PCE Working Group
Internet Draft
Intended status: Standard Track
Expires: August 13, 2014

Zafar Ali
Siva Sivabalan
Clarence Filsfils
Cisco Systems
Robert Varga
Pantheon Technologies
Victor Lopez
Oscar Gonzalez de Dios
Telefonica I+D
Xian Zhang
Huawei
February 14, 2014

Path Computation Element Communication Protocol (PCEP) Extensions for remote-initiated GMPLS LSP Setup
draft-ali-pce-remote-initiated-gmpls-lsp-03.txt

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 http://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 13, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Abstract

Draft [I-D. draft-crabbe-pce-pce-initiated-lsp] specifies procedures that can be used for creation and deletion of PCE-initiated LSPs in the active stateful PCE model. However, this specification focuses on MPLS networks, and does not cover remote instantiation of paths in GMPLS-controlled networks. This document complements [I-D. draft-crabbe-pce-pce-initiated-lsp] by addressing the requirements for remote-initiated GMPLS LSPs.

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].

Table of Contents

1. Introduction 3
2. Requirements for Remote-Initiated GMPLS LSPs 3
3. PCEP Extensions for Remote-Initiated GMPLS LSPs 4
  3.1. Generalized Endpoint in LSP Initiate Message 4
  3.2. GENERALIZED-BANDWIDTH object in LSP Initiate Message .5
  3.3. Protection Attributes in LSP Initiate Message 5
  3.4. ERO in LSP Initiate Object 5
    3.4.1. ERO with explicit label control 5
    3.4.2. ERO with Path Keys 6
    3.4.3. Switch Layer Object 6
  3.5. LSP delegation and cleanup 7
4. Security Considerations 7
5. IANA Considerations 7
  5.1. PCEP-Error Object 7
6. Acknowledgments 7
7. References 7
  7.1. Normative References 7

7.2. Informative References 8

1. Introduction

The Path Computation Element communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform route computations in response to Path Computation Clients (PCCs) requests. PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model draft [I-D. draft-ietf-pce-stateful-pce] describes a set of extensions to PCEP to enable active control of MPLS-TE and GMPLS network.

[I-D. draft-crabbe-pce-pce-initiated-lsp] describes the setup and teardown of PCE-initiated LSPs under the active stateful PCE model, without the need for local configuration on the PCC. This enables realization of a dynamic network that is centrally controlled and deployed. However, this specification is focused on MPLS networks, and does not cover the GMPLS networks (e.g., WSON, OTN, SONET/ SDH, etc. technologies). This document complements [I-D. draft-crabbe-pce-pce-initiated-lsp] by addressing the requirements for remote-initiated GMPLS LSPs. These requirements are covered in Section 2 of this draft. The PCEP extensions for remote initiated GMPLS LSPs are specified in Section 3.

2. Requirements for Remote-Initiated GMPLS LSPs

[I-D. draft-crabbe-pce-pce-initiated-lsp] specifies procedures that can be used for creation and deletion of PCE-initiated LSPs under the active stateful PCE model. However, this specification does not address GMPLS requirements outlined in the following:

- GMPLS support multiple switching capabilities on per TE link basis. GMPLS LSP creation requires knowledge of LSP switching capability (e.g., TDM, L2SC, OTN-TDM, LSC, etc.) to be used [RFC3471], [RFC3473].

- GMPLS LSP creation requires knowledge of the encoding type (e.g., lambda photonic, Ethernet, SONET/ SDH, G709 OTN, etc.) to be used by the LSP [RFC3471], [RFC3473].

- GMPLS LSP creation requires information of the generalized payload (G-PID) to be carried by the LSP [RFC3471], [RFC3473].

- GMPLS LSP creation requires specification of data flow specific traffic parameters (also known as Tspec), which are technology specific.

- GMPLS also specifics support for asymmetric bandwidth requests [RFC6387].

- GMPLS extends the addressing to include unnumbered interface identifiers, as defined in [RFC3477].

- In some technologies path calculation is tightly coupled with label selection along the route. For example, path calculation in a WDM network may include lambda continuity and/ or lambda feasibility constraints and hence a path computed by the PCE is associated with a specific lambda (label). Hence, in such networks, the label information needs to be provided to a PCC in order for a PCE to initiate GMPLS LSPs under the active stateful PCE model. I.e., explicit label control may be required.

- GMPLS specifics protection context for the LSP, as defined in [RFC4872] and [RFC4873].

3. PCEP Extensions for Remote-Initiated GMPLS LSPs

LSP initiate (PCInitiate) message defined in [I-D. draft-crabbe-pce-pce-initiated-lsp] needs to be extended to include GMPLS specific PCEP objects as follows:

3.1. Generalized Endpoint in LSP Initiate Message

This document does not modify the usage of END-POINTS object for PCE initiated LSPs as specified in [I-D. draft-crabbe-pce-pce-initiated-lsp]. It augments the usage as specified below.

END-POINTS object has been extended by [I-D. draft-ietf-pcep-gmpls-ext] to include a new object type called "Generalized Endpoint". PCInitiate message sent by a PCE to a PCC to trigger a GMPLS LSP instantiation SHOULD include the END-POINTS with Generalized Endpoint object type. Furthermore, the END-POINTS object MUST contain "label request" TLV. The label request TLV is used to specify the switching type, encoding type and GPID of the LSP being instantiated by the PCE.

The unnumbered endpoint TLV can be used to specify unnumbered endpoint addresses for the LSP being instantiated by the PCE. The END-POINTS MAY contain other TLVs defined in [I-D. draft-ietf-pcep-gmpls-ext].

If the END-POINTS Object of type Generalized Endpoint is missing the label request TLV, the PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value= TBA (LSP request TLV missing).

If the PCC does not support the END-POINTS Object of type Generalized Endpoint, the PCC MUST send a PCErr message with

Error-type = 3 (Unknown Object), Error-value = 2(unknown object type).

3.2. GENERALIZED-BANDWIDTH object in LSP Initiate Message

LSP initiate message defined in [I-D. draft-crabbe-pce-pce-initiated-lsp] can optionally include the BANDWIDTH object. However, the following possibilities cannot be represented in the BANDWIDTH object:

- Asymmetric bandwidth (different bandwidth in forward and reverse direction), as described in [RFC6387].

- Technology specific GMPLS parameters (e.g., Tspec for SDH/SONET, G.709, ATM, MEF, etc.) are not supported.

GENERALIZED-BANDWIDTH object has been defined in [I-D. draft-ietf-pcep-gmpls-ext] to address the above-mentioned limitation of the BANDWIDTH object.

This document specifies the use of GENERALIZED-BANDWIDTH object in PCInitiate message. Specifically, GENERALIZED-BANDWIDTH object MAY be included in the PCInitiate message. The GENERALIZED-BANDWIDTH object in PCInitiate message is used to specify technology specific Tspec and asymmetrical bandwidth values for the LSP being instantiated by the PCE.

3.3. Protection Attributes in LSP Initiate Message

This document does not modify the usage of LSPA object for PCE initiated LSPs as specified in [I-D. draft-crabbe-pce-pce-initiated-lsp]. It augments the usage of LSPA object in LSP Initiate Message to carry the end-to-end protection context this also includes the protection state information.

The LSP Protection Information TLV of LSPA in the PCInitiate message can be used to specify protection attributes of the LSP being instantiated by the PCE.

3.4. ERO in LSP Initiate Object

This document does not modify the usage of ERO object for PCE initiated LSPs as specified in [I-D. draft-crabbe-pce-pce-initiated-lsp]. It augments the usage as specified in the following sections.

3.4.1. ERO with explicit label control

As mentioned earlier, there are technologies and scenarios where active stateful PCE requires explicit label control in order to instantiate an LSP.

Explicit label control (ELC) is a procedure supported by RSVP-TE, where the outgoing label(s) is (are) encoded in the ERO. [I-D. draft-ietf-pcep-gmpls-ext] extends the <ERO> object of PCEP to include explicit label control. The ELC procedure enables the PCE to provide such label(s) directly in the path ERO.

The extended ERO object in PCInitiate message can be used to specify label along with ERO to PCC for the LSP being instantiated by the active stateful PCE.

3.4.2. ERO with Path Keys

There are many scenarios in packet and optical networks where the route information of an LSP may not be provided to the PCC for confidentiality reasons. A multi-domain or multi-layer network is an example of such networks. Similarly, a GMPLS User-Network Interface (UNI) [RFC4208] is also an example of such networks.

In such scenarios, ERO containing the entire route cannot be provided to PCC (by PCE). Instead, PCE provides an ERO with Path Keys to the PCC. For example, in the case UNI interface between the router and the optical nodes, the ERO in the LSP Initiate Message may be constructed as follows:

- The first hop is a strict hop that provides the egress interface information at PCC. This interface information is used to get to a network node that can extend the rest of the ERO. (Please note that in the cases where the network node is not directly connected with the PCC, this part of ERO may consist of multiple hops and may be loose).
- The following(s) hop in the ERO may provide the network node with the path key [RFC5520] that can be resolved to get the contents of the route towards the destination.
- There may be further hops but these hops may also be encoded with the path keys (if needed).

This document does not change encoding or processing roles for the path keys, which are defined in [RFC5520].

3.4.3. Switch Layer Object

[draft-ietf-pce-inter-layer-ext-07] specifies the SWITCH-LAYER object which defines and specifies the switching layer (or layers) in which a path MUST or MUST NOT be established. A switching layer is expressed as a switching type and encoding type. [I-D. draft-ietf-pcep-gmpls-ext], which defines the GMPLS

extensions for PCEP, suggests using the SWITCH-LAYER object. Thus, SWITCH-LAYER object can be used in the PCInitiate message to specify the switching layer (or layers) of the LSP being remotely initiated.

3.5. LSP delegation and cleanup

LSP delegation and cleanup procedure specified in [I-D. draft-ietf-pcep-gmpls-ext] are equally applicable to GMPLS LSPs and this document does not modify the associated usage.

4. Security Considerations

To be added in future revision of this document.

5. IANA Considerations

5.1. PCEP-Error Object

This document defines the following new Error-Value:

Error-Type	Error Value
6	Error-value=TBA: LSP Request TLV missing

6. Acknowledgments

The authors would like to thank George Swallow and Jan Medved for their comments.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[I-D. draft-crabbe-pce-pce-initiated-lsp] Crabbe, E., Minei, I., Sivabalan, S., Varga, R., "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-crabbe-pce-pce-initiated-lsp, work in progress.

[RFC5440] Vasseur, JP., Ed., and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 5623, September 2009.

[RFC 6107] Shiomoto, K., Ed., and A. Farrel, Ed., "Procedures for Dynamically Signaled Hierarchical Label Switched Paths", RFC 6107, February 2011.

7.2. Informative References

[RFC3471] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description", RFC 3471, January 2003.

[RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, January 2003.

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, January 2003.

[RFC4872] Lang, J., Ed., Rekhter, Y., Ed., and D. Papadimitriou, Ed., "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, May 2007.

[RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel, "GMPLS Segment Recovery", RFC 4873, May 2007.

[RFC4208] Swallow, G., Drake, J., Ishimatsu, H., and Y. Rekhter, "Generalized Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model", RFC 4208, October 2005.

[RFC5520] Bradford, R., Ed., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", RFC 5520, April 2009.

Author’s Addresses

Zafar Ali
Cisco Systems
Email: zali@cisco.com

Siva Sivabalan
Cisco Systems
Email: msiva@cisco.com

Clarence Filsfils

Cisco Systems
Email: cfilsfil@cisco.com

Robert Varga
Pantheon Technologies

Victor Lopez
Telefonica I+D
Email: vlopez@tid.es

Oscar Gonzalez de Dios
Telefonica I+D
Email: ogondio@tid.es

Xian Zhang
Huawei Technologies
Email: zhang.xian@huawei.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: August 18, 2014

S. Alvarez
S. Sivabalan
Z. Ali
Cisco Systems, Inc.
L. Tomotaki
Verizon
V. Lopez
Telefonica I+D

February 14, 2014

PCE Path Profiles
draft-alvarez-pce-path-profiles-01

Abstract

This document describes extensions to the Path Computation Element (PCE) Communication Protocol (PCEP) to signal path profile identifiers. A profile represents a list of path parameters or policies that a PCEP peer may invoke on a remote peer using an opaque identifier. When a path computation client (PCC) initiates a path computation request, the PCC can signal profile identifiers to invoke path parameters or policies defined on the PCE which would influence the path computation. Similarly, when a PCE initiates or updates a path, the PCE can signal profile identifiers to invoke path parameters or policies defined on the PCC which would influence the path setup.

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 http://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 18, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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 2
2. Path Profiles 3
3. Procedures 3
3.1. Capability Advertisement 3
3.2. PCC-Initiated Paths 3
3.2.1. Point-to-Point Paths 4
3.2.2. Point-to-Multipoint Paths 5
3.3. PCE-Initiated Paths 5
4. Object Extensions 6
4.1. OPEN Object 6
4.2. PATH-PROFILE Object 7
5. Error Codes for PATH-PROFILE Object 8
6. Acknowledgements 8
7. IANA Considerations 8
8. Security Considerations 8
9. References 8
9.1. Normative References 8
9.2. Informative References 9
Authors' Addresses 10

1. Introduction

1.1. 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 RFC 2119 [RFC2119].

2. Path Profiles

A path profile represents a list of path parameters or policies that a PCEP peer may invoke on a remote peer using a profile identifier. The receiving peer interprets the identifier according to a local path profile definition. The PATH-PROFILE object defined in Section 4.2 can signal one or more profile identifiers. PCEP carries profile identifiers as opaque values. PCEP peers do not exchange the details of a path profile. The PCE may be stateful or stateless.

3. Procedures

3.1. Capability Advertisement

PCEP peers advertise their capability to support path profile identifiers during the session initialization phase. They include the PATH-PROFILE-CAPABILITY TLV defined in Section 4.1 as part of the OPEN object. A PCEP peer can only signal path profile identifiers if both peers advertised this capability. A peer MUST send a PCErr message with Error-Type=4 (Not supported object), Error-value=1 (Not supported object class) and close the session if it receives a message with a path profile identifier, it supports the extensions in this document and both peers did not advertise this capability.

3.2. PCC-Initiated Paths

A PCC MAY include a PATH-PROFILE object when sending a PCReq message. The PCE uses the path profile identifier to select path parameters or path policies to fulfill the request. The means by which the PCC learns about a particular path profile identifier and decides to include it in a PCReq message are outside the scope of this document. Similarly, the means by which the PCE selects a set of parameters or policies based on the profile identifier for a specific request are outside the scope of this document. The P flag of the PATH-PROFILE object MUST be set.

A PCE may receive a path computation request with an unknown or invalid path profile identifier. The PCE sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=1 (Unknown path profile) if the path profile identifier is not known to the PCE. The PCE sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=2 (Invalid path profile) if the PCE knows about the path profile identifier, but considers the request invalid. As an example, the profile may be invalid because of the path type, the PCEP session type or the originating PCC. The PCEP-ERROR object SHOULD include the path profile identifiers that generated the error condition.

The PCE will determine whether to consider any additional optional objects included in a PCReq message based on policy. As illustrated in Section 3.2.1 and Section 3.2.2, the PCC MAY include other optional objects along with a PATH-PROFILE object as part of a path computation request. The PCC will use the processing-rule (P) flag in the common object header to signal whether it considers those objects mandatory or optional when the PCE performs path computation. Those objects may overlap with the path parameters that the PCE associates with the path profile identifier.

PCE policy may place different kinds of restrictions on PCReq messages that include a PATH-PROFILE object and additional parameters. A PCE MUST send an error message if it receives a request with optional objects signaled as mandatory (P flag = 1) for path computation and PCE policy does not allow such behavior from the originating PCC. In that case, the PCE sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=3 (Unexpected mandatory object). If the objects are signaled as optional (P flag = 0) for path computation, the PCE will decide based on policy whether to consider them or not. When sending the PCRep message for the request, the PCE will use the ignore (I) flag in the common object header to indicate to the PCC whether an object was ignored.

3.2.1. Point-to-Point Paths

[RFC5440] defines the basic structure of a PCReq message for point-to-point paths. This documents extends the message format as follows:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>


where:

   <svec-list>::=<SVEC>[<svec-list>]
   <request-list>::=<request>[<request-list>]

   <request>::= <RP>
                <END-POINTS>
                [<PATH-PROFILE>]
                [<path-computation>]


where:
```

<path-computation> is the list of optional objects used for path computation as defined initially in [RFC5440] and modified in subsequent PCEP extensions.

If present in a PCReq message, the PATH-PROFILE object MUST be the first optional object in the request portion of the message.

3.2.2. Point-to-Multipoint Paths

[RFC6006] defines the basic structure of a PCReq message for point-to-multipoint paths. This documents extends the message format as follows:

TBD

3.3. PCE-Initiated Paths

A PCE MAY include a PATH-PROFILE object when sending a PCInitiate message as defined in [I-D.ietf-pce-pce-initiated-lsp]. The PCC uses the path profile identifier to select path parameters or path policies to be applied during the instantiation of the path. The means by which the PCE learns about a particular path profile identifier and decides to include it in a PCInitiate message are outside the scope of this document. Similarly, the means by which the PCC selects a set of parameters or policies based on the profile identifier for a specific path are outside the scope of this document.

A PCC may receive a path instantiation request with an unknown or invalid path profile identifier. The PCC sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=1 (Unknown path profile) if the path profile identifier is not known to the PCC. The PCC sends a PCErr message with Error-Type=[TBA] (PATH-PROFILE Error), Error-value=2 (Invalid path profile) if the PCC knows about the path profile identifier, but considers the request invalid. As an example, the profile may be invalid because of the path type, the PCEP session type or the originating PCE. The PCEP-ERROR object SHOULD include the path profile identifiers that generated the error condition.

[I-D.ietf-pce-pce-initiated-lsp] defines the basic structure of a PCInitiate message. This documents extends the message format 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>
                                        [PATH-PROFILE>
                                        [<attribute-list>]

  <PCE-initiated-lsp-deletion> ::= <SRP>
                                   <LSP>



where:

<attribute-list> is defined in [RFC5440] and extended by PCEP
extensions.
```

4. Object Extensions

4.1. OPEN Object

This documents defines a new optional PATH-PROFILE-CAPABILITY TLV in the OPEN object.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBA]          |            Length=4           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Reserved           |             Flags             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PATH-PROFILE-CAPABILITY TLV

Figure 1

Reserved (16 bits):

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

Flags (16 bits):
Unassigned bits are considered reserved. They MUST be set to zero on transmission and ignored on receipt. No flags are currently defined.

4.2. PATH-PROFILE Object

The PATH-PROFILE object may be carried in PCReq, PCInitiate and PCUpd messages.

PATH-PROFILE Object-Class is [TBA].

PATH-PROFILE Object-Type is 1.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                            TLVs                            //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PATH-PROFILE Object

Figure 2

The PATH-PROFILE object has a variable length and contains one or more PATH-PROFILE-ID TLVs.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Type=[TBD]        |             Length=6          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Reserved   |      Flags    |       Path Profile Id         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Path Profile Id  (cont)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PATH-PROFILE-ID TLV

Figure 3

Reserved (8 bits):

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

Flags (8 bits):
Unassigned bits are considered reserved. They MUST be set to zero on transmission and ignored on receipt. No flags are currently defined.

Path Profile Id (32 bits):
(non-zero) unsigned path profile identifier.

5. Error Codes for PATH-PROFILE Object

Error-Type	Meaning	Error-Value
<TBA>	PATH-PROFILE Error	1: Unknown path profile
		2: Invalid path profile
		3: Unexpected mandatory object

6. Acknowledgements

The authors would like to thank Clarence Filsfils for his valuable comments.

7. IANA Considerations

IANA is requested to assign the following code points.

PATH-PROFILE-CAPABILITY TLV

PATH-PROFILE Object-Class

PATH-PROFILE Object-Type

PATH-PROFILE Error-Type

8. Security Considerations

TBD

9. References

9.1. Normative References

[I-D.ali-pce-remote-initiated-gmpls-lsp]
Ali, Z., Sivabalan, S., Filsfils, C., Varga, R., Lopez, V., Dios, O., and X. Zhang, "Path Computation Element Communication Protocol (PCEP) Extensions for remote-initiated GMPLS LSP Setup", draft-ali-pce-remote-initiated-gmpls-lsp-02 (work in progress), October 2013.

[I-D.ietf-pce-gmpls-pcep-extensions]
Margaria, C., Dios, O., and F. Zhang, "PCEP extensions for GMPLS", draft-ietf-pce-gmpls-pcep-extensions-08 (work in progress), July 2013.

[I-D.ietf-pce-pce-initiated-lsp]
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-00 (work in progress), December 2013.

[I-D.ietf-pce-stateful-pce]
Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

[I-D.sivabalan-pce-segment-routing]
Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E., and R. Raszuk, "PCEP Extensions for Segment Routing", draft-sivabalan-pce-segment-routing-02 (work in progress), October 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

9.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

Authors' Addresses

Santiago Alvarez
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134
USA

Email: saalvare@cisco.com

Siva Sivabalan
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, ON K2K-3E8
Canada

Email: msiva@cisco.com

Zafar Ali
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, ON K2K-3E8
Canada

Email: zali@cisco.com

Luis Tomotaki
Verizon
400 International
Richardson, TX 75081
US

Email: luis.tomotaki@verizon.com

Victor Lopez
Telefonica I+D
c/ Don Ramon de la Cruz 84
Madrid 28006
Spain

Email: vlopez@tid.es

PCE Working Group
Internet-Draft
Intended status: Experimental
Expires: August 18, 2014

U. Palle
Avantika. S
D. Dhody
Huawei Technologies

February 14, 2014

PCEP Extensions for Supporting Multiple Sources and Destinations
draft-avantika-pce-multi-src-dest-01

Abstract

The Path Computation Element (PCE) provides functions of path computation in support of traffic engineering in networks controlled by Multi-Protocol Label Switching (MPLS) and Generalized MPLS (GMPLS).

This document provides extensions for the Path Computation Element Protocol (PCEP) to support multiple sources and destinations in a single path request.

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 http://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 18, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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. Terminology 3
3. Motivation 4
3.1. Example 5
4. PCEP Requirements 6
5. Extension to PCEP 6
5.1. The New MP2MP END-POINTS Object 6
6. Other Considerations 8
6.1. Identification of Source-Destination Pair in PCRep Message 8
6.2. Request-ID 9
6.3. Backward Compatibility 9
6.4. Overloading the PCE 9
7. Security Considerations 9
8. Manageability Considerations 9
8.1. Control of Function and Policy 9
8.2. Information and Data Models 10
8.3. Liveness Detection and Monitoring 10
8.4. Verify Correct Operations 10
8.5. Requirements On Other Protocols 10
8.6. Impact On Network Operations 10
9. IANA Considerations 10
9.1. New END-POINTS Object Types 10
10. Acknowledgments 10
11. References 10
11.1. Normative References 11
11.2. Informative References 11
Appendix A. Evaluation of Message Size Reduction 12

1. Introduction

[RFC5440] specifies the Path Computation Element Communication Protocol (PCEP) for communications between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between two PCEs, in compliance with [RFC4657].

As per [RFC5440], a single Path Computation Request (PCReq) message may carry more than one path computation request. Each request is uniquely identified by a request-id number. In some scenarios (refer Section 3), there is a need to send multiple requests with the same

constraints and attributes to the PCE. Currently these requests are either sent in a separate PCReq messages or clubbed together in one (or more) PCReq messages. In either case, the constraints and attributes need to be encoded separately for each request even though they are exactly identical.

Also note that, the PCE may choose to respond to each of the request independently in a separate Path Computation Reply (PCRep) messages or choose to bundle them in one (or more) PCRep messages. In some scenarios (refer Section 3) one should wait for responses for all request before proceeding further.

A mechanism to request path computation between multiple sources and destinations in a single path computation request would be helpful.

Note that the scope of this work is point-to-point (P2P) path computation and is unrelated to MP2MP LSP ([RFC6388]).

1.1. 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 [RFC2119].

2. Terminology

The following terminology is used in this document.

LSR: Label Switch Router.

MPLS: Multiprotocol Label Switching.

NMS: Network Management System.

P2P: Point-to-Point.

PCC: Path Computation Client. Any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.

PCEP: Path Computation Element Communication Protocol.

3. Motivation

Following key scenarios are identified where a mechanism to request path computation between multiple sources and destinations in a single path computation request would be helpful.

Hierarchical PCE: [RFC6805] describes the procedure for inter-domain path computation using Hierarchical PCE. In case of end to end path computation by Parent PCE, it needs to issue multiple path computation requests to child PCEs. In case of transit domain(s), the Parent PCE issues requests from each entry boundary node to each exit boundary node to the child PCE(s). Similarly for ingress domain, the Parent PCE issues requests from source to each exit boundary node, where as for egress domain, the Parent PCE issues requests from each entry boundary node to the destination. All requests to a particular child PCE, need to be encoded separately even though they are exactly identical (they have the same constraints and attributes). Also the Parent PCE needs to wait for responses for all requests before proceeding further.

Inter-Layer PCE: [RFC5623] describes inter-layer path computation framework. In case of cooperating PCEs per layer, where each PCE has topology visibility restricted to its own layer and collaborate to compute an end-to-end path across layers. The higher layer PCE may need to issue multiple requests to lower layer PCE requesting paths from each entry boundary node to each exit boundary node. All these requests need to be encoded separately even though they are exactly identical (they have the same constraints and attributes). Also the higher layer PCE needs to wait for responses for all requests before proceeding further.

Management-Based PCE: [RFC4655] describes a case where PCC is not necessarily an LSR, but for example, maybe a NMS or a planning tool etc. Such a PCC may issue multiple requests to PCE with identical constraints and attributes to select among the several source-destination pairs.

Using Multiple P2P Path Computations for MP2MP TE LSP: In case where, for establishing a MP2MP TE LSP tunnel, multiple P2P path computation requests are sent to the PCE, one for each source-destination pair with identical constraints and attributes.

In these scenarios, a mechanism to request path computation between multiple sources and destinations in a single path computation request would be helpful.

3.1. Example

Consider the topology example mentioned in Section 4.6 of [RFC6805] -

```
 ----------------------------------------------------------------
|   Domain 5                                                     |
|                              -----                             |
|                             |PCE 5|                            |
|                              -----                             |
|                                                                |
|    ----------------     ----------------     ----------------  |
|   | Domain 1       |   | Domain 2       |   | Domain 3       | |
|   |                |   |                |   |                | |
|   |        -----   |   |        -----   |   |        -----   | |
|   |       |PCE 1|  |   |       |PCE 2|  |   |       |PCE 3|  | |
|   |        -----   |   |        -----   |   |        -----   | |
|   |                |   |                |   |                | |
|   |            ----|   |----        ----|   |----            | |
|   |           |BN11+---+BN21|      |BN23+---+BN31|           | |
|   |   -        ----|   |----        ----|   |----        -   | |
|   |  |S|           |   |                |   |           |D|  | |
|   |   -        ----|   |----        ----|   |----        -   | |
|   |           |BN12+---+BN22|      |BN24+---+BN32|           | |
|   |            ----|   |----        ----|   |----            | |
|   |                |   |                |   |                | |
|   |         ----   |   |                |   |   ----         | |
|   |        |BN13|  |   |                |   |  |BN33|        | |
|    -----------+----     ----------------     ----+-----------  |
|                \                                /               |
|                 \      ----------------        /                |
|                  \    |                |      /                 |
|                   \   |----        ----|     /                  |
|                    ----+BN41|      |BN42+----                   |
|                        |----        ----|                        |
|                        |                |                        |
|                        |        -----   |                        |
|                        |       |PCE 4|  |                        |
|                        |        -----   |                        |
|                        |                |                        |
|                        | Domain 4       |                        |
|                         ----------------                         |
|                                                                  |
 ----------------------------------------------------------------
```

Figure 1: An example topology

The following 11 individual requests are generated by parent PCE (PCE 5) -

Domain 1: S-BN11; S-BN12; S-BN13;

Domain 2: BN21-BN23; BN21-BN24; BN22-BN23; BN22-BN24;

Domain 3: BN31-D; BN32-D; BN33-D;

Domain 4: BN41-BN42;

The above requests for each domain, need to be encoded separately even though they are exactly identical. A mechanism to request them together in a single path computation request would be helpful, in which case total 4 requests would be generated by parent PCE.

4. PCEP Requirements

Following key requirements are identified for PCEP to enable multiple sources and destinations in a single path computation request:

1. It MUST be possible for a single Path Computation Request to list multiple sources and destinations.

2. It MUST be possible for a single Path Computation Reply to be sent for multiple sources and destinations.

3. It MUST NOT be possible to set different constraints, traffic parameters, or quality-of-service requirements for different source and destination pair within a single computation request.

5. Extension to PCEP

This document extends the existing END-POINTS object [RFC5440] and [RFC6006] by defining two new END-POINTS object types.

5.1. The New MP2MP END-POINTS Object

The END-POINTS object is used in a PCReq message to specify the source IP address and the destination IP address of the path for which a path computation is requested. This document extends the existing END-POINT object to support multiple sources and destinations in a single path request.

Two new MP2MP END-POINTS objects for IPv4 and IPv6 are defined.

The format of the MP2MP END-POINTS object body for IPv4 (Object-Type=5 (TBD)) is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      No of Sources  (2 bytes) |  No of Destinations  (2 bytes)|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Source IPv4 address 1 (4 bytes)               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                             ......                            ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Source IPv4 address m (4 bytes)               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Destination IPv4 address 1 (4 bytes)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                             ......                            ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Destination IPv4 address n (4 bytes)             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: The new MP2MP END-POINTS Object Body Format for IPv4

The format of the MP2MP END-POINTS object body for IPv6 (Object-Type=6 (TBD)) is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      No of Sources  (2 bytes) |  No of Destinations  (2 bytes)|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                 Source IPv6 address 1 (16 bytes)              |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                            ......                             ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                 Source IPv6 address m (16 bytes)              |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|              Destination IPv6 address 1 (16 bytes)            |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                            ......                             ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|              Destination IPv6 address n (16 bytes)            |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: The new MP2MP END-POINTS Object Body Format for IPv6

The MP2MP END-POINTS object body has a variable length. These are multiples of 4 bytes for IPv4, and multiples of 16 bytes for IPv6, plus 4 bytes.

On receiving MP2MP END-POINTS object, PCE computes m*n P2P paths, i.e, point to point path is computed between each combination of source and destination received in MP2MP END-POINTS object.

6. Other Considerations

6.1. Identification of Source-Destination Pair in PCRep Message

Since the END-POINTS object is not carried in the PCRep message ([RFC5440]), the implementation supporting this extension SHOULD encode the source and the destination as the first and the last hop

in the ERO. This is done to easily identify that which ERO corresponds to which source-destination pair.

6.2. Request-ID

As per [RFC5440], each request is uniquely identified by a request-id number.

Since a single request is used for multiple sources and destinations sharing the same request-id number, along with request and response, the request-id number in RP object used in other PCEP messages (PCNtf, PCErr ...) applies to all sources and destinations (in the single request).

6.3. Backward Compatibility

If PCE receives new END-POINTS type in path request and it understands the END-POINTS type, but the PCE is not capable of/ support multiple sources and destinations in a single path request, the PCE MUST send a PCErr message with a PCEP-ERROR Object Error-Type = 4 (Not supported object) [RFC5440]. The path computation request MUST then be cancelled.

If the PCE does not understand the new END-POINTS type, then the PCE MUST send a PCErr message with a PCEP-ERROR Object Error-Type = 3 (Unknown object) [RFC5440].

