, in the case of multiple (redundant) PCEs, when the headend receives a path from the primary PCE, it needs to be able to report the complete path information - including the Algorithm - to the backup PCE so that in HA scenarios, the backup PCE can verify the prefix SIDs appropriately.

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. A new TLV is introduced for this purpose.

Refer to [RFC8665] and [RFC8667] for details about the prefix SID Algorithm.

This document is extending:

- * the SR PCE Capability Sub-TLV and the SR-ERO subobject - defined in [RFC8664]
- * the SRv6 PCE Capability sub-TLV and the SRv6-ERO subobject - defined in [I-D.ietf-pce-segment-routing-ipv6]

A new TLV for signalling SID Algorithm constraint to the PCE is also introduced, to be carried inside the LSPA object, which is defined in [RFC5440].

The mechanisms described in this document are equally applicable to both SR-MPLS and SRv6.

2. Terminology

The following terminologies are used in this document:

ERO: Explicit Route Object

IGP: Interior Gateway Protocol

NAI: Node or Adjacency Identifier.

PCE: Path Computation Element

PCEP: Path Computation Element Protocol.

SID: Segment Identifier.

SR: Segment Routing.

SR-TE: Segment Routing Traffic Engineering.

LSP: Label Switched Path.

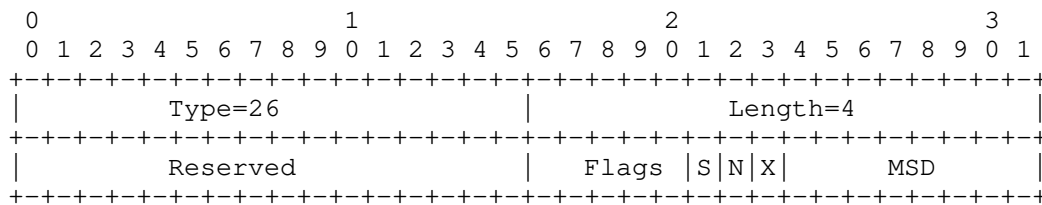
LSPA: Label Switched Path Attributes.

3. Object Formats

3.1. OPEN Object

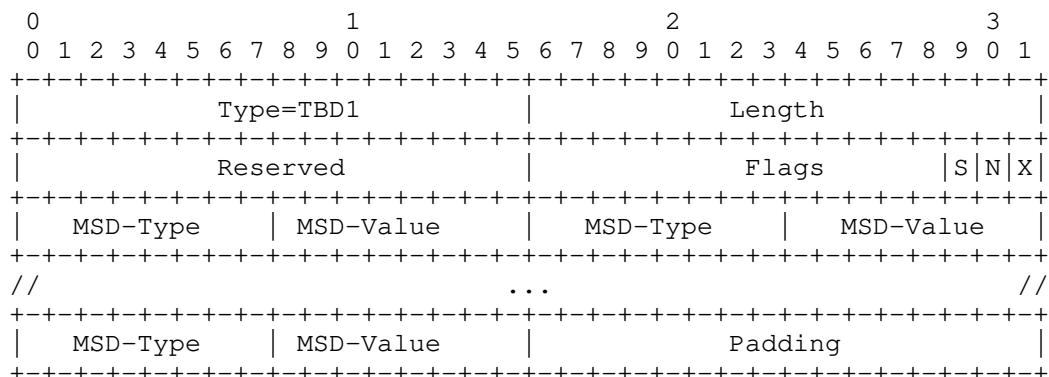
3.1.1. SR PCE Capability Sub-TLV

A new flag S is proposed in the SR PCE Capability Sub-TLV introduced in Section 4.1.2 of [RFC8664] in Path Computation Element Communication Protocol (PCEP) to indicate support for SID Algorithm field in the SR-ERO subobject.



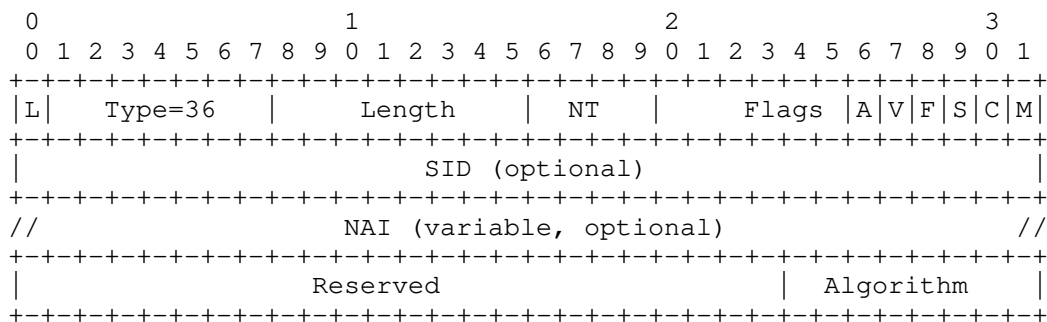
3.1.2. SRv6 PCE Capability sub-TLV

A new flag S is proposed in the SRv6 PCE Capability sub-TLV introduced in 4.1.1 of [I-D.ietf-pce-segment-routing-ipv6] in Path Computation Element Communication Protocol (PCEP) to indicate support for SID Algorithm field in the SRv6-ERO subobject.



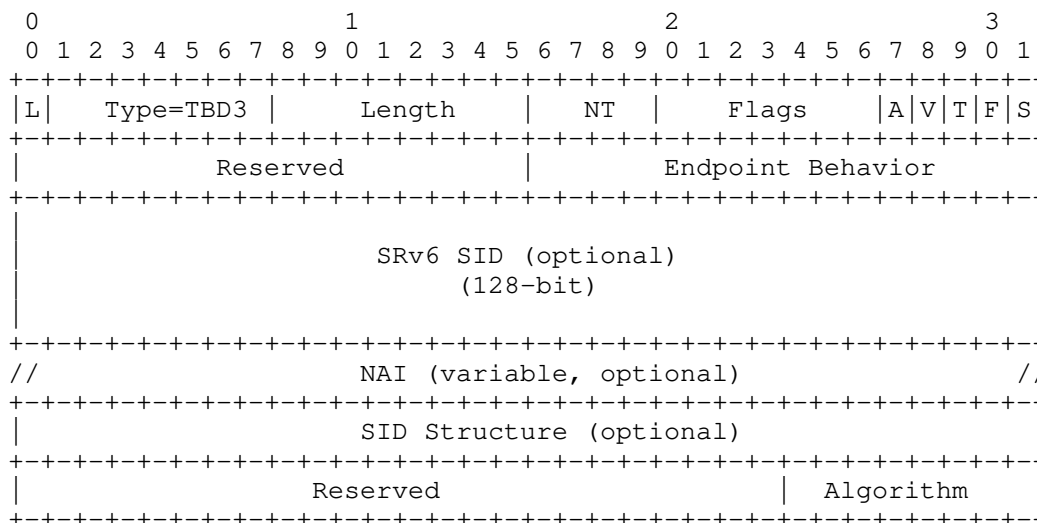
3.2. SR-ERO Subobject

The SR-ERO subobject encoding is extended with new flag "A" to indicate if the Algorithm field is included after other optional fields.



3.3. SRv6-ERO Subobject

The SRv6-ERO subobject encoding is extended with new flag "A" to indicate if the Algorithm field is included after other optional fields.



3.4. LSPA Object

A new TLV for the LSPA Object with TLV type=TBD3 is introduced to carry the SID Algorithm constraint. This TLV SHOULD only be used when PST (Path Setup type) = SR or SRv6.

The format of the SID Algorithm TLV is as follows:

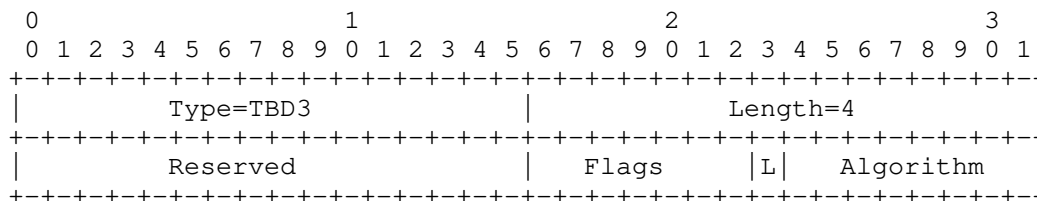


Figure 1: SID Algorithm TLV Format

The code point for the TLV type is TBD3. The TLV length is 4 octets.

The 32-bit value is formatted as follows.

Reserved: MUST be set to zero by the sender and MUST be ignored by the receiver.

Flags: This document defines the following flag bits. The other bits MUST be set to zero by the sender and MUST be ignored by the receiver.

- * L (Loose): If set to 1, the PCE MAY insert SIDs with a different Algorithm, but it MUST prefer the specified Algorithm whenever possible.

Algorithm: SID Algorithm the PCE MUST take into account while computing a path for the LSP.

4. Operation

4.1. SR-ERO and SRv6-ERO Encoding

PCEP speaker MAY set the A flag and include the Algorithm field in SR-ERO or SRv6-ERO subobject if the S flag was advertised by both PCEP speakers.

If PCEP peer receives SR-ERO subobject with the A flag set or with the SID Algorithm included, but the S flag was not advertised, then such PCEP message must be rejected with PCErrror as described in Section 7.2 of [RFC5440]

The Algorithm field MUST be included after optional SID, NAI or SID structure and length of SR-ERO or SRv6-ERO subobject MUST be increased with additional 4 bytes for Reserved and Algorithm field.

4.2. SID Algorithm Constraint

In order to signal a specific SID Algorithm constraint to the PCE, the headend MUST encode the SID ALGORITHM TLV inside the LSPA object.

When the PCE receives a SID Algorithm constraint, it MUST only take prefix SIDs with the specified Algorithm into account during path computation. However, if the L flag is set in the SID Algorithm TLV, the PCE MAY insert prefix SIDs with a different Algorithm in order to successfully compute a path.

If the PCE is unable to find a path with the given SID Algorithm constraint, it MUST bring the LSP down.

SID Algorithm does not replace the Objective Function defined in [RFC5541]. The SID Algorithm constraint acts as a filter, restricting which SIDs may be used as a result of the path computation function.

5. Security Considerations

No additional security measure is required.

6. IANA Considerations

6.1. SR Capability Flag

IANA maintains a sub-registry, named "SR Capability Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flags field of the SR-PCE-CAPABILITY TLV. IANA is requested to make the following assignment:

Value	Description	Reference
TBD1	SID Algorithm Capability	This document

Table 1

6.2. SRv6 PCE Capability Flag

IANA was requested in [I-D.ietf-pce-segment-routing-ipv6] to create a sub-registry, named "SRv6 PCE Capability Flags", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flags field of SRv6-PCE-CAPABILITY sub-TLV. IANA is requested to make the following assignment:

Value	Description	Reference
TBD2	SID Algorithm Capability	This document

Table 2

6.3. SR-ERO Flag

IANA maintains a sub-registry, named "SR-ERO Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flags field of the SR-ERO Subobject. IANA is requested to make the following assignment:

Value	Description	Reference
TBD3	SID Algorithm Flag	This document

Table 3

6.4. SRv6-ERO Flag

IANA was requested in [I-D.ietf-pce-segment-routing-ipv6], named "SRv6-ERO Flag Field", within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flags field of the SRv6-ERO subobject. IANA is requested to make the following assignment:

Value	Description	Reference
TBD4	SID Algorithm Flag	This document

Table 4

6.5. PCEP TLV Types

IANA is requested to allocate a new TLV type for the new LSPA TLV specified in this document.

Value	Description	Reference
TBD5	SID Algorithm	This document

Table 5

7. Normative References

[I-D.ietf-pce-segment-routing-ipv6]
 Li, C., Negi, M., Sivabalan, S., Koldychev, M.,
 Kaladharan, P., and Y. Zhu, "PCEP Extensions for Segment
 Routing leveraging the IPv6 data plane", Work in Progress,
 Internet-Draft, draft-ietf-pce-segment-routing-ipv6-09, 27
 May 2021, <<https://www.ietf.org/internet-drafts/draft-ietf-pce-segment-routing-ipv6-09.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>>.
- [RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, DOI 10.17487/RFC5541, June 2009, <<https://www.rfc-editor.org/info/rfc5541>>.
- [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>>.
- [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>>.
- [RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <<https://www.rfc-editor.org/info/rfc8665>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.

Appendix A. Contributors

Mike Koldychev
Cisco Systems
Kanata, Ontario
Canada

Email: mkoldych@cisco.com

Authors' Addresses

Alex Tokar
Cisco Systems, Inc.
Eurovea Central 3.
Pribinova 10
811 09 Bratislava
Slovakia

Email: atokar@cisco.com

Samuel Sidor
Cisco Systems, Inc.
Eurovea Central 3.
Pribinova 10
811 09 Bratislava
Slovakia

Email: ssidor@cisco.com

Shaofu Peng
ZTE Corporation
No.50 Software Avenue
Nanjing
Jiangsu, 210012
China

Email: peng.shaofu@zte.com.cn

Siva Sivabalan
Ciena
385 Terry Fox Drive
Kanata Ontario K2K 0L1
Canada

Email: msiva282@gmail.com

Tarek Saad
Juniper Networks

Email: tsaad@juniper.net

Shuping Peng
Huawei Technologies
Huawei Campus, No. 156 Beiqing Rd.
Beijing
100095
China

Email: pengshuping@huawei.com

Mahendra Singh Negi
RtBrick Inc
Bangalore
Karnataka
India

Email: mahend.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 5, 2021

Z. Li
S. Peng
Huawei Technologies
M. Negi
RtBrick Inc
Q. Zhao
Etheric Networks
C. Zhou
HPE
November 1, 2020

PCEP Procedures and Protocol Extensions for Using PCE as a Central
Controller (PCECC) of SR-LSPs
draft-zhao-pce-pcep-extension-pce-controller-sr-08

Abstract

The Path Computation Element (PCE) is a core component of Software-Defined Networking (SDN) systems. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands.