6.4. Overloading the PCE

The new END-POINTS type could be used to issue multiple computations together hence overloading the PCE. The PCE MUST allow for the use of policies to accept/reject such a request.

7. Security Considerations

This document defines new END-POINTS types which does not add any new security concerns beyond those discussed in [RFC5440].

8. Manageability Considerations

8.1. Control of Function and Policy

TBD.

8.2. Information and Data Models

TBD.

8.3. Liveness Detection and Monitoring

TBD.

8.4. Verify Correct Operations

TBD.

8.5. Requirements On Other Protocols

TBD.

8.6. Impact On Network Operations

TBD.

9. IANA Considerations

IANA assigns values to PCEP parameters in registries defined in [RFC5440]. IANA is requested to make the following additional assignments.

9.1. New END-POINTS Object Types

Two new END-POINTS Object-Types are defined in this document. IANA is requested to make the following Object-Type allocations from the "PCEP Objects" sub-registry:

```
Object-Class Value    4
Name                  END-POINTS
Object-Type           5: MP2MP IPv4
                      6: MP2MP IPv6
                      7-15: Unassigned
Reference             This.I-D
```

10. Acknowledgments

Thanks to Quintin Zhao for his suggestions.

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4657] Ash, J. and J. Le Roux, "Path Computation Element (PCE) Communication Protocol Generic Requirements", RFC 4657, September 2006.

11.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 5623, September 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

[RFC6388] Wijnands, IJ., Minei, I., Kompella, K., and B. Thomas, "Label Distribution Protocol Extensions for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths", RFC 6388, November 2011.

[RFC6805] King, D. and A. Farrel, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, November 2012.

Appendix A. Evaluation of Message Size Reduction

Consider the domain 2 in Figure 1, where 4 path requests need to be encoded (from 2 entry boundary nodes to 2 exit boundary nodes with the exact same constraints).

Following figure illustrates the existing mechanism of carrying multiple requests in single PCReq message (as per [RFC5440]):

```
          +--------------+ +--------------+ +--------------+
          |RP            | |RP            | |RP            |
+-------+ |END-POINTS(8) | |END-POINTS    | |END-POINTS    |
|COMMON |+|BANDWIDTH     |+|BANDWIDTH     |+|BANDWIDTH     |
|HEADER | |METRIC        | |METRIC        | |METRIC        |
+-------+ |METRIC        | |METRIC        | |METRIC        |
          +--------------+ +--------------+ +--------------+
4 Bytes     36 Bytes         36 Bytes         36 Bytes

          +--------------+
          |RP            |
          |END-POINTS(20)|
        + |BANDWIDTH     | = 148 Bytes
          |METRIC        |
          |METRIC        |
          +--------------+
            36 Bytes
```

Combining multiple requests into a single request by using MP2MP END-POINTS object is illustrated below:

```
          +--------------+
          |RP            |
+-------+ |END-POINTS(20)|
|COMMON |+|BANDWIDTH     | = 54 Bytes
|HEADER | |METRIC        |
+-------+ |METRIC        |
          +--------------+
4 Bytes     48 Bytes
```

There is message size reduction of 64% in this example for just one domain.

Authors' Addresses

Udayasree Palle
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: udayasree.palle@huawei.com

Avantika
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: avantika.sushilkumar@huawei.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.ietf@gmail.com

Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: April 24, 2014

H. Chen
Huawei Technologies
A. Liu
Ericsson
F. Xu
Verizon
M. Toy
Comcast
V. Liu
China Mobile

October 21, 2013

Extensions to PCEP for Distributing Label Cross Domains
draft-chen-pce-label-x-domains-00.txt

Abstract

This document specifies extensions to PCEP for distributing labels crossing domains for an inter-domain Point-to-Point (P2P) or Point-to-Multipoint (P2MP) Traffic Engineering (TE) Label Switched Path (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 http://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 April 24, 2014.

Copyright Notice

Copyright (c) 2013 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 (http://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. Conventions Used in This Document 4
4. Label Distribution 4
4.1. An Exmaple 4
5. Extensions to PCEP 5
5.1. RP Object Extension 5
5.2. Label Object 6
5.3. LSP Tunnel Object 7
5.4. Request Message Extension 9
5.5. Reply Message Extension 9
6. Procedures 10
6.1. Distributing Label in Ordered Setup 10
6.2. Distributing Label in Path Computation 10
7. Security Considerations 11
8. IANA Considerations 11
8.1. Request Parameter Bit Flags 11
9. Acknowledgement 11
10. References 11
10.1. Normative References 11
10.2. Informative References 12
Authors' Addresses 12

1. Introduction

After a path crossing multiple domains is computed, an inter-domain Traffic Engineering (TE) Label Switched Path (LSP) tunnel may be set up along the path by a number of tunnel central controllers (TCCs). Each of the domains through which the path goes may be controlled by a tunnel central controller (TCC), which sets up the segment of the TE LSP tunnel in the domain. When the TCC sets up the segment of the TE LSP tunnel in its domain that is not a domain containing the tail end of the tunnel, it needs a label from a domain, which is next to it along the path.

This document specifies extensions to PCEP and various procedures for distributing a label from a domain to its previous domain along the path for the TE LSP tunnel crossing multiple domains.

2. Terminology

ABR: Area Border Router. Routers used to connect two IGP areas (areas in OSPF or levels in IS-IS).

ASBR: Autonomous System Border Router. Routers used to connect together ASes of the same or different service providers via one or more inter-AS links.

Boundary Node (BN): a boundary node is either an ABR in the context of inter-area Traffic Engineering or an ASBR in the context of inter-AS Traffic Engineering.

Entry BN of domain(n): a BN connecting domain(n-1) to domain(n) along a determined sequence of domains.

Exit BN of domain(n): a BN connecting domain(n) to domain(n+1) along a determined sequence of domains.

Inter-area TE LSP: A TE LSP that crosses an IGP area boundary.

Inter-AS TE LSP: A TE LSP that crosses an AS boundary.

LSP: Label Switched Path.

LSR: Label Switching Router.

PCC: Path Computation Client. Any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or

network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.

PCE(i) is a PCE with the scope of domain(i).

TED: Traffic Engineering Database.

This document uses terminologies defined in RFC5440.

3. 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 RFC2119.

4. Label Distribution

The Label Distribution may be provided by the PCE-based path computation. A PCE responsible for a domain computes a path segment for the domain, which is from an entry boundary to an exit boundary (or an egress) node of the domain. The PCE gets an label from the entry boundary node and adds an label object containing the label in the reply message to be sent to the requesting PCC (or another PCE).

When a PCE or PCC receives a reply message containing an label object, it removes the object from the message. The PCE may store the information in the label object or send the information to another component such as a Tunnel Central Controller (TCC).

4.1. An Exmaple

Figure 1 below illustrates a simple two-AS topology. There is a PCE responsible for the path computation in each AS. A path computation is requested from the Tunnel Central Controller (TCC), acting as the PCC, which sends the path computation request to PCE-1. PCE-1 is unable to compute an end-to-end path and invokes PCE-2 (possibly using the techniques described in [RFC5441]). PCE-2 computes a path segment from entry boundary node ASBR-2 of the right domain to the egress as {ASBR-2, C, D, Egress}. In addition to placing this path segment in the reply message to PCE-1, PCE-2 gets an label from the entry boundary node ASBR-2 and adds an label object containing the label and optionally the ASBR-2 into the reply message.

```
    -------------------------------    ------------------------------
   |             -------           |  |    -------                   |
   |        +-->| PCE-1 |<---------+--+-->| PCE-2 |                  |
   |        |    -------           |  |    -------                   |
   |        v                      |  |    ^                         |
   |      -----                    |  |    |                         |
   |     | TCC |                   |  |    |                         |
   |     | PCC |                   |  |    |                         |
   |      -----                    |  |    v                         |
   |  -------    -    -    ------  |  |  ------    -    -    ------  |
   | |Ingress|--|A|--|B|--|ASBR-1|-+--+-|ASBR-2|--|C|--|D|--|Egress| |
   |  -------    -    -    ------  |  |  ------    -    -    ------  |
   |                               |  |                              |
    -------------------------------  ------------------------------
```

Figure 1: Example of Label Distribution

When PCE-1 receives the reply message containing the label object from PCE-2, it removes the object from the message. PCE-1 may store the information in the label object or send the information to another component such as a Tunnel Central Controller (TCC). TCC may set up the segment of the LSP tunnel from Ingress to ASBR-2 using the label in the label object from ASBR-2.

5. Extensions to PCEP

This section describes the extensions to PCEP for distributing labels crossing domains for an inter-domain Point-to-Point (P2P) or Point-to-Multipoint (P2MP) Traffic Engineering (TE) Label Switched Path (LSP). The extensions include the definition of a new flag in the RP object, tunnel information and label in a PCReq/PCRep message.

5.1. RP Object Extension

The following flags are added into the RP Object:

An L bit is added in the flag bits field of the RP object to tell a receiver of a PCReq/PCRep message that the message is for distributing labels crossing domains for an inter-domain LSP.

- o L (Label distribution bit - 1 bit):

  0: This indicates that this is not a PCReq/PCRep message for distributing labels crossing domains.

  1: This indicates that this is a PCReq or PCRep message for distributing labels crossing domains.

The IANA request is referenced in Section below (Request Parameter Bit Flags) of this document.

This L bit with the N bit defined in RFC6006 can indicate whether the PCReq/PCRep message is for distributing labels for an MPLS TE P2P LSP or an MPLS TE P2MP LSP.

- o L = 1 and N = 0: This indicates that this is a PCReq/PCRep message for distributing labels for a P2P LSP.

- o L = 1 and N = 1: This indicates that this is a PCReq/PCRep message for distributing labels for a P2MP LSP.

The C bit is added in the flag bits field of the RP object to tell the receiver of a PCReq/PCRep message that the message is for creating the segment of the LSP tunnel in a domain before distributing labels from this domain to its previous domain.

- o C (LSP tunnel Creation bit - 1 bit):

  - 0: This indicates that this is not a PCReq/PCRep message for creating the segment of the LSP tunnel.

  - 1: This indicates that this is a PCReq/PCRep message for creating the segment of the LSP tunnel in the domain before distributing labels to its previous domain.

The IANA request is referenced in Section below (Request Parameter Bit Flags) of this document.

5.2. Label Object

The format of a label object body (Object-Type=2) is illustrated below, which comprises a label and an optional node sub object. The node sub object contains a boundary node IP address, from which the label is allocated and distributed.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Label                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Node IPv4/IPv6 sub object (optional)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format of the node IPv4 address sub object (Type=1) is as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|   Type(1)   |   Length (8)  |     Node IPv4 address         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Node IPv4 address (cont)    |        Reserved               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The format of the node IPv6 address sub object (Type=2) is illustrated below:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|   Type(1)   |  Length (20)  |     Node IPv6 address         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                  Node IPv6 address (cont)                     |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Node IPv6 address (cont)    |        Reserved               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

5.3. LSP Tunnel Object

The LSP tunnel object contains the information that may be used to identify an LSP tunnel. An LSP tunnel may be a P2P or P2MP LSP tunnel. It may be an IPv4 or IPv6 LPS tunnel. Thus there are four types of LSP tunnels: 1) P2P LSP IPv4 tunnel, 2) P2P LSP IPv6 tunnel, 3) P2MP LSP IPv4 tunnel, and 4) P2MP LSP IPv6 tunnel.

The format of the P2P LSP IPv4/6 tunnel object body is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       P2P LSP Tunnel Egress IPv4/6 Address (4/16 bytes)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved              |            Tunnel ID          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Extended Tunnel ID (4/16 bytes)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved              |              LSP ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Controller ID (4/16 bytes)                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- o P2P LSP Tunnel Egress IPv4/6 Address:
  IPv4/6 address of the egress of the tunnel.
- o Tunnel ID:
  A 16-bit identifier that is constant over the life of the tunnel.
- o Extended Tunnel ID:
  A 4/16-byte identifier that is constant over the life of the tunnel.
- o LSP ID:
  A 16-bit identifier to allow resources sharing.
- o Controller ID:
  A 4/16-byte identifier for the controller responsible for the head segment of the tunnel.

The format of the P2MP LSP IPv4/6 tunnel object body is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            P2MP ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved              |            Tunnel ID          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Extended Tunnel ID (4/16 bytes)                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Reserved              |              LSP ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Controller ID (4/16 bytes)                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

- o P2MP ID:
  A 32-bit number unique within the ingress of LSP tunnel.
- o Tunnel ID:
  A 16-bit identifier that is constant over the life of the tunnel.
- o Extended Tunnel ID:
  A 4/16-byte identifier that is constant over the life of the tunnel.
- o LSP ID:
  A 16-bit identifier to allow resources sharing.
- o Controller ID:
  A 16-byte identifier for the controller responsible for the head segment of the tunnel.

5.4. Request Message Extension

Figure below illustrates the format of a request message with a optional LSP tunnel object:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>
<request-list>::=<request>[<request-list>]
<request>::= <RP> <END-POINTS> [<OF>] [<LSPA>] [<BANDWIDTH>]
             [<metric-list>] [<RRO>[<BANDWIDTH>]] [<IRO>]
             [<LOAD-BALANCING>]
             [<LSP-tunnel>]
```

Figure 2: Format for Request Message

5.5. Reply Message Extension

Below is the format of a reply message with an optional Label object:

```
<PCReq Message>::= <Common Header>
                   <response-list>
<response-list>::=<response>[<response-list>]
<response>::= <RP>
              [<NO-PATH>]
              [<attribute-list>]
              [<path-list>]
<path-list>::=<path>[<path-list>]
<path>::= <ERO><attribute-list>[<LSP-tunnel>][<Label>]
```

Figure 3: Format for Reply Message

6. Procedures

Ther may be a number of procedures for distributing labels crossing domains.

6.1. Distributing Label in Ordered Setup

Suppose that a path for an MPLS TE LSP tunnel crossing multiple domains is computed by PCEs and a sequence of domains (D1, D2, ..., Dn) through which the path goes are controlled by a sequence of Tunnel Central Controllers TCCs (TCC1, TCC2, ..., TCCn) respectively. The method or procedure for distributing a label in ordered setup may comprise the following steps:

Step 1: TCCi (i = 1, ..., n-1) sends TCCj (j = i + 1) a request for establishing the TE LSP tunnel.

Step 2: TCCn (e.g., TCC3) allocates a label from the enter border node (e.g., border node R) of domain Dn (e.g., D3) and sends TCCn-1 (e.g., TCC2) a reply containing the label after establishing the TE LSP tunnel segment (e.g., from node R to U) in domain Dn (e.g., D3).

Step 3: TCCj (j = n-1, ..., 2) receives a reply containing a first label from TCCj+1, allocates a second label from the enter border node of domain Dj, establishes the TE LSP tunnel segment in Dj and sends TCCi (i = j - 1) a reply containing the label.

Step 4: TCC1 receives a reply containing a label from TCC2 and establishes the TE LSP tunnel segment in D1. At this point, the TE LSP tunnel crossing multiple domains is established.

6.2. Distributing Label in Path Computation

Suppose that a path for an MPLS TE LSP tunnel crossing multiple domains is computed by PCEs and a sequence of domains (D1, D2, ..., Dn) through which the path goes are controlled by a sequence of PCEs (PCE1, PCE2, ..., PCEn) as TCCs respectively. The method or procedure for distributing a label in path computation may comprise the following steps:

Step 1: After PCEn (e.g., PCE3) receives a path request for computing the path and determines that the path segment of the path in domain Dn (e.g., D3) is on the best path, it allocates a label from the enter border node (e.g., R) of domain Dn (e.g., D3) on the path, establishes the TE LSP tunnel segment in domain Dn and sends PCEn-1 (e.g., PCE2) a path reply containing the label.

Step 2: When PCEj (j = n-1, ..., 2) receives a path reply containing a first label from PCEj+1 and determines that the path segment of the path in domain Dj (e.g., D2) is on the best path, it allocates a second label from the enter border node of domain Dj, establishes the TE LSP tunnel segment in Dj and sends PCEi (i = j - 1) a path reply containing the second label.

Step 3: After PCE1 receives a path reply containing a label from PCE2 and determines the path segment in domain D1, it establishes the TE LSP tunnel segment in D1. At this point, the TE LSP tunnel crossing multiple domains is established.

7. Security Considerations

The mechanism described in this document does not raise any new security issues for the PCEP protocols.

8. IANA Considerations

This section specifies requests for IANA allocation.

8.1. Request Parameter Bit Flags

A new RP Object Flag has been defined in this document. IANA is requested to make the following allocation from the "PCEP RP Object Flag Field" Sub-Registry:

Bit	Description	Reference
18	Label Distribution (L-bit)	This I-D
19	LSP tunnel Creation (C-bit)	This I-D

9. Acknowledgement

The author would like to thank people for their valuable comments on this draft.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

10.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC5862] Yasukawa, S. and A. Farrel, "Path Computation Clients (PCC) - Path Computation Element (PCE) Requirements for Point-to-Multipoint MPLS-TE", RFC 5862, June 2010.

Authors' Addresses

Huaimo Chen
Huawei Technologies
Boston, MA
US

Email: huaimo.chen@huawei.com

Autumn Liu
Ericsson
CA
USA

Email: autumn.liu@ericsson.com

Fengman Xu
Verizon
2400 N. Glenville Dr
Richardson, TX 75082
USA

Email: fengman.xu@verizon.com

Mehmet Toy
Comcast
1800 Bishops Gate Blvd.
Mount Laurel, NJ 08054
USA

Email: mehmet_toy@cable.comcast.com

Vic Liu
China Mobile
No.32 Xuanwumen West Street, Xicheng District
Beijing, 100053
China

Email: liuzhiheng@chinamobile.com

PCE Working Group
Internet-Draft
Intended status: Experimental
Expires: March 28, 2014

D. Dhody
Q. Wu
U. Palle
X. Zhang
Huawei Technologies

September 24, 2013

PCE support for Domain Diversity
draft-dwpz-pce-domain-diverse-00

Abstract

The Path Computation Element (PCE) may be used for computing path for services that traverse multi-area and multi-AS Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineered (TE) networks.

Path computation should facilitate the selection of paths with domain diversity. This document examines the existing mechanisms to do so and further propose some extensions to Path Computation Element Protocol (PCEP).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://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 28, 2014.

Copyright Notice

Copyright (c) 2013 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 (http://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. Terminology 3
3. Domain Diversity 3
3.1. Per Domain Path Computation 4
3.2. Backward-Recursive PCE-based Computation 4
3.3. Hierarchical PCE 4
3.3.1. End to End Path 5
3.3.2. Domain-Sequence 5
4. Extension to PCEP 5
4.1. SVEC Object 5
4.2. Transit Domain Identifier 6
4.3. Minimize Shared Domains 6
5. Security Considerations 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 7
6.4. Verify Correct Operations 7
6.5. Requirements On Other Protocols 7
6.6. Impact On Network Operations 7
7. IANA Considerations 7
8. Acknowledgments 8
9. References 8
9.1. Normative References 8
9.2. Informative References 8
Appendix A. Contributor Addresses 9

1. Introduction

The ability to compute shortest constrained TE LSPs in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks across multiple domains has been identified as a key requirement. In this context, a domain is a collection of network elements within a common sphere of address management or path computational responsibility such as an Interior Gateway Protocol (IGP) area or an Autonomous Systems (AS).

In a multi-domain environment, Domain Diversity is defined in [RFC6805]. A pair of paths are domain-diverse if they do not traverse any of the same transit domains. Domain diversity may be maximized for a pair of paths by selecting paths that have the smallest number of shared domains. Path computation should facilitate the selection of domain diverse paths as a way to reduce the risk of shared failure and automatically helps to ensure path diversity for most of the route of a pair of LSPs.

This document examine a way to achieve domain diversity with existing inter-domain path computation mechanism like per-domain path computation technique [RFC5152], Backward Recursive Path Computation (BRPC) mechanism [RFC5441] and Hierarchical PCE [RFC6805]. This document also considers synchronized dependent path computations as well as non-synchronized path computation. Since independent and synchronized path computation cannot be used to apply diversity, it is not discussed in this document.

1.1. 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 [RFC2119].

2. Terminology

The terminology is as per [RFC5440].

3. Domain Diversity

As described in [RFC6805], a set of paths are considered to be domain diverse if they do not share any transit domains, apart from ingress and egress domains.

Some additional parameters to consider would be -

Minimize shared domain: When a fully domain diverse path is not possible, PCE could be requested to minimize the number of shared transit domains. This can also be termed as maximizing partial domain diversity.

Boundary Nodes: TBD

3.1. Per Domain Path Computation

The per domain path computation technique [RFC5152] defines a method where the path is computed during the signaling process (on a per-domain basis). The entry Boundary Node (BN) of each domain is responsible for performing the path computation for the section of the LSP that crosses the domain, or for requesting that a PCE for that domain computes that piece of the path.

Non-Synchronized Path Computation: Path computations are performed in a serialized and independent fashion. After the setup of primary path, a domain diverse path can be signaled by encoding the transit domain identifiers in XRO or EXRS using domain sub-objects defined in [DOMAIN-SUBOBJ] and [RFC3209] in RSVP-TE. Note that the head end LSR should be aware of transit domain identifiers of the primary path to be able to do so.

Synchronized Path Computation: Not Applicable.

3.2. Backward-Recursive PCE-based Computation

The BRPC [RFC5441] technique involves cooperation and communication between PCEs in order to compute an optimal end-to-end path across multiple domains. The sequence of domains to be traversed maybe known before the path computation, but it can also be used when the domain path is unknown and determined during path computation.

Non-Synchronized Path Computation: Path computations are performed in a serialized and independent fashion. After the path computation and setup of primary path, a domain diverse path computation request is sent by PCC to the PCE, by encoding the transit domain identifiers in XRO or EXRS using domain sub-objects defined in [PCE-DOMAIN] and [RFC3209] in PCEP. Note that the PCC should be aware of transit domain identifiers of the primary path to be able to do so.

Synchronized Path Computation: Not Applicable. [Since different transit domain PCEs are involved , there is no way to achieve synchronization for domain diverse paths]. BTW [RFC5440] describes other diversity parameters (node, link etc).

3.3. Hierarchical PCE

In H-PCE [RFC6805] architecture, the parent PCE is used to compute a multi-domain path based on the domain connectivity information. The parent PCE may be requested to provide a end to end path or only the sequence of domains.

3.3.1. End to End Path

Non-Synchronized Path Computation: Path computations are performed in a serialized and independent fashion. After the path computation and setup of primary path, a domain diverse path computation request is sent to the parent PCE, by encoding the transit domain identifiers in XRO or EXRS using domain sub-objects defined in [PCE-DOMAIN] and [RFC3209] in PCEP. Note that the PCC should be aware of transit domain identifiers of the primary path to be able to do so. The parent PCE should provide a domain diverse end to end path.

Synchronized Path Computation: Child PCE should be able to request dependent and synchronized domain diverse end to end paths from its parent PCE. A new flag is added in SVEC object for this (Refer Section 4.1).

3.3.2. Domain-Sequence

Non-Synchronized Path Computation: Path computations are performed in a serialized and independent fashion. After the primary path computation using H-PCE (involving domain-sequence selection by parent PCE and end-to-end path computation via BRPC or Per-Domain mechanisms) and setup, a domain diverse path computation request is sent to the parent PCE, by encoding the transit domain identifiers in XRO or EXRS using domain sub-objects defined in [PCE-DOMAIN] and [RFC3209] in PCEP. Note that the PCC should be aware of transit domain identifiers of the primary path to be able to do so. The parent PCE should provide a diverse domain sequence.

Synchronized Path Computation: Child PCE should be able to request dependent and synchronized diverse domain-sequence(s) from its parent PCE. A new flag is added in SVEC object for this (Refer Section 4.1). The parent PCE should reply with diverse domain sequence(s) encoded in ERO as described in [PCE-DOMAIN].

4. Extension to PCEP

4.1. SVEC Object

[RFC5440] defines SVEC object which includes flags for the potential dependency between the set of path computation requests (Link, Node and SRLG diverse). This document proposes a new flag O for domain diversity.

The format of the SVEC object body is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Reserved    |                   Flags           |O|P|D|S|N|L|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Request-ID-number #1                      |
//                                                             //
|                     Request-ID-number #M                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: SVEC Body Object Format

Following new bit is added in the Flags field:

* O (Domain diverse) bit: when set, this indicates that the computed paths corresponding to the requests specified by the following RP objects MUST NOT have any transit domain(s) in common.

The Domain Diverse O-bit can be used in Hierarchical PCE path computation to compute synchronized domain diverse end to end path or diverse domain sequences as described in Section 3.3.

When domain diverse O bit is set, it is applied to the transit domains. The other bit in SVEC object (N, L etc) is set, should still be applied in the ingress and egress domain.

4.2. Transit Domain Identifier

In case of non-synchronized path computation, Ingress node (i.e. a PCC) should be aware of transit domain identifiers of the primary path. So during the path computation or signaling of the primary path, the transit domain should be identified.

A possible solution for path computation could be a flag in RP object requesting domain identifier to be returned in the PCEP path reply message. Further details - TBD

4.3. Minimize Shared Domains

A new Objective function (OF) [RFC5541] code for synchronized path computation requests is proposed:

MCTD

* Name: Minimize the number of Common Transit Domains.

* Objective Function Code: TBD

* Description: Find a set of paths such that it passes through the least number of common transit domains.

The MCTD OF can be used in Hierarchical PCE path computation to request synchronized domain diverse end to end paths or diverse domain sequences as described in Section 3.3.

For non synchronized diverse domain path computation the X bit in XRO or EXRS [RFC5521] sub-objects can be used, where X bit set as 1 indicates that the domain specified SHOULD be excluded from the path computed by the PCE, but MAY be included subject to PCE policy and the absence of a viable path that meets the other constraints and excludes the domain.

5. Security Considerations

TBD.

6. Manageability Considerations

6.1. Control of Function and Policy

TBD.

6.2. Information and Data Models

TBD.

6.3. Liveness Detection and Monitoring

TBD.

6.4. Verify Correct Operations

TBD.

6.5. Requirements On Other Protocols

TBD.

6.6. Impact On Network Operations

TBD.

7. IANA Considerations

TBD.

8. Acknowledgments

We would like to thank Qilei Wang for starting this discussion in the mailing list.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2. 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, December 2001.

[RFC5152] Vasseur, JP., Ayyangar, A., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, February 2008.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5441] Vasseur, JP., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", RFC 5441, April 2009.

[RFC5521] Oki, E., Takeda, T., and A. Farrel, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions", RFC 5521, April 2009.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, June 2009.

[RFC6805] King, D. and A. Farrel, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, November 2012.

[DOMAIN-SUBOBJ]
Dhody, D., Palle, U., Kondreddy, V., and R. Casellas, "Domain Subobjects for Resource ReserVation Protocol -

Traffic Engineering (RSVP-TE). (draft-dhody-ccamp-rsvp-te-domain-subobjects)", July 2013.

[PCE-DOMAIN]
Dhody, D., Palle, U., and R. Casellas, "Standard Representation Of Domain Sequence. (draft-ietf-pce-pcep-domain-sequence)", July 2013.

Appendix A. Contributor Addresses

Ramon Casellas
CTTC - Centre Tecnologic de Telecomunicacions de Catalunya
Av. Carl Friedrich Gauss n7
Castelldefels, Barcelona 08860
SPAIN

EMail: ramon.casellas@cttc.es

Avantika
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: avantika.sushilkumar@huawei.com

Authors' Addresses

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.ietf@gmail.com

Qin Wu
Huawei Technologies
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

EMail: bill.wu@huawei.com

Udayasree Palle
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: udayasree.palle@huawei.com

Xian Zhang
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R.China

EMail: zhang.xian@huawei.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 14, 2020

C. Margaria, Ed.
Juniper
O. Gonzalez de Dios, Ed.
Telefonica Investigacion y Desarrollo
F. Zhang, Ed.
Huawei Technologies

December 12, 2019

PCEP extensions for GMPLS
draft-ietf-pce-gmpls-pcep-extensions-16

Abstract

A Path Computation Element (PCE) provides path computation functions for Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks. Additional requirements for GMPLS are identified in RFC7025.

This memo provides extensions to the Path Computation Element communication Protocol (PCEP) for the support of the GMPLS control plane to address those requirements.

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 June 14, 2020.

Copyright Notice

Copyright (c) 2019 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. Terminology 3
1.2. PCEP Requirements for GMPLS 5
1.3. Requirements Applicability 5
1.3.1. Requirements on Path Computation Request 6
1.3.2. Requirements on Path Computation Response 7
1.4. Existing Support for GMPLS in Base PCEP Objects and its Limitations 7
2. PCEP Objects and Extensions 10
2.1. GMPLS Capability Advertisement 10
2.1.1. GMPLS Computation TLV in the Existing PCE Discovery Protocol 10
2.1.2. OPEN Object Extension GMPLS-CAPABILITY TLV 10
2.2. RP Object Extension 11
2.3. BANDWIDTH Object Extensions 12
2.4. LOAD-BALANCING Object Extensions 14
2.5. END-POINTS Object Extensions 16
2.5.1. Generalized Endpoint Object Type 17
2.5.2. END-POINTS TLV Extensions 20
2.6. IRO Extension 24
2.7. XRO Extension 24
2.8. LSPA Extensions 26
2.9. NO-PATH Object Extension 26
2.9.1. Extensions to NO-PATH-VECTOR TLV 27
3. Additional Error-Types and Error-Values Defined 27
4. Manageability Considerations 29
4.1. Control of Function through Configuration and Policy . . 29
4.2. Information and Data Models 29
4.3. Liveness Detection and Monitoring 29
4.4. Verifying Correct Operation 30
4.5. Requirements on Other Protocols and Functional Components 30
4.6. Impact on Network Operation 30
5. IANA Considerations 30
5.1. PCEP Objects 30
5.2. Endpoint type field in Generalized END-POINTS Object . . 31
5.3. New PCEP TLVs 32
5.4. RP Object Flag Field 32
5.5. New PCEP Error Codes 32
5.6. New NO-PATH-VECTOR TLV Fields 33

5.7. New Subobject for the Include Route Object 34
5.8. New Subobject for the Exclude Route Object 34
5.9. New GMPLS-CAPABILITY TLV Flag Field 35
6. Security Considerations 35
7. Contributing Authors 36
8. Acknowledgments 38
9. References 38
9.1. Normative References 38
9.2. Informative References 42
Appendix A. LOAD-BALANCING Usage for SDH Virtual Concatenation . 43
Authors' Addresses 43

1. Introduction

Although [RFC4655] defines the PCE architecture and framework for both MPLS and GMPLS networks, most preexisting PCEP RFCs [RFC5440], [RFC5521], [RFC5541], [RFC5520] are focused on MPLS networks, and do not cover the wide range of GMPLS networks. This document complements these RFCs by addressing the extensions required for GMPLS applications and routing requests, for example for Optical Transport Network (OTN) and Wavelength Switched Optical Network (WSON) networks.

The functional requirements to be addressed by the PCEP extensions to support these applications are fully described in [RFC7025] and [RFC7449].

1.1. Terminology

This document uses terminologies from the PCE architecture document [RFC4655], the PCEP documents including [RFC5440], [RFC5521], [RFC5541], [RFC5520], [RFC7025] and [RFC7449], and the GMPLS documents such as [RFC3471], [RFC3473] and so on. Note that it is expected the reader is familiar with these documents. The following abbreviations are used in this document

ODU ODU Optical Channel Data Unit [G.709-v3]

OTN Optical Transport Network [G.709-v3]

L2SC Layer-2 Switch Capable [RFC3471]

TDM Time-Division Multiplex Capable [RFC3471]

LSC Lambda Switch Capable [RFC3471]

SONET Synchronous Optical Networking

SDH Synchronous Digital Hierarchy

PCC Path Computation Client

RSVP-TE Resource Reservation Protocol - Traffic Engineering

LSP Label Switched Path

TE-LSP Traffic Engineering LSP

IRO Include Route Object

ERO Explicit Route Object

XRO eXclude Route Object

RRO Record Route Object

LSPA LSP Attribute

SRLG Shared Risk Link Group

NVC Number of Virtual Components [RFC4328][RFC4606]

NCC Number of Contiguous Components [RFC4328][RFC4606]

MT Multiplier [RFC4328][RFC4606]

RCC Requested Contiguous Concatenation [RFC4606]

PCReq Path Computation Request [RFC5440]

PCRep Path Computation Reply [RFC5440]

MEF Metro Ethernet Forum

SSON Spectrum-Switched Optical Network

P2MP Point to Multi-Point

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.

1.2. PCEP Requirements for GMPLS

The document [RFC7025] describes the set of PCEP requirements to support GMPLS TE-LSPs. This document assumes a significant familiarity with [RFC7025] and existing PCEP extensions. As a short overview, those requirements can be broken down into the following categories.

- o Which data flow is switched by the LSP: a combination of Switching type (for instance L2SC or TDM), LSP Encoding type (e.g., Ethernet, SONET/SDH) and sometimes the Signal Type (e.g., in case of TDM/LSC switching capability).

- o Data flow specific traffic parameters, which are technology specific. For instance, in SDH/SONET and [G.709-v3] OTN networks the Concatenation Type and the Concatenation Number have an influence on the switched data and on which link it can be supported

- o Support for asymmetric bandwidth requests.

- o Support for unnumbered interface identifiers, as defined in [RFC3477]

- o Label information and technology specific label(s) such as wavelength labels as defined in [RFC6205]. A PCC should also be able to specify a label restriction similar to the one supported by RSVP-TE in [RFC3473].

- o Ability to indicate the requested granularity for the path ERO: node, link or label. This is to allow the use of the explicit label control feature of RSVP-TE.

The requirements of [RFC7025] apply to several objects conveyed by PCEP, this is described in Section 1.3. Some of the requirements of [RFC7025] are already supported in existing documents, as described in Section 1.4.

This document describes a set of PCEP extensions, including new object types, TLVs, encodings, error codes and procedures, in order to fulfill the aforementioned requirements not covered in existing RFCs.

1.3. Requirements Applicability

This section follows the organization of [RFC7025] Section 3 and indicates, for each requirement, the affected piece of information carried by PCEP and its scope.

1.3.1. Requirements on Path Computation Request

(1) Switching capability/type: as described in [RFC3471] this piece of information is used with the Encoding Type and Signal Type to fully describe the switching technology and data carried by the TE-LSP. This is applicable to the TE-LSP itself and also to the TE-LSP endpoint (Carried in the END-POINTS object for MPLS networks in [RFC5440]) when considering multiple network layers. Inter-layer path computation requirements are addressed in in [RFC8282] which addressing the TE-LSP itself, but the TE-LSP endpoints are not addressed.

(2) Encoding type: see (1).

(3) Signal type: see (1).

(4) Concatenation type: this parameter and the Concatenation Number (5) are specific to some TDM (SDH and ODU) switching technology. They MUST be described together and are used to derive the requested resource allocation for the TE-LSP. It is scoped to the TE-LSP and is related to the [RFC5440] BANDWIDTH object in MPLS networks. See [RFC4606] and [RFC4328] about concatenation information.

(5) Concatenation number: see (4).

(6) Technology-specific label(s): as described in [RFC3471] the GMPLS Labels are specific to each switching technology. They can be specified on each link and also on the TE-LSP endpoints , in WSON networks for instance, as described in [RFC6163]. The label restriction can apply to endpoints and on each hop, the related PCEP objects are END-POINTS, IRO, XRO and RRO.

(7) End-to-End (E2E) path protection type: as defined in [RFC4872], this is applicable to the TE-LSP. In MPLS networks the related PCEP object is LSPA (carrying local protection information).

(8) Administrative group: as defined in [RFC3630], this information is already carried in the LSPA object.

(9) Link protection type: as defined in [RFC4872], this is applicable to the TE-LSP and is carried in association with the E2E path protection type.

(10) Support for unnumbered interfaces: as defined in [RFC3477]. Its scope and related objects are the same as labels

(11) Support for asymmetric bandwidth requests: as defined [RFC6387], the scope is similar to (4)

(12) Support for explicit label control during the path computation. This affects the TE-LSP and amount of information returned in the ERO.

(13) Support of label restrictions in the requests/responses: This is described in (6).

1.3.2. Requirements on Path Computation Response

(1) Path computation with concatenation: This is related to Path Computation request requirement (4). In addition there is a specific type of concatenation called virtual concatenation that allows different routes to be used between the endpoints. It is similar to the semantic and scope of the LOAD-BALANCING in MPLS networks.

(2) Label constraint: The PCE should be able to include Labels in the path returned to the PCC, the related object is the ERO object.

(3) Roles of the routes: as defined in [RFC4872], this is applicable to the TE-LSP and is carried in association with the E2E path protection type.

1.4. Existing Support for GMPLS in Base PCEP Objects and its Limitations

The support provided by specifications in [RFC8282] and [RFC5440] for the requirements listed in [RFC7025] is summarized in Table 1 and Table 2. In some cases the support may not be complete, as noted, and additional support need to be provided in this specification.

Req.	Name	Support
1	Switching capability/type	SWITCH-LAYER (RFC8282)
2	Encoding type	SWITCH-LAYER (RFC8282)
3	Signal type	SWITCH-LAYER (RFC8282)
4	Concatenation type	No
5	Concatenation number	No
6	Technology-specific label	(Partial) ERO (RFC5440)
7	End-to-End (E2E) path protection type	No
8	Administrative group	LSPA (RFC5440)
9	Link protection type	No
10	Support for unnumbered interfaces	(Partial) ERO (RFC5440)
11	Support for asymmetric bandwidth requests	No
12	Support for explicit label control during the path computation	No
13	Support of label restrictions in the requests/responses	No

Table 1: RFC7025 Section 3.1 requirements support

Req.	Name	Support
1	Path computation with concatenation	No
2	Label constraint	No
3	Roles of the routes	No

Table 2: RFC7025 Section 3.2 requirements support

As described in Section 1.3 PCEP as of [RFC5440], [RFC5521] and [RFC8282], supports the following objects, included in requests and responses, related to the described requirements.

From [RFC5440]:

- o END-POINTS: related to requirements (1, 2, 3, 6, 10 and 13). The object only supports numbered endpoints. The context specifies whether they are node identifiers or numbered interfaces.

- o BANDWIDTH: related to requirements (4, 5 and 11). The data rate is encoded in the bandwidth object (as IEEE 32 bit float). [RFC5440] does not include the ability to convey an encoding proper to all GMPLS-controlled networks.

- o ERO: related to requirements (6, 10, 12 and 13). The ERO content is defined in RSVP in [RFC3209][RFC3473][RFC3477][RFC7570] and supports all the requirements already.

- o LSPA: related to requirements (7, 8 and 9). The requirement 8 (setup and holding priorities) is already supported.

From [RFC5521]:

- o XRO:

  - * This object allows excluding (strict or not) resources and is related to requirements (6, 10 and 13). It also includes the requested diversity (node, link or SRLG).

  - * When the F bit is set, the request indicates that the existing path has failed and the resources present in the RRO can be reused.

From [RFC8282]:

- o SWITCH-LAYER: addresses requirements (1, 2 and 3) for the TE-LSP and indicates which layer(s) should be considered. The object can be used to represent the RSVP-TE generalized label request. It does not address the endpoints case of requirements (1, 2 and 3).

- o REQ-ADAP-CAP: indicates the adaptation capabilities requested, can also be used for the endpoints in case of mono-layer computation

The gaps in functional coverage of the base PCEP objects are:

The BANDWIDTH and LOAD-BALANCING objects do not describe the details of the traffic request (requirements 4 and 5, for example NVC, multiplier) in the context of GMPLS networks, for instance TDM or OTN networks.

The END-POINTS object does not allow specifying an unnumbered interface, nor potential label restrictions on the interface (requirements 6, 10 and 13). Those parameters are of interest in case of switching constraints.

The Include/eXclude Route Objects (IRO/XRO) do not allow the inclusion/exclusion of labels (requirements 6, 10 and 13).

Base attributes do not allow expressing the requested link protection level and/or the end-to-end protection attributes.

The PCEP extensions defined later in this document to cover the gaps are:

Two new object types are defined for the BANDWIDTH object (Generalized bandwidth, Generalized bandwidth of existing TE-LSP for which a reoptimization is requested).

A new object type is defined for the LOAD-BALANCING object (Generalized Load Balancing).

A new object type is defined for the END-POINTS object (Generalized Endpoint).

A new TLV is added to the Open message for capability negotiation.

A new TLV is added to the LSPA object.

The Label TLV is now allowed in the IRO and XRO objects.

In order to indicate the used routing granularity in the response, a new flag in the RP object is added.

2. PCEP Objects and Extensions

This section describes the necessary PCEP objects and extensions. The PCReq and PCRep messages are defined in [RFC5440]. This document does not change the existing grammars.

2.1. GMPLS Capability Advertisement

2.1.1. GMPLS Computation TLV in the Existing PCE Discovery Protocol

IGP-based PCE Discovery (PCED) is defined in [RFC5088] and [RFC5089] for the OSPF and IS-IS protocols. Those documents have defined bit 0 in PCE-CAP-FLAGS Sub-TLV of the PCED TLV as "Path computation with GMPLS link constraints". This capability is optional and can be used to detect GMPLS-capable PCEs. PCEs that set the bit to indicate support of GMPLS path computation MUST follow the procedures in Section 2.1.2 to further qualify the level of support during PCEP session establishment.

2.1.2. OPEN Object Extension GMPLS-CAPABILITY TLV

In addition to the IGP advertisement, a PCEP speaker MUST be able to discover the other peer GMPLS capabilities during the Open message exchange. This capability is also useful to avoid misconfigurations. This document defines a GMPLS-CAPABILITY TLV for use in the OPEN object to negotiate the GMPLS capability. The inclusion of this TLV

in the Open message indicates that the PCEP speaker support the PCEP extensions defined in the document. A PCEP speaker that is able to support the GMPLS extensions defined in this document MUST include the GMPLS-CAPABILITY TLV on the Open message. If one of the PCEP peers does not include the GMPLS-CAPABILITY TLV in the Open message, the peers MUST NOT make use of the objects and TLVs defined in this document.

If the PCEP speaker supports the extensions of this specification but did not advertise the GMPLS-CAPABILITY capability, upon receipt of a message from the PCE including an extension defined in this document, it MUST generate a PCEP Error (PCErr) with Error-Type=10 (Reception of an invalid object) and Error-value=TBA-42 (Missing GMPLS-CAPABILITY TLV), and it SHOULD terminate the PCEP session.

IANA has allocated value TBA-1 from the "PCEP TLV Type Indicators" sub-registry, as documented in Section 5.3 ("New PCEP TLVs"). The description is "GMPLS-CAPABILITY". Its format is shown in the following figure.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Type=TBA-1      |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Flags                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

No Flags are defined in this document, they are reserved for future use.

2.2. RP Object Extension

Explicit label control (ELC) is a procedure supported by RSVP-TE, where the outgoing labels are encoded in the ERO. As a consequence, the PCE can provide such labels directly in the path ERO. Depending on policies or switching layer, it can be necessary for the PCC to use explicit label control or explicit link ids, thus it needs to indicate in the PCReq which granularity it is expecting in the ERO. This corresponds to requirement 12 of [RFC7025]. The possible granularities can be node, link or label. The granularities are inter-dependent, in the sense that link granularity implies the presence of node information in the ERO; similarly, a label granularity implies that the ERO contains node, link and label information.

A new 2-bit routing granularity (RG) flag (Bits TBA-13) is defined in the RP object. The values are defined as follows

0: reserved
1: node
2: link
3: label

Table 3: RG flag

The flag in the RP object indicates the requested route granularity. The PCE SHOULD follow this granularity and MAY return a NO-PATH if the requested granularity cannot be provided. The PCE MAY return any granularity on the route based on its policy. The PCC can decide if the ERO is acceptable based on its content.

If a PCE honored the requested routing granularity for a request, it MUST indicate the selected routing granularity in the RP object included in the response. Otherwise, the PCE MUST use the reserved RG to leave the check of the ERO to the PCC. The RG flag is backward-compatible with [RFC5440]: the value sent by an implementation (PCC or PCE) not supporting it will indicate a reserved value.

2.3. BANDWIDTH Object Extensions

From [RFC5440] the object carrying the requested size for the TE-LSP is the BANDWIDTH object. The object types 1 and 2 defined in [RFC5440] do not describe enough information to describe the TE-LSP bandwidth in GMPLS networks. The BANDWIDTH object encoding has to be extended to allow the object to express the bandwidth as described in [RFC7025]. RSVP-TE extensions for GMPLS provide a set of encodings allowing such representation in an unambiguous way, this is encoded in the RSVP-TE TSpec and FlowSpec objects. This document extends the BANDWIDTH object with new object types reusing the RSVP-TE encoding.

The following possibilities are supported by the extended encoding:

- o Asymmetric bandwidth (different bandwidth in forward and reverse direction), as described in [RFC6387]

- o GMPLS (SDH/SONET, G.709, ATM, MEF, etc.) parameters.

This corresponds to requirements 3, 4, 5 and 11 of [RFC7025] Section 3.1.

This document defines two Object Types for the BANDWIDTH object:

TBA-2 Generalized bandwidth

TBA-3 Generalized bandwidth of an existing TE-LSP for which a reoptimization is requested

The definitions below apply for Object Type TBA-2 and TBA-3. The body is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Bandwidth Spec Length      | Rev. Bandwidth Spec Length    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Bw Spec Type  |   Reserved                                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                      Generalized Bandwidth                    ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~           Optional:  Reverse Generalized Bandwidth            ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                        Optional TLVs                          ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The BANDWIDTH object type TBA-2 and TBA-3 have a variable length. The 16-bit Bandwidth Spec Length field indicates the length of the Generalized Bandwidth field. The Bandwidth Spec Length MUST be strictly greater than 0. The 16-bit Reverse Bandwidth Spec Length field indicates the length of the Reverse Generalized Bandwidth field. The Reverse Bandwidth Spec Length MAY be equal to 0.

The Bw Spec Type field determines which type of bandwidth is represented by the object.

The Bw Spec Type corresponds to the RSVP-TE SENDER_TSPEC (Object Class 12) C-Types

The encoding of the fields Generalized Bandwidth and Reverse Generalized Bandwidth is the same as the Traffic Parameters carried in RSVP-TE, it can be found in the following references. It is to be noted that the RSVP-TE traffic specification MAY also include TLVs (e.g., [RFC6003] different from the PCEP TLVs).

Bw Spec Type	Name	Reference
2	Intserv	[RFC2210]
4	SONET/SDH	[RFC4606]
5	G.709	[RFC4328]
6	Ethernet	[RFC6003]
7	OTN-TDM	[RFC7139]
8	SSON	[RFC7792]

Table 4: Generalized Bandwidth and Reverse Generalized Bandwidth field encoding

When a PCC requests a bi-directional path with symmetric bandwidth, it SHOULD only specify the Generalized Bandwidth field, and set the Reverse Bandwidth Spec Length to 0. When a PCC needs to request a bi-directional path with asymmetric bandwidth, it SHOULD specify the different bandwidth in the forward and reverse directions with a Generalized Bandwidth and Reverse Generalized Bandwidth fields.

The procedure described in [RFC5440] for the PCRep is unchanged: a PCE MAY include the BANDWIDTH objects in the response to indicate the BANDWIDTH of the path.

As specified in [RFC5440] in the case of the reoptimization of a TE-LSP, the bandwidth of the existing TE-LSP MUST also be included in addition to the requested bandwidth if and only if the two values differ. The Object Type TBA-3 MAY be used instead of the previously specified object type 2 to indicate the existing TE-LSP bandwidth originally specified with object type TBA-2. A PCC that requested a path with a BANDWIDTH object of object type 1 MUST use object type 2 to represent the existing TE-LSP BANDWIDTH.

OPTIONAL TLVs MAY be included within the object body to specify more specific bandwidth requirements. No TLVs for the Object Type TBA-2 and TBA-3 are defined by this document.

2.4. LOAD-BALANCING Object Extensions

The LOAD-BALANCING object [RFC5440] is used to request a set of at most Max-LSP TE-LSP having in total the bandwidth specified in BANDWIDTH, with each TE-LSP having at least a specified minimum bandwidth. The LOAD-BALANCING follows the bandwidth encoding of the BANDWIDTH object, and thus the existing definition from [RFC5440] does not describe enough details for the bandwidth specification expected by GMPLS.

Similarly to the BANDWIDTH object, a new object type is defined to allow a PCC to represent the bandwidth types supported by GMPLS networks.

This document defines the Generalized Load Balancing object type TBA-4 for the LOAD-BALANCING object. The Generalized Load Balancing object type has a variable length.

The format of the Generalized Load Balancing object type is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Bandwidth Spec Length      | Reverse Bandwidth Spec Length |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Bw Spec Type  |  Max-LSP      | Reserved                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Min Bandwidth Spec                                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Min Reverse Bandwidth Spec (optional)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                Optional   TLVs                                ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Bandwidth Spec Length (16 bits): the total length of the Min Bandwidth Spec field. The length MUST be strictly greater than 0.

Reverse Bandwidth Spec Length (16 bits): the total length of the Min Reverse Bandwidth Spec field. It MAY be equal to 0.

Bw Spec Type (8 bits): the bandwidth specification type, it corresponds to the RSVP-TE SENDER_TSPEC (Object Class 12) C-Types.

Max-LSP (8 bits): maximum number of TE-LSPs in the set.

Min Bandwidth Spec (variable): specifies the minimum bandwidth specification of each element of the TE-LSP set.

Min Reverse Bandwidth Spec (variable): specifies the minimum reverse bandwidth specification of each element of the TE-LSP set.

The encoding of the fields Min Bandwidth Spec and Min Reverse Bandwidth Spec is the same as in RSVP-TE SENDER_TSPEC object, it can be found in Table 4 from Section 2.3 from this document.

When a PCC requests a bi-directional path with symmetric bandwidth while specifying load balancing constraints it SHOULD specify the Min Bandwidth Spec field, and set the Reverse Bandwidth Spec Length to 0. When a PCC needs to request a bi-directional path with asymmetric bandwidth while specifying load balancing constraints, it MUST specify the different bandwidth in forward and reverse directions through a Min Bandwidth Spec and Min Reverse Bandwidth Spec fields.

OPTIONAL TLVs MAY be included within the object body to specify more specific bandwidth requirements. No TLVs for the Generalized Load Balancing object type are defined by this document.

The semantic of the LOAD-BALANCING object is not changed. If a PCC requests the computation of a set of TE-LSPs with at most N TE-LSPs so that it can carry generalized bandwidth X , each TE-LSP must at least transport bandwidth B, it inserts a BANDWIDTH object specifying X as the required bandwidth and a LOAD-BALANCING object with the Max-LSP and Min Bandwidth Spec fields set to N and B, respectively. When the BANDWIDTH and Min Bandwidth Spec can be summarized as scalars, the sum of all TE-LSPs bandwith in the set is greater than X. The mapping of X over N path with (at least) bandwidth B is technology and possibly node specific. Each standard definition of the transport technology is defining those mappings and are not repeated in this document. A simplified example for SDH is described in Appendix A

In all other cases, including for technologies based on statistical multiplexing (e.g., InterServ, Ethernet), the exact bandwidth management (e.g., Ethernet’s Excessive Rate) is left to the PCE’s policies, according to the operator’s configuration. If required, further documents may introduce a new mechanism to finely express complex load balancing policies within PCEP.

The BANDWITH and LOAD-BALANCING Bw Spec Type can be different depending on the endpoint nodes architecture. When the PCE is not able to handle those two Bw Spec Type, it MUST return a NO-PATH with the bit "LOAD-BALANCING could not be performed with the bandwidth constraits " set in the NO-PATH-VECTOR TLV.

2.5. END-POINTS Object Extensions

The END-POINTS object is used in a PCEP request message to specify the source and the destination of the path for which a path computation is requested. From [RFC5440], the source IP address and the destination IP address are used to identify those. A new Object Type is defined to address the following possibilities:

- o Different source and destination endpoint types.

- o Label restrictions on the endpoint.
- o Specification of unnumbered endpoints type as seen in GMPLS networks.

The Object encoding is described in the following sections.

In path computation within a GMPLS context the endpoints can:

- o Be unnumbered as described in [RFC3477].
- o Have labels associated to them, specifying a set of constraints on the allocation of labels.
- o Have different switching capabilities

The IPv4 and IPv6 endpoints are used to represent the source and destination IP addresses. The scope of the IP address (Node or numbered Link) is not explicitly stated. It is also possible to request a Path between a numbered link and an unnumbered link, or a P2MP path between different type of endpoints.

This document defines the Generalized Endpoint object type TBA-5 for the END-POINTS object. This new type also supports the specification of constraints on the endpoint label to be used. The PCE might know the interface restrictions but this is not a requirement. This corresponds to requirements 6 and 10 of [RFC7025].

2.5.1. Generalized Endpoint Object Type

The Generalized Endpoint object type format consists of a body and a list of TLVs scoped to this object. The TLVs give the details of the endpoints and are described in Section 2.5.2. For each Endpoint Type, a different grammar is defined. The TLVs defined to describe an endpoint are:

1. IPv4 address endpoint.
2. IPv6 address endpoint.
3. Unnumbered endpoint.
4. Label request.
5. Label set.

The Label set TLV is used to restrict or suggest the label allocation in the PCE. This TLV expresses the set of restrictions which may

apply to signaling. Label restriction support can be an explicit or a suggested value (Label set describing one label, with the L bit respectively cleared or set), mandatory range restrictions (Label set with L bit cleared) and optional range restriction (Label set with L bit set). Endpoints label restriction may not be part of the RRO or IRO. They can be included when following [RFC4003] in signaling for egress endpoint, but ingress endpoint properties can be local to the PCC and not signaled. To support this case the label set allows indication which label are used in case of reoptimization. The label range restrictions are valid in GMPLS-controlled networks, either by PCC policy or depending on the switching technology used, for instance on given Ethernet or ODU equipment having limited hardware capabilities restricting the label range. Label set restriction also applies to WSON networks where the optical senders and receivers are limited in their frequency tunability ranges, consequently restricting the possible label ranges on the interface in GMPLS. The END-POINTS Object with Generalized Endpoint object type is encoded as follow:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Reserved                                 | Endpoint Type |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                            TLVs                               ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Reserved bits SHOULD be set to 0 when a message is sent and ignored when the message is received.

The Endpoint Type is defined as follow:

Value	Type	Meaning
0	Point-to-Point	
1	Point-to-Multipoint	New leaves to add
2		Old leaves to remove
3		Old leaves whose path can be modified/reoptimized
4		Old leaves whose path has to be left unchanged
5-244	Reserved	
245-255	Experimental range	

Table 5: Generalized Endpoint endpoint types

The Endpoint Type is used to cover both point-to-point and different point-to-multipoint endpoints. A PCE may accept only Endpoint Type 0: Endpoint Types 1-4 apply if the PCE implementation supports P2MP path calculation. A PCE not supporting a given Endpoint Type SHOULD respond with a PCErr with Error-Type=4 (Not supported object), Error-value=TBA-15 (Unsupported endpoint type in END-POINTS Generalized Endpoint object type). As per [RFC5440], a PCE unable to process Generalized Endpoints may respond with Error-Type=3 (Unknown Object), Error-value=2 (Unrecognized object Type) or Error-Type=4 (Not supported object), Error-value=2 (Not supported object Type). The TLVs present in the request object body MUST follow the following [RFC5511] grammar:

```
<generalized-endpoint-tlvs>::=
  <p2p-endpoints> | <p2mp-endpoints>

<p2p-endpoints> ::=
  <endpoint> [<endpoint-restriction-list>]
  <endpoint> [<endpoint-restriction-list>]

<p2mp-endpoints> ::=
  <endpoint> [<endpoint-restriction-list>]
  <endpoint> [<endpoint-restriction-list>]
  [<endpoint> [<endpoint-restriction-list>]]...
```

For endpoint type Point-to-Point, 2 endpoint TLVs MUST be present in the message. The first endpoint is the source and the second is the destination.

For endpoint type Point-to-Multipoint, several END-POINT objects MAY be present in the message and the exact meaning depending on the endpoint type defined for the object. The first endpoint TLV is the root and other endpoints TLVs are the leaves. The root endpoint MUST be the same for all END-POINTS objects for that P2MP tree request. If the root endpoint is not the same for all END-POINTS, a PCErr with Error-Type=17 (P2MP END-POINTS Error), Error-value=4 (The PCE cannot satisfy the request due to inconsistent END-POINTS) MUST be returned. The procedure defined in [RFC8306] Section 3.10 also apply to the Generalized Endpoint with Point-to-Multipoint endpoint types.

An endpoint is defined as follows:

```
<endpoint>::=<IPV4-ADDRESS>|<IPV6-ADDRESS>|<UNNUMBERED-ENDPOINT>
<endpoint-restriction-list> ::=               <endpoint-restriction>
             [<endpoint-restriction-list>]

<endpoint-restriction> ::=
                 [<LABEL-REQUEST>][<label-restriction-list>]

<label-restriction-list> ::= <label-restriction>
                             [<label-restriction-list>]
<label-restriction> ::= <LABEL-SET>
```

The different TLVs are described in the following sections. A PCE MAY support any or all of IPV4-ADDRESS, IPV6-ADDRESS, and UNNUMBERED-ENDPOINT TLVs. When receiving a PCReq, a PCE unable to resolve the identifier in one of those TLVs MUST respond using a PCRep with NO-PATH and set the bit "Unknown destination" or "Unknown source" in the NO-PATH-VECTOR TLV. The response SHOULD include the END-POINTS object with only the unsupported TLV(s).

A PCE MAY support either or both of the LABEL-REQUEST and LABEL-SET TLVs. If a PCE finds a non-supported TLV in the END-POINTS the PCE MUST respond with a PCErr message with Error-Type=4 (Not supported object) and Error-value=TBA-15 (Unsupported TLV present in END-POINTS Generalized Endpoint object type) and the message SHOULD include the END-POINTS object in the response with only the endpoint and endpoint restriction TLV it did not understand. A PCE supporting those TLVs but not being able to fulfil the label restriction MUST send a response with a NO-PATH object which has the bit "No endpoint label resource" or "No endpoint label resource in range" set in the NO-PATH-VECTOR TLV. The response SHOULD include an END-POINTS object containing only the TLV(s) related to the constraints the PCE could not meet.

2.5.2. END-POINTS TLV Extensions

All endpoint TLVs have the standard PCEP TLV header as defined in [RFC5440] Section 7.1. For the Generalized Endpoint Object Type the TLVs MUST follow the ordering defined in Section 2.5.1.

2.5.2.1. IPV4-ADDRESS TLV

This TLV represents a numbered endpoint using IPv4 numbering, the format of the IPv4-ADDRESS TLV value (TLV-Type=TBA-6) is as follows:

```
 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 address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This TLV MAY be ignored, in which case a PCRep with NO-PATH SHOULD be returned, as described in Section 2.5.1.