PCE was developed to derive paths for MPLS Label Switched Paths (LSPs), which are supplied to the head end of the LSP using the Path Computation Element Communication Protocol (PCEP). But SDN has a broader applicability than signaled (G)MPLS traffic-engineered (TE) networks, and the PCE may be used to determine paths in a range of use cases. PCEP has been proposed as a control protocol for use in these environments to allow the PCE to be fully enabled as a central controller.

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. Thus, the LSP can be calculated/set up/initiated and the label forwarding entries can also be downloaded through a centralized PCE server to each network device along the path while leveraging the existing PCE technologies as much as possible.

This document specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers, in addition to computing the paths for packet flows in a segment routing (SR) network and telling the edge routers what instructions to attach to packets as they enter the network.

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 May 5, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	5
2. Terminology	5
3. PCECC SR	5
4. PCEP Requirements	6
5. Procedures for Using the PCE as a Central Controller (PCECC) in Segment Routing	6
5.1. Stateful PCE Model	6
5.2. New LSP Functions	6
5.3. PCECC Capability Advertisement	7
5.4. PCEP session IP address and TED Router ID	7
5.5. LSP Operations	8
5.5.1. PCECC Segment Routing (SR)	8
5.5.1.1. PCECC SR Node/Prefix SID allocation	8

5.5.1.2.	PCECC SR Adjacency Label allocation	10
5.5.1.3.	Redundant PCEs	12
5.5.1.4.	Re Delegation and Clean up	12
5.5.1.5.	Synchronization of Label Allocations	13
5.5.1.6.	PCC-Based Allocations	13
5.5.1.7.	Binding SID	13
6.	PCEP Messages	14
6.1.	Central Control Instructions	14
6.1.1.	The PCInitiate Message	14
6.1.2.	The PCRpt message	15
7.	PCEP Objects	16
7.1.	OPEN Object	16
7.1.1.	PCECC Capability sub-TLV	16
7.2.	SR-TE Path Setup	17
7.3.	CCI Object	17
7.4.	FEC Object	19
8.	Implementation Status	21
8.1.	Huawei's Proof of Concept based on ONOS	22
9.	Security Considerations	22
10.	Manageability Considerations	22
10.1.	Control of Function and Policy	22
10.2.	Information and Data Models	23
10.3.	Liveness Detection and Monitoring	23
10.4.	Verify Correct Operations	23
10.5.	Requirements On Other Protocols	23
10.6.	Impact On Network Operations	23
11.	IANA Considerations	23
11.1.	PCECC-CAPABILITY sub-TLV	23
11.2.	PCEP Object	24
11.3.	PCEP-Error Object	24
11.4.	CCI Object Flag Field for SR	24
12.	Acknowledgments	25
13.	References	25
13.1.	Normative References	25
13.2.	Informative References	27
Appendix A.	Contributor Addresses	30
Authors'	Addresses	31

1. Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload the path computation function from routers in an MPLS traffic-engineered network. Since then, the role and function of the PCE has grown to cover a number of other uses (such as GMPLS [RFC7025]) and to allow delegated control [RFC8231] and PCE-initiated use of network resources [RFC8281].

According to [RFC7399], Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system, called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place traffic flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths for MPLS Label Switched Paths (LSPs), paths were requested by network elements (known as Path Computation Clients (PCCs)), and the results of the path computations were supplied to network elements using the Path Computation Element Communication Protocol (PCEP) [RFC5440]. This protocol was later extended to allow a PCE to send unsolicited requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE and PCC. [RFC8283] further examines the motivations and applicability for PCEP as a Southbound Interface (SBI), and introduces the implications for the protocol. [I-D.ietf-teas-pcecc-use-cases] describes the use cases for the PCE-based Central Controller (PCECC) architecture.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify the procedures and PCEP extensions for using the PCE as the central controller for static LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path.

Segment Routing (SR) technology leverages the source routing and tunneling paradigms. A source node can choose a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as a set of "segments" advertised by link-state routing protocols (IS-IS or OSPF). [RFC8402] provides an introduction to SR architecture. The corresponding IS-IS and OSPF extensions are specified in [RFC8667] and [RFC8665], respectively. It relies on a series of forwarding instructions being placed in the header of a packet. The segment routing architecture supports operations that can be used to steer packet flows in a network, thus providing a form of traffic engineering. [RFC8664] specify the SR specific PCEP extensions.

PCECC may further use PCEP for SR SID (Segment Identifier) distribution on the SR nodes with some benefits.

This document specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers (SR SID distribution in this case), in addition to computing the paths for packet flows in a segment routing network and telling the edge routers what instructions to attach to packets as they enter the network.

Only SR using MPLS dataplane (SR-MPLS) is in the scope of this document. Refer [I-D.dhody-pce-pcep-extension-pce-controller-srv6] for use of PCECC technique for SR in IPv6 (SRv6) dataplane.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

Terminologies used in this document is the same as described in the draft [RFC8283] and [I-D.ietf-teas-pcecc-use-cases].

3. PCECC SR

[RFC8664] specifies extensions to PCEP that allow a stateful PCE to compute, update, or initiate SR-TE paths. An ingress node of an SR-TE path appends all outgoing packets with a list of MPLS labels (SIDs). This is encoded in SR-ERO subobject, capable of carrying a label (SID) as well as the identity of the node/adjacency label (SID).

The notion of segment and SID is defined in [RFC8402], which fits the MPLS architecture [RFC3031] as the label which is managed by a local allocation process of LSR (similarly to other MPLS signaling protocols) [RFC8660]. The SR information such as node/adjacency label (SID) is flooded via IGP as specified in [RFC8667] and [RFC8665].

As per [RFC8283], PCE as a central controller can allocate and provision the node/prefix/adjacency label (SID) via PCEP.

The rest of the processing is similar to existing stateful PCE with SR mechanism.

For the purpose of this document, it is assumed that the label range to be used by a PCE is set on both PCEP peers. Further, a global label range is assumed to be set on all PCEP peers in the SR domain. This document also allows a case where the label space is maintained by PCC itself, and the labels are allocated by the PCC, in this case, the PCE should request the allocation from PCC as described in Section 5.5.1.6.

4. PCEP Requirements

Following key requirements for PCECC-SR should be considered when designing the PCECC-based solution:

- o A PCEP speaker supporting this draft needs to have the capability to advertise its PCECC-SR capability to its peers.
- o PCEP procedures need to allow for PCC-based label/SID allocations.
- o PCEP procedures need means to update (or clean up) the label-map entry to the PCC.
- o PCEP procedures need to provide a mean to synchronize the SR labels allocations between the PCE to the PCC via PCEP messages.

5. Procedures for Using the PCE as a Central Controller (PCECC) in Segment Routing

5.1. Stateful PCE Model

Active stateful PCE is described in [RFC8231]. PCE as a Central Controller (PCECC) reuses the existing active stateful PCE mechanism as much as possible to control the LSPs.

5.2. New LSP Functions

Several new functions are required in PCEP to support PCECC as described in [I-D.ietf-pce-pcep-extension-for-pce-controller]. This document reuses the existing messages to support PCECC-SR.

The PCEP messages PCRpt, PCInitiate, PCUpd are used to send LSP Reports, LSP setup, and LSP update respectively. The extended PCInitiate message described in [I-D.ietf-pce-pcep-extension-for-pce-controller] is used to download or clean up central controller's instructions (CCIs) (SR SID in the scope of this document). The extended PCRpt message described in [I-D.ietf-pce-pcep-extension-for-pce-controller] is also used to report the CCIs (SR SIDs) from PCC to PCE.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify an object called CCI for the encoding of the central controller's instructions. This document extends the CCI by defining a new object-type for segment routing. The PCEP messages are extended in this document to handle the PCECC operations for SR.

5.3. PCECC Capability Advertisement

During PCEP Initialization Phase, PCEP Speakers (PCE or PCC) advertise their support of PCECC extensions. A PCEP Speaker includes the "PCECC Capability" sub-TLV, described in [I-D.ietf-pce-pcep-extension-for-pce-controller].

A new S-bit is added in the PCECC-CAPABILITY sub-TLV to indicate support for PCECC-SR. A PCC MUST set the S-bit in the PCECC-CAPABILITY sub-TLV and include the SR-PCE-CAPABILITY sub-TLV ([RFC8664]) in the OPEN Object (inside the PATH-SETUP-TYPE-CAPABILITY TLV) to support the PCECC SR extensions defined in this document. If the S-bit is set in the PCECC-CAPABILITY sub-TLV and the SR-PCE-CAPABILITY sub-TLV is not advertised in the OPEN Object, PCE SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TB4 (SR capability was not advertised) and terminate the session.

The rest of the processing is as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

5.4. PCEP session IP address and TED Router ID

A PCE may construct its Traffic Engineering Database (TED) by participating in the IGP ([RFC3630] and [RFC5305] for MPLS-TE; [RFC4203] and [RFC5307] for GMPLS). An alternative is offered by BGP-LS [RFC7752] and [I-D.dhodylee-pce-pcep-ls].

A PCEP [RFC5440] speaker could use any local IP address while creating a TCP session. It is important to link the session IP address with the Router ID in TED for successful PCECC operations.

During PCEP Initialization Phase, the PCC SHOULD advertise the TE mapping information by including the "Node Attributes TLV" [I-D.dhodylee-pce-pcep-ls] with "IPv4/IPv6 Router-ID of Local Node", in the OPEN Object for this purpose. [RFC7752] describes the usage as auxiliary Router-IDs that the IGP might be using, e.g., for TE purposes. If there are more than one auxiliary Router-ID of a given type, then multiple TLVs are used to encode them.

If "IPv4/IPv6 Router-ID" TLV is not present, the TCP session IP address is directly used for mapping purpose.

5.5. LSP Operations

[RFC8664] specify the PCEP extension to allow a stateful PCE to compute and initiate SR-TE paths, as well as a PCC to request a path subject to certain constraint(s) and optimization criteria in SR networks.

The Path Setup Type for segment routing (PST=1) is used on the PCEP session with the Ingress as per [RFC8664].

5.5.1. PCECC Segment Routing (SR)

Segment Routing (SR) as described in [RFC8402] depends on "segments" that are advertised by Interior Gateway Protocols (IGPs). The SR-node allocates and advertises the SID (node, adj, etc) and flood them via the IGP. This document proposes a new mechanism where PCE allocates the SID (label/index/SID) centrally and uses PCEP to advertise them. In some deployments, PCE (and PCEP) are better suited than IGP because of the centralized nature of PCE and direct TCP based PCEP sessions to the node.

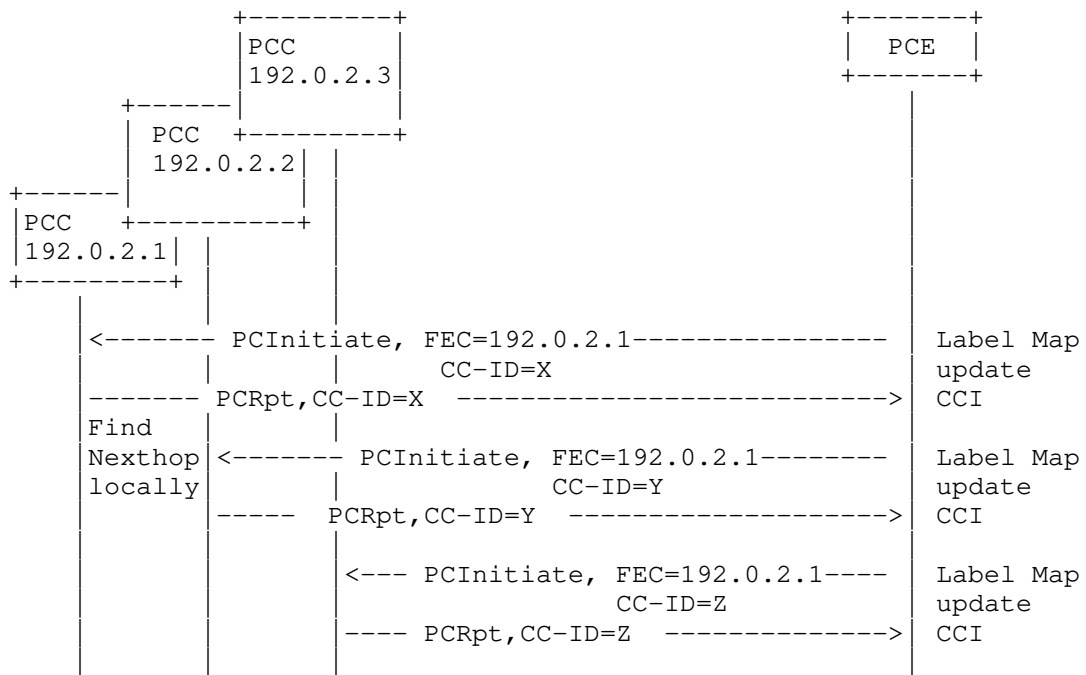
5.5.1.1. PCECC SR Node/Prefix SID allocation

Each node (PCC) is allocated a node-SID by the PCECC. The PCECC sends PCInitiate message to update the label map of each node to all the nodes in the domain. The TE router ID is determined from the TED or from "IPv4/IPv6 Router-ID" Sub-TLV [I-D.dhodylee-pce-pcep-ls], in the OPEN Object Section 5.4.

It is RECOMMENDED that PCEP session with PCECC-SR capability to use a different session IP address during TCP session establishment than the node Router ID in TEDB, to make sure that the PCEP session does not get impacted by the SR Node/Prefix Label maps (Section 5.4).

If a node (PCC) receives a PCInitiate message with a CCI encoding a SID, out of the range set aside for the SR Global Block (SRGB), it MUST send a PCErr message with Error-type=TBD (PCECC failure) and Error-value=TBD (Label out of range) (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]) and MUST include the SRP object to specify the error is for the corresponding central control instruction via the PCInitiate message.

On receiving the label map, each node (PCC) uses the local routing information to determine the next-hop and download the label forwarding instructions accordingly. The PCInitiate message in this case does not use the LSP object but uses a new FEC object defined in this document.

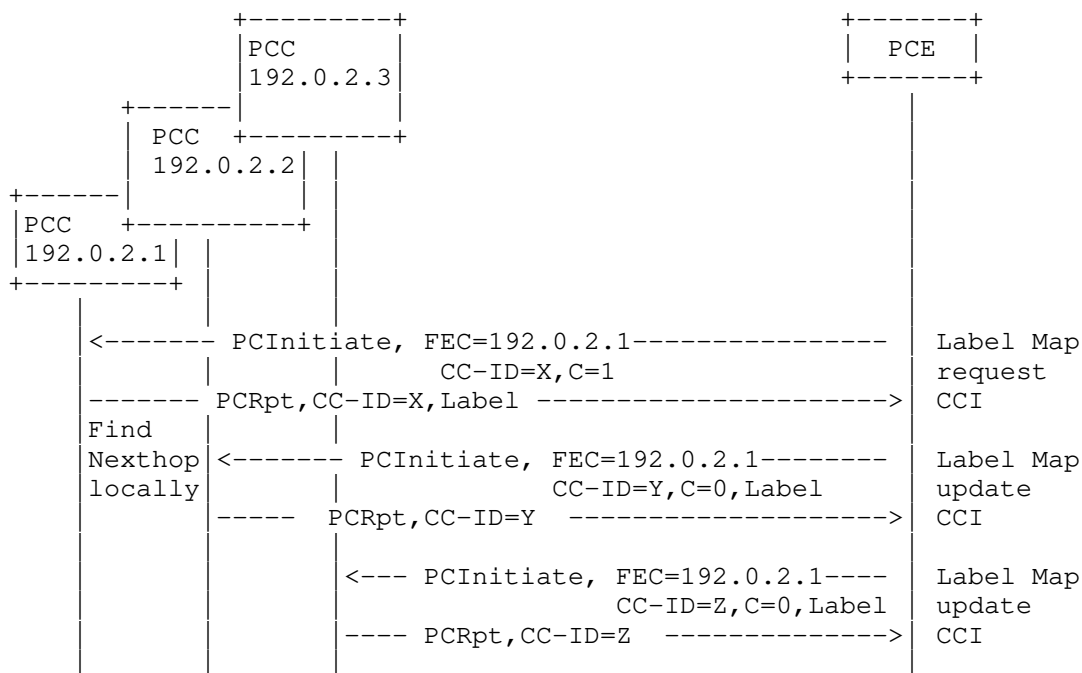


The forwarding behavior and the end result is similar to IGP based "Node-SID" in SR. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node as per [RFC8402].

PCE relies on the Node/Prefix Label clean up using the same PCInitiate message as per [RFC8281].

The above example Figure 1 depicts the FEC and PCEP speakers that uses IPv4 address. Similarly an IPv6 address (such as 2001:DB8::1) can be used during PCEP session establishment in the FEC object as described in this specification.

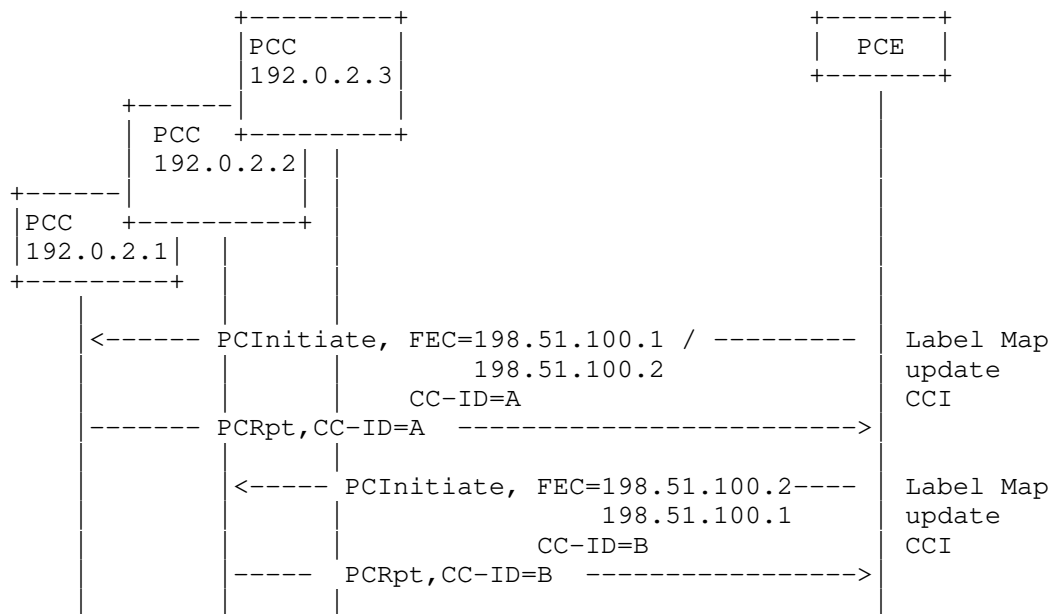
In the case where the label/SID allocation is made by the PCC itself (see Section 5.5.1.6), the PCE could request an allocation to be made by the PCC, and where the PCC would send a PCRpt with the allocated label/SID encoded in the CC-ID object as shown in Figure 2.



It should be noted that in this example, the request is made to the node 192.0.2.1 with C bit set in the CCI object to indicate that the allocation needs to be done by this PCC and it responds with the allocated label/SID to the PCE. The PCE would further inform the other PCCs in the network about the label-map allocation without setting the C bit.

5.5.1.2. PCECC SR Adjacency Label allocation

For PCECC-SR, apart from node-SID, Adj-SID is used where each adjacency is allocated an Adj-SID by the PCECC. The PCECC sends the PCInitiate message to update the label map of each adjacency to the corresponding nodes in the domain. Each node (PCC) download the label forwarding instructions accordingly. Similar to SR Node/Prefix Label allocation, the PCInitiate message in this case does not use the LSP object but uses the new FEC object defined in this document.



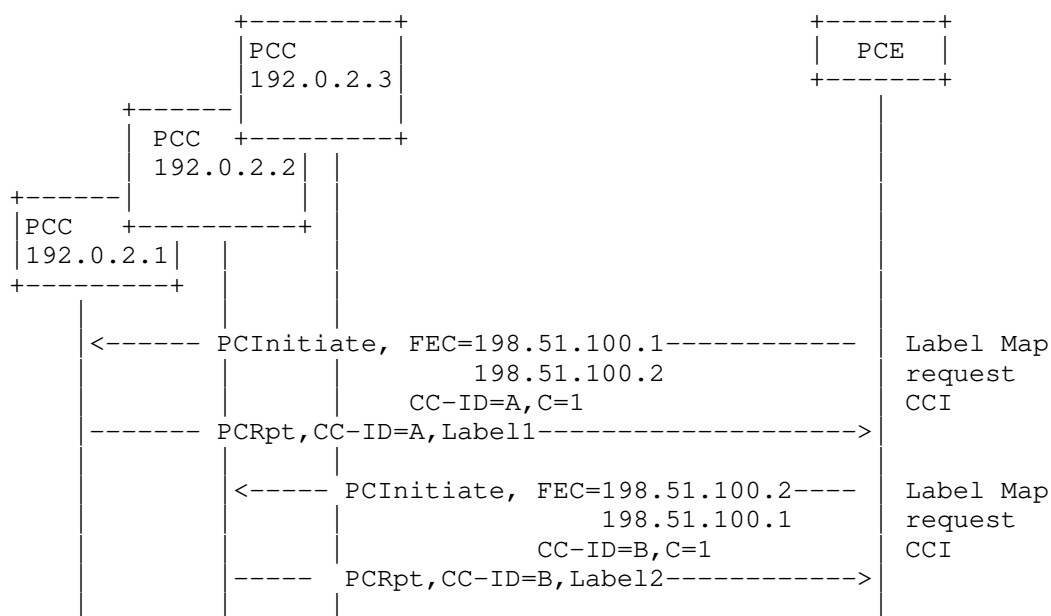
The forwarding behavior and the end result is similar to IGP based "Adj-SID" in SR.

PCE relies on the Adj label clean up using the same PCInitiate message as per [RFC8281].

The above example Figure 3 depicts FEC object and PCEP speakers that uses an IPv4 address. Similarly an IPv6 address (such as 2001:DB8::1, 2001:DB8::2) can be used during the PCEP session establishment in the FEC object as described in this specification.

The handling of adjacencies on the LAN subnetworks is specified in [RFC8402]. PCECC MUST assign Adj-SID for every pair of routers in the LAN. The rest of the protocol mechanism remains the same.

In the case where the label/SID map allocation is made by the PCC itself (see Section 5.5.1.6), the PCE could request an allocation to be made by the PCC, and where the PCC would send a PCRpt with the allocated label/SID encoded in the CC-ID object as shown in Figure 4.



In this example, the request is made to the node 192.0.2.1 with the C bit set in the CCI object to indicate that the allocation needs to be done by this PCC for the adjacency (198.51.100.1 - 198.51.100.2) and it responds with the allocated label/SID to the PCE. Similarly, another request is made to the node 192.0.2.2 with the C bit set in the CCI object to indicate that the allocation needs to be done by this PCC for the adjacency (198.51.100.2 - 198.51.100.1).

5.5.1.3. Redundant PCEs

[I-D.litkowski-pce-state-sync] describes the synchronization mechanism between the stateful PCEs. The SR SIDs allocated by a PCE MUST also be synchronized among PCEs for PCECC SR state synchronization. Note that the SR SIDs are independent of the SR-TE LSPs, and remains intact till any topology change. The redundant PCEs MUST have a common view of all SR SIDs allocated in the domain.

5.5.1.4. Re Delegation and Clean up

[I-D.ietf-pce-pcep-extension-for-pce-controller] describes the action needed for CCIs for the static LSPs on a terminated session. Same holds true for the CCI for SR SID as well.

5.5.1.5. Synchronization of Label Allocations

[I-D.ietf-pce-pcep-extension-for-pce-controller] describes the synchronization of Central Controller's Instructions (CCI) via LSP state synchronization as described in [RFC8231] and [RFC8232]. Same procedures are applied for the CCI for SR SID as well.

5.5.1.6. PCC-Based Allocations

The PCE can request the PCC to allocate the label/SID using the PCInitiate message. The C flag in the CCI object is set to 1 to indicate that the allocation needs to be done by the PCC. The PCC would allocate the SID/Label/Index and would report to the PCE using the PCRpt message.

If the value of the SID/Label/Index is 0 and the C flag is set to 1, it indicates that the PCE is requesting the allocation to be done by the PCC. If the SID/Label/Index is 'n' and the C flag is set to 1 in the CCI object, it indicates that the PCE requests a specific value 'n' for the SID/Label/Index. If the allocation is successful, the PCC should report via PCRpt message with the CCI object. Else, it MUST send a PCErr message with Error-Type = TBD ("PCECC failure") and Error Value = TBD ("Invalid CCI") (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]). If the value of the SID/Label/Index in the CCI object is valid, but the PCC is unable to allocate it, it MUST send a PCErr message with Error-Type = TBD ("PCECC failure") and Error Value = TBD ("Unable to allocate the specified CCI") (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]).

If the PCC wishes to withdraw or modify the previously assigned label/SID, it MUST send a PCRpt message without any SID/Label/Index or with the SID/Label/Index containing the new value respectively in the CCI object. The PCE would further trigger the removal of the central controller instruction as per this document.

5.5.1.7. Binding SID

A PCECC can allocate and provision the node/prefix/adjacency label (SID) via PCEP. Another SID called binding SID is described in [I-D.ietf-pce-binding-label-sid], the PCECC mechanism can also be used to allocate the binding SID.

A procedure for binding label/SID allocation is described in [I-D.ietf-pce-pcep-extension-for-pce-controller] and is applicable for all path setup types (including SR paths).