2.5.2.2. IPV6-ADDRESS TLV

This TLV represents a numbered endpoint using IPV6 numbering, the format of the IPv6-ADDRESS TLV value (TLV-Type=TBA-7) is as follows:

```
 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 address (16 bytes)                          |
|                                                               |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This TLV MAY be ignored, in which case a PCRep with NO-PATH SHOULD be returned, as described in Section 2.5.1.

2.5.2.3. UNNUMBERED-ENDPOINT TLV

This TLV represents an unnumbered interface. This TLV has the same semantic as in [RFC3477]. The TLV value is encoded as follows (TLV-Type=TBA-8)

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          LSR's Router ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Interface ID (32 bits)                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This TLV MAY be ignored, in which case a PCRep with NO-PATH SHOULD be returned, as described in Section 2.5.1.

2.5.2.4. LABEL-REQUEST TLV

The LABEL-REQUEST TLV indicates the switching capability and encoding type of the following label restriction list for the endpoint. The value format and encoding is the same as described in [RFC3471]

Section 3.1 Generalized label request. The LABEL-REQUEST TLV uses TLV-Type=TBA-9. The Encoding Type indicates the encoding type, e.g., SONET/SDH/GigE etc., of the LSP with which the data is associated. The Switching type indicates the type of switching that is being requested on the endpoint. G-PID identifies the payload. This TLV and the following one are defined to satisfy requirement 13 of [RFC7025] for the endpoint. It is not directly related to the TE-LSP label request, which is expressed by the SWITCH-LAYER object.

On the path calculation request only the GENERALIZED-BANDWIDTH and SWITCH-LAYER need to be coherent, the endpoint labels could be different (supporting a different LABEL-REQUEST). Hence the label restrictions include a Generalized label request in order to interpret the labels. This TLV MAY be ignored, in which case a PCRep with NO-PATH SHOULD be returned, as described in Section 2.5.1.

2.5.2.5. LABEL-SET TLV

Label or label range restrictions can be specified for the TE-LSP endpoints. Those are encoded using the LABEL-SET TLV. The label value need to be interpreted with a description on the Encoding and switching type. The REQ-ADAP-CAP object from [RFC8282] can be used in case of mono-layer request, however in case of multilayer it is possible to have more than one object, so it is better to have a dedicated TLV for the label and label request. These TLVs MAY be ignored, in which case a response with NO-PATH SHOULD be returned, as described in Section 2.5.1. TLVs are encoded as follows (following [RFC5440]):

- o LABEL-SET TLV, Type=TBA-10. The TLV Length is variable, Encoding follows [RFC3471] Section 3.5 "Label set" with the addition of a U bit, O bit and L bit. The L bit is used to represent a suggested set of labels, following the semantic of SUGGESTED_LABEL defined by [RFC3471].

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Action     |    Reserved |L|O|U|        Label Type         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Subchannel 1                          |
|                             ...                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
:                              :                                :
:                              :                                :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Subchannel N                          |
|                             ...                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

A LABEL-SET TLV represents a set of possible labels that can be used on an interface. If the L bit is cleared, the label allocated on the first endpoint MUST be within the label set range. The action parameter in the Label set indicates the type of list provided. These parameters are described by [RFC3471] Section 3.5.1.

The U, O and L bits have the following meaning:

U: Upstream direction: The U bit is set for upstream (revers) direction in case of bidirectional LSP.
O: Old Label: set when the TLV represent the old (previously allocated) label in case of re-optimization. The R bit of the RP object MUST be set to 1. If the L bit is set, this bit SHOULD be set to 0 and ignored on receipt. When this bit is set, the Action field MUST be set to 0 (Inclusive List) and the Label Set MUST contain one subchannel.
L: Loose Label: set when the TLV indicates to the PCE a set of preferred (ordered) labels to be used. The PCE MAY use those labels for label allocation.

Labels TLV bits

Several LABEL_SET TLVs MAY be present with the O bit cleared, LABEL_SET TLVs with L bit set can be combined with a LABEL_SET TLV with L bit cleared. There MUST NOT be more than two LABEL_SET TLVs present with the O bit set. If there are two LABEL_SET TLVs present, there MUST NOT be more than one with the U bit set, and there MUST NOT be more than one with the U bit cleared. For a given U bit value, if more than one LABEL_SET TLV with the O bit set is present, the first TLV MUST be processed and the following TLVs with the same U and O bit MUST be ignored.

A LABEL-SET TLV with the O and L bit set MUST trigger a PCErr message with Error-Type=10 (Reception of an invalid object) Error-value=TBA-25 (Wrong LABEL-SET TLV present with O and L bit set).

A LABEL-SET TLV with the O bit set and an Action Field not set to 0 (Inclusive list) or containing more than one subchannel MUST trigger a PCErr message with Error-Type=10 (Reception of an invalid object) Error-value=TBA-26 (Wrong LABEL-SET TLV present with O bit and wrong format).

If a LABEL-SET TLV is present with O bit set, the R bit of the RP object MUST be set, otherwise a PCErr message MUST be sent with Error-Type=10 (Reception of an invalid object) Error-value=TBA-24 (LABEL-SET TLV present with O bit set but without R bit set in RP).

2.6. IRO Extension

The IRO as defined in [RFC5440] is used to include specific objects in the path. RSVP-TE allows the inclusion of a label definition. In order to fulfill requirement 13 of [RFC7025] the IRO needs to support the new subobject type as defined in [RFC3473]:

Type	Sub-object
TBA-38	LABEL

The Label subobject MUST follow a subobject identifying a link, currently an IP address subobject (Type 1 or 2) or an interface ID (type 4) subobject. If an IP address subobject is used, then the given IP address MUST be associated with a link. More than one label subobject MAY follow each link subobject. The procedure associated with this subobject is as follows.

If the PCE is able to allocate labels (e.g., via explicit label control) the PCE MUST allocate one label from within the set of label values for the given link. If the PCE does not assign labels, then it sends a response with a NO-PATH object, containing a NO-PATH-VECTOR TLV with the bit ’No label resource in range’ set.

2.7. XRO Extension

The XRO as defined in [RFC5521] is used to exclude specific objects in the path. RSVP-TE allows the exclusion of certain labels ([RFC6001]). In order to fulfill requirement 13 of [RFC7025] Section 3.1, the PCEP’s XRO needs to support a new subobject to enable label exclusion.

The encoding of the XRO Label subobject follows the encoding of the Label ERO subobject defined in [RFC3473] and XRO subobject defined in

[RFC5521]. The XRO Label subobject represent one Label and is defined as follows:

XRO Subobject Type TBA-39: Label Subobject.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X| Type=TBA-39 |    Length     |U|   Reserved  |   C-Type      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Label                             |
|                              ...                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

X (1 bit): as per [RFC5521]. The X-bit indicates whether the exclusion is mandatory or desired. 0 indicates that the resource specified MUST be excluded from the path computed by the PCE. 1 indicates that the resource specified SHOULD be excluded from the path computed by the PCE, but MAY be included subject to PCE policy and the absence of a viable path that meets the other constraints and excludes the resource.

Type (7 bits): The Type of the XRO Label subobject is TBA-39.

Length (8 bits): see [RFC5521], the total length of the subobject in bytes (including the Type and Length fields). The Length is always divisible by 4.

U (1 bit): see [RFC3471] Section 6.1.

C-Type (8 bits): the C-Type of the included Label Object as defined in [RFC3473].

Label: see [RFC3471].

The Label subobject MUST follow a subobject identifying a link, currently an IP address subobject (Type 1 or 2) or an interface ID (type 4) subobject. If an IP address subobject is used, then the given IP address MUST be associated with a link. More than one label subobject MAY follow each link subobject.

Type	Sub-object
3	LABEL

2.8. LSPA Extensions

The LSPA carries the LSP attributes. In the end-to-end recovery context, this also includes the protection state information. A new TLV is defined to fulfil requirement 7 of [RFC7025] Section 3.1 and requirement 3 of [RFC7025] Section 3.2. This TLV contains the information of the PROTECTION object defined by [RFC4872] and can be used as a policy input. The LSPA object MAY carry a PROTECTION-ATTRIBUTE TLV defined as: Type TBA-12: PROTECTION-ATTRIBUTE

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Type                  |  Length                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S|P|N|O|  Reserved | LSP Flags |     Reserved      | Link Flags|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|I|R|   Reserved    | Seg.Flags |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The content is as defined in [RFC4872] Section 14, [RFC4873] Section 6.1.

LSP (protection) Flags or Link flags field can be used by a PCE implementation for routing policy input. The other attributes are only meaningful for a stateful PCE.

This TLV is OPTIONAL and MAY be ignored by the PCE. If ignored by the PCE, it MUST NOT include the TLV in the LSPA of the response. When the TLV is used by the PCE, a LSPA object and the PROTECTION-ATTRIBUTE TLV MUST be included in the response. Fields that were not considered MUST be set to 0.

2.9. NO-PATH Object Extension

The NO-PATH object is used in PCRep messages in response to an unsuccessful path computation request (the PCE could not find a path satisfying the set of constraints). In this scenario, PCE MUST include a NO-PATH object in the PCRep message. The NO-PATH object MAY carry the NO-PATH-VECTOR TLV that specifies more information on the reasons that led to a negative reply. In case of GMPLS networks there could be some additional constraints that led to the failure such as protection mismatch, lack of resources, and so on. Several new flags have been defined in the 32-bit flag field of the NO-PATH-VECTOR TLV but no modifications have been made in the NO-PATH object.

2.9.1. Extensions to NO-PATH-VECTOR TLV

The modified NO-PATH-VECTOR TLV carrying the additional information is as follows:

Bit number TBA-32 - Protection Mismatch (1-bit). Specifies the mismatch of the protection type in the PROTECTION-ATTRIBUTE TLV in the request.

Bit number TBA-33 - No Resource (1-bit). Specifies that the resources are not currently sufficient to provide the path.

Bit number TBA-34 - Granularity not supported (1-bit). Specifies that the PCE is not able to provide a path with the requested granularity.

Bit number TBA-35 - No endpoint label resource (1-bit). Specifies that the PCE is not able to provide a path because of the endpoint label restriction.

Bit number TBA-36 - No endpoint label resource in range (1-bit). Specifies that the PCE is not able to provide a path because of the endpoint label set restriction.

Bit number TBA-37 - No label resource in range (1-bit). Specifies that the PCE is not able to provide a path because of the label set restriction.

3. Additional Error-Types and Error-Values Defined

A PCEP-ERROR object is used to report a PCEP error and is characterized by an Error-Type that specifies the type of error while Error-value that provides additional information about the error. An additional error type and several error values are defined to represent some of the errors related to the newly identified objects related to GMPLS networks. For each PCEP error, an Error-Type and an Error-value are defined. Error-Type 1 to 10 are already defined in [RFC5440]. Additional Error-values are defined for Error-Types 4 and 10. A new Error-Type is defined (value TBA-27).

The Error-Type TBA-27 (path computation failure) is used to reflect constraints not understood by the PCE, for instance when the PCE is not able to understand the generalized bandwidth. If the constraints are understood, but the PCE is unable to find with those constraints, the NO-PATH is to be used.

```
   Error-Type Error-value

       4      Not supported
              object
              value=TBA-14: Bandwidth Object type TBA-2 or TBA-3 not
                            supported
              value=TBA-15: Unsupported endpoint type in
                            END-POINTS Generalized Endpoint
                            object type
              value=TBA-16: Unsupported TLV present in END-POINTS
                            Generalized Endpoint object type
              value=TBA-17: Unsupported granularity in the RP object
                            flags
       10     Reception of
              an invalid
              object
              value=TBA-18: Bad Bandwidth Object type TBA-2(Generalized
                            bandwidth) or TBA-3( Generalized bandwidth
                            of existing TE-LSP for which a
                            reoptimization is requested)
              value=TBA-20: Unsupported LSP Protection Flags in
                            PROTECTION-ATTRIBUTE TLV
              value=TBA-21: Unsupported Secondary LSP Protection Flags
                            in PROTECTION-ATTRIBUTE TLV
              value=TBA-22: Unsupported Link Protection Type in
                            PROTECTION-ATTRIBUTE TLV
              value=TBA-24: LABEL-SET TLV present with 0 bit set but
                            without R bit set in RP
              value=TBA-25: Wrong LABEL-SET
                            TLV present with
                            0 and L bit set
              value=TBA-26: Wrong LABEL-SET with O bit set and wrong
                            format
              value=TBA-42: Missing GMPLS-CAPABILITY TLV
     TBA-27   Path
              computation
              failure
              value=0:      Unassigned
              value=TBA-28: Unacceptable request message
              value=TBA-29: Generalized bandwidth value not supported
              value=TBA-30: Label Set constraint could not be
                            met
              value=TBA-31: Label constraint could not be
                            met
```

4. Manageability Considerations

This section follows the guidance of [RFC6123].

4.1. Control of Function through Configuration and Policy

This document makes no change to the basic operation of PCEP and so the requirements described in [RFC5440] Section 8.1. also apply to this document. In addition to those requirements a PCEP implementation may allow the configuration of the following parameters:

Accepted RG in the RP object.

Default RG to use (overriding the one present in the PCReq)

Accepted BANDWIDTH object type TBA-2 and TBA-3 parameters in request, default mapping to use when not specified in the request

Accepted LOAD-BALANCING object type TBA-4 parameters in request.

Accepted endpoint type and allowed TLVs in object END-POINTS with object type Generalized Endpoint.

Accepted range for label restrictions in label restriction in END-POINTS, or IRO or XRO objects

PROTECTION-ATTRIBUTE TLV acceptance and suppression.

The configuration of the above parameters is applicable to the different sessions as described in [RFC5440] Section 8.1 (by default, per PCEP peer, etc.).

4.2. Information and Data Models

This document makes no change to the basic operation of PCEP and so the requirements described in [RFC5440] Section 8.2. also apply to this document. This document does not introduce any new ERO sub objects, so that the, ERO information model is already covered in [RFC4802].

4.3. Liveness Detection and Monitoring

This document makes no change to the basic operation of PCEP and so there are no changes to the requirements for liveness detection and monitoring set out in [RFC4657] and [RFC5440] Section 8.3.

4.4. Verifying Correct Operation

This document makes no change to the basic operations of PCEP and considerations described in [RFC5440] Section 8.4. New errors defined by this document should satisfy the requirement to log error events.

4.5. Requirements on Other Protocols and Functional Components

No new Requirements on Other Protocols and Functional Components are made by this document. This document does not require ERO object extensions. Any new ERO subobject defined in the TEAS or CCAMP working group can be adopted without modifying the operations defined in this document.

4.6. Impact on Network Operation

This document makes no change to the basic operations of PCEP and considerations described in [RFC5440] Section 8.6. In addition to the limit on the rate of messages sent by a PCEP speaker, a limit MAY be placed on the size of the PCEP messages.

5. IANA Considerations

IANA assigns values to the PCEP objects and TLVs. IANA is requested to make some allocations for the newly defined objects and TLVs defined in this document. Also, IANA is requested to manage the space of flags that are newly added in the TLVs.

5.1. PCEP Objects

As described in Section 2.3, Section 2.4 and Section 2.5.1 new Objects types are defined. IANA is requested to make the following Object-Type allocations from the "PCEP Objects" sub-registry.

```
Object      5
Class
Name        BANDWIDTH
Object-Type TBA-2: Generalized bandwidth
            TBA-3: Generalized bandwidth of an existing TE-LSP for
            which a reoptimization is requested
Reference   This document (Section 2.3)

Object      14
Class
Name        LOAD-BALANCING
Object-Type TBA-4: Generalized Load Balancing

Reference   This document (Section 2.4)
Object      4
Class
Name        END-POINTS
Object-Type TBA-5: Generalized Endpoint
Reference   This document (Section 2.5)
```

5.2. Endpoint type field in Generalized END-POINTS Object

IANA is requested to create a registry to manage the Endpoint Type field of the END-POINTS object, Object Type Generalized Endpoint and manage the code space.

New endpoint type in the Reserved range are assigned by Standards Action [RFC8126]. Each endpoint type should be tracked with the following attributes:

- o Endpoint type
- o Description
- o Defining RFC

New endpoint type in the Experimental range are for experimental use; these will not be registered with IANA and MUST NOT be mentioned by RFCs.

The following values have been defined by this document. (Section 2.5.1, Table 5):

Value	Type	Meaning
0	Point-to-Point	
1	Point-to-Multipoint	New leaves to add
2		Old leaves to remove
3		Old leaves whose path can be modified/reoptimized
4		Old leaves whose path has to be left unchanged
5-244	Unassigned	
245-255	Experimental range	

5.3. New PCEP TLVs

IANA manages the PCEP TLV code point registry (see [RFC5440]). This is maintained as the "PCEP TLV Type Indicators" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry. IANA is requested to do the following allocation. Note: TBA-11 is not used

Value	Meaning	Reference
TBA-6	IPV4-ADDRESS	This document (Section 2.5.2.1)
TBA-7	IPV6-ADDRESS	This document (Section 2.5.2.2)
TBA-8	UNNUMBERED-ENDPOINT	This document (Section 2.5.2.3)
TBA-9	LABEL-REQUEST	This document (Section 2.5.2.4)
TBA-10	LABEL-SET	This document (Section 2.5.2.5)
TBA-12	PROTECTION-ATTRIBUTE	This document (Section 2.8)
TBA-1	GMPLS-CAPABILITY	This document (Section 2.1.2)

5.4. RP Object Flag Field

As described in Section 2.2 new flag are defined in the RP Object Flag IANA is requested to make the following Object-Type allocations from the "RP Object Flag Field" sub-registry.

Bit	Description	Reference
TBA-13	routing granularity (2 bits) (RG)	This document, Section 2.2

5.5. New PCEP Error Codes

As described in Section 3, new PCEP Error-Types and Error-values are defined. IANA is requested to make the following allocation in the "PCEP-ERROR Object Error Types and Values" registry.

Error	name	Reference
Type=4	Not supported object	[RFC5440]
Value=TBA-14:	Bandwidth Object type TBA-2 or TBA-3 not supported	This Document
Value=TBA-15:	Unsupported endpoint type in END-POINTS Generalized Endpoint object type	This Document
Value=TBA-16:	Unsupported TLV present in END-POINTS Generalized Endpoint object type	This Document
Value=TBA-17:	Unsupported granularity in the RP object flags	This Document
Type=10	Reception of an invalid object	[RFC5440]
Value=TBA-18:	Bad Bandwidth Object type TBA-2(Generalized bandwidth) or TBA-3(Generalized bandwidth of existing TE-LSP for which a reoptimization is requested)	This Document
Value=TBA-20:	Unsupported LSP Protection Flags in PROTECTION-ATTRIBUTE TLV	This Document
Value=TBA-21:	Unsupported Secondary LSP Protection Flags in PROTECTION-ATTRIBUTE TLV	This Document
Value=TBA-22:	Unsupported Link Protection Type in PROTECTION-ATTRIBUTE TLV	This Document
Value=TBA-24:	LABEL-SET TLV present with 0 bit set but without R bit set in RP	This Document
Value=TBA-25:	Wrong LABEL-SET TLV present with 0 and L bit set	This Document
Value=TBA-26:	Wrong LABEL-SET with O bit set and wrong format	This Document
Value=TBA-42:	Missing GMPLS-CAPABILITY TLV	This Document
Type=TBA-27	Path computation failure	This Document
Value=0	Unassigned	This Document
Value=TBA-28:	Unacceptable request message	This Document
Value=TBA-29:	Generalized bandwidth value not supported	This Document
Value=TBA-30:	Label Set constraint could not be met	This Document
Value=TBA-31:	Label constraint could not be met	This Document

5.6. New NO-PATH-VECTOR TLV Fields

As described in Section 2.9.1, new NO-PATH-VECTOR TLV Flag Fields have been defined. IANA is requested to do the following allocations in the "NO-PATH-VECTOR TLV Flag Field" sub-registry.

Bit number TBA-32 - Protection Mismatch (1-bit). Specifies the mismatch of the protection type of the PROTECTION-ATTRIBUTE TLV in the request.

Bit number TBA-33 - No Resource (1-bit). Specifies that the resources are not currently sufficient to provide the path.

Bit number TBA-34 - Granularity not supported (1-bit). Specifies that the PCE is not able to provide a path with the requested granularity.

Bit number TBA-35 - No endpoint label resource (1-bit). Specifies that the PCE is not able to provide a path because of the endpoint label restriction.

Bit number TBA-36 - No endpoint label resource in range (1-bit). Specifies that the PCE is not able to provide a path because of the endpoint label set restriction.

Bit number TBA-37 - No label resource in range (1-bit). Specifies that the PCE is not able to provide a path because of the label set restriction.

Bit number TBA-40 - LOAD-BALANCING could not be performed with the bandwidth constraits (1 bit). Specifies that the PCE is not able to provide a path because it could not map the BANDWIDTH into the parameters specified by the LOAD-BALANCING.

5.7. New Subobject for the Include Route Object

The "PCEP Parameters" registry contains a subregistry "IRO Subobjects" with an entry for the Include Route Object (IRO).

IANA is requested to add a further subobject that can be carried in the IRO as follows:

Subobject type		Reference
TBA-38	Label subobject	This Document

5.8. New Subobject for the Exclude Route Object

The "PCEP Parameters" registry contains a subregistry "XRO Subobjects" with an entry for the XRO object (Exclude Route Object).

IANA is requested to add a further subobject that can be carried in the XRO as follows:

Subobject type		Reference
TBA-39	Label subobject	This Document

5.9. New GMPLS-CAPABILITY TLV Flag Field

IANA is requested to create a sub-registry to manage the Flag field of the GMPLS-CAPABILITY TLV within the "Path Computation Element Protocol (PCEP) Numbers" registry.

New bit numbers 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 initial contents of the sub-registry are empty, with all bits marked unassigned

6. Security Considerations

GMPLS controls multiple technologies and types of network elements. The LSPs that are established using GMPLS, whose paths can be computed using the PCEP extensions to support GMPLS described in this document, can carry a high volume of traffic and can be a critical part of a network infrastructure. The PCE can then play a key role in the use of the resources and in determining the physical paths of the LSPs and thus it is important to ensure the identity of PCE and PCC, as well as the communication channel. In many deployments there will be a completely isolated network where an external attack is of very low probability. However, there are other deployment cases in which the PCC-PCE communication can be more exposed and there could be more security considerations. Three main situations in case of an attack in the GMPLS PCE context could happen:

- o PCE Identity theft: A legitimate PCC could request a path for a GMPLS LSP to a malicious PCE, which poses as a legitimate PCE. The answer can make that the LSP traverses some geographical place known to the attacker where confidentiality (sniffing), integrity (traffic modification) or availability (traffic drop) attacks could be performed by use of an attacker-controlled middlebox device. Also, the resulting LSP can omit constraints given in the requests (e.g., excluding certain fibers, avoiding some SRLGs) which could make that the LSP which will be later set-up can look perfectly fine, but will be in a risky situation. Also, the result can lead to the creation of an LSP that does not provide the desired quality and gives less resources than necessary.

- o PCC Identity theft: A malicious PCC, acting as a legitimate PCC, requesting LSP paths to a legitimate PCE can obtain a good knowledge of the physical topology of a critical infrastructure. It could get to know enough details to plan a later physical attack.

- o Message inspection: As in the previous case, knowledge of an infrastructure can be obtained by sniffing PCEP messages.

The security mechanisms can provide authentication and confidentiality for those scenarios where the PCC-PCE communication cannot be completely trusted. [RFC8253] provides origin verification, message integrity and replay protection, and ensures that a third party cannot decipher the contents of a message.

In order to protect against the malicious PCE case the PCC SHOULD have policies in place to accept or not the path provided by the PCE. Those policies can verify if the path follows the provided constraints. In addition, technology specific data plane mechanism can be used (following [RFC5920] Section 5.8) to verify the data plane connectivity and deviation from constraints.

The document [RFC8253] describes the usage of Transport Layer Security (TLS) to enhance PCEP security. The document describes the initiation of the TLS procedures, the TLS handshake mechanisms, the TLS methods for peer authentication, the applicable TLS ciphersuites for data exchange, and the handling of errors in the security checks. PCE and PCC SHOULD use [RFC8253] mechanism to protect against malicious PCC and PCE.

Finally, as mentioned by [RFC7025] the PCEP extensions to support GMPLS should be considered under the same security as current PCE work and this extension will not change the underlying security issues. However, given the critical nature of the network infrastructures under control by GMPLS, the security issues described above should be seriously considered when deploying a GMPLS-PCE based control plane for such networks. For more information on the security considerations on a GMPLS control plane, not only related to PCE/PCEP, [RFC5920] provides an overview of security vulnerabilities of a GMPLS control plane.

7. Contributing Authors

Elie Sfeir
Coriant
St Martin Strasse 76
Munich, 81541
Germany

Email: elie.sfeir@coriant.com

Franz Rambach
Nockherstrasse 2-4,
Munich 81541
Germany

Phone: +49 178 8855738
Email: franz.rambach@cgi.com

Francisco Javier Jimenez Chico
Telefonica Investigacion y Desarrollo
C/ Emilio Vargas 6
Madrid, 28043
Spain

Phone: +34 91 3379037
Email: fjjc@tid.es

Huawei Technologies

Suresh BR
Shenzhen
China
Email: sureshbr@huawei.com

Young Lee
1700 Alma Drive, Suite 100
Plano, TX 75075
USA

Phone: (972) 509-5599 (x2240)
Email: ylee@huawei.com

SenthilKumar S
Shenzhen
China
Email: senthilkumars@huawei.com

Jun Sun
Shenzhen
China
Email: johnsun@huawei.com

CTTC - Centre Tecnologic de Telecomunicacions de Catalunya

Ramon Casellas
PMT Ed B4 Av. Carl Friedrich Gauss 7
08860 Castelldefels (Barcelona)
Spain
Phone: (34) 936452916
Email: ramon.casellas@cttc.es

8. Acknowledgments

The research of Ramon Casellas, Francisco Javier Jimenez Chico, Oscar Gonzalez de Dios, Cyril Margaria, and Franz Rambach leading to these results has received funding from the European Community's Seventh Framework Program FP7/2007-2013 under grant agreement no 247674 and no 317999.

The authors would like to thank Julien Meuric, Lyndon Ong, Giada Lander, Jonathan Hardwick, Diego Lopez, David Sinicrope, Vincent Roca, Dhruv Dhody, Adrian Farrel and Tianran Zhou for their review and useful comments to the document.

Thanks to Alisa Cooper, Benjamin Kaduk, Elwun-davies, Martin Vigoureux, Roman Danyliw, and Suresh Krishnan for the IESG comments

9. References

9.1. Normative References

[G.709-v3]
ITU-T, "Interfaces for the optical transport network, Recommendation G.709/Y.1331", June 2016, <https://www.itu.int/rec/T-REC-G.709-201606-I/en>.

[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>.

[RFC2210] Wroclawski, J., "The Use of RSVP with IETF Integrated Services", RFC 2210, DOI 10.17487/RFC2210, September 1997, <https://www.rfc-editor.org/info/rfc2210>.

[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>.

[RFC3471] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description", RFC 3471, DOI 10.17487/RFC3471, January 2003, <https://www.rfc-editor.org/info/rfc3471>.

[RFC3473] Berger, L., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, DOI 10.17487/RFC3473, January 2003, <https://www.rfc-editor.org/info/rfc3473>.

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, DOI 10.17487/RFC3477, January 2003, <https://www.rfc-editor.org/info/rfc3477>.

[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>.

[RFC4003] Berger, L., "GMPLS Signaling Procedure for Egress Control", RFC 4003, DOI 10.17487/RFC4003, February 2005, <https://www.rfc-editor.org/info/rfc4003>.

[RFC4328] Papadimitriou, D., Ed., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Extensions for G.709 Optical Transport Networks Control", RFC 4328, DOI 10.17487/RFC4328, January 2006, <https://www.rfc-editor.org/info/rfc4328>.

[RFC4606] Mannie, E. and D. Papadimitriou, "Generalized Multi-Protocol Label Switching (GMPLS) Extensions for Synchronous Optical Network (SONET) and Synchronous Digital Hierarchy (SDH) Control", RFC 4606, DOI 10.17487/RFC4606, August 2006, <https://www.rfc-editor.org/info/rfc4606>.

[RFC4802] Nadeau, T., Ed. and A. Farrel, Ed., "Generalized Multiprotocol Label Switching (GMPLS) Traffic Engineering Management Information Base", RFC 4802, DOI 10.17487/RFC4802, February 2007, <https://www.rfc-editor.org/info/rfc4802>.

[RFC4872] Lang, J., Ed., Rekhter, Y., Ed., and D. Papadimitriou, Ed., "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, DOI 10.17487/RFC4872, May 2007, <https://www.rfc-editor.org/info/rfc4872>.

[RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel, "GMPLS Segment Recovery", RFC 4873, DOI 10.17487/RFC4873, May 2007, <https://www.rfc-editor.org/info/rfc4873>.

[RFC5088] Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R. Zhang, "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5088, DOI 10.17487/RFC5088, January 2008, <https://www.rfc-editor.org/info/rfc5088>.

[RFC5089] Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R. Zhang, "IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5089, DOI 10.17487/RFC5089, January 2008, <https://www.rfc-editor.org/info/rfc5089>.

[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>.

[RFC5520] Bradford, R., Ed., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", RFC 5520, DOI 10.17487/RFC5520, April 2009, <https://www.rfc-editor.org/info/rfc5520>.

[RFC5521] Oki, E., Takeda, T., and A. Farrel, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions", RFC 5521, DOI 10.17487/RFC5521, April 2009, <https://www.rfc-editor.org/info/rfc5521>.

[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>.

[RFC6001] Papadimitriou, D., Vigoureux, M., Shiomoto, K., Brungard, D., and JL. Le Roux, "Generalized MPLS (GMPLS) Protocol Extensions for Multi-Layer and Multi-Region Networks (MLN/MRN)", RFC 6001, DOI 10.17487/RFC6001, October 2010, <https://www.rfc-editor.org/info/rfc6001>.

[RFC6003] Papadimitriou, D., "Ethernet Traffic Parameters", RFC 6003, DOI 10.17487/RFC6003, October 2010, <https://www.rfc-editor.org/info/rfc6003>.

[RFC6205] Otani, T., Ed. and D. Li, Ed., "Generalized Labels for Lambda-Switch-Capable (LSC) Label Switching Routers", RFC 6205, DOI 10.17487/RFC6205, March 2011, <https://www.rfc-editor.org/info/rfc6205>.

[RFC6387] Takacs, A., Berger, L., Caviglia, D., Fedyk, D., and J. Meuric, "GMPLS Asymmetric Bandwidth Bidirectional Label Switched Paths (LSPs)", RFC 6387, DOI 10.17487/RFC6387, September 2011, <https://www.rfc-editor.org/info/rfc6387>.

[RFC7139] Zhang, F., Ed., Zhang, G., Belotti, S., Ceccarelli, D., and K. Pithewan, "GMPLS Signaling Extensions for Control of Evolving G.709 Optical Transport Networks", RFC 7139, DOI 10.17487/RFC7139, March 2014, <https://www.rfc-editor.org/info/rfc7139>.

[RFC7570] Margaria, C., Ed., Martinelli, G., Balls, S., and B. Wright, "Label Switched Path (LSP) Attribute in the Explicit Route Object (ERO)", RFC 7570, DOI 10.17487/RFC7570, July 2015, <https://www.rfc-editor.org/info/rfc7570>.

[RFC7792] Zhang, F., Zhang, X., Farrel, A., Gonzalez de Dios, O., and D. Ceccarelli, "RSVP-TE Signaling Extensions in Support of Flexi-Grid Dense Wavelength Division Multiplexing (DWDM) Networks", RFC 7792, DOI 10.17487/RFC7792, March 2016, <https://www.rfc-editor.org/info/rfc7792>.

[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>.

[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>.

[RFC8282] Oki, E., Takeda, T., Farrel, A., and F. Zhang, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 8282, DOI 10.17487/RFC8282, December 2017, <https://www.rfc-editor.org/info/rfc8282>.

[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>.

9.2. 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>.

[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>.

[RFC5920] Fang, L., Ed., "Security Framework for MPLS and GMPLS Networks", RFC 5920, DOI 10.17487/RFC5920, July 2010, <https://www.rfc-editor.org/info/rfc5920>.

[RFC6123] Farrel, A., "Inclusion of Manageability Sections in Path Computation Element (PCE) Working Group Drafts", RFC 6123, DOI 10.17487/RFC6123, February 2011, <https://www.rfc-editor.org/info/rfc6123>.

[RFC6163] Lee, Y., Ed., Bernstein, G., Ed., and W. Imajuku, "Framework for GMPLS and Path Computation Element (PCE) Control of Wavelength Switched Optical Networks (WSONs)", RFC 6163, DOI 10.17487/RFC6163, April 2011, <https://www.rfc-editor.org/info/rfc6163>.

[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>.

[RFC7449] Lee, Y., Ed., Bernstein, G., Ed., Martensson, J., Takeda, T., Tsuritani, T., and O. Gonzalez de Dios, "Path Computation Element Communication Protocol (PCEP) Requirements for Wavelength Switched Optical Network (WSON) Routing and Wavelength Assignment", RFC 7449, DOI 10.17487/RFC7449, February 2015, <https://www.rfc-editor.org/info/rfc7449>.

Appendix A. LOAD-BALANCING Usage for SDH Virtual Concatenation

For example a request for one co-signaled n x VC-4 TE-LSP will not use the LOAD-BALANCING. In case the VC-4 components can use different paths, the BANDWIDTH with object type TBA-2 will contain a traffic specification indicating the complete n x VC-4 traffic specification and the LOAD-BALANCING the minimum co-signaled VC-4. For an SDH network, a request to have a TE-LSP group with 10 VC-4 containers, each path using at minimum 2 x VC-4 containers, can be represented with a BANDWIDTH object with OT=TBA-2, Bw Spec Type set to 4, the content of the Generalized Bandwidth is ST=6, RCC=0, NCC=0, NVC=10, MT=1. The LOAD-BALANCING, OT=TBA-4 with Bw Spec Type set to 4, Max-LSP=5, Min Bandwidth Spec is (ST=6, RCC=0, NCC=0, NVC=2, MT=1). The PCE can respond with a response with maximum 5 paths, each of them having a BANDWIDTH OT=TBA-2 and Generalized Bandwidth matching the Min Bandwidth Spec from the LOAD-BALANCING object of the corresponding request.

Authors' Addresses

Cyril Margaria (editor)
Juniper

Email: cmargaria@juniper.net

Oscar Gonzalez de Dios (editor)
Telefonica Investigacion y Desarrollo
C/ Ronda de la Comunicacion
Madrid 28050
Spain

Phone: +34 91 4833441
Email: oscar.gonzalezdedios@telefonica.com

Fatai Zhang (editor)
Huawei Technologies
F3-5-B R&D Center, Huawei Base
Bantian, Longgang District
Shenzhen 518129
P.R.China

Email: zhangfatai@huawei.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 8, 2018

E. Crabbe
Individual Contributor
I. Minei
Google, Inc.
S. Sivabalan
Cisco Systems, Inc.
R. Varga
Pantheon Technologies SRO

October 5, 2017

PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model
draft-ietf-pce-pce-initiated-lsp-11

Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

The extensions for stateful PCE provide active control of Multiprotocol Label Switching (MPLS) Traffic Engineering Label Switched Paths (TE LSP) via PCEP, for a model where the PCC delegates control over one or more locally configured LSPs to the PCE. This document describes the creation and deletion of PCE-initiated LSPs 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 [RFC2119].

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 http://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 April 8, 2018.

Copyright Notice

Copyright (c) 2017 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 (http://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. Architectural Overview 4
  3.1. Motivation 4
  3.2. Operation Overview 5
4. Support of PCE-initiated LSPs 6
  4.1. STATEFUL-PCE-CAPABILITY TLV 6
5. PCE-initiated LSP Instantiation and Deletion 7
  5.1. The LSP Initiate Request 7
  5.2. The R flag in the SRP Object 8
  5.3. LSP Instantiation 9
    5.3.1. The Create Flag 11
    5.3.2. The SPEAKER-ENTITY-ID TLV 11
  5.4. LSP Deletion 12
6. LSP Delegation and Cleanup 12
7. LSP State Synchronization 13
8. Implementation Status 14
9. IANA Considerations 14
  9.1. PCEP Messages 14
  9.2. LSP Object 15
  9.3. SRP object 15
  9.4. STATEFUL-PCE-CAPABILITY TLV 15
  9.5. PCEP-Error Object 15
10. Security Considerations 16
  10.1. Malicious PCE 16
  10.2. Malicious PCC 17
11. Acknowledgements 17
12. References 17
  12.1. Normative References 17

12.2. Informative References 18
Authors' Addresses 18

1. Introduction

[RFC5440] describes the Path Computation Element Communication Protocol (PCEP). PCEP defines the communication between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between PCE and PCE, enabling computation of Multiprotocol Label Switching (MPLS) for Traffic Engineering Label Switched Path (TE LSP) characteristics.

[RFC8231] specifies a set of extensions to PCEP to enable stateful control of TE LSPs between and across PCEP sessions in compliance with [RFC4657]. It includes

- o mechanisms to effect LSP state synchronization between PCCs and PCEs
- o delegation of control of LSPs to PCEs
- o PCE control of timing and sequence of path computations within and across PCEP sessions

It focuses on a model where LSPs are configured on the PCC and control over them is delegated to the PCE.

This document describes the setup, maintenance and teardown of PCE-initiated LSPs under the stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic network that is centrally controlled and deployed.

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.

This document uses the following terms defined in [RFC8231]: Redelegation Timeout Interval, State Timeout Interval, LSP State Report, LSP Update Request.

The following terms are defined in this document:

PCE-initiated LSP: LSP that is instantiated as a result of a request from the PCE.

The message formats in this document are specified using Routing Backus-Naur Form (RBNF) encoding as specified in [RFC5511].

3. Architectural Overview

3.1. Motivation

[RFC8231] provides active control over LSPs that are locally configured on the PCC. This model relies on the Label Edge Router (LER) taking an active role in delegating locally configured LSPs to the PCE, and is well suited in environments where the LSP placement is fairly static. However, in environments where the LSP placement needs to change in response to application demands, it is useful to support dynamic creation and tear down of LSPs. The ability for a PCE to trigger the creation of LSPs on demand can be seamlessly integrated into a controller-based network architecture, where intelligence in the controller can determine when and where to set up paths.

A possible use case is a software-defined network, where applications request network resources and paths from the network infrastructure. For example, an application can request a path with certain constraints between two LSRs by contacting the PCE. The PCE can compute a path satisfying the constraints, and instruct the head end LSR to instantiate and signal it. When the path is no longer required by the application, the PCE can request its teardown.

Another use case is dynamically adjusting aggregate bandwidth between two points in the network using multiple LSPs. This functionality is very similar to auto-bandwidth, but allows for providing the desired capacity through multiple LSPs. This approach overcomes two of the limitations auto-bandwidth can experience: 1) growing the capacity between the endpoints beyond the capacity of individual links in the path and 2) achieving good bin-packing through use of several small LSPs instead of a single large one. The number of LSPs varies based on the demand, and LSPs are created and deleted dynamically to satisfy the bandwidth requirements.

Another use case is demand engineering, where a PCE with visibility into both the network state and the demand matrix can anticipate and optimize how traffic is distributed across the infrastructure. Such optimizations may require creating new paths across the infrastructure.

3.2. Operation Overview

This document defines the new I flag in the STATEFUL-PCE-CAPABILITY TLV to indicate that the sender supports PCE-initiated LSPs (see details in Section 4.1). A PCC or PCE sets this flag in the Open message during the PCEP Initialization Phase to indicate that it supports the procedures of this document.

This document defines a new PCEP message, the LSP Initiate Request (PCInitiate) message, which a PCE can send to a PCC to request the initiaton or deletion of an LSP. The decision when to instantiate or delete a PCE-initiated LSP is out of the scope of this document.

The PCE sends a PCInitiate message to the PCC to request the initiation of an LSP. The PCC creates the LSP using the attributes communicated by the PCE and local values for any unspecified parameters. The PCC generates an LSP State Report (PCRpt) for the LSP, carrying a newly assigned PLSP-ID for the LSP and delegating the LSP to the PCE via the Delegate flag in the LSP object.

The PCE can update the attributes of the LSP by sending subsequent PCUpd messages. Subsequent LSP State Report (PCRpt) and LSP Update Request (PCUpd) messages that the PCC and PCE, respectively, send for the LSP will carry the PCC-assigned PLSP-ID, which uniquely identifies the LSP. See details in Section 5.3.

The PCE sends a PCInitiate message to the PCC to request the deletion of an LSP. To indicate a delete operation, this document defines the new R flag in the SRP object in the PCInitiate message, as described in Section 5.2. As a result of the deletion request, the PCC removes the LSP and sends a PCRpt for the removed state. See details in Section 5.4.

Figure 1 illustrates these message exchanges.

```
         +-+-+                            +-+-+
         |PCC|                            |PCE|
         +-+-+                            +-+-+
           |                                |
           |<--PCInitiate-------------------| (Initiate LSP)
           |                                |
           |---PCRpt, PLSP_ID=1, D=1------->| (Confirm initiation)
           |            .                   |
           |            .                   |
           |                                |
           |<--PCUpd, PLSP_ID=1-------------| (Update LSP)
           |                                |
           |---PCRpt, PLSP_ID=1, D=1------->| (Confirm update)
           |            .                   |
           |            .                   |
           |                                |
           |<--PCInitiate, PLSP_ID=1, R=1---| (Delete LSP)
           |                                |
           |---PCRpt, PLSP_ID=1, R=1------->| (Confirm delete)
```

Figure 1: PCE-Initiated LSP lifecycle

4. Support of PCE-initiated LSPs

A PCEP speaker indicates its ability to support PCE-initiated LSPs during the PCEP Initialization phase, as follows. When the PCEP session is created, it sends an Open message with an OPEN object that contains the STATEFUL-PCE-CAPABILITY TLV, defined in [RFC8231]. A new flag, the I (LSP-INSTANTIATION-CAPABILITY) flag, is introduced to this TLV to indicate support for instantiation of PCE-initiated LSPs. A PCE can initiate LSPs only for PCCs that advertised this capability. A PCC will follow the procedures described in this document only on sessions where the PCE advertised the I flag.

4.1. STATEFUL-PCE-CAPABILITY TLV

The format of the STATEFUL-PCE-CAPABILITY TLV is defined in [RFC8231] and included here for easy reference with the addition of the new I flag.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Type            |             Length=4          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Flags                                          |I|S|U|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: STATEFUL-PCE-CAPABILITY TLV format

A new flag is defined to indicate the sender’s support for LSP instantiation by a PCE:

I (LSP-INSTANTIATION-CAPABILITY - 1 bit): If set to 1 by a PCC, the I Flag indicates that the PCC allows instantiation of an LSP by a PCE. If set to 1 by a PCE, the I flag indicates that the PCE supports instantiating LSPs. The LSP-INSTANTIATION-CAPABILITY flag must be set by both PCC and PCE in order to enable PCE-initiated LSP instantiation.

5. PCE-initiated LSP Instantiation and Deletion

To initiate an LSP, a PCE sends a PCInitiate message to a PCC. The message format, objects and TLVs are discussed separately below for the creation and the deletion cases.

5.1. The LSP Initiate Request

An LSP Initiate Request (PCInitiate) message is a PCEP message sent by a PCE to a PCC to trigger LSP instantiation or deletion. The Message-Type field of the PCEP common header for the PCInitiate message is set to 12. The PCInitiate message MUST include the SRP and the LSP objects, and MAY contain other objects, as discussed later in this section.

The format of a 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-instantiation> ::= <SRP>
                                      <LSP>
                                      [<END-POINTS>]
                                      <ERO>
                                      [<attribute-list>]

<PCE-initiated-lsp-deletion> ::= <SRP>
                                 <LSP>
```

Where:

<attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

The LSP object is defined in [RFC8231]. The END-POINTS and ERO objects are defined in [RFC5440].

The SRP object is defined in [RFC8231]. The SRP Object contains an SRP-ID-number which is unique within a PCEP session. The PCE increments the last-used SRP-ID-number before it sends each PCInitiate message. The PCC MUST echo the value of the SRP-ID-number in PCErr and PCRpt messages that it sends as a result of the PCInitiate to allow the PCE to correlate them with the corresponding PCInitiate message.

5.2. The R flag in the SRP Object

The format of the SRP object is is defined in [RFC8231] and included here for easy reference with the addition of the new R flag.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Flags                              |R|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       SRP-ID-number                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                      Optional TLVs                          //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: The SRP Object format

A new flag is defined to indicate a delete operation initiated by the PCE:

R (LSP-REMOVE - 1 bit): If set to 0, it indicates a request to create an LSP. If set to 1, it indicates a request to remove an LSP.

5.3. LSP Instantiation

The LSP is instantiated by sending a PCInitiate message. The LSP is set up using RSVP-TE. Extensions for other setup methods are outside the scope of this draft.

The PCInitiate message, when used to instantiate an LSP, MUST contain an LSP object with the reserved PLSP-ID 0. The LSP Object MUST include the SYMBOLIC-PATH-NAME TLV, which is used to correlate between the PCC-assigned PLSP-ID and the LSP.

The PCInitiate message, when used to instantiate an LSP, MUST contain an Explicit Route Object (ERO) for the LSP.

For an instantiation request of an RSVP-signaled LSP, the destination address may be needed. The PCC MAY determine it from a provided object (e.g., ERO) or a local decision. Alternatively, the END-POINTS object MAY be included to explicitly convey the destination addresses to be used in the RSVP-TE signaling. The source address MUST either be specified or left for the PCC to choose by setting it to "0.0.0.0" (if the destination is an IPv4 address) or "::" (if the destination is an IPv6 address).

The PCE MAY include various attributes as per [RFC5440]. The PCC MUST use these values in the LSP instantiation, and local values for unspecified parameters. After the LSP setup, the PCC MUST send a

PCRpt to the PCE, reflecting these values. The SRP object in the PCRpt message MUST echo the value of the PCInitiate message that triggered the setup. LSPs that were instantiated as a result of a PCInitiate message MUST have the Create flag (Section 5.3.1) set in the LSP object.

If the PCC receives a PCInitiate message with a non-zero PLSP-ID and the R flag in the SRP object set to zero, then it MUST send a PCErr message with Error-type=19 (Invalid Operation) and Error-value=8 (Non-zero PLSP-ID in the PCInitiate message).

If the PCC receives a PCInitiate message without an ERO and the R flag in the SRP object set to zero, then it MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=9 (ERO Object missing).

If the PCC receives a PCInitiate message without a SYMBOLIC-PATH-NAME TLV, then it MUST send a PCErr message with Error-type=10 (Invalid object) and Error-value=8 (SYMBOLIC-PATH-NAME TLV missing).

The PCE MUST NOT provide a symbolic path name that conflicts with the symbolic path name of any existing LSP in the PCC. (Existing LSPs may be either statically configured, or initiated by another PCE). If there is a conflict with the symbolic path name of an existing LSP, the PCC MUST send a PCErr message with Error-type=23 (Bad Parameter value) and Error-value=1 (SYMBOLIC-PATH-NAME in use). The only exception to this rule is for LSPs for which the State Timeout Interval timer is running (see Section 6).

If the PCC determines that the LSP parameters proposed in the PCInitiate message are unacceptable, it MUST send a PCErr message with Error-type=24 (PCE instantiation error) and Error-value=1 (Unacceptable instantiation parameters). If the PCC encounters an internal error during the processing of the PCInitiate message, it MUST send a PCErr message with Error-type=24 (PCE instantiation error) and Error-value=2 (Internal error).

A PCC MUST relay to the PCE errors it encounters in the setup of PCE-initiated LSP by sending a PCErr message with Error-type=24 (PCE instantiation error) and Error-value=3 (Signaling error). The PCErr message MUST echo the SRP-ID-number of the PCInitiate message. The PCEP-ERROR object SHOULD include the RSVP_ERROR_SPEC TLV (if an RSVP ERROR_SPEC object was returned to the PCC by a downstream node). After the LSP is set up, errors in RSVP signaling are reported in PCRpt messages, as described in [RFC8231].

On successful completion of the LSP instantiation, the PCC MUST send a PCRpt message. The LSP object message MUST contain a non-zero

PLSP-ID that uniquely identifies the LSP within this PCC, and MUST have the Create flag (Section 5.3.1) and Delegate flag set. The SRP object MUST contain an SRP-ID-number that echoes the value from the PCInitiate message that triggered the setup. The PCRpt MUST include the attributes that the PCC used to instantiate the LSP.

A PCC SHOULD be able to place a limit on either the number of LSPs or the percentage of resources that are allocated to honor PCE-initiated LSP requests. As soon as that limit is reached, the PCC MUST send a PCErr message with Error-type=19 (Invalid Operation) and Error-value=6 (PCE-initiated LSP limit reached) and is free to drop any incoming PCInitiate messages without additional processing.

Similarly, the PCE SHOULD be able to place a limit on either the number of PCInitiate messages pending for a particular PCC, or on the time it waits for a response (positive or negative) to a PCInitiate message from a PCC and MAY take further action (such as closing the session or removing all its LSPs) if this limit is reached.

5.3.1. The Create Flag

The LSP object is defined in [RFC8231] and included here for easy reference with the addition of the new C flag.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                PLSP-ID                |Flags  |C|  O  |A|R|S|D|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        TLVs                                  //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: The LSP Object format

A new flag, the Create (C) flag is introduced. On a PCRpt message, the C Flag set to 1 indicates that this LSP was created via a PCInitiate message. The C Flag MUST be set to 1 on each PCRpt message for the duration of existence of the LSP. The Create flag allows PCEs to be aware of which LSPs were PCE-initiated (a state that would otherwise only be known by the PCC and the PCE that initiated them).

5.3.2. The SPEAKER-ENTITY-ID TLV

The optional SPEAKER-ENTITY-ID TLV defined in [RFC8232] MAY be included in the LSP object in a PCRpt message, as an optional TLV for LSPs for which the C flag is 1. The SPEAKER-ENTITY-ID TLV identifies

the PCE which initiated the creation of the LSP on all PCEP sessions, a state that would otherwise only be known by the PCC and the PCE that initiated the LSP. If the TLV appears in a PCRpt for an LSP for which the C flag is 0, the LSP MUST be ignored and the PCE MUST send a PCErr message with Error-type=23 ("Bad parameter value") and Error-value=2 ("Speaker identity included for an LSP that is not PCE-initiated").

5.4. LSP Deletion

A PCE can initiate the removal of a PCE-initiated LSP by sending a PCInitiate message with an LSP object carrying the PLSP-ID of the LSP to be removed and an SRP object with the R flag set (see Section 5.2). A PLSP-ID of zero removes all LSPs with the C flag set to 1 (in their LSP object) that are delegated to the PCE.

If the PLSP-ID is unknown, the PCC MUST send a PCErr message with Error-type=19 ("Invalid operation") and Error-value=3 ("Unknown PLSP-ID") ([RFC8231]).

If the PLSP-ID specified in the PCInitiate message is not delegated to the PCE, the PCC MUST send a PCErr message with Error-type=19 ("Invalid operation") and Error-value=1 ("LSP is not delegated") ([RFC8231]).

If the PLSP-ID specified in the PCInitiate message was not created by a PCE, the PCC MUST send a PCErr message with Error-type=19 ("Invalid operation") and Error-value=9 ("LSP is not PCE-initiated").

Following the removal of the LSP, the PCC MUST send a PCRpt as described in [RFC8231]. The SRP object in the PCRpt MUST include the SRP-ID-number from the PCInitiate message that triggered the removal. The R flag in the SRP object MUST be set.

6. LSP Delegation and Cleanup

The PCC MUST delegate PCE-initiated LSPs to the PCE upon instantiation. The PCC MUST set the delegation bit to 1 in the PCRpt that includes the assigned PLSP-ID.

The PCC MUST NOT revoke the delegation for a PCE-initiated LSP on an active PCEP session. Therefore, all PCRpt messages from the PCC to the PCE that owns the delegation MUST have the delegation bit set to 1. If the PCE that owns the delegation receives a PCRpt message with the delegation bit set to 0 then it MUST send a PCErr message with Error-type=19 ("Invalid Operation") and Error-value=7 ("Delegation for PCE-initiated LSP cannot be revoked"). The PCE MAY further react by closing the session.

Control over a PCE-initiated LSP can revert to the PCC in two ways. A PCE MAY return a delegation to the PCC to allow for LSP transfer between PCEs. Alternatively, the PCC gains control of an LSP if the PCEP session that it was delegated on fails and the Redelegation Timeout Interval timer expires. In both cases, the LSP becomes an orphan until the expiration of the State Timeout Interval timer ([RFC8231]).

The PCC MAY attempt to redelegate an orphaned LSP by following the procedures of [RFC8231]. Alternatively, if the orphaned LSP was PCE-initiated, then a PCE MAY obtain control over it, as follows.

A PCE (either the original or one of its backups) sends a PCInitiate message, including just the SRP and LSP objects, and carrying the PLSP-ID of the LSP it wants to take control of. If the PCC receives a PCInitiate message with a PLSP-ID pointing to an orphaned PCE-initiated LSP, then it MUST redelegate that LSP to the PCE. Any other non-zero PLSP-ID MUST result in the generation of a PCErr message using the rules described in Section 5.4. The State Timeout Interval timer for the LSP is stopped upon the redelegation. After obtaining control of the LSP, the PCE may remove it using the procedures described in this document.

The State Timeout Interval timer ensures that a PCE crash does not result in automatic and immediate disruption for the services using PCE-initiated LSPs. PCE-initiated LSPs are not removed immediately upon PCE failure. Instead, they are cleaned up on the expiration of this timer. This allows for network cleanup without manual intervention. The PCC MUST support removal of PCE-initiated LSPs as one of the behaviors applied on expiration of the State Timeout Interval timer. The behavior MUST be picked based on local policy, and can result either in LSP removal, or in reverting to operator-defined default parameters.

7. LSP State Synchronization

LSP State Synchronization procedures are described in section 5.4 of [RFC8231]. During State Synchronization, a PCC reports the state of its LSPs to the PCE using PCRpt messages, setting the SYNC flag in the LSP Object. For PCE-initiated LSPs, the PCC MUST also set the Create Flag in the LSP Object and MAY include the SPEAKER-ENTITY-ID TLV identifying the PCE that requested the LSP creation. At the end of state synchronization, the PCE SHOULD send a PCInitiate message to initiate any missing LSPs and/or remove any LSPs that are not wanted. Under some circumstances, depending on the deployment, it might be preferable for a PCE not to send this PCInitiate immediately, or at all. For example, the PCC may be a slow device, or the operator might prefer not to disrupt active flows.

8. Implementation Status

This section to be removed by the RFC editor.

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 RFC 7942, "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".

Two vendors are implementing the extensions described in this draft and have included the functionality in releases that will be shipping in the near future. An additional entity is working on implementing these extensions in the scope of research projects.

9. IANA Considerations

This document requests IANA actions to allocate code points for the protocol elements defined in this document.

9.1. PCEP Messages

IANA is requested to confirm the early allocation of the following new message type within the "PCEP Messages" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:

Value	Meaning	Reference
12	LSP Initiate Request	This document

Note to IANA: The early allocation was done for a message called "Initiate". This name has changed to "LSP Initiate Request" as above.

9.2. LSP Object

[RFC8231] defines the LSP Object and requests that IANA creates a registry to manage the value of the LSP Object's Flag field. IANA is requested to allocate a new bit in the LSP Object Flag Field registry, as follows:

Bit	Description	Reference
4	Create	This document

9.3. SRP object

This document requests that a new sub-registry, named "SRP Object Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the SRP object. 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), description and defining RFC.

The following values are defined in this document:

Bit	Description	Reference
31	LSP-Remove	This document

9.4. STATEFUL-PCE-CAPABILITY TLV

[RFC8231] defines the STATEFUL-PCE-CAPABILITY TLV and requests that IANA creates a registry to manage the value of the STATEFUL-PCE-CAPABILITY TLV's Flag field. IANA is requested to allocate a new bit in the STATEFUL-PCE-CAPABILITY TLV Flag Field registry, as follows:

Bit	Description	Reference
29	I (LSP-INSTANTIATION-CAPABILITY)	This document

9.5. PCEP-Error Object

IANA is requested to confirm the early allocation of the following new error types and error values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:

```
    Error-Type  Meaning
       10       Invalid Object

                 Error-value=8:  SYMBOLIC-PATH-NAME TLV missing
       19       Invalid operation

                 Error-value=6:  PCE-initiated LSP limit reached
                 Error-value=7:  Delegation for PCE-initiated LSP cannot
                                 be revoked
                 Error-value=8:  Non-zero PLSP-ID in PCInitiate message
                 Error-value=9:  LSP is not PCE-initiated
                 Error-value=10: PCE-initiated operation-frequency limit
                                 reached
       23       Bad parameter value

                 Error-value=1:  SYMBOLIC-PATH-NAME in use
                 Error-value=2:  Speaker identity included for an LSP
                                 that is not PCE-initiated
       24       LSP instantiation error

                 Error-value=1:  Unacceptable instantiation parameters
                 Error-value=2:  Internal error
                 Error-value=3:  Signaling error
```

10. Security Considerations

The security considerations described in [RFC8231] apply to the extensions described in this document. Additional considerations related to a malicious PCE are introduced.

10.1. Malicious PCE

The LSP instantiation mechanism described in this document allows a PCE to generate state on the PCC and throughout the network. As a result, it introduces a new attack vector: an attacker may flood the PCC with LSP instantiation requests and consume network and LSR resources, either by spoofing messages or by compromising the PCE itself.

A PCC can protect itself from such an attack by imposing a limit on either the number of LSPs or the percentage of resources that are allocated to honor PCE-initiated LSP requests. As soon as that limit is reached, the PCC MUST send a PCErr message with Error-type=19 ("Invalid Operation") and Error-value=6 ("PCE-initiated LSP limit reached") and is free to drop any incoming PCInitiate messages for LSP instantiation without additional processing.

Rapid flaps triggered by the PCE can also be an attack vector. A PCC can protect itself from such an attack by imposing a limit on the number of flaps per unit of time that it allows a PCE to generate. As soon as that limit is reached, a PCC MUST send a PCErr message with Error-type=19 ("Invalid Operation") and Error-value=10 ("PCE-initiated operation frequency reached") and is free to treat the session as having reached the limit in terms of resources allocated to honor PCE-initiated LSP requests, either permanently or for a locally-defined cool-off period.

10.2. Malicious PCC

The LSP instantiation mechanism described in this document requires the PCE to keep state for LSPs that it instantiates and relies on the PCC responding (with either a state report or an error message) to requests for LSP instantiation. A malicious PCC or one that reached the limit of the number of PCE-initiated LSPs, can ignore PCE requests and consume PCE resources. A PCE can protect itself by imposing a limit on the number of requests pending, or by setting a timeout and it MAY take further action such as closing the session or removing all the LSPs it initiated.

11. Acknowledgements

We would like to thank Jan Medved, Ambrose Kwong, Ramon Casellas, Cyril Margaria, Dhruv Dhody, Raveendra Trovi and Jon Hardwick for their contributions to this document.

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>.

[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>.

[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>.

12.2. 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>.

[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>.

[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>.

Authors' Addresses

Edward Crabbe
Individual Contributor

Email: edward.crabbe@gmail.com

Ina Minei
Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: inaminei@google.com

Siva Sivabalan
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134
US

Email: msiva@cisco.com

Robert Varga
Pantheon Technologies SRO
Mlynske Nivy 56
Bratislava 821 05
Slovakia

Email: robert.varga@pantheon.tech

PCE Working Group
Internet-Draft
Intended status: Experimental
Expires: July 11, 2014

D. Dhody
U. Palle
Huawei Technologies
R. Casellas
CTTC

January 7, 2014

Standard Representation Of Domain-Sequence
draft-ietf-pce-pcep-domain-sequence-04

Abstract

The ability to compute shortest constrained Traffic Engineering Label Switched Paths (TE LSPs) in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks across multiple domains has been identified as a key requirement. In this context, a domain is a collection of network elements within a common sphere of address management or path computational responsibility such as an Interior Gateway Protocol (IGP) area or an Autonomous Systems (AS). This document specifies a standard representation and encoding of a Domain-Sequence, which is defined as an ordered sequence of domains traversed to reach the destination domain to be used by Path Computation Elements (PCEs) to compute inter-domain shortest constrained paths across a predetermined sequence of domains . This document also defines new subobjects to be used to encode domain identifiers.

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 http://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 July 11, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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 4
2. Terminology 4
3. Detail Description 5
3.1. Domains 5
3.2. Domain-Sequence 5
3.3. Standard Representation 6
3.4. Include Route Object (IRO) 7
3.4.1. Subobjects 7
3.4.2. Option (A): New IRO Object Type 9
3.4.2.1. Handling of the Domain-Sequence IRO object . . . 11
3.4.3. Option B: Existing IRO Object Type 12
3.4.4. Comparison 13
3.5. Exclude Route Object (XRO) 14
3.5.1. Subobjects 14
3.5.1.1. Autonomous system 14
3.5.1.2. IGP Area 15
3.6. Explicit Exclusion Route Subobject (EXRS) 16
3.7. Explicit Route Object (ERO) 17
4. Other Considerations 17
4.1. Inter-Area Path Computation 18
4.2. Inter-AS Path Computation 20
4.2.1. Example 1 20
4.2.2. Example 2 22
4.3. Boundary Node and Inter-AS-Link 24
4.4. PCE Serving multiple Domains 24
4.5. P2MP 25
4.6. Hierarchical PCE 25
4.7. Relationship to PCE Sequence 27
4.8. Relationship to RSVP-TE 27
5. IANA Considerations 28

5.1. PCEP Objects 28
5.2. New Subobjects 28
5.3. Error Object Field Values 28
6. Security Considerations 29
7. Manageability Considerations 29
7.1. Control of Function and Policy 29
7.2. Information and Data Models 29
7.3. Liveness Detection and Monitoring 29
7.4. Verify Correct Operations 29
7.5. Requirements On Other Protocols 30
7.6. Impact On Network Operations 30
8. Acknowledgments 30
9. References 30
9.1. Normative References 30
9.2. Informative References 30

1. Introduction

A PCE may be used to compute end-to-end paths across multi-domain environments using a per-domain path computation technique [RFC5152]. The so called backward recursive path computation (BRPC) mechanism [RFC5441] defines a PCE-based path computation procedure to compute inter-domain constrained (G)MPLS TE LSPs. However, both per-domain and BRPC techniques assume that the sequence of domains to be crossed from source to destination is known, either fixed by the network operator or obtained by other means. Also for inter-domain point-to-multi-point (P2MP) tree computation, [PCE-P2MP-PROCEDURES] assumes the domain-tree is known in priori.

The list of domains (domain-sequence) in a point-to-point (P2P) path or a point-to-multi-point (P2MP) tree is usually a constraint in the path computation request. The PCE determines the next PCE to forward the request based on the domain-sequence. In a multi-domain path computation, a PCC MAY indicate the sequence of domains to be traversed using the Include Route Object (IRO) defined in [RFC5440].

When the sequence of domains is not known in advance, the Hierarchical PCE (H-PCE) [RFC6805] architecture and mechanisms can be used to determine the end-to-end Domain-Sequence.

This document defines a standard way to represent and encode a Domain-Sequence in various deployment scenarios including P2P, P2MP and H-PCE.

The Domain-Sequence (the set of domains traversed to reach the destination domain) is either administratively predetermined or discovered by some means (H-PCE) that is outside of the scope of this document.

[RFC5440] defines the Include Route Object (IRO) and the Explicit Route Object (ERO); [RFC5521] defines the Exclude Route Object (XRO) and the Explicit Exclusion Route Subobject (EXRS); The use of Autonomous System (AS) (albeit with a 2-Byte AS number) as an abstract node representing domain is defined in [RFC3209], this document specifies new subobjects to include or exclude domains such as an IGP area or an Autonomous Systems (4-Byte as per [RFC4893]).

Further, the domain identifier may simply act as delimiter to specify where the domain boundary starts and ends.

This is a companion document to Resource ReserVation Protocol - Traffic Engineering (RSVP-TE) extensions for the domain identifiers [DOMAIN-SUBOBJ].

1.1. 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 [RFC2119].

2. Terminology

The following terminology is used in this document.

ABR: OSPF Area Border Router. Routers used to connect two IGP areas.

AS: Autonomous System.

ASBR: Autonomous System Boundary Router.

BN: Boundary Node, Can be an ABR or ASBR.

BRPC: Backward Recursive Path Computation

Domain: As per [RFC4655], any collection of network elements within a common sphere of address management or path computational responsibility. Examples of domains include Interior Gateway Protocol (IGP) areas and Autonomous Systems (ASs).

Domain-Sequence: An ordered sequence of domains traversed to reach the destination domain.

ERO: Explicit Route Object

H-PCE: Hierarchical PCE

IGP: Interior Gateway Protocol. Either of the two routing protocols, Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS).

IRO: Include Route Object

IS-IS: Intermediate System to Intermediate System.

OSPF: Open Shortest Path First.

PCC: Path Computation Client: any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.

P2MP: Point-to-Multipoint

P2P: Point-to-Point

RSVP: Resource Reservation Protocol

TE LSP: Traffic Engineering Label Switched Path.

3. Detail Description

3.1. Domains

[RFC4726] and [RFC4655] define domain as a separate administrative or geographic environment within the network. A domain may be further defined as a zone of routing or computational ability. Under these definitions a domain might be categorized as an AS or an IGP area. Each AS can be made of several IGP areas. In order to encode a Domain-Sequence, it is required to uniquely identify a domain in the Domain-Sequence. A domain can be uniquely identified by area-id or AS or both.

3.2. Domain-Sequence

A domain-sequence is an ordered sequence of domains traversed to reach the destination domain.

A domain-sequence can be applied as a constraint and carried in path computation request to PCE(s). A domain-sequence can also be the result of a path computation. For example, in the case of H-PCE

[RFC6805] Parent PCE MAY send the Domain-Sequence as a result in a path computation reply.

In this context, the ordered nature of a domain-sequence is considered to be important. In a P2P path, the domains listed appear in the order that they are crossed. In a P2MP path, the domain tree is represented as list of domain sequences.

A domain-sequence enables a PCE to select the next PCE to forward the path computation request based on the domain information.

A PCC or PCE MAY add an additional constraints covering which Boundary Nodes (ABR or ASBR) or Border links (Inter-AS-link) MUST be traversed while defining a Domain-Sequence.

Thus a Domain-Sequence MAY be made up of one or more of -

- o AS Number
- o Area ID
- o Boundary Node ID
- o Inter-AS-Link Address

Consequently, a Domain-Sequence can be used:

1. by a PCE in order to discover or select the next PCE in a collaborative path computation, such as in BRPC [RFC5441];
2. by the Parent PCE to return the Domain-Sequence when unknown, this can further be an input to BRPC procedure [RFC6805];
3. by a PCC (or PCE) to constraint the domains used in a H-PCE path computation, explicitly specifying which domains to be expanded;
4. by a PCE in per-domain path computation model [RFC5152] to identify the next domain(s);

3.3. Standard Representation

Domain-Sequence MAY appear in PCEP Messages, notably in -

- o Include Route Object (IRO): As per [RFC5440], used to specify set of network elements that MUST be traversed. These subobjects are used to specify the domain-sequence that MUST be traversed to reach the destination.