6. PCEP Messages

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable-length body made of a set of objects that can be either mandatory or optional. An object is said to be mandatory in a PCEP message when the object must be included for the message to be considered valid. For each PCEP message type, a set of rules is defined that specify the set of objects that the message can carry. An implementation **MUST** form the PCEP messages using the object ordering specified in this document.

6.1. Central Control Instructions

6.1.1. The PCInitiate Message

The PCInitiate message defined in [RFC8281] and extended in [I-D.ietf-pce-pcep-extension-for-pce-controller] is further extended to support SR based central control instructions.

The format of the extended PCInitiate message is as follows:

```

<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>) |
                                         (<FEC>
                                          <CCI>)

```

```

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

```

Where:

<PCE-initiated-lsp-instantiation> and
 <PCE-initiated-lsp-deletion> are as per
 [RFC8281].

The LSP and SRP object is defined in [RFC8231].

When the PCInitiate message is used to distribute SR SIDs, the SRP, the FEC and the CCI objects MUST be present. The error handling for missing SRP or CCI object is as per [I-D.ietf-pce-pcep-extension-for-pce-controller]. If the FEC object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=TBD5 (FEC object missing).

To clean up, the R (remove) bit in the SRP object and the corresponding FEC and the CCI object are included.

6.1.2. The PCRpt message

The PCRpt message can be used to report the SR central controller instructions received from the PCECC during the state synchronization phase or as an acknowledgment to the PCInitiate message.

The format of the PCRpt message is as follows:

```
<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>) |
                              (<FEC>
                               <CCI>)
```

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

Where:

<path> is as per [RFC8231] and the LSP and SRP object are also defined in [RFC8231].

When PCRpt message is used to report the label map allocations, the FEC and CCI objects MUST be present. The error handling for the missing CCI object is as per [I-D.ietf-pce-pcep-extension-for-pce-controller]. If the FEC object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=TBD5 (FEC object missing).

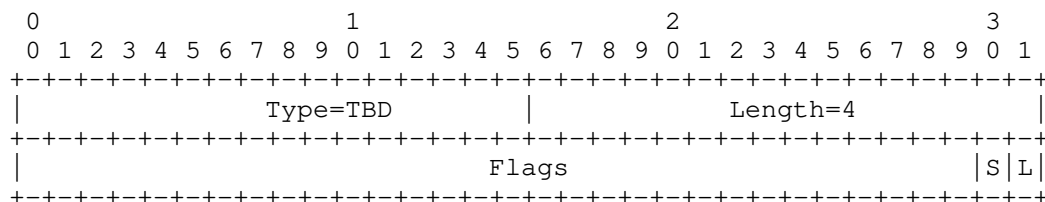
7. PCEP Objects

7.1. OPEN Object

7.1.1. PCECC Capability sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defined the PCECC-CAPABILITY sub-TLV.

A new S-bit is defined in PCECC-CAPABILITY sub-TLV for PCECC-SR:



[Editor's Note - The above figure is included for ease of the reader but should be removed before publication.]

S (PCECC-SR-CAPABILITY - 1 bit - TBD1): If set to 1 by a PCEP speaker, it indicates that the PCEP speaker is capable of PCECC-SR capability and the PCE allocates the Node and Adj label/SID on this session.

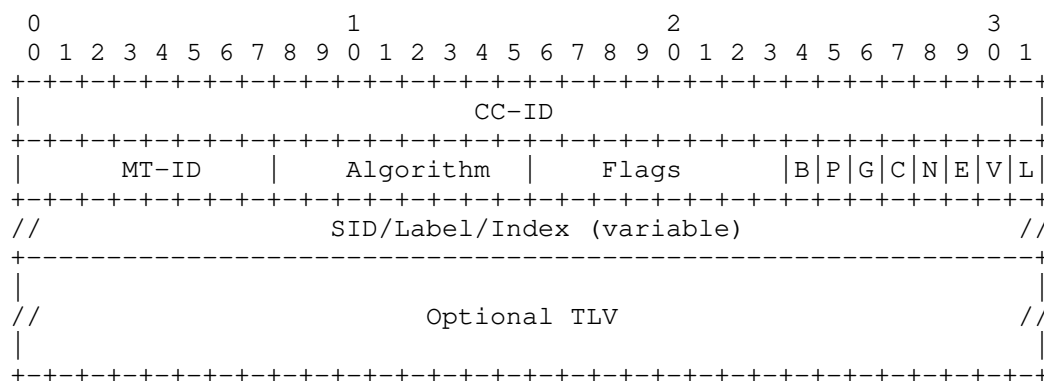
7.2. SR-TE Path Setup

The PATH-SETUP-TYPE TLV is defined in [RFC8408]. A PST value of 1 is used when Path is setup via SR mode as per [RFC8664]. The procedure for SR-TE path setup as specified in [RFC8664] remains unchanged.

7.3. CCI Object

The Central Control Instructions (CCI) Object used by the PCE to specify the controller instructions is defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]. This document defines another object-type for SR-MPLS purpose.

CCI Object-Type is TBD6 for SR-MPLS as below -



The field CC-ID is as described in [I-D.ietf-pce-pcep-extension-for-pce-controller]. Following new fields are defined for CCI Object-Type TBD6 -

MT-ID: Multi-Topology ID (as defined in [RFC4915]).

Algorithm: Single octet identifying the algorithm the SID is associated with. See [RFC8665].

Flags: is used to carry any additional information pertaining to the CCI. The following bits are defined -

- * L-Bit (Local/Global): If set, then the value/index carried by the CCI object has local significance. If not set, then the value/index carried by this object has global significance.
- * V-Bit (Value/Index): If set, then the CCI carries an absolute value. If not set, then the CCI carries an index.
- * E-Bit (Explicit-Null): If set, any upstream neighbor of the node that advertised the SID MUST replace the SID with the Explicit-NULL label (0 for IPv4) before forwarding the packet.
- * N-Bit (No-PHP): If set, then the penultimate hop MUST NOT pop the SID before delivering packets to the node that advertised the SID.
- * C-Bit (PCC Allocation): If the bit is set to 1, it indicates that the allocation needs to be done by the PCC for this central controller instruction. A PCE set this bit to request the PCC to make an allocation from its SR label/ID space. A PCC would set this bit to indicate that it has allocated the CC-ID and report it to the PCE.
- * Following bits are applicable when the SID represents an Adj-SID only, it MUST be ignored for others -
 - + G-Bit (Group): When set, the G-Flag indicates that the Adj-SID refers to a group of adjacencies (and therefore MAY be assigned to other adjacencies as well).
 - + P-Bit (Persistent): When set, the P-Flag indicates that the Adj-SID is persistently allocated, i.e., the Adj-SID value remains consistent across router restart and/or interface flap.
 - + B-Bit (Backup): If set, the Adj-SID refers to an adjacency that is eligible for protection (e.g., using IP Fast Reroute

or MPLS-FRR (MPLS-Fast Reroute) as described in Section 2.1 of [RFC8402].

- + All unassigned bits MUST be set to zero at transmission and ignored at receipt.

SID/Label/Index: According to the V and L flags, it contains either:

A 32-bit index defining the offset in the SID/Label space advertised by this router.

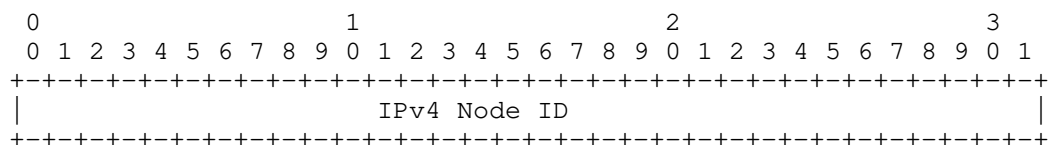
A 24-bit label where the 20 rightmost bits are used for encoding the label value.

7.4. FEC Object

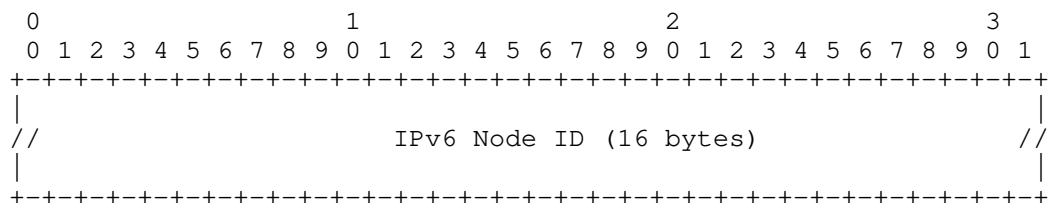
The FEC Object is used to specify the FEC information and MAY be carried within PCInitiate or PCRpt message.

FEC Object-Class is TBD3.

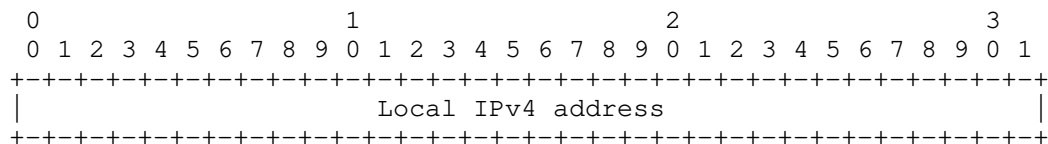
FEC Object-Type is 1 'IPv4 Node ID'.



FEC Object-Type is 2 'IPv6 Node ID'.



FEC Object-Type is 3 'IPv4 Adjacency'.



```

|----- Remote IPv4 address -----|
+-----+

```

FEC Object-Type is 4 'IPv6 Adjacency'.

```

      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
+-----+
|
//               Local IPv6 address (16 bytes)               //
|
+-----+
|
//               Remote IPv6 address (16 bytes)               //
|
+-----+

```

FEC Object-Type is 5 'Unnumbered Adjacency with IPv4 NodeIDs'.

```

      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
+-----+
|               Local Node-ID               |
+-----+
|               Local Interface ID           |
+-----+
|               Remote Node-ID               |
+-----+
|               Remote Interface ID          |
+-----+

```

FEC Object-Type is 6 'Linklocal IPv6 Adjacency'.

```

      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
+-----+
//               Local IPv6 address (16 octets)               //
+-----+
|               Local Interface ID           |
+-----+
//               Remote IPv6 address (16 octets)               //
+-----+
|               Remote Interface ID          |
+-----+

```


The FEC objects are as follows:

IPv4 Node ID: where IPv4 Node ID is specified as an IPv4 address of the Node. FEC Object-type is 1, and the Object-Length is 4 in this case.

IPv6 Node ID: where IPv6 Node ID is specified as an IPv6 address of the Node. FEC Object-type is 2, and the Object-Length is 16 in this case.

IPv4 Adjacency: where Local and Remote IPv4 address is specified as pair of IPv4 addresses of the adjacency. FEC Object-type is 3, and the Object-Length is 8 in this case.

IPv6 Adjacency: where Local and Remote IPv6 address is specified as pair of IPv6 addresses of the adjacency. FEC Object-type is 4, and the Object-Length is 32 in this case.

Unnumbered Adjacency with IPv4 NodeID: where a pair of Node ID / Interface ID tuple is used. FEC Object-type is 5, and the Object-Length is 16 in this case.

Linklocal IPv6 Adjacency: where a pair of (global IPv6 address, interface ID) tuple is used. FEC object-type is 6, and the Object-Length is 40 in this case.

8. Implementation Status

[Note to the RFC Editor - remove this section before publication, as well as remove the reference to RFC 7942.]

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. Huawei's Proof of Concept based on ONOS

The PCE function was developed in the ONOS open source platform. This extension was implemented on a private version as a proof of concept for PCECC.

- o Organization: Huawei
- o Implementation: Huawei's PoC based on ONOS
- o Description: PCEP as a southbound plugin was added to ONOS. To support PCECC-SR, an earlier version of this I-D was implemented. Refer <https://wiki.onosproject.org/display/ONOS/PCEP+Protocol>
- o Maturity Level: Prototype
- o Coverage: Partial
- o Contact: satishk@huawei.com

9. Security Considerations

The security considerations described in [I-D.ietf-pce-pcep-extension-for-pce-controller] apply to the extensions described in this document.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

10. Manageability Considerations

10.1. Control of Function and Policy

A PCE or PCC implementation SHOULD allow to configure to enable/disable PCECC SR capability as a global configuration. The implementation SHOULD also allow setting the local IP address used by the PCEP session.

10.2. Information and Data Models

[RFC7420] describes the PCEP MIB, this MIB can be extended to get the PCECC SR capability status.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to enable/disable PCECC SR capability.

10.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

10.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

10.5. Requirements On Other Protocols

PCEP extensions defined in this document do not put new requirements on other protocols.

10.6. Impact On Network Operations

PCEP implementation SHOULD allow a limit to be placed on the rate of PCLabelUpd messages sent by PCE and processed by PCC. It SHOULD also allow sending a notification when a rate threshold is reached.

11. IANA Considerations

11.1. PCECC-CAPABILITY sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defines the PCECC-CAPABILITY sub-TLV and requests that IANA to create a new sub-registry to manage the value of the PCECC-CAPABILITY sub-TLV's Flag field.