- o Exclude Route Object (XRO): As per [RFC5521], used to specify certain abstract nodes that MUST be excluded from whole path. These subobjects are used to specify certain domains that MUST be avoided to reach the destination.

- o Explicit Exclusion Route Subobject (EXRS): As per [RFC5521], used to specify exclusion of certain abstract nodes between a specific pair of nodes. EXRS are a subobject inside the IRO. These subobjects are used to specify the domains that must be excluded between two abstract nodes.

- o Explicit Route Object (ERO): As per [RFC5440], used to specify a computed path in the network. For example, in the case of H-PCE [RFC6805] Parent PCE MAY send the Domain-Sequence as a result in a path computation reply using ERO.

3.4. Include Route Object (IRO)

As per [RFC5440], IRO (Include Route Object) can be used to specify that the computed path MUST traverse a set of specified network elements or abstract nodes.

3.4.1. Subobjects

Some subobjects are defined in [RFC3209], [RFC3473], [RFC3477] and [RFC4874], but new subobjects related to Domain-Sequence are needed.

The following subobject types are used in IRO.

Type	Subobject
1	IPv4 prefix
2	IPv6 prefix
4	Unnumbered Interface ID
32	Autonomous system number (2 Byte)
33	Explicit Exclusion (EXRS)

This document extends the above list to support 4-Byte AS numbers and IGP Areas.

Type	Subobject
TBD	Autonomous system number (4 Byte)
TBD	OSPF Area id
TBD	ISIS Area id

- Autonomous system

[RFC3209] already defines 2 byte AS number.

To support 4 byte AS number as per [RFC4893] following subobject is defined:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |         Reserved              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          AS-ID (4 bytes)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

L: The L bit is an attribute of the subobject as defined in [RFC3209].

Type: (TBA by IANA) indicating a 4-Byte AS Number.

Length: 8 (Total length of the subobject in bytes).

Reserved: Zero at transmission, ignored at receipt.

AS-ID: The 4-Byte AS Number. Note that if 2-Byte AS numbers are in use, the low order bits (16 through 31) should be used and the high order bits (0 through 15) should be set to zero.

- IGP Area

Since the length and format of Area-id is different for OSPF and ISIS, following two subobjects are defined:

For OSPF, the area-id is a 32 bit number. The subobject is encoded as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |         Reserved              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    OSPF Area Id (4 bytes)                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

L: The L bit is an attribute of the subobject as defined in [RFC3209].

Type: (TBA by IANA) indicating a 4-Byte OSPF Area ID.

Length: 8 (Total length of the subobject in bytes).

Reserved: Zero at transmission, ignored at receipt.

OSPF Area Id: The 4-Byte OSPF Area ID.

For IS-IS, the area-id is of variable length and thus the length of the Subobject is variable. The Area-id is as described in IS-IS by ISO standard [ISO10589]. The subobject is encoded as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |  Area-Len     |  Reserved     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                        IS-IS Area ID                        //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

L: The L bit is an attribute of the subobject as defined in [RFC3209].

Type: (TBA by IANA) indicating IS-IS Area ID.

Length: Variable. As per [RFC3209], the total length of the subobject in bytes, including the L, Type and Length fields. The Length MUST be at least 4, and MUST be a multiple of 4.

Area-Len: Variable (Length of the actual (non-padded) IS-IS Area Identifier in octets; Valid values are from 2 to 11 inclusive).

Reserved: Zero at transmission, ignored at receipt.

IS-IS Area Id: The variable-length IS-IS area identifier. Padded with trailing zeroes to a four-byte boundary.

3.4.2. Option (A): New IRO Object Type

[RFC5440] in its description of IRO does not require the subobjects to be in a given particular order. When considering a Domain-Sequence, the domain relative ordering is a basic criterion and, as such, this option suggest a new IRO object type.

IRO Object-Class is 10.
IRO Object-Type is TBD. (2 suggested value to IANA)

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                        (Subobjects)                         //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Subobjects: The IRO is made of subobjects identical to the ones defined in [RFC3209], [RFC3473], and [RFC3477], where the IRO subobject type is identical to the subobject type defined in the related documents. Some new subobjects related to Domain-Sequence are also added in this document as mentioned in Section 3.4.

[RFC3209] defines subobjects for IPv4, IPv6 and unnumbered Interface ID, which in the context of domain-sequence is used to specify Boundary Node (ABR/ASBR) and Inter-AS-Links. The subobjects for AS Number (2 or 4 Byte) and IGP Area is used to specify the domain identifiers in the domain-sequence.

The new IRO Object-Type used to define the domain-sequence would handle the L bit (Loose / Strict) in the subobjects similar to [RFC3209].

Further we have following options:

* Option (A.1): New IRO Object Type for Domain-Sequence object only. A new IRO Object Type is used to specify the ordered sequence of domains (Domain-Sequence) only. The PCReq message is modified to allow encoding of both types of IRO; one with IRO Type 1 [RFC5440] used to specify the intra-domain abstract nodes and resources and the second IRO Type with the new subobjects as described in this section to specify the domain-sequence. All other rules of PCEP objects and message processing (ex. P bit handling of Common Object Header) is as per [RFC5440].

* Option (A.2): New IRO Object Type for both intra and inter-domain (domain-sequence). A new IRO Object Type is used to include both intra nodes and inter-domains nodes but the sequence of domain is strict. The intra-domain nodes can still be ordered. In case of inter-domain path computation, only the new IRO type is used which contains the specific intra domain network nodes as well as inter-domain abstract nodes or domains. The inter-domain abstract nodes are encoded in the sequence they must be traversed but the intra-

domain elements MAY be an unordered set. There is no need to change the PCReq message format.

3.4.2.1. Handling of the Domain-Sequence IRO object

An IRO object containing Domain-Sequence subobjects constraints or defines the domains involved in a multi-domain path computation, typically involving two or more collaborative PCEs.

A Domain-Sequence can have varying degrees of granularity; it is possible to have a Domain-Sequence composed of, uniquely, AS identifiers. It is also possible to list the involved areas for a given AS.

In any case, the mapping between domains and responsible PCEs is not defined in this document. It is assumed that a PCE that needs to obtain a "next PCE" from a Domain-Sequence is able to do so (e.g. via administrative configuration, or discovery).

A PCC builds a Domain-Sequence IRO to encode the Domain-Sequence, that is all domains that it wishes the cooperating PCEs to traverse in order to compute the end to end path.

For each inclusion, the PCC clears the L-bit to indicate that the PCE is required to include the domain, or sets the L-bit to indicate that the PCC simply desires that the domain be included in the domain-sequence.

When a PCE receives a PCEP Request message with an IRO, it looks for a Domain-Sequence IRO (new type) to see if a domain-sequence is specified. If the message contains more than one Domain-Sequence IRO (new type), it MUST use the first one in the message and MUST ignore subsequent instances.

If a PCE does not recognize the Domain-Sequence IRO (new type), it MUST return a PCErr message with Error-Type "Unknown Object" and Error-value "Unrecognized object Type" as described in [RFC5440].

If a PCE is unwilling or unable to process the Domain-Sequence IRO (new type), it MUST return a PCErr message with the Error-Type "Not supported object" and follow the relevant procedures described in [RFC5440].

If a PCE that supports the Domain-Sequence IRO (new type) and encounters a subobject that it does not support or recognize, it MUST act according to the setting of the L-bit in the subobject. If the L-bit is clear, the PCE MUST respond with a PCErr with Error-Type "Unrecognized subobject" and set the Error-Value to the subobject

type code. If the L-bit is set, the PCE MAY respond with a PCErr as already stated or MAY ignore the subobject: this choice is a local policy decision.

If a PCE parses a Domain-Sequence IRO (new type), it MUST act according to the requirements expressed in the subobject. That is, if the L-bit is clear, the PCE(s) MUST produce a path that follows domain-sequence nodes in order identified by the subobjects in the path. If the L-bit is set, the PCE(s) SHOULD produce a path along the Domain-Sequence unless it is not possible to construct a path complying with the other constraints expressed in the request.

A successful path computation reported in a PCEP response message MUST include an ERO to specify the path that has been computed as specified in [RFC5440] following the sequence of domains.

In a PCEP response message, PCE MAY also supply a Domain-Sequence IRO (new type) with the NO-PATH object indicating that the set of elements of the request’s Domain-Sequence IRO prevented the PCE from finding a path.

Subobject types for AS and IGP Area affect the next domain selection as well as finding the PCE serving that domain.

Note that a particular domain in the domain-sequence can be identified by :-

- o A single IGP Area: Only the IGP (OSPF or ISIS) Area subobject is used to identify the next domain. (Refer Figure 1)

- o A single AS: Only the AS subobject is used to identify the next domain. (Refer Figure 2)

- o Both an AS and an IGP Area: Combination of both AS and Area are used to identify the next domain. In this case the order is AS Subobject followed by Area. (Refer Figure 3)

Subobject representing Boundary Node or Inter-AS-Link MUST be applied during the final path computation procedure as before.

3.4.3. Option B: Existing IRO Object Type

The IRO (Include Route Object) [RFC5440] is an optional object used to specify a set of network elements that the computed path MUST traverse.

The new subobjects denoting the domain-sequence are carried in the same IRO Type 1, and all the rules of processing as specified in [RFC5440] are applied.

Note the following differences :-

- o Order: Since there is no inherent order specified in the encoding of the subobjects in IRO Type 1 [RFC5440], it is the job of the PCE to figure out the optimal order of the domains to be crossed to reach the destination domain. Note that in case of doubt, or when applicable, the PCE can still apply the ordering as specified in the request message. Further PCE may have to crankback and try multiple permutations before figuring out the correct sequence.

- o Loose / Strict (L-Bit): [RFC5440] state that the L bit of the subobjects within an IRO Type 1 [RFC5440] has no meaning. This will be applicable for subobjects denoting domain-sequence as well.

- o Scope: Coexistence of intra-domain nodes, boundary nodes and domain nodes in the same IRO List. It is the job of PCE to figure out the scope and apply the processing rules accordingly. The nodes in the IRO which are recognized by the PCE are handled locally and others are forwarded to next PCE hoping they would handle them, the aggregating PCE (Ingress PCE or Parent) would make sure that all nodes in IRO are handled correctly.

3.4.4. Comparison

	Option (A.1): New IRO Object Type for Domain-Sequence object only	Option (A.2): New IRO Object Type for both intra and inter-domain	Option B: Existing IRO Object Type
Order	Yes	Yes	No
L/X bit	Yes	Yes	No
Msg Format Unchanged	No	Yes	Yes
Seperation of scope	Yes	Yes*	No

* becasue of the ordered nature, intra-domain nodes would be first in the new IRO type

[Editor’s Note: Based on our opinion and the feedback received so far, we feel the option A.2 should be selected.]

3.5. Exclude Route Object (XRO)

The Exclude Route Object (XRO) [RFC5521] is an optional object used to specify exclusion of certain abstract nodes or resources from the whole path.

3.5.1. Subobjects

The following subobject types are defined to be used in XRO as defined in [RFC3209], [RFC3477], [RFC4874], and [RFC5521].

Type	Subobject
1	IPv4 prefix
2	IPv6 prefix
4	Unnumbered Interface ID
32	Autonomous system number (2 Byte)
34	SRLG
64	IPv4 Path Key
65	IPv6 Path Key

This document extends the above list to support 4-Byte AS numbers and IGP Areas.

Type	Subobject
TBD	Autonomous system number (4 Byte)
TBD	OSPF Area id
TBD	ISIS Area id

3.5.1.1. Autonomous system

The new subobjects to support 4 byte AS and IGP (OSPF / ISIS) Area MAY also be used in the XRO to specify exclusion of certain domains in the path computation procedure.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|    Type     |     Length    |         Reserved              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          AS-ID (4 bytes)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The X-bit indicates whether the exclusion is mandatory or desired.

0: indicates that the AS specified MUST be excluded from the path computed by the PCE(s).

1: indicates that the AS specified SHOULD be avoided from the inter-domain path computed by the PCE(s), but MAY be included subject to PCE policy and the absence of a viable path that meets the other constraints.

All other fields are consistent with the definition in Section 3.4.

3.5.1.2. IGP Area

Since the length and format of Area-id is different for OSPF and ISIS, following two subobjects are defined:

For OSPF, the area-id is a 32 bit number. The subobject is encoded as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|    Type     |     Length    |         Reserved              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    OSPF Area Id (4 bytes)                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The X-bit indicates whether the exclusion is mandatory or desired.

0: indicates that the OSFF Area specified MUST be excluded from the path computed by the PCE(s).

1: indicates that the OSFF Area specified SHOULD be avoided from the inter-domain path computed by the PCE(s), but MAY be included subject to PCE policy and the absence of a viable path that meets the other constraints.

All other fields are consistent with the definition in Section 3.4.

For IS-IS, the area-id is of variable length and thus the length of the subobject is variable. The Area-id is as described in IS-IS by ISO standard [ISO10589]. The subobject is encoded as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|X|    Type     |     Length    |  Area-Len     |  Reserved     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                       IS-IS Area ID                         //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The X-bit indicates whether the exclusion is mandatory or desired.

0: indicates that the ISIS Area specified MUST be excluded from the path computed by the PCE(s).

1: indicates that the ISIS Area specified SHOULD be avoided from the inter-domain path computed by the PCE(s), but MAY be included subject to PCE policy and the absence of a viable path that meets the other constraints.

All other fields are consistent with the definition in Section 3.4.

If a PCE that supports XRO and encounters a subobject that it does not support or recognize, it MUST act according to the setting of the X-bit in the subobject. If the X-bit is clear, the PCE MUST respond with a PCErr with Error-Type "Unrecognized subobject" and set the Error-Value to the subobject type code. If the X-bit is set, the PCE MAY respond with a PCErr as already stated or MAY ignore the subobject: this choice is a local policy decision.

All the other processing rules are as per [RFC5521].

3.6. Explicit Exclusion Route Subobject (EXRS)

Explicit Exclusion Route Subobject (EXRS) [RFC5521] is used to specify exclusion of certain abstract nodes between a specific pair of nodes.

The EXRS subobject may carry any of the subobjects defined for inclusion in the XRO, thus the new subobjects to support 4 byte AS and IGP (OSPF / ISIS) Area MAY also be used in the EXRS. The meanings of the fields of the new XRO subobjects are unchanged when the subobjects are included in an EXRS, except that scope of the exclusion is limited to the single hop between the previous and subsequent elements in the IRO.

All the processing rules are as per [RFC5521].

3.7. Explicit Route Object (ERO)

The Explicit Route Object (ERO) [RFC5440] is used to specify a computed path in the network. PCEP ERO subobject types correspond to RSVP-TE ERO subobject types as defined in [RFC3209], [RFC3473], [RFC3477], [RFC4873], [RFC4874], and [RFC5520].

Type	Subobject
1	IPv4 prefix
2	IPv6 prefix
3	Label
4	Unnumbered Interface ID
32	Autonomous system number (2 Byte)
33	Explicit Exclusion (EXRS)
37	Protection
64	IPv4 Path Key
65	IPv6 Path Key

This document extends the above list to support 4-Byte AS numbers and IGP Areas.

Type	Subobject
TBD	Autonomous system number (4 Byte)
TBD	OSPF Area id
TBD	ISIS Area id

The new subobjects to support 4 byte AS and IGP (OSPF / ISIS) Area MAY also be used in the ERO to specify an abstract node (a group of nodes whose internal topology is opaque to the ingress node of the LSP). Using this concept of abstraction, an explicitly routed LSP can be specified as a sequence of domains.

In case of Hierarchical PCE [RFC6805], a Parent PCE MAY be requested to find the domain-sequence. Refer example in Section 4.6.

The format of the new ERO subobjects is similar to new IRO subobjects, refer Section 3.4.

4. Other Considerations

The examples in this section are for illustration purposes only; to show how the new subobjects may be encoded.

4.1. Inter-Area Path Computation

In an inter-area path computation where the ingress and the egress nodes belong to different IGP areas within the same AS, the Domain-Sequence MAY be represented using a ordered list of Area subobjects. The AS number MAY be skipped, as area information is enough to select the next PCE.

```
 +-------------------+                          +-------------------+
 |                   |                          |                   |
 |           +--+    |                          |     +--+          |
 |  +--+     |  |    |                          |     |  |          |
 |  |  |     +--+    |                          |     +--+   +--+   |
 |  +--*             +                          +            |  |   |
 |                   |                          |            +--+   |
 |         *--+      +                          +                   |
 |         |  |      |                          |     +--+          |
 |         +--+      |                          |     |  |          |
 |                   |+-------------------------+|     +--+          |
 |                  ++++                       +-++                 |
 |                  ||||         +--+          | ||                 |
 | Area 2           ++++         |  |          +-++  Area 4         |
 +-------------------+|          +--+            |+-------------------+
                      |                          |
                      |                +--+      |
                      |    +--+        |  |      |
                      |    |  |        +--+      |
                      |    +--+                  |
                      |                          |
                      |                          |
                      |                          |
                      |                          |
                      |           +--+           |
                      |           |  |           |
                      |           +--+           |
  +------------------+|                          |+--------------------+
  |                  ++-+                       +-++                  |
  |                  || |                       | ||                  |
  |                  ++-+    Area 0             +-++                  |
  |                  |+-------------------------+|     +--+          |
  |          +--+    |                           |     |  |          |
  |          |  |    |                           |     +--+          |
  | +--+     +--+    |                           |                   |
  | |  |             +                           +            +--+   |
  | +--+             |                           |            |  |   |
  |                  +                           +            +--+   |
  |       +--+       |                           |                   |
  |       |  |       |                           |     +--+          |
  |       +--+       |                           |     |  |          |
  |                  |                           |     +--+          |
  |                  |                           |                   |
  | Area 1           |                           |  Area 5           |
  +------------------+                           +--------------------+
```

Figure 1: Inter-Area Path Computation

AS Number is 100.

This could be represented in the <IRO> as:

```
+---------+ +---------+ +---------+ +---------+
|IRO      | |Sub      | |Sub      | |Sub      |
|Object   | |Object   | |Object   | |Object   |
|Header   | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         |
|         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+


+---------+ +---------+ +---------+ +---------+ +---------+
|IRO      | |Sub      | |Sub      | |Sub      | |Sub      |
|Object   | |Object AS| |Object   | |Object   | |Object   |
|Header   | |100      | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         | |         |
|         | |         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+ +---------+
```

AS is optional and it MAY be skipped. PCE should be able to understand both notations.

4.2. Inter-AS Path Computation

In inter-AS path computation, where ingress and egress belong to different AS, the Domain-Sequence is represented using an ordered list of AS subobjects. The Domain-Sequence MAY further include decomposed area information in Area subobjects.

4.2.1. Example 1

As shown in Figure 2, where AS to be made of a single area, the area subobject MAY be skipped in the Domain-Sequence as AS is enough to uniquely identify the next domain and PCE.

```
                                      +---------------------------------+
                                      |AS 200                           |
                                      |            +------+             |
                                      |            |      |             |
      +------------------------+      |            |      |   +------+  |
      | AS 100                 |      |            +------+   |      |  |
      |               +------+ |      | +------+              |      |  |
      |               |      +-+-----+-+      |              +------+  |
      |               |      | |      | |      |                        |
      |               +------+ |      | +------+                        |
      | +------+               |      |              +------+            |
      | |      |               |      |              |      |            |
      | |      |               |      |              |      |            |
      | +------+               |      |              +------+            |
      |                        |      |                                  |
      |               +------+ |      | +------+                        |
      |               |      +-+-----+-+      |               +------+ |
      |               |      | |      | |      |               |      | |
      |               +------+ |      | +------+               |      | |
      |                        |      |                        +------+ |
      |                        |      |                                 |
      |                        |      |                                 |
      |       +------+         |      |              +------+           |
      |       |      |         |      |              |      |           |
      |       |PCE   |         |      |              |PCE   |           |
      |       +------+         |      |              +------+           |
      |                        |      |                                 |
      +------------------------+      |                                 |
                                      +---------------------------------+
```

Figure 2: Inter-AS Path Computation

Both AS are made of Area 0.

This could be represented in the <IRO> as:

```
+---------+  +---------+  +---------+
|IRO      |  |Sub      |  |Sub      |
|Object   |  |Object AS|  |Object AS|
|Header   |  |100      |  |200      |
|         |  |         |  |         |
|         |  |         |  |         |
+---------+  +---------+  +---------+
```

```
+---------+  +---------+  +---------+  +---------+  +---------+
|IRO      |  |Sub      |  |Sub      |  |Sub      |  |Sub      |
|Object   |  |Object AS|  |Object   |  |Object AS|  |Object   |
|Header   |  |100      |  |Area 0   |  |200      |  |Area 0   |
|         |  |         |  |         |  |         |  |         |
|         |  |         |  |         |  |         |  |         |
+---------+  +---------+  +---------+  +---------+  +---------+
```

Area subobject is optional and it MAY be skipped. PCE should be able to understand both notations.

4.2.2. Example 2

As shown in Figure 3, where AS 200 is made up of multiple areas and multiple domain-sequence exist, PCE MAY include both AS and Area subobject to uniquely identify the next domain and PCE.

```
                 |
                 |  +-------------+                +----------------+
                 |  |Area 2       |                |Area 4          |
                 |  |         +--+|                |          +--+  |
                 |  |         |  ||                |          |  |  |
                 |  |  +--+   +--+|                |   +--+   +--+  |
                 |  |  |  |       |                |   |  |         |
                 |  |  *--+       |                |   +--+         |
                 |  | /      +--+ |                |          +--+  |
                 |  |/       |  | |                |          |  |  |
                 |  /        +--+ |                |   +--+   +--+  |
                 | /|  +--+       |+---------------+|   |  |         |
                 |/ |  |  |       ++-+          +-++   +--+         |
+-------------+/  |  +--+       || |          | ||                 |
|             /|  |             ++-+          +-++                 |
|         +--*||  +-------------+|              |+----------------+
|         |  |||                 |     +--+     |
|         +--+||                 |     |  |     |
|    +--+     ||                 |     +--+     |
|    |  |     ||                 |              |
|    +--+     ||                 |              |
```

```
   |             ||                 |     +--+     |
   |+--+         ||                 |     |  |     |
   ||  |         ||                 |     +--+     |
   |+--+         ||                 |              |
   |             ||                 |     +--+     |
   |    +--+     ||  +------------+ |     |  |     |+----------------+
   |    |  |     ||  |Area 3      +-++    +--+   +-++ Area 5         |
   |    +--+     ||  |            | ||           | ||                |
   |             ||  |            +-++           +-++                |
   |         +--+||  |       +--+ | |  Area 0      ||   +--+         |
   |         |  |||  |       |  | | +--------------+|   |  |         |
   |         +--*||  |       +--+ |                 |   +--+         |
   |             \|  |            |                 |          +--+  |
   |Area 1       |\  |   +--+     |                 |   +--+   |  |  |
   +-------------+|\ |   |  |     |                 |   |  |   +--+  |
                  | \|   +--+  +--+                 |   +--+         |
                  |  \         |  |                 |                |
                  |  |\        +--+                 |          +--+  |
                  |  | \ +--+     |                 |          |  |  |
                  |  |  \|  |     |                 |          +--+  |
                  |  |   *--+     |                 |                |
                  |  |            |                 |                |
                  |  +------------+                 +----------------+
                  |
                  |
       AS 100     |  AS 200
                  |
```

Figure 3: Inter-AS Path Computation

The Domain-Sequence can be carried in the IRO as shown below:

```
+-------+ +-------+ +-------+ +-------+ +-------+ +-------+ +-------+
|IRO    | |Sub    | |Sub    | |Sub    | |Sub    | |Sub    | |Sub    |
|Object | |Object | |Object | |Object | |Object | |Object | |Object |
|Header | |AS 100 | |Area 1 | |AS 200 | |Area 3 | |Area 0 | |Area 4 |
|       | |       | |       | |       | |       | |       | |       |
+-------+ +-------+ +-------+ +-------+ +-------+ +-------+ +-------+
```

The combination of both an AS and an Area uniquely identify a domain in the Domain-Sequence.

Note that an Area domain identifier always belongs to the previous AS that appears before it or, if no AS subobjects are present, it is assumed to be the current AS.

If the area information cannot be provided, PCE MAY forward the path computation request to the next PCE based on AS alone. If multiple PCEs are responsible, PCE MAY apply local policy to select the next PCE.

4.3. Boundary Node and Inter-AS-Link

A PCC or PCE MAY add additional constraints covering which Boundary Nodes (ABR or ASBR) or Border links (Inter-AS-link) MUST be traversed while defining a Domain-Sequence. In which case the Boundary Node or Link MAY be encoded as a part of the domain-sequence using the existing subobjects.

Boundary Nodes (ABR / ASBR) can be encoded using the IPv4 or IPv6 prefix subobjects usually the loopback address of 32 and 128 prefix length respectively. An Inter-AS link can be encoded using the IPv4 or IPv6 prefix subobjects or unnumbered interface subobjects.

For Figure 1, an ABR to be traversed can be specified as:

```
+---------+ +---------+ +---------++---------+ +---------+
|IRO      | |Sub      | |Sub      ||Sub      | |Sub      |
|Object   | |Object   | |Object   ||Object   | |Object   |
|Header   | |Area 2   | |IPv4     ||Area 0   | |Area 4   |
|         | |         | |x.x.x.x  ||         | |         |
|         | |         | |         ||         | |         |
+---------+ +---------+ +---------++---------+ +---------+
```

For Figure 2, an inter-AS-link to be traversed can be specified as:

```
+---------+  +---------+ +---------+ +---------+ +---------+
|IRO      |  |Sub      | |Sub      | |Sub      | |Sub      |
|Object   |  |Object AS| |Object   | |Object   | |Object AS|
|Header   |  |100      | |IPv4     | |IPv4     | |200      |
|         |  |         | |x.x.x.x  | |x.x.x.x  | |         |
|         |  |         | |         | |         | |         |
+---------+  +---------+ +---------+ +---------+ +---------+
```

4.4. PCE Serving multiple Domains

A single PCE MAY be responsible for multiple domains; for example PCE function deployed on an ABR. A PCE which can support 2 adjacent domains can internally handle this situation without any impact on the neighboring domains.

4.5. P2MP

In case of inter-domain P2MP path computation, (Refer [PCE-P2MP-PROCEDURES]) the path domain tree is nothing but a series of Domain Sequences, as shown in the below figure:

```
D1-D3-D6, D1-D3-D5 and D1-D2-D4.
            D1
           /  \
          D2  D3
         /   /  \
        D4  D5  D6
```

All rules of processing as applied to P2P can be applied to P2MP as well.

In case of P2MP, different destinations MAY have different Domain-Sequence within the domain tree, it requires domain-sequence to be attached per destination. (Refer [PCE-P2MP-PER-DEST])

4.6. Hierarchical PCE

As per [RFC6805], consider a case as shown in Figure 4 consisting of multiple child PCEs and a parent PCE.

```
                                 +--------+
                                 | Parent |
                                 | PCE    |
                                 +--------+

 +-------------------+                          +-------------------+
 |           +--+    |                          |     +--+          |
 |  +--+     |  |    |                          |     |  |          |
 |  |  |     +--+    |                          |     +--+   +--+   |
 |  +--*             +                          +            |  |   |
 |                   |                          |            +--+   |
 |         *--+      +                          +                   |
 |         |  |      |                          |     +--+          |
 |         +--+      |                          |     |  |          |
 |                   |+-------------------------+|     +--+          |
 |                  ++++                       +-++                 |
 |                  ||||         +--+          | ||                 |
 | Area 2           ++++         |  |          +-++  Area 4         |
 +-------------------+|          +--+            |+-------------------+
                      |                +--+      |
                      |    +--+        |  |      |
                      |    |  |        +--+      |
                      |    +--+                  |
                      |                          |
                      |           +--+           |
                      |           |  |           |
                      |           +--+           |
  +------------------+|                          |+--------------------+
  |                  ++-+                       +-++                  |
  |                  || |                       | ||                  |
  |                  ++-+    Area 0             +-++                  |
  |                  |+-------------------------+|     +--+          |
  |          +--+    |                           |     |  |          |
  |          |  |    |                           |     +--+          |
  | +--+     +--+    |                           |                   |
  | |  |             +                           +            +--+   |
  | +--+             |                           |            |  |   |
  |                  +                           +            +--+   |
  |       +--+       |                           |                   |
  |       |  |       |                           |     +--+          |
  |       +--+       |                           |     |  |          |
  |                  |                           |     +--+          |
  | Area 1           |                           |  Area 5           |
  +------------------+                           +--------------------+
```

Figure 4: Hierarchical PCE

In H-PCE, the Ingress PCE PCE(1) can request the parent PCE to determine the Domain-Sequence and return it in the PCEP response, using the ERO Object. The ERO can contain an ordered sequence of subobjects such as AS and Area (OSPF/ISIS) subobjects. In this case, the Domain-Sequence appear as:

```
+---------+ +---------+ +---------+ +---------+
|ERO      | |Sub      | |Sub      | |Sub      |
|Object   | |Object   | |Object   | |Object   |
|Header   | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         |
|         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+


+---------+ +---------+ +---------+ +---------+ +---------+
|ERO      | |Sub      | |Sub      | |Sub      | |Sub      |
|Object   | |Object AS| |Object   | |Object   | |Object   |
|Header   | |100      | |Area 2   | |Area 0   | |Area 4   |
|         | |         | |         | |         | |         |
|         | |         | |         | |         | |         |
+---------+ +---------+ +---------+ +---------+ +---------+
```

Note that, in the case of ERO objects, no new PCEP object type is required since the ordering constraint is assumed.

4.7. Relationship to PCE Sequence

Instead of a domain-sequence, a sequence of PCEs MAY be enforced by policy on the PCC, and this constraint can be carried in the PCReq message (as defined in [RFC5886]).

Note that PCE-Sequence can be used along with domain-sequence in which case PCE-Sequence SHOULD have higher precedence in selecting the next PCE in the inter-domain path computation procedures. Note that Domain-Sequence IRO constraints should still be checked as per the rules of processing IRO.

4.8. Relationship to RSVP-TE

[RFC3209] already describes the notion of abstract nodes, where an abstract node is a group of nodes whose internal topology is opaque to the ingress node of the LSP. It further defines a subobject for AS but with a 2-Byte AS Number.

[DOMAIN-SUBOBJ] extends the notion of abstract nodes by adding new subobjects for IGP Areas and 4-byte AS numbers. These subobjects MAY

be included in Explicit Route Object (ERO), Exclude Route object (XRO) or Explicit Exclusion Route Subobject (EXRS) in RSVP-TE.

In any case subobject type defined in RSVP-TE are identical to the subobject type defined in the related documents in PCEP.

5. IANA Considerations

5.1. PCEP Objects

The "PCEP Parameters" registry contains a subregistry "PCEP Objects". IANA is requested to make the following allocations from this registry.

Object Class	Name	Reference
10	IRO Object-Type (TBA): Domain-Sequence	[RFC5440] [This I.D.]

5.2. New Subobjects

The "PCEP Parameters" registry contains a subregistry "PCEP Objects" with an entry for the Include Route Object (IRO), Exclude Route Object (XRO) and Explicit Route Object (ERO). IANA is requested to add further subobjects as follows:

7 ERO
10 IRO
17 XRO

Subobject Type		Reference
TBA	4 byte AS number	[This I.D.]
TBA	OSPF Area ID	[This I.D.]
TBA	IS-IS Area ID	[This I.D.]

5.3. Error Object Field Values

The "PCEP Parameters" registry contains a subregistry "Error Types and Values". IANA is requested to make the following allocations from this subregistry

ERROR Type	Meaning	Reference
TBA	"Unrecognized subobject" Error-Value: type code	[This I.D.]

6. Security Considerations

This document specifies a standard representation of Domain-Sequence and new subobjects, which MAY be used in inter-domain PCE scenarios as explained in other RFC and drafts. The new subobjects and Domain-Sequence mechanisms defined in this document allow finer and more specific control of the path computed by a cooperating PCE(s). Such control increases the risk if a PCEP message is intercepted, modified, or spoofed because it allows the attacker to exert control over the path that the PCE will compute or to make the path computation impossible. Therefore, the security techniques described in [RFC5440] are considered more important.

Note, however, that the Domain-Sequence mechanisms also provide the operator with the ability to route around vulnerable parts of the network and may be used to increase overall network security.

7. Manageability Considerations

7.1. Control of Function and Policy

Several local policy decisions should be made at the PCE. Firstly, the exact behavior with regard to desired inclusion and exclusion of domains must be available for examination by an operator and may be configurable. Second, the behavior on receipt of an unrecognized subobjects with the L or X-bit set should be configurable and must be available for inspection. The inspection and control of these local policy choices may be part of the PCEP MIB module.

7.2. Information and Data Models

A MIB module for management of the PCEP is being specified in a separate document [PCEP-MIB]. That MIB module allows examination of individual PCEP messages, in particular requests, responses and errors. The MIB module MUST be extended to include the ability to view the domain-sequence extensions defined in this document.

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].

7.5. Requirements On Other Protocols

In case of per-domain path computation [RFC5152], where the full path of an inter-domain TE LSP cannot be or is not determined at the ingress node, and signaling message may use domain identifiers. The Subobjects defined in this document SHOULD be supported by RSVP-TE. [DOMAIN-SUBOBJ] extends the notion of abstract nodes by adding new subobjects for IGP Areas and 4-byte AS numbers.

Apart from this, mechanisms defined in this document do not imply any requirements on other protocols in addition to those already listed in [RFC5440].

7.6. 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].

8. Acknowledgments

We would like to thank Adrian Farrel, Pradeep Shastry, Suresh Babu, Quintin Zhao, Fatai Zhang, Daniel King, Oscar Gonzalez, Chen Huaimo, Venugopal Reddy, Reeja Paul Sandeep Boina and Avantika for their useful comments and suggestions.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9.2. 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, December 2001.

[RFC3473] Berger, L., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, January 2003.

[RFC3477] Kompella, K. and Y. Rekhter, "Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)", RFC 3477, January 2003.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC4726] Farrel, A., Vasseur, J., and A. Ayyangar, "A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering", RFC 4726, November 2006.

[RFC4873] Berger, L., Bryskin, I., Papadimitriou, D., and A. Farrel, "GMPLS Segment Recovery", RFC 4873, May 2007.

[RFC4874] Lee, CY., Farrel, A., and S. De Cnodder, "Exclude Routes - Extension to Resource ReserVation Protocol-Traffic Engineering (RSVP-TE)", RFC 4874, April 2007.

[RFC4893] Vohra, Q. and E. Chen, "BGP Support for Four-octet AS Number Space", RFC 4893, May 2007.

[RFC5152] Vasseur, JP., Ayyangar, A., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, February 2008.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5441] Vasseur, JP., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", RFC 5441, April 2009.

[RFC5520] Bradford, R., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", RFC 5520, April 2009.

[RFC5521] Oki, E., Takeda, T., and A. Farrel, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Route Exclusions", RFC 5521, April 2009.

[RFC5886] Vasseur, JP., Le Roux, JL., and Y. Ikejiri, "A Set of Monitoring Tools for Path Computation Element (PCE)-Based Architecture", RFC 5886, June 2010.

[RFC6805] King, D. and A. Farrel, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, November 2012.

[PCE-P2MP-PROCEDURES]
Zhao, Q., Dhody, D., Ali, Z., Saad,, T., Sivabalan,, S., and R. Casellas, "PCE-based Computation Procedure To Compute Shortest Constrained P2MP Inter-domain Traffic Engineering Label Switched Paths (draft-ietf-pce-pcep-inter-domain-p2mp-procedures)", July 2013.

[PCEP-MIB]
Koushik, A., Emile, S., Zhao, Q., King, D., and J. Hardwick, "PCE communication protocol(PCEP) Management Information Base", July 2013.

[PCE-P2MP-PER-DEST]
Dhody, D., Palle, U., and V. Kondreddy, "Supporting explicit inclusion or exclusion of abstract nodes for a subset of P2MP destinations in Path Computation Element Communication Protocol (PCEP). (draft-dhody-pce-pcep-p2mp-per-destination)", October 2013.

[DOMAIN-SUBOBJ]
Dhody, D., Palle, U., Kondreddy, V., and R. Casellas, "Domain Subobjects for Resource ReserVation Protocol - Traffic Engineering (RSVP-TE). (draft-dhody-ccamp-rsvp-te-domain-subobjects)", January 2014.

[ISO10589]
ISO, "Intermediate system to Intermediate system routing information exchange protocol for use in conjunction with the Protocol for providing the Connectionless-mode Network Service (ISO 8473)", ISO/IEC 10589:2002, 1992.

Authors' Addresses

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.ietf@gmail.com

Udayasree Palle
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: udayasree.palle@huawei.com

Ramon Casellas
CTTC
Av. Carl Friedrich Gauss n7
Castelldefels, Barcelona 08860
SPAIN

EMail: ramon.casellas@cttc.es

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 21, 2017

E. Crabbe
Oracle
I. Minei
Google, Inc.
J. Medved
Cisco Systems, Inc.
R. Varga
Pantheon Technologies SRO

June 19, 2017

PCEP Extensions for Stateful PCE
draft-ietf-pce-stateful-pce-21

Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

Although PCEP explicitly makes no assumptions regarding the information available to the PCE, it also makes no provisions for PCE control of timing and sequence of path computations within and across PCEP sessions. This document describes a set of extensions to PCEP to enable stateful control of MPLS-TE and GMPLS LSPs via PCEP.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://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 December 21, 2017.

Copyright Notice

Copyright (c) 2017 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 (http://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 4
2. Terminology 4
3. Motivation and Objectives for Stateful PCE 5
3.1. Motivation 5
3.1.1. Background 5
3.1.2. Why a Stateful PCE? 6
3.1.3. Protocol vs. Configuration 7
3.2. Objectives 7
4. New Functions to Support Stateful PCEs 8
5. Overview of Protocol Extensions 9
5.1. LSP State Ownership 9
5.2. New Messages 9
5.3. Error Reporting 10
5.4. Capability Advertisement 10
5.5. IGP Extensions for Stateful PCE Capabilities Advertisement 11
5.6. State Synchronization 12
5.7. LSP Delegation 15
5.7.1. Delegating an LSP 15
5.7.2. Revoking a Delegation 16
5.7.3. Returning a Delegation 18
5.7.4. Redundant Stateful PCEs 18
5.7.5. Redelegation on PCE Failure 19
5.8. LSP Operations 19
5.8.1. Passive Stateful PCE Path Computation Request/Response 19
5.8.2. Switching from Passive Stateful to Active Stateful . 21
5.8.3. Active Stateful PCE LSP Update 22
5.9. LSP Protection 23
5.10. PCEP Sessions 23
6. PCEP Messages 23
6.1. The PCRpt Message 24
6.2. The PCUpd Message 26
6.3. The PCErr Message 28
6.4. The PCReq Message 29

6.5. The PCRep Message 30
7. Object Formats 30
7.1. OPEN Object 30
7.1.1. Stateful PCE Capability TLV 30
7.2. SRP Object 31
7.3. LSP Object 33
7.3.1. LSP-IDENTIFIERS TLVs 35
7.3.2. Symbolic Path Name TLV 38
7.3.3. LSP Error Code TLV 39
7.3.4. RSVP Error Spec TLV 40
8. IANA Considerations 41
8.1. PCE Capabilities in IGP Advertisements 41
8.2. PCEP Messages 41
8.3. PCEP Objects 42
8.4. LSP Object 42
8.5. PCEP-Error Object 43
8.6. Notification Object 43
8.7. PCEP TLV Type Indicators 44
8.8. STATEFUL-PCE-CAPABILITY TLV 44
8.9. LSP-ERROR-CODE TLV 45
9. Manageability Considerations 45
9.1. Control Function and Policy 45
9.2. Information and Data Models 46
9.3. Liveness Detection and Monitoring 47
9.4. Verifying Correct Operation 47
9.5. Requirements on Other Protocols and Functional Components 47
9.6. Impact on Network Operation 47
10. Security Considerations 48
10.1. Vulnerability 48
10.2. LSP State Snooping 48
10.3. Malicious PCE 49
10.4. Malicious PCC 49
11. Contributing Authors 49
12. Acknowledgements 50
13. References 50
13.1. Normative References 50
13.2. Informative References 51
Authors' Addresses 53

1. Introduction

[RFC5440] describes the Path Computation Element Communication Protocol (PCEP). PCEP defines the communication between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between PCEs, enabling computation of Multiprotocol Label Switching (MPLS) for Traffic Engineering Label Switched Path (TE LSP) characteristics. Extensions for support of Generalized MPLS (GMPLS) in PCEP are defined in [I-D.ietf-pce-gmpls-pcep-extensions]

This document specifies a set of extensions to PCEP to enable stateful control of LSPs within and across PCEP sessions in compliance with [RFC4657]. It includes mechanisms to effect Label Switched Path (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.

Extensions to permit the PCE to drive creation of an LSP are defined in [I-D.ietf-pce-pce-initiated-lsp], which specifies PCE-initiated LSP creation.

1.1. 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 [RFC2119].

2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer, PCEP Speaker.

This document uses the following terms defined in [RFC4655]: TED.

This document uses the following terms defined in [RFC3031]: LSP.

This document uses the following terms defined in [RFC8051]: Stateful PCE, Passive Stateful PCE, Active Stateful PCE, Delegation, LSP State Database.

The following terms are defined in this document:

Revocation: an operation performed by a PCC on a previously delegated LSP. Revocation revokes the rights granted to the PCE in the delegation operation.

Redelegation Timeout Interval: the period of time a PCC waits for, when a PCEP session is terminated, before revoking LSP delegation to a PCE and attempting to redelegate LSPs associated with the terminated PCEP session to an alternate PCE. The Redelegation Timeout Interval is a PCC-local value that can be either operator-configured or dynamically computed by the PCC based on local policy.

State Timeout Interval: the period of time a PCC waits for, when a PCEP session is terminated, before flushing LSP state associated with that PCEP session and reverting to operator-defined default parameters or behaviors. The State Timeout Interval is a PCC-

local value that can be either operator-configured or dynamically computed by the PCC based on local policy.

LSP State Report: an operation to send LSP state (Operational / Admin Status, LSP attributes configured at the PCC and set by a PCE, etc.) from a PCC to a PCE.

LSP Update Request: an operation where an Active Stateful PCE requests a PCC to update one or more attributes of an LSP and to re-signal the LSP with updated attributes.

SRP-ID-number: a number used to correlate errors and LSP State Reports to LSP Update Requests. It is carried in the SRP (Stateful PCE Request Parameters) Object described in Section 7.2.

Within this document, PCEP communications are described through PCC-PCE relationship. The PCE architecture also supports the PCE-PCE communication, by having the requesting PCE fill the role of a PCC, as usual.

The message formats in this document are specified using Routing Backus-Naur Format (RBNF) encoding as specified in [RFC5511].

3. Motivation and Objectives for Stateful PCE

3.1. Motivation

[RFC8051] presents several use cases, demonstrating scenarios that benefit from the deployment of a stateful PCE. The scenarios apply equally to MPLS-TE and GMPLS deployments.

3.1.1. Background

Traffic engineering has been a goal of the MPLS architecture since its inception ([RFC3031], [RFC2702], [RFC3346]). In the traffic engineering system provided by [RFC3630], [RFC5305], and [RFC3209] information about network resources utilization is only available as total reserved capacity by traffic class on a per interface basis; individual LSP state is available only locally on each LER for its own LSPs. In most cases, this makes good sense, as distribution and retention of total LSP state for all LERs within in the network would be prohibitively costly.

Unfortunately, this visibility in terms of global LSP state may result in a number of issues for some demand patterns, particularly within a common setup and hold priority. This issue affects online traffic engineering systems.

A sufficiently over-provisioned system will by definition have no issues routing its demand on the shortest path. However, lowering the degree to which network over-provisioning is required in order to run a healthy, functioning network is a clear and explicit promise of MPLS architecture. In particular, it has been a goal of MPLS to provide mechanisms to alleviate congestion scenarios in which "traffic streams are inefficiently mapped onto available resources; causing subsets of network resources to become over-utilized while others remain underutilized" ([RFC2702]).

3.1.2. Why a Stateful PCE?

[RFC4655] defines a stateful PCE to be one in which the PCE maintains "strict synchronization between the PCE and not only the network states (in term of topology and resource information), but also the set of computed paths and reserved resources in use in the network." [RFC4655] also expressed a number of concerns with regard to a stateful PCE, specifically:

- o Any reliable synchronization mechanism would result in significant control plane overhead

- o Out-of-band TED synchronization would be complex and prone to race conditions

- o Path calculations incorporating total network state would be highly complex

In general, stress on the control plane will be directly proportional to the size of the system being controlled and the tightness of the control loop, and indirectly proportional to the amount of over-provisioning in terms of both network capacity and reservation overhead.

Despite these concerns in terms of implementation complexity and scalability, several TE algorithms exist today that have been demonstrated to be extremely effective in large TE systems, providing both rapid convergence and significant benefits in terms of optimality of resource usage [MXMN-TE]. All of these systems share at least two common characteristics: the requirement for both global visibility of a flow (or in this case, a TE LSP) state and for ordered control of path reservations across devices within the system being controlled. While some approaches have been suggested in order to remove the requirements for ordered control (See [MPLS-PC]), these approaches are highly dependent on traffic distribution, and do not allow for multiple simultaneous LSP priorities representing diffserv classes.

The use cases described in [RFC8051] demonstrate a need for visibility into global inter-PCC LSP state in PCE path computations, and for PCE control of sequence and timing in altering LSP path characteristics within and across PCEP sessions.

3.1.3. Protocol vs. Configuration

Note that existing configuration tools and protocols can be used to set LSP state, such as a Command Line Interface (CLI) tool. However, this solution has several shortcomings:

- o Scale & Performance: configuration operations often have transactional semantics which are typically heavyweight and often require processing of additional configuration portions beyond the state being directly acted upon, with corresponding cost in CPU cycles, negatively impacting both PCC stability LSP update rate capacity.

- o Security: when a PCC opens a configuration channel allowing a PCE to send configuration, a malicious PCE may take advantage of this ability to take over the PCC. In contrast, the PCEP extensions described in this document only allow a PCE control over a very limited set of LSP attributes.

- o Interoperability: each vendor has a proprietary information model for configuring LSP state, which limits interoperability of a stateful PCE with PCCs from different vendors. The PCEP extensions described in this document allow for a common information model for LSP state for all vendors.

- o Efficient State Synchronization: configuration channels may be heavyweight and unidirectional, therefore efficient state synchronization between a PCC and a PCE may be a problem.

3.2. Objectives

The objectives for the protocol extensions to support stateful PCE described in this document are as follows:

- o Allow a single PCC to interact with a mix of stateless and stateful PCEs simultaneously using the same protocol, i.e. PCEP.

- o Support efficient LSP state synchronization between the PCC and one or more active or passive stateful PCEs.

- o Allow a PCC to delegate control of its LSPs to an active stateful PCE such that a given LSP is under the control of a single PCE at any given time.

  * A PCC may revoke this delegation at any time during the lifetime of the LSP. If LSP delegation is revoked while the PCEP session is up, the PCC MUST notify the PCE about the revocation.

  * A PCE may return an LSP delegation at any point during the lifetime of the PCEP session. If LSP delegation is returned by the PCE while the PCEP session is up, the PCE MUST notify the PCC about the returned delegation.

- o Allow a PCE to control computation timing and update timing across all LSPs that have been delegated to it.

- o Enable uninterrupted operation of PCC's LSPs in the event of a PCE failure or while control of LSPs is being transferred between PCEs.

4. New Functions to Support Stateful PCEs

Several new functions are required in PCEP to support stateful PCEs. A function can be initiated either from a PCC towards a PCE (C-E) or from a PCE towards a PCC (E-C). The new functions are:

Capability advertisement (E-C,C-E): both the PCC and the PCE must announce during PCEP session establishment that they support PCEP Stateful PCE extensions defined in this document.

LSP state synchronization (C-E): after the session between the PCC and a stateful PCE is initialized, the PCE must learn the state of a PCC's LSPs before it can perform path computations or update LSP attributes in a PCC.

LSP Update Request (E-C): a PCE requests modification of attributes on a PCC's LSP.

LSP State Report (C-E): a PCC sends an LSP state report to a PCE whenever the state of an LSP changes.

LSP control delegation (C-E,E-C): a PCC grants to a PCE the right to update LSP attributes on one or more LSPs; the PCE becomes the authoritative source of the LSP's attributes as long as the delegation is in effect (See Section 5.7); the PCC may withdraw the delegation or the PCE may give up the delegation at any time.

Similarly to [RFC5440], no assumption is made about the discovery method used by a PCC to discover a set of PCEs (e.g., via static configuration or dynamic discovery) and on the algorithm used to select a PCE.

5. Overview of Protocol Extensions

5.1. LSP State Ownership

In PCEP (defined in [RFC5440]), LSP state and operation are under the control of a PCC (a PCC may be an LSR or a management station). Attributes received from a PCE are subject to PCC's local policy. The PCEP extensions described in this document do not change this behavior.

An active stateful PCE may have control of a PCC's LSPs that were delegated to it, but the LSP state ownership is retained by the PCC. In particular, in addition to specifying values for LSP's attributes, an active stateful PCE also decides when to make LSP modifications.

Retaining LSP state ownership on the PCC allows for:

- o a PCC to interact with both stateless and stateful PCEs at the same time

- o a stateful PCE to only modify a small subset of LSP parameters, i.e. to set only a small subset of the overall LSP state; other parameters may be set by the operator, for example through command line interface (CLI) commands

- o a PCC to revert delegated LSP to an operator-defined default or to delegate the LSPs to a different PCE, if the PCC get disconnected from a PCE with currently delegated LSPs

5.2. New Messages

In this document, we define the following new PCEP messages:

Path Computation State Report (PCRpt): a PCEP message sent by a PCC to a PCE to report the status of one or more LSPs. Each LSP State Report in a PCRpt message MAY contain the actual LSP's path, bandwidth, operational and administrative status, etc. An LSP Status Report carried on a PCRpt message is also used in delegation or revocation of control of an LSP to/from a PCE. The PCRpt message is described in Section 6.1.

Path Computation Update Request (PCUpd): a PCEP message sent by a PCE to a PCC to update LSP parameters, on one or more LSPs. Each LSP Update Request on a PCUpd message MUST contain all LSP parameters that a PCE wishes to be set for a given LSP. An LSP Update Request carried on a PCUpd message is also used to return LSP delegations if at any point PCE no longer desires control of an LSP. The PCUpd message is described in Section 6.2.

The new functions defined in Section 4 are mapped onto the new messages as shown in the following table.

Function	Message
Capability Advertisement (E-C,C-E)	Open
State Synchronization (C-E)	PCRpt
LSP State Report (C-E)	PCRpt
LSP Control Delegation (C-E,E-C)	PCRpt, PCUpd
LSP Update Request (E-C)	PCUpd

Table 1: New Function to Message Mapping

5.3. Error Reporting

Error reporting is done using the procedures defined in [RFC5440], and reusing the applicable error types and error values of [RFC5440] wherever appropriate. The current document defines new error values for several error types to cover failures specific to stateful PCE.

5.4. Capability Advertisement

During PCEP Initialization Phase, PCEP Speakers (PCE or PCC) advertise their support of stateful PCEP extensions. A PCEP Speaker includes the "Stateful PCE Capability" TLV, described in Section 7.1.1, in the OPEN Object to advertise its support for PCEP stateful extensions. The Stateful Capability TLV includes the ’LSP Update’ Flag that indicates whether the PCEP Speaker supports LSP parameter updates.

The presence of the Stateful PCE Capability TLV in PCC’s OPEN Object indicates that the PCC is willing to send LSP State Reports whenever LSP parameters or operational status changes.

The presence of the Stateful PCE Capability TLV in PCE’s OPEN message indicates that the PCE is interested in receiving LSP State Reports whenever LSP parameters or operational status changes.

The PCEP extensions for stateful PCEs MUST NOT be used if one or both PCEP Speakers have not included the Stateful PCE Capability TLV in their respective OPEN message. If the PCEP Speaker on the PCC supports the extensions of this draft but did not advertise this capability, then upon receipt of PCUpd message from the PCE, it MUST generate a PCErr with error-type 19 (Invalid Operation), error-value 2 (Attempted LSP Update Request if the stateful PCE capability was not advertised)(see Section 8.5) and it SHOULD terminate the PCEP

session. If the PCEP Speaker on the PCE supports the extensions of this draft but did not advertise this capability, then upon receipt of a PCRpt message from the PCC, it MUST generate a PCErr with error-type 19 (Invalid Operation), error-value 5 (Attempted LSP State Report if stateful PCE capability was not advertised) (see Section 8.5) and it SHOULD terminate the PCEP session.

LSP delegation and LSP update operations defined in this document may only be used if both PCEP Speakers set the LSP-UPDATE-CAPABILITY Flag in the "Stateful Capability" TLV to 'Updates Allowed (U Flag = 1)'. If this is not the case and LSP delegation or LSP update operations are attempted, then a PCErr with error-type 19 (Invalid Operation) and error-value 1 (Attempted LSP Update Request for a non-delegated LSP) (see Section 8.5) MUST be generated. Note that, even if one of the PCEP speakers does not set the LSP-UPDATE-CAPABILITY flag in its "Stateful Capability" TLV, a PCE can still operate as a passive stateful PCE by accepting LSP State Reports from the PCC in order to build and maintain an up to date view of the state of the PCC's LSPs.

5.5. IGP Extensions for Stateful PCE Capabilities Advertisement

When PCCs are LSRs participating in the IGP (OSPF or IS-IS), and PCEs are either LSRs or servers also participating in the IGP, an effective mechanism for PCE discovery within an IGP routing domain consists of utilizing IGP advertisements. Extensions for the advertisement of PCE Discovery Information are defined for OSPF and for IS-IS in [RFC5088] and [RFC5089] respectively.

The PCE-CAP-FLAGS sub-TLV, defined in [RFC5089], is an optional sub-TLV used to advertise PCE capabilities. It MAY be present within the PCED sub-TLV carried by OSPF or IS-IS. [RFC5088] and [RFC5089] provide the description and processing rules for this sub-TLV when carried within OSPF and IS-IS, respectively.

The format of the PCE-CAP-FLAGS sub-TLV is included below for easy reference:

Type: 5

Length: Multiple of 4.

Value: This contains an array of units of 32 bit flags with the most significant bit as 0. Each bit represents one PCE capability.

PCE capability bits are defined in [RFC5088]. This document defines new capability bits for the stateful PCE as follows:

Bit	Capability
11	Active Stateful PCE capability
12	Passive Stateful PCE capability

Note that while active and passive stateful PCE capabilities may be advertised during discovery, PCEP Speakers that wish to use stateful PCEP MUST negotiate stateful PCEP capabilities during PCEP session setup, as specified in the current document. A PCC MAY initiate stateful PCEP capability negotiation at PCEP session setup even if it did not receive any IGP PCE capability advertisements.

5.6. State Synchronization

The purpose of State Synchronization is to provide a checkpoint-in-time state replica of a PCC's LSP state in a PCE. State Synchronization is performed immediately after the Initialization phase ([RFC5440]).

During State Synchronization, a PCC first takes a snapshot of the state of its LSPs state, then sends the snapshot to a PCE in a sequence of LSP State Reports. Each LSP State Report sent during State Synchronization has the SYNC Flag in the LSP Object set to 1. The set of LSPs for which state is synchronized with a PCE is determined by the PCC's local configuration (see more details in Section 9.1) and MAY also be determined by stateful PCEP capabilities defined in other documents, such as [I-D.ietf-pce-stateful-sync-optimizations].

The end of synchronization marker is a PCRpt message with the SYNC Flag set to 0 for an LSP Object with PLSP-ID equal to the reserved value 0 (see Section 7.3). In this case, the LSP Object SHOULD NOT include the SYMBOLIC-PATH-NAME TLV and SHOULD include the LSP-IDENTIFIERS TLV with the special value of all zeroes. The PCRpt message MUST include an empty ERO as its intended path and SHOULD NOT include the optional RRO object for its actual path. If the PCC has no state to synchronize, it SHOULD only send the end of synchronization marker.

A PCE SHOULD NOT send PCUpd messages to a PCC before State Synchronization is complete. A PCC SHOULD NOT send PCReq messages to a PCE before State Synchronization is complete. This is to allow the PCE to get the best possible view of the network before it starts computing new paths.

Either the PCE or the PCC MAY terminate the session using the PCEP session termination procedures during the synchronization phase. If the session is terminated, the PCE MUST clean up state it received from this PCC. The session reestablishment MUST be re-attempted per

the procedures defined in [RFC5440], including use of a back-off timer.

If the PCC encounters a problem which prevents it from completing the LSP state synchronization, it MUST send a PCErr message with error-type 20 (LSP State Synchronization Error) and error-value 5 (indicating an internal PCC error) to the PCE and terminate the session.

The PCE does not send positive acknowledgements for properly received synchronization messages. It MUST respond with a PCErr message with error-type 20 (LSP State Synchronization Error) and error-value 1 (indicating an error in processing the PCRpt) (see Section 8.5) if it encounters a problem with the LSP State Report it received from the PCC and it MUST terminate the session.

A PCE implementing a limit on the resources a single PCC can occupy, MUST send a PCNtf message with Notification Type 4 (Stateful PCE resource limit exceeded) and Notification Value 1 (Entering resource limit exceeded state) in response to the PCRpt message triggering this condition in the synchronization phase and MUST terminate the session.

The successful State Synchronization sequence is shown in Figure 1.

```
+-+-+                     +-+-+
|PCC|                     |PCE|
+-+-+                     +-+-+
  |                         |
  |-----PCRpt, SYNC=1----->| (Sync start)
  |                         |
  |-----PCRpt, SYNC=1----->|
  |            .            |
  |            .            |
  |            .            |
  |-----PCRpt, SYNC=1----->|
  |            .            |
  |            .            |
  |            .            |
  |                         |
  |-----PCRpt, SYNC=0----->| (End of sync marker
  |                         |  LSP State Report
  |                         |  for PLSP-ID=0)
  |                         | (Sync done)
```

Figure 1: Successful state synchronization

The sequence where the PCE fails during the State Synchronization phase is shown in Figure 2.

```
                  +-+-+                    +-+-+
                  |PCC|                    |PCE|
                  +-+-+                    +-+-+
                    |                        |
                    |-----PCRpt, SYNC=1----->|
                    |                        |
                    |-----PCRpt, SYNC=1----->|
                    |            .           |
                    |            .           |
                    |            .           |
                    |-----PCRpt, SYNC=1----->|
                    |                        |
                    |-PCRpt, SYNC=1          |
                    |         \    ,-PCErr    |
                    |          \  /          |
                    |           \/           |
                    |           /\           |
                    |          /   '-------->| (Ignored)
                    |<--------'              |
```

Figure 2: Failed state synchronization (PCE failure)

The sequence where the PCC fails during the State Synchronization phase is shown in Figure 3.

```
                  +-+-+                    +-+-+
                  |PCC|                    |PCE|
                  +-+-+                    +-+-+
                    |                        |
                    |-----PCRpt, SYNC=1----->|
                    |                        |
                    |-----PCRpt, SYNC=1----->|
                    |            .           |
                    |            .           |
                    |            .           |
                    |-------- PCErr=? ------>|
                    |                        |
```

Figure 3: Failed state synchronization (PCC failure)

Optimizations to the synchronization procedures and alternate mechanisms of providing the synchronization function are outside the scope of this document and are discussed elsewhere (see [I-D.ietf-pce-stateful-sync-optimizations]).

5.7. LSP Delegation

If during Capability advertisement both the PCE and the PCC have indicated that they support LSP Update, then the PCC may choose to grant the PCE a temporary right to update (a subset of) LSP attributes on one or more LSPs. This is called "LSP Delegation", and it MAY be performed at any time after the Initialization phase, including during the State Synchronization phase.

A PCE MAY return an LSP delegation at any time if it no longer wishes to update the LSP’s state. A PCC MAY revoke an LSP delegation at any time. Delegation, Revocation, and Return are done individually for each LSP.

In the event of a delegation being rejected or returned by a PCE, the PCC SHOULD react based on local policy. It can, for example, either retry delegating to the same PCE using an exponentially increasing timer or delegate to an alternate PCE.

5.7.1. Delegating an LSP

A PCC delegates an LSP to a PCE by setting the Delegate flag in LSP State Report to 1. If the PCE does not accept the LSP Delegation, it MUST immediately respond with an empty LSP Update Request which has the Delegate flag set to 0. If the PCE accepts the LSP Delegation, it MUST set the Delegate flag to 1 when it sends an LSP Update Request for the delegated LSP (note that this may occur at a later time). The PCE MAY also immediately acknowledge a delegation by sending an empty LSP Update Request which has the Delegate flag set to 1.

The delegation sequence is shown in Figure 4.

```
                      +-+-+                    +-+-+
                      |PCC|                    |PCE|
                      +-+-+                    +-+-+
                        |                        |
                        |---PCRpt, Delegate=1--->| LSP Delegated
                        |                        |
                        |---PCRpt, Delegate=1--->|
                        |            .           |
                        |            .           |
                        |            .           |
                        |<--(PCUpd,Delegate=1)---| Delegation confirmed
                        |                        |
                        |---PCRpt, Delegate=1--->|
                        |                        |
```

Figure 4: Delegating an LSP

Note that for an LSP to remain delegated to a PCE, the PCC MUST set the Delegate flag to 1 on each LSP State Report sent to the PCE.

5.7.2. Revoking a Delegation

5.7.2.1. Explicit Revocation

When a PCC decides that a PCE is no longer permitted to modify an LSP, it revokes that LSP’s delegation to the PCE. A PCC may revoke an LSP delegation at any time during the LSP’s life time. A PCC revoking an LSP delegation MAY immediately remove the updated parameters provided by the PCE and revert to the operator-defined parameters, but to avoid traffic loss, it SHOULD do so in a make-before-break fashion. If the PCC has received but not yet acted on PCUpd messages from the PCE for the LSP whose delegation is being revoked, then it SHOULD ignore these PCUpd messages when processing the message queue. All effects of all messages for which processing started before the revocation took place MUST be allowed to complete and the result MUST be given the same treatment as any LSP that had been previously delegated to the PCE (e.g. the state MAY immediately revert to the operator-defined parameters).