IANA is requested to allocate a new bit in the PCECC-CAPABILITY sub-TLV Flag Field sub-registry, as follows:

Bit	Description	Reference
TBD1	SR	This document

11.2. PCEP Object

IANA is requested to allocate new code-points for the new FEC object and a new Object-Type for CCI object in "PCEP Objects" sub-registry as follows:

Object-Class Value	Name	Object-Type	Reference
TBD3	FEC	1: IPv4 Node ID	This document
		2: IPv6 Node ID	This document
		3: IPv4 Adjacency	This document
		4: IPv6 Adjacency	This document
		5: Unnumbered Adjacency with IPv4 NodeID	This document
		6: Linklocal IPv6 Adjacency	This document
TBD	CCI	TBD6: SR-MPLS	This document

11.3. PCEP-Error Object

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

Error-Type	Meaning	
6	Mandatory Object missing.	
19	Error-value = TBD5 : Invalid operation.	FEC object missing
	Error-value = TBD4 :	SR capability was not advertised

11.4. CCI Object Flag Field for SR

IANA is requested to create a new sub-registry to manage the Flag field of the CCI Object-Type=TBD6 for SR called "CCI Object Flag Field for SR". New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (counting from bit 0 as the most significant bit)
- o Capability description
- o Defining RFC

Following bits are defined for the CCI Object flag field for SR in this document as follows:

Bit	Description	Reference
0-7	Unassigned	This document
8	B-Bit - Backup	This document
9	P-Bit - Persistent	This document
10	G-Bit - Group	This document
11	C-Bit - PCC Allocation	This document
12	N-Bit - No-PHP	This document
13	E-Bit - Explicit-Null	This document
14	V-Bit - Value/Index	This document
15	L-Bit - Local/Global	This document

12. Acknowledgments

We would like to thank Robert Tao, Changjing Yan, Tieying Huang and Avantika for their useful comments and suggestions.

13. References

13.1. Normative References

- [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>>.
- [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <<https://www.rfc-editor.org/info/rfc3630>>.
- [RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, DOI 10.17487/RFC4203, October 2005, <<https://www.rfc-editor.org/info/rfc4203>>.
- [RFC4915] Psenak, P., Mirtorabi, S., Roy, A., Nguyen, L., and P. Pillay-Esnault, "Multi-Topology (MT) Routing in OSPF", RFC 4915, DOI 10.17487/RFC4915, June 2007, <<https://www.rfc-editor.org/info/rfc4915>>.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <<https://www.rfc-editor.org/info/rfc5305>>.
- [RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <<https://www.rfc-editor.org/info/rfc5307>>.

- [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>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [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>>.
- [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>>.
- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.

[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>>.

[I-D.ietf-pce-pcep-extension-for-pce-controller]
Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-07 (work in progress), September 2020.

13.2. Informative References

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <<https://www.rfc-editor.org/info/rfc3031>>.

[RFC4655] Farrel, A., Vasseur, J., 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>>.

[RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C. Margaria, "Requirements for GMPLS Applications of PCE", RFC 7025, DOI 10.17487/RFC7025, September 2013, <<https://www.rfc-editor.org/info/rfc7025>>.

[RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.

[RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <<https://www.rfc-editor.org/info/rfc7420>>.

[RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.

- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.
- [RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8660] Bashandy, A., Ed., Filsfils, C., Ed., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with the MPLS Data Plane", RFC 8660, DOI 10.17487/RFC8660, December 2019, <<https://www.rfc-editor.org/info/rfc8660>>.
- [RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <<https://www.rfc-editor.org/info/rfc8665>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.
- [I-D.ietf-teas-pcecc-use-cases] Li, Z., Khasanov, B., Dhody, D., Zhao, Q., Ke, Z., Fang, L., Zhou, C., Communications, T., Rachitskiy, A., and A. Gulida, "The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).", draft-ietf-teas-pcecc-use-cases-06 (work in progress), September 2020.

[I-D.ietf-pce-pcep-yang]

Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-14 (work in progress), July 2020.

[I-D.ietf-pce-binding-label-sid]

Filsfils, C., Sivabalan, S., Tantsura, J., Hardwick, J., Previdi, S., and C. Li, "Carrying Binding Label/Segment-ID in PCE-based Networks.", draft-ietf-pce-binding-label-sid-03 (work in progress), June 2020.

[I-D.litkowski-pce-state-sync]

Litkowski, S., Sivabalan, S., Li, C., and H. Zheng, "Inter Stateful Path Computation Element (PCE) Communication Procedures.", draft-litkowski-pce-state-sync-08 (work in progress), July 2020.

[I-D.dhodylee-pce-pcep-ls]

Dhody, D., Peng, S., Lee, Y., Ceccarelli, D., and A. Wang, "PCEP extensions for Distribution of Link-State and TE Information", draft-dhodylee-pce-pcep-ls-17 (work in progress), July 2020.

[I-D.dhody-pce-pcep-extension-pce-controller-srv6]

Li, Z., Peng, S., Geng, X., and M. Negi, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) for SRv6", draft-dhody-pce-pcep-extension-pce-controller-srv6-04 (work in progress), July 2020.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Satish Karunanithi
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: satishk@huawei.com

Adrian Farrel
Juniper Networks, Inc
UK

EMail: adrian@olddog.co.uk

Xuesong Geng
Huawei Technologies
China

Email: gengxuesong@huawei.com

Udayasree Palle

EMail: udayasreereddy@gmail.com

Katherine Zhao
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

EMail: katherine.zhao@huawei.com

Boris Zhang
Telus Ltd.
Toronto
Canada

EMail: boris.zhang@telus.com

Alex Tokar
Cisco Systems
Slovak Republic

EMail: atokar@cisco.com

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: pengshuping@huawei.com

Mahendra Singh Negi
RtBrick Inc
N-17L, 18th Cross Rd, HSR Layout
Bangalore, Karnataka 560102
India

EMail: mahend.ietf@gmail.com

Quintin Zhao
Etheric Networks
1009 S CLAREMONT ST
SAN MATEO, CA 94402
USA

EMail: qzhao@ethericnetworks.com

Chao Zhou
HPE

EMail: chaozhou_us@yahoo.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 29, 2021

Z. Li
S. Peng
Huawei Technologies
M. Negi
RtBrick Inc
Q. Zhao
Etheric Networks
C. Zhou
HPE
November 25, 2020

PCEP Procedures and Protocol Extensions for Using PCE as a Central
Controller (PCECC) for Segment Routing (SR) MPLS Segment Identifier
(SID) Allocation and Distribution.
draft-zhao-pce-pcep-extension-pce-controller-sr-09

Abstract

The Path Computation Element (PCE) is a core component of Software-Defined Networking (SDN) systems. It can compute optimal paths for traffic across a network and can also update the paths to reflect changes in the network or traffic demands.

PCE was developed to derive paths for MPLS Label Switched Paths (LSPs), which are supplied to the head end of the LSP using the Path Computation Element Communication Protocol (PCEP). But SDN has a broader applicability than signaled (G)MPLS traffic-engineered (TE) networks, and the PCE may be used to determine paths in a range of use cases. PCEP has been proposed as a control protocol for use in these environments to allow the PCE to be fully enabled as a central controller.

A PCE-based Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. Thus, the LSP can be calculated/set up/initiated and the label forwarding entries can also be downloaded through a centralized PCE server to each network device along the path while leveraging the existing PCE technologies as much as possible.

This document specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers, in addition to computing the paths for packet flows in a segment routing (SR) network and telling the edge routers what instructions to attach to packets as they enter the network. PCECC is further enhanced for SR SID (Segment Identifier) allocation and distribution.

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 May 29, 2021.

Copyright Notice

Copyright (c) 2020 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
1.1. Requirements Language	5
2. Terminology	5
3. PCECC SR	5
4. PCEP Requirements	6
5. Procedures for Using the PCE as a Central Controller (PCECC) in Segment Routing	6
5.1. Stateful PCE Model	6
5.2. New LSP Functions	6
5.3. PCECC Capability Advertisement	7
5.4. PCEP session IP address and TED Router ID	7
5.5. LSP Operations	8
5.5.1. PCECC Segment Routing (SR)	8
5.5.1.1. PCECC SR Node/Prefix SID allocation	8

5.5.1.2.	PCECC SR Adjacency Label allocation	10
5.5.1.3.	Redundant PCEs	12
5.5.1.4.	Re Delegation and Clean up	12
5.5.1.5.	Synchronization of Label Allocations	13
5.5.1.6.	PCC-Based Allocations	13
5.5.1.7.	Binding SID	13
6.	PCEP Messages	14
6.1.	Central Control Instructions	14
6.1.1.	The PCInitiate Message	14
6.1.2.	The PCRpt message	15
7.	PCEP Objects	16
7.1.	OPEN Object	16
7.1.1.	PCECC Capability sub-TLV	16
7.2.	SR-TE Path Setup	17
7.3.	CCI Object	17
7.4.	FEC Object	19
8.	Implementation Status	21
8.1.	Huawei's Proof of Concept based on ONOS	22
9.	Security Considerations	22
10.	Manageability Considerations	22
10.1.	Control of Function and Policy	22
10.2.	Information and Data Models	23
10.3.	Liveness Detection and Monitoring	23
10.4.	Verify Correct Operations	23
10.5.	Requirements On Other Protocols	23
10.6.	Impact On Network Operations	23
11.	IANA Considerations	23
11.1.	PCECC-CAPABILITY sub-TLV	23
11.2.	PCEP Object	24
11.3.	PCEP-Error Object	24
11.4.	CCI Object Flag Field for SR	24
12.	Acknowledgments	25
13.	References	25
13.1.	Normative References	25
13.2.	Informative References	27
Appendix A.	Contributor Addresses	30
Authors'	Addresses	31

1. Introduction

The Path Computation Element (PCE) [RFC4655] was developed to offload the path computation function from routers in an MPLS traffic-engineered network. Since then, the role and function of the PCE has grown to cover a number of other uses (such as GMPLS [RFC7025]) and to allow delegated control [RFC8231] and PCE-initiated use of network resources [RFC8281].

According to [RFC7399], Software-Defined Networking (SDN) refers to a separation between the control elements and the forwarding components so that software running in a centralized system, called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place traffic flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system (including an SDN system) is presented in [RFC7491].

In early PCE implementations, where the PCE was used to derive paths for MPLS Label Switched Paths (LSPs), paths were requested by network elements (known as Path Computation Clients (PCCs)), and the results of the path computations were supplied to network elements using the Path Computation Element Communication Protocol (PCEP) [RFC5440]. This protocol was later extended to allow a PCE to send unsolicited requests to the network for LSP establishment [RFC8281].

[RFC8283] introduces the architecture for PCE as a central controller as an extension of the architecture described in [RFC4655] and assumes the continued use of PCEP as the protocol used between PCE and PCC. [RFC8283] further examines the motivations and applicability for PCEP as a Southbound Interface (SBI), and introduces the implications for the protocol. [I-D.ietf-teas-pcecc-use-cases] describes the use cases for the PCE-based Central Controller (PCECC) architecture.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify the procedures and PCEP extensions for using the PCE as the central controller for static LSPs, where LSPs can be provisioned as explicit label instructions at each hop on the end-to-end path.

Segment Routing (SR) technology leverages the source routing and tunneling paradigms. A source node can choose a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as a set of "segments" advertised by link-state routing protocols (IS-IS or OSPF). [RFC8402] provides an introduction to SR architecture. The corresponding IS-IS and OSPF extensions are specified in [RFC8667] and [RFC8665], respectively. It relies on a series of forwarding instructions being placed in the header of a packet. The segment routing architecture supports operations that can be used to steer packet flows in a network, thus providing a form of traffic engineering. [RFC8664] specify the SR specific PCEP extensions.

PCECC may further use PCEP for SR SID (Segment Identifier) allocation and distribution on the SR nodes with some benefits.

This document specifies the procedures and PCEP extensions when a PCE-based controller is also responsible for configuring the forwarding actions on the routers (SR SID allocation and distribution in this case), in addition to computing the paths for packet flows in a segment routing network and telling the edge routers what instructions to attach to packets as they enter the network.

Only SR using MPLS dataplane (SR-MPLS) is in the scope of this document. Refer [I-D.dhody-pce-pcep-extension-pce-controller-srv6] for use of PCECC technique for SR in IPv6 (SRv6) dataplane.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

Terminologies used in this document is the same as described in the draft [RFC8283] and [I-D.ietf-teas-pcecc-use-cases].

3. PCECC SR

[RFC8664] specifies extensions to PCEP that allow a stateful PCE to compute, update, or initiate SR-TE paths. An ingress node of an SR-TE path appends all outgoing packets with a list of MPLS labels (SIDs). This is encoded in SR-ERO subobject, capable of carrying a label (SID) as well as the identity of the node/adjacency label (SID).

The notion of segment and SID is defined in [RFC8402], which fits the MPLS architecture [RFC3031] as the label which is managed by a local allocation process of LSR (similarly to other MPLS signaling protocols) [RFC8660]. The SR information such as node/adjacency label (SID) is flooded via IGP as specified in [RFC8667] and [RFC8665].

As per [RFC8283], PCE as a central controller can allocate and provision the node/prefix/adjacency label (SID) via PCEP.

The rest of the processing is similar to existing stateful PCE with SR mechanism.

For the purpose of this document, it is assumed that the label range to be used by a PCE is set on both PCEP peers. Further, a global label range is assumed to be set on all PCEP peers in the SR domain. This document also allows a case where the label space is maintained by PCC itself, and the labels are allocated by the PCC, in this case, the PCE should request the allocation from PCC as described in Section 5.5.1.6.

4. PCEP Requirements

Following key requirements for PCECC-SR should be considered when designing the PCECC-based solution:

- o A PCEP speaker supporting this draft needs to have the capability to advertise its PCECC-SR capability to its peers.
- o PCEP procedures need to allow for PCC-based label/SID allocations.
- o PCEP procedures need means to update (or clean up) the label-map entry to the PCC.
- o PCEP procedures need to provide a mean to synchronize the SR labels allocations between the PCE to the PCC via PCEP messages.

5. Procedures for Using the PCE as a Central Controller (PCECC) in Segment Routing

5.1. Stateful PCE Model

Active stateful PCE is described in [RFC8231]. PCE as a Central Controller (PCECC) reuses the existing active stateful PCE mechanism as much as possible to control the LSPs.

5.2. New LSP Functions

Several new functions are required in PCEP to support PCECC as described in [I-D.ietf-pce-pcep-extension-for-pce-controller]. This document reuses the existing messages to support PCECC-SR.

The PCEP messages PCRpt, PCInitiate, PCUpd are used to send LSP Reports, LSP setup, and LSP update respectively. The extended PCInitiate message described in [I-D.ietf-pce-pcep-extension-for-pce-controller] is used to download or clean up central controller's instructions (CCIs) (SR SID in the scope of this document). The extended PCRpt message described in [I-D.ietf-pce-pcep-extension-for-pce-controller] is also used to report the CCIs (SR SIDs) from PCC to PCE.

[I-D.ietf-pce-pcep-extension-for-pce-controller] specify an object called CCI for the encoding of the central controller's instructions. This document extends the CCI by defining a new object-type for segment routing. The PCEP messages are extended in this document to handle the PCECC operations for SR.

5.3. PCECC Capability Advertisement

During PCEP Initialization Phase, PCEP Speakers (PCE or PCC) advertise their support of PCECC extensions. A PCEP Speaker includes the "PCECC Capability" sub-TLV, described in [I-D.ietf-pce-pcep-extension-for-pce-controller].

A new S-bit is added in the PCECC-CAPABILITY sub-TLV to indicate support for PCECC-SR. A PCC MUST set the S-bit in the PCECC-CAPABILITY sub-TLV and include the SR-PCE-CAPABILITY sub-TLV ([RFC8664]) in the OPEN Object (inside the PATH-SETUP-TYPE-CAPABILITY TLV) to support the PCECC SR extensions defined in this document. If the S-bit is set in the PCECC-CAPABILITY sub-TLV and the SR-PCE-CAPABILITY sub-TLV is not advertised in the OPEN Object, PCE SHOULD send a PCErr message with Error-Type=19 (Invalid Operation) and Error-value=TBd4 (SR capability was not advertised) and terminate the session.

The rest of the processing is as per [I-D.ietf-pce-pcep-extension-for-pce-controller].

5.4. PCEP session IP address and TED Router ID

A PCE may construct its Traffic Engineering Database (TED) by participating in the IGP ([RFC3630] and [RFC5305] for MPLS-TE; [RFC4203] and [RFC5307] for GMPLS). An alternative is offered by BGP-LS [RFC7752] and [I-D.dhodylee-pce-pcep-ls].

A PCEP [RFC5440] speaker could use any local IP address while creating a TCP session. It is important to link the session IP address with the Router ID in TED for successful PCECC operations.

During PCEP Initialization Phase, the PCC SHOULD advertise the TE mapping information by including the "Node Attributes TLV" [I-D.dhodylee-pce-pcep-ls] with "IPv4/IPv6 Router-ID of Local Node", in the OPEN Object for this purpose. [RFC7752] describes the usage as auxiliary Router-IDs that the IGP might be using, e.g., for TE purposes. If there are more than one auxiliary Router-ID of a given type, then multiple TLVs are used to encode them.

If "IPv4/IPv6 Router-ID" TLV is not present, the TCP session IP address is directly used for mapping purpose.

5.5. LSP Operations

[RFC8664] specify the PCEP extension to allow a stateful PCE to compute and initiate SR-TE paths, as well as a PCC to request a path subject to certain constraint(s) and optimization criteria in SR networks.

The Path Setup Type for segment routing (PST=1) is used on the PCEP session with the Ingress as per [RFC8664].

5.5.1. PCECC Segment Routing (SR)

Segment Routing (SR) as described in [RFC8402] depends on "segments" that are advertised by Interior Gateway Protocols (IGPs). The SR-node allocates and advertises the SID (node, adj, etc) and flood them via the IGP. This document proposes a new mechanism where PCE allocates the SID (label/index/SID) centrally and uses PCEP to advertise them. In some deployments, PCE (and PCEP) are better suited than IGP because of the centralized nature of PCE and direct TCP based PCEP sessions to the node.

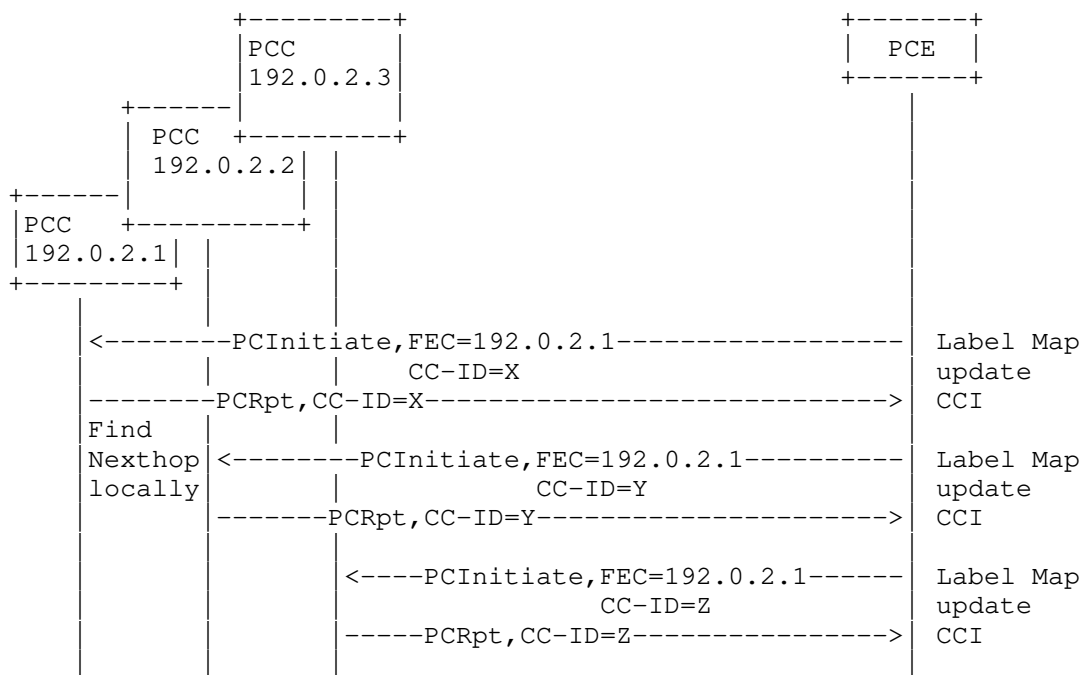
5.5.1.1. PCECC SR Node/Prefix SID allocation

Each node (PCC) is allocated a node-SID by the PCECC. The PCECC sends PCInitiate message to update the label map of each node to all the nodes in the domain. The TE router ID is determined from the TED or from "IPv4/IPv6 Router-ID" Sub-TLV [I-D.dhodylee-pce-pcep-ls], in the OPEN Object Section 5.4.

It is RECOMMENDED that PCEP session with PCECC-SR capability to use a different session IP address during TCP session establishment than the node Router ID in TEDB, to make sure that the PCEP session does not get impacted by the SR Node/Prefix Label maps (Section 5.4).

If a node (PCC) receives a PCInitiate message with a CCI encoding a SID, out of the range set aside for the SR Global Block (SRGB), it MUST send a PCErr message with Error-type=TBD (PCECC failure) and Error-value=TBD (Label out of range) (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]) and MUST include the SRP object to specify the error is for the corresponding central control instruction via the PCInitiate message.

On receiving the label map, each node (PCC) uses the local routing information to determine the next-hop and download the label forwarding instructions accordingly. The PCInitiate message in this case does not use the LSP object but uses a new FEC object defined in this document.

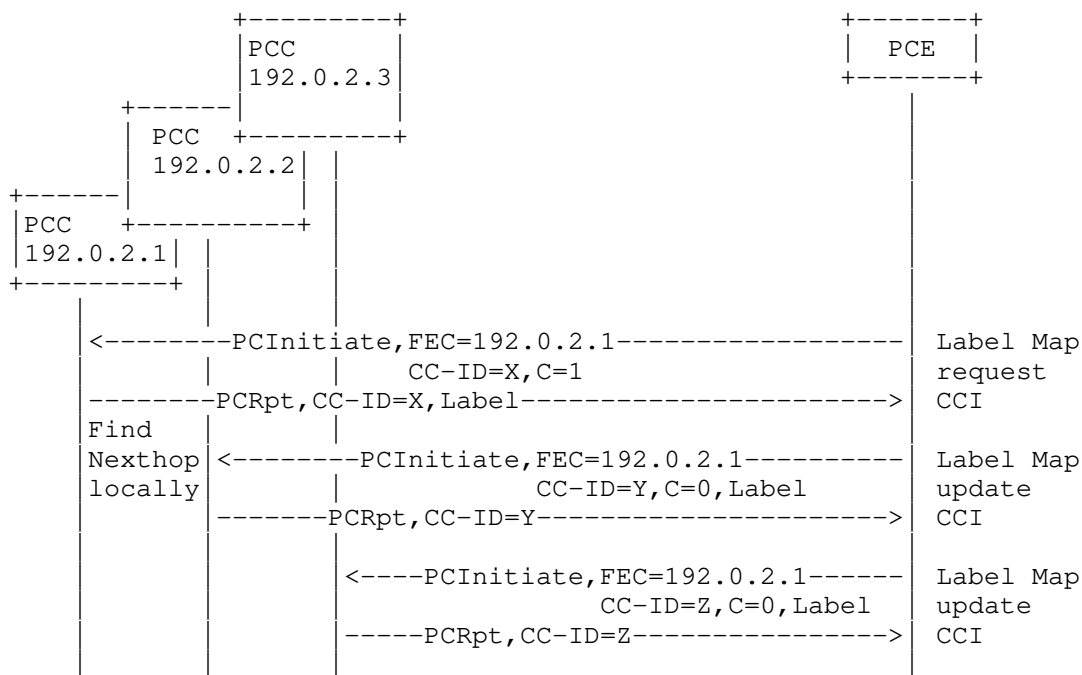


The forwarding behavior and the end result is similar to IGP based "Node-SID" in SR. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node as per [RFC8402].

PCE relies on the Node/Prefix Label clean up using the same PCInitiate message as per [RFC8281].

The above example Figure 1 depicts the FEC and PCEP speakers that uses IPv4 address. Similarly an IPv6 address (such as 2001:DB8::1) can be used during PCEP session establishment in the FEC object as described in this specification.

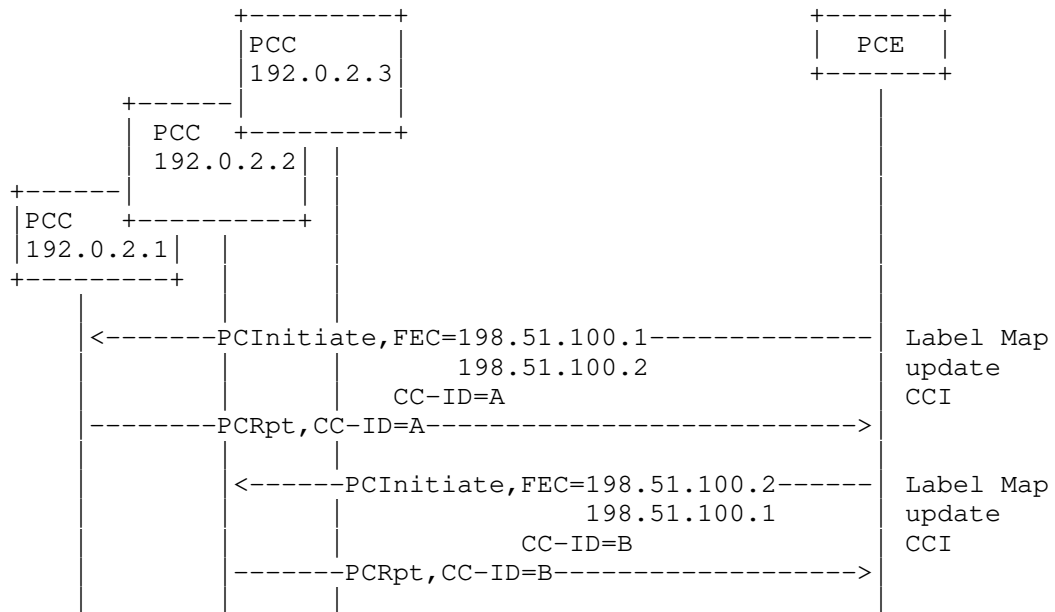
In the case where the label/SID allocation is made by the PCC itself (see Section 5.5.1.6), the PCE could request an allocation to be made by the PCC, and where the PCC would send a PCRpt with the allocated label/SID encoded in the CC-ID object as shown in Figure 2.



It should be noted that in this example, the request is made to the node 192.0.2.1 with C bit set in the CCI object to indicate that the allocation needs to be done by this PCC and it responds with the allocated label/SID to the PCE. The PCE would further inform the other PCCs in the network about the label-map allocation without setting the C bit.

5.5.1.2. PCECC SR Adjacency Label allocation

For PCECC-SR, apart from node-SID, Adj-SID is used where each adjacency is allocated an Adj-SID by the PCECC. The PCECC sends the PCInitiate message to update the label map of each adjacency to the corresponding nodes in the domain. Each node (PCC) download the label forwarding instructions accordingly. Similar to SR Node/Prefix Label allocation, the PCInitiate message in this case does not use the LSP object but uses the new FEC object defined in this document.



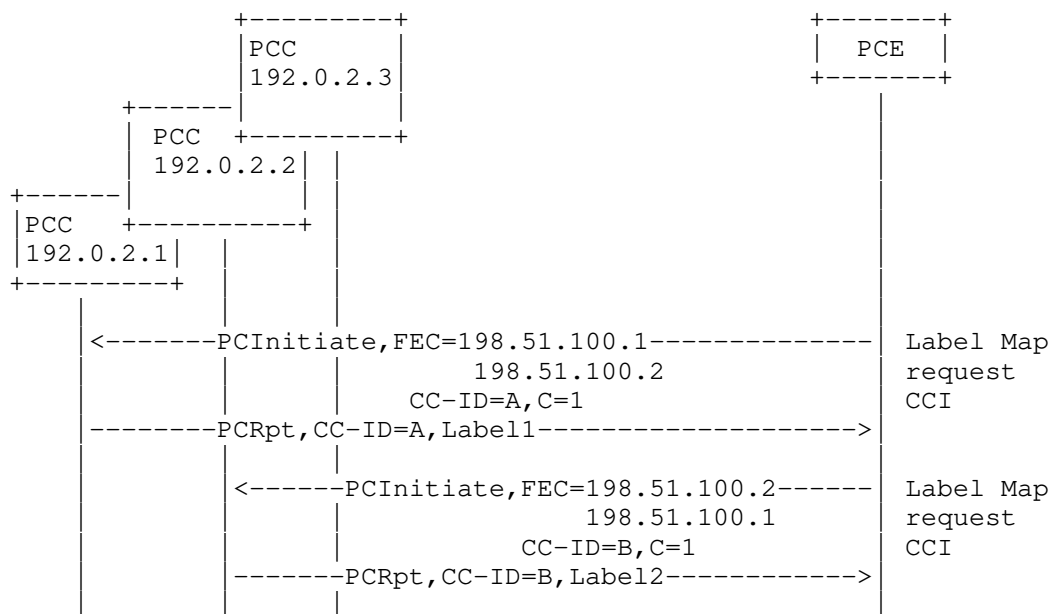
The forwarding behavior and the end result is similar to IGP based "Adj-SID" in SR.

PCE relies on the Adj label clean up using the same PCInitiate message as per [RFC8281].

The above example Figure 3 depicts FEC object and PCEP speakers that uses an IPv4 address. Similarly an IPv6 address (such as 2001:DB8::1, 2001:DB8::2) can be used during the PCEP session establishment in the FEC object as described in this specification.

The handling of adjacencies on the LAN subnetworks is specified in [RFC8402]. PCECC MUST assign Adj-SID for every pair of routers in the LAN. The rest of the protocol mechanism remains the same.

In the case where the label/SID map allocation is made by the PCC itself (see Section 5.5.1.6), the PCE could request an allocation to be made by the PCC, and where the PCC would send a PCRpt with the allocated label/SID encoded in the CC-ID object as shown in Figure 4.



In this example, the request is made to the node 192.0.2.1 with the C bit set in the CCI object to indicate that the allocation needs to be done by this PCC for the adjacency (198.51.100.1 - 198.51.100.2) and it responds with the allocated label/SID to the PCE. Similarly, another request is made to the node 192.0.2.2 with the C bit set in the CCI object to indicate that the allocation needs to be done by this PCC for the adjacency (198.51.100.2 - 198.51.100.1).

5.5.1.3. Redundant PCEs

[I-D.litkowski-pce-state-sync] describes the synchronization mechanism between the stateful PCEs. The SR SIDs allocated by a PCE MUST also be synchronized among PCEs for PCECC SR state synchronization. Note that the SR SIDs are independent of the SR-TE LSPs, and remains intact till any topology change. The redundant PCEs MUST have a common view of all SR SIDs allocated in the domain.

5.5.1.4. Re Delegation and Clean up

[I-D.ietf-pce-pcep-extension-for-pce-controller] describes the action needed for CCIs for the static LSPs on a terminated session. Same holds true for the CCI for SR SID as well.

5.5.1.5. Synchronization of Label Allocations

[I-D.ietf-pce-pcep-extension-for-pce-controller] describes the synchronization of Central Controller's Instructions (CCI) via LSP state synchronization as described in [RFC8231] and [RFC8232]. Same procedures are applied for the CCI for SR SID as well.

5.5.1.6. PCC-Based Allocations

The PCE can request the PCC to allocate the label/SID using the PCInitiate message. The C flag in the CCI object is set to 1 to indicate that the allocation needs to be done by the PCC. The PCC would allocate the SID/Label/Index and would report to the PCE using the PCRpt message.

If the value of the SID/Label/Index is 0 and the C flag is set to 1, it indicates that the PCE is requesting the allocation to be done by the PCC. If the SID/Label/Index is 'n' and the C flag is set to 1 in the CCI object, it indicates that the PCE requests a specific value 'n' for the SID/Label/Index. If the allocation is successful, the PCC should report via PCRpt message with the CCI object. Else, it MUST send a PCErr message with Error-Type = TBD ("PCECC failure") and Error Value = TBD ("Invalid CCI") (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]). If the value of the SID/Label/Index in the CCI object is valid, but the PCC is unable to allocate it, it MUST send a PCErr message with Error-Type = TBD ("PCECC failure") and Error Value = TBD ("Unable to allocate the specified CCI") (defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]).

If the PCC wishes to withdraw or modify the previously assigned label/SID, it MUST send a PCRpt message without any SID/Label/Index or with the SID/Label/Index containing the new value respectively in the CCI object. The PCE would further trigger the removal of the central controller instruction as per this document.

5.5.1.7. Binding SID

A PCECC can allocate and provision the node/prefix/adjacency label (SID) via PCEP. Another SID called binding SID is described in [I-D.ietf-pce-binding-label-sid], the PCECC mechanism can also be used to allocate the binding SID.

A procedure for binding label/SID allocation is described in [I-D.ietf-pce-pcep-extension-for-pce-controller] and is applicable for all path setup types (including SR paths).

6. PCEP Messages

As defined in [RFC5440], a PCEP message consists of a common header followed by a variable-length body made of a set of objects that can be either mandatory or optional. An object is said to be mandatory in a PCEP message when the object must be included for the message to be considered valid. For each PCEP message type, a set of rules is defined that specify the set of objects that the message can carry. An implementation **MUST** form the PCEP messages using the object ordering specified in this document.

6.1. Central Control Instructions

6.1.1. The PCInitiate Message

The PCInitiate message defined in [RFC8281] and extended in [I-D.ietf-pce-pcep-extension-for-pce-controller] is further extended to support SR based central control instructions.

The format of the extended PCInitiate message is as follows:

```
<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>) |
                                          (<FEC>
                                           <CCI>)
```

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

Where:

<PCE-initiated-lsp-instantiation> and
<PCE-initiated-lsp-deletion> are as per
[RFC8281].

The LSP and SRP object is defined in [RFC8231].

When the PCInitiate message is used to distribute SR SIDs, the SRP, the FEC and the CCI objects MUST be present. The error handling for missing SRP or CCI object is as per [I-D.ietf-pce-pcep-extension-for-pce-controller]. If the FEC object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=TBD5 (FEC object missing).

To clean up, the R (remove) bit in the SRP object and the corresponding FEC and the CCI object are included.

6.1.2. The PCRpt message

The PCRpt message can be used to report the SR central controller instructions received from the PCECC during the state synchronization phase or as an acknowledgment to the PCInitiate message.

The format of the PCRpt message is as follows:

```
<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>) |
                              (<FEC>
                               <CCI>)
```

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

Where:

<path> is as per [RFC8231] and the LSP and SRP object are also defined in [RFC8231].

When PCRpt message is used to report the label map allocations, the FEC and CCI objects MUST be present. The error handling for the missing CCI object is as per [I-D.ietf-pce-pcep-extension-for-pce-controller]. If the FEC object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=TBD5 (FEC object missing).

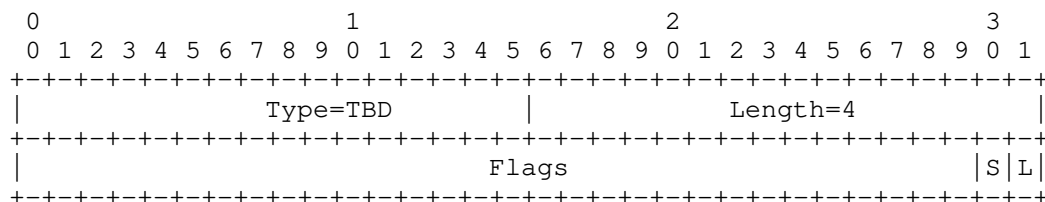
7. PCEP Objects

7.1. OPEN Object

7.1.1. PCECC Capability sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defined the PCECC-CAPABILITY sub-TLV.

A new S-bit is defined in PCECC-CAPABILITY sub-TLV for PCECC-SR:



[Editor's Note - The above figure is included for ease of the reader but should be removed before publication.]

S (PCECC-SR-CAPABILITY - 1 bit - TBD1): If set to 1 by a PCEP speaker, it indicates that the PCEP speaker is capable of PCECC-SR capability and the PCE allocates the Node and Adj label/SID on this session.

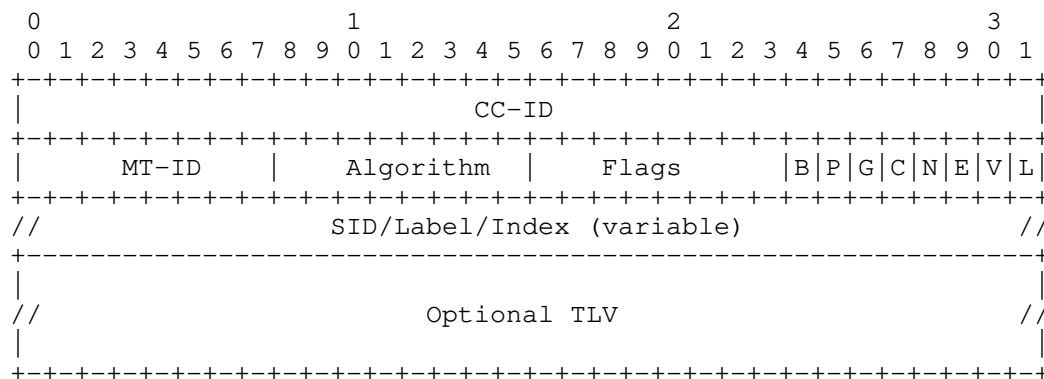
7.2. SR-TE Path Setup

The PATH-SETUP-TYPE TLV is defined in [RFC8408]. A PST value of 1 is used when Path is setup via SR mode as per [RFC8664]. The procedure for SR-TE path setup as specified in [RFC8664] remains unchanged.

7.3. CCI Object

The Central Control Instructions (CCI) Object used by the PCE to specify the controller instructions is defined in [I-D.ietf-pce-pcep-extension-for-pce-controller]. This document defines another object-type for SR-MPLS purpose.

CCI Object-Type is TBD6 for SR-MPLS as below -



The field CC-ID is as described in [I-D.ietf-pce-pcep-extension-for-pce-controller]. Following new fields are defined for CCI Object-Type TBD6 -

MT-ID: Multi-Topology ID (as defined in [RFC4915]).

Algorithm: Single octet identifying the algorithm the SID is associated with. See [RFC8665].

Flags: is used to carry any additional information pertaining to the CCI. The following bits are defined -

- * L-Bit (Local/Global): If set, then the value/index carried by the CCI object has local significance. If not set, then the value/index carried by this object has global significance.
- * V-Bit (Value/Index): If set, then the CCI carries an absolute value. If not set, then the CCI carries an index.
- * E-Bit (Explicit-Null): If set, any upstream neighbor of the node that advertised the SID MUST replace the SID with the Explicit-NULL label (0 for IPv4) before forwarding the packet.
- * N-Bit (No-PHP): If set, then the penultimate hop MUST NOT pop the SID before delivering packets to the node that advertised the SID.
- * C-Bit (PCC Allocation): If the bit is set to 1, it indicates that the allocation needs to be done by the PCC for this central controller instruction. A PCE set this bit to request the PCC to make an allocation from its SR label/ID space. A PCC would set this bit to indicate that it has allocated the CC-ID and report it to the PCE.
- * Following bits are applicable when the SID represents an Adj-SID only, it MUST be ignored for others -
 - + G-Bit (Group): When set, the G-Flag indicates that the Adj-SID refers to a group of adjacencies (and therefore MAY be assigned to other adjacencies as well).
 - + P-Bit (Persistent): When set, the P-Flag indicates that the Adj-SID is persistently allocated, i.e., the Adj-SID value remains consistent across router restart and/or interface flap.
 - + B-Bit (Backup): If set, the Adj-SID refers to an adjacency that is eligible for protection (e.g., using IP Fast Reroute

or MPLS-FRR (MPLS-Fast Reroute) as described in Section 2.1 of [RFC8402].

- + All unassigned bits MUST be set to zero at transmission and ignored at receipt.

SID/Label/Index: According to the V and L flags, it contains either:

A 32-bit index defining the offset in the SID/Label space advertised by this router.

A 24-bit label where the 20 rightmost bits are used for encoding the label value.

7.4. FEC Object

The FEC Object is used to specify the FEC information and MAY be carried within PCInitiate or PCRpt message.

FEC Object-Class is TBD3.

FEC Object-Type is 1 'IPv4 Node ID'.

```

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
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     IPv4 Node ID                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

FEC Object-Type is 2 'IPv6 Node ID'.

```

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
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     IPv6 Node ID (16 bytes)                                     |
//                                     //
+-----+-----+-----+-----+-----+-----+-----+-----+

```

FEC Object-Type is 3 'IPv4 Adjacency'.

```

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
+-----+-----+-----+-----+-----+-----+-----+-----+
|                                     Local IPv4 address                                     |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

```

|----- Remote IPv4 address -----|
+-----+

```

FEC Object-Type is 4 'IPv6 Adjacency'.

```

      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
+-----+
|
//               Local IPv6 address (16 bytes)               //
|
+-----+
|
//               Remote IPv6 address (16 bytes)               //
|
+-----+

```

FEC Object-Type is 5 'Unnumbered Adjacency with IPv4 NodeIDs'.

```

      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
+-----+
|               Local Node-ID               |
+-----+
|               Local Interface ID           |
+-----+
|               Remote Node-ID              |
+-----+
|               Remote Interface ID          |
+-----+

```

FEC Object-Type is 6 'Linklocal IPv6 Adjacency'.

```

      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
+-----+
//               Local IPv6 address (16 octets)               //
+-----+
|               Local Interface ID           |
+-----+
//               Remote IPv6 address (16 octets)               //
+-----+
|               Remote Interface ID          |
+-----+

```


The FEC objects are as follows:

IPv4 Node ID: where IPv4 Node ID is specified as an IPv4 address of the Node. FEC Object-type is 1, and the Object-Length is 4 in this case.

IPv6 Node ID: where IPv6 Node ID is specified as an IPv6 address of the Node. FEC Object-type is 2, and the Object-Length is 16 in this case.

IPv4 Adjacency: where Local and Remote IPv4 address is specified as pair of IPv4 addresses of the adjacency. FEC Object-type is 3, and the Object-Length is 8 in this case.

IPv6 Adjacency: where Local and Remote IPv6 address is specified as pair of IPv6 addresses of the adjacency. FEC Object-type is 4, and the Object-Length is 32 in this case.

Unnumbered Adjacency with IPv4 NodeID: where a pair of Node ID / Interface ID tuple is used. FEC Object-type is 5, and the Object-Length is 16 in this case.

Linklocal IPv6 Adjacency: where a pair of (global IPv6 address, interface ID) tuple is used. FEC object-type is 6, and the Object-Length is 40 in this case.

8. Implementation Status

[Note to the RFC Editor - remove this section before publication, as well as remove the reference to RFC 7942.]

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. Huawei's Proof of Concept based on ONOS

The PCE function was developed in the ONOS open source platform. This extension was implemented on a private version as a proof of concept for PCECC.

- o Organization: Huawei
- o Implementation: Huawei's PoC based on ONOS
- o Description: PCEP as a southbound plugin was added to ONOS. To support PCECC-SR, an earlier version of this I-D was implemented. Refer <https://wiki.onosproject.org/display/ONOS/PCEP+Protocol>
- o Maturity Level: Prototype
- o Coverage: Partial
- o Contact: satishk@huawei.com

9. Security Considerations

The security considerations described in [I-D.ietf-pce-pcep-extension-for-pce-controller] apply to the extensions described in this document.

As per [RFC8231], it is RECOMMENDED that these PCEP extensions only be activated on authenticated and encrypted sessions across PCEs and PCCs belonging to the same administrative authority, using Transport Layer Security (TLS) [RFC8253] as per the recommendations and best current practices in [RFC7525] (unless explicitly set aside in [RFC8253]).

10. Manageability Considerations

10.1. Control of Function and Policy

A PCE or PCC implementation SHOULD allow to configure to enable/disable PCECC SR capability as a global configuration. The implementation SHOULD also allow setting the local IP address used by the PCEP session.

10.2. Information and Data Models

[RFC7420] describes the PCEP MIB, this MIB can be extended to get the PCECC SR capability status.

The PCEP YANG module [I-D.ietf-pce-pcep-yang] could be extended to enable/disable PCECC SR capability.

10.3. Liveness Detection and Monitoring

Mechanisms defined in this document do not imply any new liveness detection and monitoring requirements in addition to those already listed in [RFC5440].

10.4. Verify Correct Operations

Mechanisms defined in this document do not imply any new operation verification requirements in addition to those already listed in [RFC5440] and [RFC8231].

10.5. Requirements On Other Protocols

PCEP extensions defined in this document do not put new requirements on other protocols.

10.6. Impact On Network Operations

PCEP implementation SHOULD allow a limit to be placed on the rate of PCLabelUpd messages sent by PCE and processed by PCC. It SHOULD also allow sending a notification when a rate threshold is reached.

11. IANA Considerations

11.1. PCECC-CAPABILITY sub-TLV

[I-D.ietf-pce-pcep-extension-for-pce-controller] defines the PCECC-CAPABILITY sub-TLV and requests that IANA to create a new sub-registry to manage the value of the PCECC-CAPABILITY sub-TLV's Flag field.

IANA is requested to allocate a new bit in the PCECC-CAPABILITY sub-TLV Flag Field sub-registry, as follows:

Bit	Description	Reference
TBD1	SR	This document

11.2. PCEP Object

IANA is requested to allocate new code-points for the new FEC object and a new Object-Type for CCI object in "PCEP Objects" sub-registry as follows:

Object-Class Value	Name	Object-Type	Reference
TBD3	FEC	1: IPv4 Node ID	This document
		2: IPv6 Node ID	This document
		3: IPv4 Adjacency	This document
		4: IPv6 Adjacency	This document
		5: Unnumbered Adjacency with IPv4 NodeID	This document
		6: Linklocal IPv6 Adjacency	This document
TBD	CCI	TBD6: SR-MPLS	This document

11.3. PCEP-Error Object

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

Error-Type	Meaning	
6	Mandatory Object missing.	
19	Error-value = TBD5 : Invalid operation.	FEC object missing
	Error-value = TBD4 :	SR capability was not advertised

11.4. CCI Object Flag Field for SR

IANA is requested to create a new sub-registry to manage the Flag field of the CCI Object-Type=TBD6 for SR called "CCI Object Flag Field for SR". New values are to be assigned by Standards Action [RFC8126]. Each bit should be tracked with the following qualities:

- o Bit number (counting from bit 0 as the most significant bit)
- o Capability description
- o Defining RFC

Following bits are defined for the CCI Object flag field for SR in this document as follows:

Bit	Description	Reference
0-7	Unassigned	This document
8	B-Bit - Backup	This document
9	P-Bit - Persistent	This document
10	G-Bit - Group	This document
11	C-Bit - PCC Allocation	This document
12	N-Bit - No-PHP	This document
13	E-Bit - Explicit-Null	This document
14	V-Bit - Value/Index	This document
15	L-Bit - Local/Global	This document

12. Acknowledgments

We would like to thank Robert Tao, Changjing Yan, Tieying Huang, Avantika, and Aijun Wang for their useful comments and suggestions.

13. References

13.1. Normative References

- [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>>.
- [RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <<https://www.rfc-editor.org/info/rfc3630>>.
- [RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, DOI 10.17487/RFC4203, October 2005, <<https://www.rfc-editor.org/info/rfc4203>>.
- [RFC4915] Psenak, P., Mirtorabi, S., Roy, A., Nguyen, L., and P. Pillay-Esnault, "Multi-Topology (MT) Routing in OSPF", RFC 4915, DOI 10.17487/RFC4915, June 2007, <<https://www.rfc-editor.org/info/rfc4915>>.
- [RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <<https://www.rfc-editor.org/info/rfc5305>>.
- [RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <<https://www.rfc-editor.org/info/rfc5307>>.

- [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>>.
- [RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <<https://www.rfc-editor.org/info/rfc7752>>.
- [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>>.
- [RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <<https://www.rfc-editor.org/info/rfc8253>>.
- [RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <<https://www.rfc-editor.org/info/rfc8281>>.
- [RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <<https://www.rfc-editor.org/info/rfc8408>>.
- [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>>.

[I-D.ietf-pce-pcep-extension-for-pce-controller]
Li, Z., Peng, S., Negi, M., Zhao, Q., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-ietf-pce-pcep-extension-for-pce-controller-08 (work in progress), November 2020.

13.2. Informative References

- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <<https://www.rfc-editor.org/info/rfc3031>>.
- [RFC4655] Farrel, A., Vasseur, J., 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>>.
- [RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., and C. Margaria, "Requirements for GMPLS Applications of PCE", RFC 7025, DOI 10.17487/RFC7025, September 2013, <<https://www.rfc-editor.org/info/rfc7025>>.
- [RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <<https://www.rfc-editor.org/info/rfc7399>>.
- [RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <<https://www.rfc-editor.org/info/rfc7420>>.
- [RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <<https://www.rfc-editor.org/info/rfc7491>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.

- [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>>.
- [RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <<https://www.rfc-editor.org/info/rfc8232>>.
- [RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <<https://www.rfc-editor.org/info/rfc8283>>.
- [RFC8402] Filssils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <<https://www.rfc-editor.org/info/rfc8402>>.
- [RFC8660] Bashandy, A., Ed., Filssils, C., Ed., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with the MPLS Data Plane", RFC 8660, DOI 10.17487/RFC8660, December 2019, <<https://www.rfc-editor.org/info/rfc8660>>.
- [RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filssils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <<https://www.rfc-editor.org/info/rfc8665>>.
- [RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filssils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <<https://www.rfc-editor.org/info/rfc8667>>.
- [I-D.ietf-teas-pcecc-use-cases]
Li, Z., Khasanov, B., Dhody, D., Zhao, Q., Ke, Z., Fang, L., Zhou, C., Communications, T., Rachitskiy, A., and A. Gulida, "The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).", draft-ietf-teas-pcecc-use-cases-06 (work in progress), September 2020.

[I-D.ietf-pce-pcep-yang]

Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-15 (work in progress), October 2020.

[I-D.ietf-pce-binding-label-sid]

Sivabalan, S., Filsfils, C., Tantsura, J., Hardwick, J., Previdi, S., and C. Li, "Carrying Binding Label/Segment-ID in PCE-based Networks.", draft-ietf-pce-binding-label-sid-05 (work in progress), October 2020.

[I-D.litkowski-pce-state-sync]

Litkowski, S., Sivabalan, S., Li, C., and H. Zheng, "Inter Stateful Path Computation Element (PCE) Communication Procedures.", draft-litkowski-pce-state-sync-09 (work in progress), November 2020.

[I-D.dhodylee-pce-pcep-ls]

Dhody, D., Peng, S., Lee, Y., Ceccarelli, D., Wang, A., and G. Mishra, "PCEP extensions for Distribution of Link-State and TE Information", draft-dhodylee-pce-pcep-ls-19 (work in progress), November 2020.

[I-D.dhody-pce-pcep-extension-pce-controller-srv6]

Li, Z., Peng, S., Geng, X., and M. Negi, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) for SRv6", draft-dhody-pce-pcep-extension-pce-controller-srv6-05 (work in progress), November 2020.

Appendix A. Contributor Addresses

Dhruv Dhody
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: dhruv.ietf@gmail.com

Satish Karunanithi
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: satishk@huawei.com

Adrian Farrel
Juniper Networks, Inc
UK

EMail: adrian@olddog.co.uk

Xuesong Geng
Huawei Technologies
China

Email: gengxuesong@huawei.com

Udayasree Palle

EMail: udayasreereddy@gmail.com

Katherine Zhao
Huawei Technologies
2330 Central Expressway
Santa Clara, CA 95050
USA

EMail: katherine.zhao@huawei.com

Boris Zhang
Telus Ltd.
Toronto
Canada

EMail: boris.zhang@telus.com

Alex Tokar
Cisco Systems
Slovak Republic

EMail: atokar@cisco.com

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

EMail: pengshuping@huawei.com

Mahendra Singh Negi
RtBrick Inc
N-17L, 18th Cross Rd, HSR Layout
Bangalore, Karnataka 560102
India

EMail: mahend.ietf@gmail.com

Quintin Zhao
Etheric Networks
1009 S CLAREMONT ST
SAN MATEO, CA 94402
USA

EMail: qzhao@ethericnetworks.com

Chao Zhou
HPE

EMail: chaozhou_us@yahoo.com