If a PCEP session with the PCE to which the LSP is delegated exists in the UP state during the revocation, the PCC MUST notify that PCE by sending an LSP State Report with the Delegate flag set to 0, as shown in Figure 5.

```
                      +-+-+                     +-+-+
                      |PCC|                     |PCE|
                      +-+-+                     +-+-+
                        |                         |
                        |---PCRpt, Delegate=1--->|
                        |                         |
                        |<--(PCUpd,Delegate=1)---| Delegation confirmed
                        |            .            |
                        |            .            |
                        |            .            |
                        |---PCRpt, Delegate=0--->| PCC revokes delegation
                        |                         |
```

Figure 5: Revoking a Delegation

After an LSP delegation has been revoked, a PCE can no longer update LSP's parameters; an attempt to update parameters of a non-delegated LSP will result in the PCC sending a PCErr message with error-type 19 (Invalid Operation), error-value 1 (attempted LSP Update Request for a non-delegated LSP) (see Section 8.5).

5.7.2.2. Revocation on Redelegation Timeout

When a PCC's PCEP session with a PCE terminates unexpectedly, the PCC MUST wait the time interval specified in Redelegation Timeout Interval before revoking LSP delegations to that PCE and attempting to redelegate LSPs to an alternate PCE. If a PCEP session with the original PCE can be reestablished before the Redelegation Timeout Interval timer expires, LSP delegations to the PCE remain intact.

Likewise, when a PCC's PCEP session with a PCE terminates unexpectedly, and the PCC does not succeed in redelegating its LSPs, the PCC MUST wait for the State Timeout Interval before flushing any LSP state associated with that PCE. Note that the State Timeout Interval timer may expire before the PCC has redelegated the LSPs to another PCE, for example if a PCC is not connected to any active stateful PCE or if no connected active stateful PCE accepts the delegation. In this case, the PCC MUST flush any LSP state set by the PCE upon expiration of the State Timeout Interval and revert to operator-defined default parameters or behaviors. This operation SHOULD be done in a make-before-break fashion.

The State Timeout Interval MUST be greater than or equal to the Redelegation Timeout Interval and MAY be set to infinity (meaning that until the PCC specifically takes action to change the parameters set by the PCE, they will remain intact).

5.7.3. Returning a Delegation

In order to keep a delegation, a PCE MUST set the Delegate flag to 1 on each LSP Update Request sent to the PCC. A PCE that no longer wishes to update an LSP's parameters SHOULD return the LSP delegation back to the PCC by sending an empty LSP Update Request which has the Delegate flag set to 0. If a PCC receives an LSP Update Request with the Delegate flag set to 0 (whether the LSP Update Request is empty or not), it MUST treat this as a delegation return.

```
+-+-+                     +-+-+
|PCC|                     |PCE|
+-+-+                     +-+-+
  |                         |
  |---PCRpt, Delegate=1--->| LSP delegated
  |            .            |
  |            .            |
  |            .            |
  |<--PCUpd, Delegate=0----| Delegation returned
  |                         |
  |---PCRpt, Delegate=0--->| No delegation for LSP
  |                         |
```

Figure 6: Returning a Delegation

If a PCC cannot delegate an LSP to a PCE (for example, if a PCC is not connected to any active stateful PCE or if no connected active stateful PCE accepts the delegation), the LSP delegation on the PCC will time out within a configurable Redelegation Timeout Interval and the PCC MUST flush any LSP state set by a PCE at the expiration of the State Timeout Interval and revert to operator-defined default parameters or behaviors.

5.7.4. Redundant Stateful PCEs

In a redundant configuration where one PCE is backing up another PCE, the backup PCE may have only a subset of the LSPs in the network delegated to it. The backup PCE does not update any LSPs that are not delegated to it. In order to allow the backup to operate in a hot-standby mode and avoid the need for state synchronization in case the primary fails, the backup receives all LSP State Reports from a PCC. When the primary PCE for a given LSP set fails, after expiry of the Redelegation Timeout Interval, the PCC SHOULD delegate to the redundant PCE all LSPs that had been previously delegated to the failed PCE. Assuming that the State Timeout Interval had been configured to be greater than the Redelegation Timeout Interval (as MANDATORY), and assuming that the primary and redundant PCEs take

similar decisions, this delegation change will not cause any changes to the LSP parameters.

5.7.5. Redelegation on PCE Failure

On failure, the goal is to: 1) avoid any traffic loss on the LSPs that were updated by the PCE that crashed 2) minimize the churn in the network in terms of ownership of the LSPs, 3) not leave any "orphan" (undelegated) LSPs and 4) be able to control when the state that was set by the PCE can be changed or purged. The values chosen for the Redelegation Timeout and State Timeout values affect the ability to accomplish these goals.

This section summarizes the behaviour with regards to LSP delegation and LSP state on a PCE failure.

If the PCE crashes but recovers within the Redelegation Timeout, both the delegation state and the LSP state are kept intact.

If the PCE crashes but does not recover within the Redelegation Timeout, the delegation state is returned to the PCC. If the PCC can redelegate the LSPs to another PCE, and that PCE accepts the delegations, there will be no change in LSP state. If the PCC cannot redelegate the LSPs to another PCE, then upon expiration of the State Timeout Interval, the state set by the PCE is removed and the LSP reverts to operator-defined parameters, which may cause a change in the LSP state. Note that an operator may choose to use an infinite State Timeout Interval if he wishes to maintain the PCE state indefinitely. Note also that flushing the state should be implemented using make-before-break to avoid traffic loss.

If there is a standby PCE, the Redelegation Timeout may be set to 0 through policy on the PCC, causing the LSPs to be redelegated immediately to the PCC, which can delegate them immediately to the standby PCE. Assuming that the PCC can redelegate the LSP to the standby PCE within the State Timeout Interval, and assuming the standby PCE takes similar decisions as the failed PCE, the LSP state will be kept intact.

5.8. LSP Operations

5.8.1. Passive Stateful PCE Path Computation Request/Response

```
                      +-+-+                    +-+-+
                      |PCC|                    |PCE|
                      +-+-+                    +-+-+
                        |                        |
   1) Path computation  |----- PCReq message --->|
      request sent to   |                        |2) Path computation
      PCE               |                        |   request received,
                        |                        |   path computed
                        |                        |
                        |<---- PCRep message ----|3) Computed paths
                        |     (Positive reply)   |   sent to the PCC
                        |     (Negative reply)   |
   4) LSP State change  |                        |
      event             |                        |
                        |                        |
   5) LSP State Report  |----- PCRpt message --->|
      sent to all       |            .           |
      stateful PCEs     |            .           |
                        |            .           |
   6) Repeat for each   |----- PCRpt message --->|
      LSP state change  |                        |
                        |                        |
```

Figure 7: Passive Stateful PCE Path Computation Request/Response

Once a PCC has successfully established a PCEP session with a passive stateful PCE and the PCC’s LSP state is synchronized with the PCE (i.e. the PCE knows about all PCC’s existing LSPs), if an event is triggered that requires the computation of a set of paths, the PCC sends a path computation request to the PCE ([RFC5440], Section 4.2.3). The PCReq message MAY contain the LSP Object to identify the LSP for which the path computation is requested.

Upon receiving a path computation request from a PCC, the PCE triggers a path computation and returns either a positive or a negative reply to the PCC ([RFC5440], Section 4.2.4).

Upon receiving a positive path computation reply, the PCC receives a set of computed paths and starts to setup the LSPs. For each LSP, it MAY send an LSP State Report carried on a PCRpt message to the PCE, indicating that the LSP’s status is "Going-up".

Once an LSP is up or active, the PCC MUST send an LSP State Report carried on a PCRpt message to the PCE, indicating that the LSP’s status is ’Up’ or ’Active’ respectively. If the LSP could not be set up, the PCC MUST send an LSP State Report indicating that the LSP is "Down’ and stating the cause of the failure. Note that due to timing constraints, the LSP status may change from ’Going-up’ to ’Up’ (or

'Down') before the PCC has had a chance to send an LSP State Report indicating that the status is 'Going-up'. In such cases, the PCC MAY choose to only send the PCRpt indicating the latest status ('Active', 'Up' or 'Down').

Upon receiving a negative reply from a PCE, a PCC MAY resend a modified request or take any other appropriate action. For each requested LSP, it SHOULD also send an LSP State Report carried on a PCRpt message to the PCE, indicating that the LSP's status is 'Down'.

There is no direct correlation between PCRep and PCRpt messages. For a given LSP, multiple LSP State Reports will follow a single PCRep message, as a PCC notifies a PCE of the LSP's state changes.

A PCC MUST send each LSP State Report to each stateful PCE that is connected to the PCC.

Note that a single PCRpt message MAY contain multiple LSP State Reports.

The passive stateful model for stateful PCEs is described in [RFC4655], Section 6.8.

5.8.2. Switching from Passive Stateful to Active Stateful

This section deals with the scenario of an LSP transitioning from a passive stateful to an active stateful mode of operation. When the LSP has no working path, prior to delegating the LSP, the PCC MUST first use the procedure defined in Section 5.8.1 to request the initial path from the PCE. This is required because the action of delegating the LSP to a PCE using a PCRpt message is not an explicit request to the PCE to compute a path for the LSP. The only explicit way for a PCC to request a path from PCE is to send a PCReq message. The PCRpt message MUST NOT be used by the PCC to attempt to request a path from the PCE.

When the LSP is delegated after its setup, it may be useful for the PCC to communicate to the PCE the locally configured intended configuration parameters, so that the PCE may reuse them in its computations. Such parameters MAY be acquired through an out of band channel, or MAY be communicated in the PCRpt message delegating the LSPs, by including them as part of the intented-attribute-list as explained in Section 6.1. An implementation MAY allow policies on the PCC to determine the configuration parameters to be sent to the PCE.

5.8.3. Active Stateful PCE LSP Update

```
                      +-+-+                    +-+-+
                      |PCC|                    |PCE|
                      +-+-+                    +-+-+
                        |                        |
   1) LSP State         |-- PCRpt, Delegate=1 -->|
      Synchronization   |            .           |
                        |            .           |2) PCE decides to
                        |            .           |   update the LSP
                        |                        |
                        |<---- PCUpd message ----|3) PCUpd message sent
                        |                        |   to PCC
                        |                        |
                        |                        |
   4) LSP State Report  |---- PCRpt message ---->|
      sent(->Going-up)  |            .           |
                        |            .           |
                        |            .           |
   5) LSP State Report  |---- PCRpt message ---->|
      sent (->Up|Down)  |                        |
                        |                        |
```

Figure 8: Active Stateful PCE

Once a PCC has successfully established a PCEP session with an active stateful PCE, the PCC’s LSP state is synchronized with the PCE (i.e. the PCE knows about all PCC’s existing LSPs). After LSPs have been delegated to the PCE, the PCE can modify LSP parameters of delegated LSPs.

To update an LSP, a PCE MUST send the PCC an LSP Update Request using a PCUpd message. The LSP Update Request contains a variety of objects that specify the set of constraints and attributes for the LSP’s path. Each LSP Update Request MUST have a unique identifier, the SRP-ID-number, carried in the SRP (Stateful PCE Request Parameters) Object described in Section 7.2. The SRP-ID-number is used to correlate errors and state reports to LSP Update Requests. A single PCUpd message MAY contain multiple LSP Update Requests.

Upon receiving a PCUpd message the PCC starts to setup LSPs specified in LSP Update Requests carried in the message. For each LSP, it MAY send an LSP State Report carried on a PCRpt message to the PCE, indicating that the LSP’s status is ’Going-up’. If the PCC decides that the LSP parameters proposed in the PCUpd message are unacceptable, it MUST report this error by including the LSP-ERROR-CODE TLV (Section 7.3.3) with LSP error-value="Unacceptable parameters" in the LSP object in the PCRpt message to the PCE. Based

on local policy, it MAY react further to this error by revoking the delegation. If the PCC receives a PCUpd message for an LSP object identified with a PLSP-ID that does not exist on the PCC, it MUST generate a PCErr with error-type 19 (Invalid Operation), error-value 3, (Attempted LSP Update Request for an LSP identified by an unknown PSP-ID) (see Section 8.5).

Once an LSP is up, the PCC MUST send an LSP State Report (PCRpt message) to the PCE, indicating that the LSP's status is 'Up'. If the LSP could not be set up, the PCC MUST send an LSP State Report indicating that the LSP is 'Down' and stating the cause of the failure. A PCC MAY compress LSP State Reports to only reflect the most up to date state, as discussed in the previous section.

A PCC MUST send each LSP State Report to each stateful PCE that is connected to the PCC.

PCErr and PCRpt messages triggered as a result of a PCUpd message MUST include the SRP-ID-number from the PCUpd. This provides correlation of requests and errors and acknowledgement of state processing. The PCC MAY compress state when processing PCUpd. In this case, receipt of a higher SRP-ID-number implicitly acknowledges processing all the updates with lower SRP-ID-number for the specific LSP (as per Section 7.2).

A PCC MUST NOT send to any PCE a Path Computation Request for a delegated LSP. Should the PCC decide it wants to issue a Path Computation Request on a delegated LSP, it MUST perform Delegation Revocation procedure first.

5.9. LSP Protection

LSP protection and interaction with stateful PCE, as well as the extensions necessary to implement this functionality will be discussed in a separate document.

5.10. PCEP Sessions

A permanent PCEP session MUST be established between a stateful PCE and the PCC. In the case of session failure, session reestablishment MUST be re-attempted per the procedures defined in [RFC5440].

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. For each PCEP message type, a set of rules is defined that specify the set of objects that the message can carry.

6.1. The PCRpt Message

A Path Computation LSP State Report message (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 PCRpt message can carry more than one LSP State Reports. A PCC can send an LSP State Report either in response to an LSP Update Request from a PCE, or asynchronously when the state of an LSP changes. The Message-Type field of the PCEP common header for the PCRpt message is 10.

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> ::= [<SRP>]
                   <LSP>
                   <path>
```

Where:

```
<path>::= <intended-path>
          [<actual-attribute-list><actual-path>]
          <intended-attribute-list>

<actual-attribute-list>::=[<BANDWIDTH>]
                           [<metric-list>]
```

Where:

<intended-path> is represented by the ERO object defined in section 7.9 of [RFC5440].

<actual-attribute-list> consists of the actual computed and signaled values of the <BANDWIDTH> and <metric-lists> objects defined in [RFC5440].

<actual-path> is represented by the RRO object defined in section 7.10 of [RFC5440].

<intended-attribute-list> is the attribute-list defined in section 6.5 of [RFC5440] and extended by PCEP extensions.

The SRP object (see Section 7.2) is OPTIONAL. If the PCRpt message is not in response to a PCupd message, the SRP object MAY be omitted. When the PCC does not include the SRP object, the PCE MUST treat this as an SRP object with an SRP-ID-number equal to the reserved value 0x00000000. The reserved value 0x00000000 indicates that the state reported is not as a result of processing a PCUpd message.

If the PCRpt message is in response to a PCUpd message, the SRP object MUST be included and the value of the SRP-ID-number in the SRP Object MUST be the same as that sent in the PCUpd message that triggered the state that is reported. If the PCC compressed several PCUpd messages for the same LSP by only processing the one with the highest number, then it should use the SRP-ID-number of that request. No state compression is allowed for state reporting, e.g. PCRpt messages MUST NOT be pruned from the PCC's egress queue even if subsequent operations on the same LSP have been completed before the PCRpt message has been sent to the TCP stack. The PCC MUST explicitly report state changes (including removal) for paths it manages.

The LSP object (see Section 7.3) is REQUIRED, and it MUST be included in each LSP State Report on the PCRpt message. If the LSP object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value 8 (LSP object missing).

If the LSP transitioned to non-operational state, the PCC SHOULD include the LSP-ERROR-TLV (Section 7.3.3) with the relevant LSP Error Code to report the error to the PCE.

The intended path, represented by the ERO object, is REQUIRED. If the ERO object is missing, the receiving PCE MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value 9 (ERO object missing). The ERO may be empty if the PCE does not have a path for a delegated LSP.

The actual path, represented by the RRO object, SHOULD be included in PCRpt by the PCC when the path is up or active, but MAY be omitted if the path is down due to a signaling error or another failure.

The intended-attribute-list maps to the attribute-list in Section 6.5 of [RFC5440] and is used to convey the requested parameters of the LSP path. This is needed in order to support the switch from passive to active stateful PCE as described in Section 5.8.2. When included as part of the intended-attribute-list, the meaning of the BANDWIDTH object is the requested bandwidth as intended by the operator. In this case, the BANDWIDTH Object-Type of 1 SHOULD be used. Similarly, to indicate a limiting constraint, the METRIC object SHOULD be included as part of the intended-attribute-list with the B flag set and with a specific metric value. To indicate the optimization metric, the METRIC object SHOULD be included as part of the intended-attribute-list with the B flag unset and the metric value set to zero. Note that the intended-attribute-list is optional and thus may be omitted. In this case, the PCE MAY use the values in the actual-attribute-list as the requested parameters for the path.

The actual-attribute-list consists of the actual computed and signaled values of the BANDWIDTH and METRIC objects defined in [RFC5440]. When included as part of the actual-attribute-list, Object-Type 2 ([RFC5440]) SHOULD be used for the BANDWIDTH object and the C flag SHOULD be set in the METRIC object ([RFC5440]).

Note that the ordering of intended-path, actual-attribute-list, actual-path and intended-attribute-list is chosen to retain compatibility with implementations of an earlier version of this standard.

A PCE may choose to implement a limit on the resources a single PCC can occupy. If a PCRpt is received that causes the PCE to exceed this limit, the PCE MUST notify the PCC using a PCNtf message with Notification Type 4 (Stateful PCE resource limit exceeded) and Notification Value 1 (Entering resource limit exceeded state) and MUST terminate the session.

6.2. The PCUpd Message

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. A PCUpd message can carry more than one LSP Update Request. The Message-Type field of the PCEP common header for the PCUpd message is 11.

The format of a PCUpd message is as follows:

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

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

    <update-request> ::= <SRP>
                         <LSP>
                         <path>
 Where:
    <path>::= <intended-path><intended-attribute-list>

 Where:
    <intended-path> is represented by the ERO object defined in
    section 7.9 of [RFC5440].
    <intended-attribute-list> is the attribute-list defined in [RFC5440]
    and extended by PCEP extensions.
```

There are three mandatory objects that MUST be included within each LSP Update Request in the PCUpd message: the SRP Object (see

Section 7.2), the LSP object (see Section 7.3) and the ERO object (as defined in [RFC5440], which represents the intended path. If the SRP object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=10 (SRP object missing). If the LSP object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=8 (LSP object missing). If the ERO object is missing, the receiving PCC MUST send a PCErr message with Error-type=6 (Mandatory Object missing) and Error-value=9 (ERO object missing).

The ERO in the PCUpd may be empty if the PCE cannot find a valid path for a delegated LSP. One typical situation resulting in this empty ERO carried in the PCUpd message is that a PCE can no longer find a strict SRLG-disjoint path for a delegated LSP after a link failure. The PCC SHOULD implement a local policy to decide the appropriate action to be taken: either tear down the LSP, or revoke the delegation and use a locally computed path, or keep the existing LSP.

A PCC only acts on an LSP Update Request if permitted by the local policy configured by the network manager. Each LSP Update Request that the PCC acts on results in an LSP setup operation. An LSP Update Request MUST contain all LSP parameters that a PCE wishes to be set for the LSP. A PCC MAY set missing parameters from locally configured defaults. If the LSP specified in the Update Request is already up, it will be re-signaled.

The PCC SHOULD minimize the traffic interruption, and MAY use the make-before-break procedures described in [RFC3209] in order to achieve this goal. If the make-before-break procedures are used, two paths will briefly co-exist. The PCC MUST send separate PCRpt messages for each, identified by the LSP-IDENTIFIERS TLV. When the old path is torn down after the head end switches over the traffic, this event MUST be reported by sending a PCRpt message with the LSP-IDENTIFIERS-TLV of the old path and the R bit set. The SRP-ID-number that the PCC associates with this PCRpt MUST be 0x00000000. Thus, a make-before-break operation will typically result in at least two PCRpt messages, one for the new path and one for the removal of the old path (more messages may be possible if intermediate states are reported).

If the path setup fails due to an RSVP signaling error, the error is reported to the PCE. The PCC will not attempt to resignal the path until it is prompted again by the PCE with a subsequent PCUpd message.

A PCC MUST respond with an LSP State Report to each LSP Update Request it processed to indicate the resulting state of the LSP in

the network (even if this processing did not result in changing the state of the LSP). The SRP-ID-number included in the PCRpt MUST match that in the PCUpd. A PCC MAY respond with multiple LSP State Reports to report LSP setup progress of a single LSP. In that case, the SRP-ID-number MUST be included for the first message, for subsequent messages the reserved value 0x00000000 SHOULD be used.

Note that a PCC MUST process all LSP Update Requests - for example, an LSP Update Request is sent when a PCE returns delegation or puts an LSP into non-operational state. The protocol relies on TCP for message-level flow control.

If the rate of PCUpd messages sent to a PCC for the same target LSP exceeds the rate at which the PCC can signal LSPs into the network, the PCC MAY perform state compression on its ingress queue. The compression algorithm is based on the fact that each PCUpd request contains the complete LSP state the PCE wishes to be set and works as follows: when the PCC starts processing a PCUpd message at the head of its ingress queue, it may search the queue forward for more recent PCUpd messages pertaining that particular LSP, prune all but the latest one from the queue and process only the last one as that request contains the most up-to-date desired state for the LSP. The PCC MUST NOT send PCRpt nor PCErr messages for requests which were pruned from the queue in this way. This compression step may be performed only while the LSP is not being signaled, e.g. if two PCUpd arrive for the same LSP in quick succession and the PCC started the signaling of the changes relevant to the first PCUpd, then it MUST wait until the signaling finishes (and report the new state via a PCRpt) before attempting to apply the changes indicated in the second PCUpd.

Note also that it is up to the PCE to handle inter-LSP dependencies; for example, if ordering of LSP set-ups is required, the PCE has to wait for an LSP State Report for a previous LSP before starting the update of the next LSP.

If the PCUpd cannot be satisfied (for example due to unsupported object or TLV), the PCC MUST respond with a PCErr message indicating the failure (see Section 7.3.3).

6.3. The PCErr Message

If the stateful PCE capability has been advertised on the PCEP session, the PCErr message MAY include the SRP object. If the error reported is the result of an LSP update request, then the SRP-ID-number MUST be the one from the PCUpd that triggered the error. If the error is unsolicited, the SRP object MAY be omitted. This is

equivalent to including an SRP object with SRP-ID-number equal to the reserved value 0x00000000.

The format of a PCErr message from [RFC5440] is extended as follows:

```
<PCErr Message> ::= <Common Header>
                  ( <error-obj-list> [<Open>] ) | <error>
                  [<error-list>]

<error-obj-list>::=<PCEP-ERROR>[<error-obj-list>]

<error>::=[<request-id-list> | <stateful-request-id-list>]
           <error-obj-list>

<request-id-list>::=<RP>[<request-id-list>]

<stateful-request-id-list>::=<SRP>[<stateful-request-id-list>]

<error-list>::=<error>[<error-list>]
```

6.4. The PCReq Message

A PCC MAY include the LSP object in the PCReq message (see Section 7.3) if the stateful PCE capability has been negotiated on a PCEP session between the PCC and a PCE.

The definition of the PCReq message from [RFC5440] is extended to optionally include the LSP object after the END-POINTS object. The encoding from [RFC5440] will become:

```
<PCReq Message>::= <Common Header>
                   [<svec-list>]
                   <request-list>
```

Where:

```
      <svec-list>::=<SVEC>[<svec-list>]
      <request-list>::=<request>[<request-list>]

      <request>::= <RP>
                   <END-POINTS>
                   [<LSP>]
                   [<LSPA>]
                   [<BANDWIDTH>]
                   [<metric-list>]
                   [<RRO>[<BANDWIDTH>]]
                   [<IRO>]
                   [<LOAD-BALANCING>]
```

6.5. The PCRep Message

A PCE MAY include the LSP object in the PCRep message (see (Section 7.3) if the stateful PCE capability has been negotiated on a PCEP session between the PCC and the PCE and the LSP object was included in the corresponding PCReq message from the PCC.

The definition of the PCRep message from [RFC5440] is extended to optionally include the LSP object after the RP object. The encoding from [RFC5440] will become:

```
<PCRep Message> ::= <Common Header>
                    <response-list>
```

Where:

```
<response-list>::=<response>[<response-list>]

<response>::=<RP>
             [<LSP>]
             [<NO-PATH>]
             [<attribute-list>]
             [<path-list>]
```

7. Object Formats

The PCEP objects defined in this document are compliant with the PCEP object format defined in [RFC5440]. The P flag and the I flag of the PCEP objects defined in the current document MUST be set to 0 on transmission and SHOULD be ignored on receipt since the P and I flags are exclusively related to path computation requests.

7.1. OPEN Object

This document defines one new optional TLV for use in the OPEN Object.

7.1.1. Stateful PCE Capability TLV

The STATEFUL-PCE-CAPABILITY TLV is an optional TLV for use in the OPEN Object for stateful PCE capability advertisement. Its format is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Type=16         |            Length=4           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                           |U|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 9: STATEFUL-PCE-CAPABILITY TLV format

The type (16 bits) of the TLV is 16. The length field is 16 bit-long and has a fixed value of 4.

The value comprises a single field - Flags (32 bits):

U (LSP-UPDATE-CAPABILITY - 1 bit): if set to 1 by a PCC, the U Flag indicates that the PCC allows modification of LSP parameters; if set to 1 by a PCE, the U Flag indicates that the PCE is capable of updating LSP parameters. The LSP-UPDATE-CAPABILITY Flag must be advertised by both a PCC and a PCE for PCUpd messages to be allowed on a PCEP session.

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

A PCEP speaker operating in passive stateful PCE mode advertises the stateful PCE capability with the U flag set to 0. A PCEP speaker operating in active stateful PCE mode advertises the stateful PCE capability with the U Flag set to 1.

Advertisement of the stateful PCE capability implies support of LSPs that are signaled via RSVP, as well as the objects, TLVs and procedures defined in this document.

7.2. SRP Object

The SRP (Stateful PCE Request Parameters) object MUST be carried within PCUpd messages and MAY be carried within PCRpt and PCErr messages. The SRP object is used to correlate between update requests sent by the PCE and the error reports and state reports sent by the PCC.

SRP Object-Class is 33.

SRP Object-Type is 1.

The format of the SRP object body is shown in Figure 10:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Flags                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         SRP-ID-number                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                       Optional TLVs                         //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 10: The SRP Object format

The SRP object body has a variable length and may contain additional TLVs.

Flags (32 bits): None defined yet.

SRP-ID-number (32 bits): The SRP-ID-number value in the scope of the current PCEP session uniquely identify the operation that the PCE has requested the PCC to perform on a given LSP. The SRP-ID-number is incremented each time a new request is sent to the PCC, and may wrap around.

The values 0x00000000 and 0xFFFFFFFF are reserved.

Optional TLVs MAY be included within the SRP object body. The specification of such TLVs is outside the scope of this document.

Every request to update an LSP receives a new SRP-ID-number. This number is unique per PCEP session and is incremented each time an operation is requested from the PCE. Thus, for a given LSP there may be more than one SRP-ID-number unacknowledged at a given time. The value of the SRP-ID-number is echoed back by the PCC in PCErr and PCRpt messages to allow for correlation between requests made by the PCE and errors or state reports generated by the PCC. If the error or report were not as a result of a PCE operation (for example in the case of a link down event), the reserved value of 0x00000000 is used for the SRP-ID-number. The absence of the SRP object is equivalent to an SRP object with the reserved value of 0x00000000. An SRP-ID-number is considered unacknowledged and cannot be reused until a PCErr or PCRpt arrives with an SRP-ID-number equal or higher for the same LSP. In case of SRP-ID-number wrapping the last SRP-ID-number before the wrapping MUST be explicitly acknowledged, to avoid a situation where SRP-ID-numbers remain unacknowledged after the wrap.

This means that the PCC may need to issue two PCUpd messages on detecting a wrap.

7.3. LSP Object

The LSP object MUST be present within PCRpt and PCUpd messages. The LSP object MAY be carried within PCReq and PCRep messages if the stateful PCE capability has been negotiated on the session. The LSP object contains a set of fields used to specify the target LSP, the operation to be performed on the LSP, and LSP Delegation. It also contains a flag indicating to a PCE that the LSP state synchronization is in progress. This document focuses on LSPs that are signaled with RSVP, many of the TLVs used with the LSP object mirror RSVP state.

LSP Object-Class is 32.

LSP Object-Type is 1.

The format of the LSP object body is shown in Figure 11:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                PLSP-ID                |    Flag |  O  |A|R|S|D|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        TLVs                                  //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 11: The LSP Object format

PLSP-ID (20 bits): A PCEP-specific identifier for the LSP. A PCC creates a unique PLSP-ID for each LSP that is constant for the lifetime of a PCEP session. The PCC will advertise the same PLSP-ID on all PCEP sessions it maintains at a given times. The mapping of the Symbolic Path Name to PLSP-ID is communicated to the PCE by sending a PCRpt message containing the SYMBOLIC-PATH-NAME TLV. All subsequent PCEP messages then address the LSP by the PLSP-ID. The values of 0 and 0xFFFFF are reserved. Note that the PLSP-ID is a value that is constant for the lifetime of the PCEP session, during which time for an RSVP-signaled LSP there might be a different RSVP identifiers (LSP-id, tunnel-id) allocated to it.

Flags (12 bits), starting from the least significant bit:

D (Delegate - 1 bit): On a PCRpt message, the D Flag set to 1 indicates that the PCC is delegating the LSP to the PCE. On a

PCUpd message, the D flag set to 1 indicates that the PCE is confirming the LSP Delegation. To keep an LSP delegated to the PCE, the PCC must set the D flag to 1 on each PCRpt message for the duration of the delegation - the first PCRpt with the D flag set to 0 revokes the delegation. To keep the delegation, the PCE must set the D flag to 1 on each PCUpd message for the duration of the delegation - the first PCUpd with the D flag set to 0 returns the delegation.

S (SYNC - 1 bit): The S Flag MUST be set to 1 on each PCRpt sent from a PCC during State Synchronization. The S Flag MUST be set to 0 in other messages sent from the PCC. When sending a PCUpd message, the PCE MUST set the S Flag to 0.

R(Remove - 1 bit): On PCRpt messages the R Flag indicates that the LSP has been removed from the PCC and the PCE SHOULD remove all state from its database. Upon receiving an LSP State Report with the R Flag set to 1 for an RSVP-signaled LSP, the PCE SHOULD remove all state for the path identified by the LSP-IDENTIFIERS TLV from its database. When the all-zeros LSP-IDENTIFIERS TLV is used, the PCE SHOULD remove all state for the PLSP-ID from its database. When sending a PCUpd message, the PCE MUST set the R Flag to 0.

A(Administrative - 1 bit): On PCRpt messages, the A Flag indicates the PCC’s target operational status for this LSP. On PCUpd messages, the A Flag indicates the LSP status that the PCE desires for this LSP. In both cases, a value of ’1’ means that the desired operational state is active, and a value of ’0’ means that the desired operational state is inactive. A PCC ignores the A flag on a PCUpd message unless the operator’s policy allows the PCE to control the corresponding LSP’s administrative state.

O(Operational - 3 bits): On PCRpt messages, the O Field represents the operational status of the LSP.

The following values are defined:

0 - DOWN: not active.

1 - UP: signalled.

2 - ACTIVE: up and carrying traffic.

3 - GOING-DOWN: LSP is being torn down, resources are being released.

4 - GOING-UP: LSP is being signalled.

5-7 - Reserved: these values are reserved for future use.

Unassigned bits are considered reserved. They MUST be set to 0 on transmission and MUST be ignored on receipt. When sending a PCUpd message, the PCE MUST set the O Field to 0.

TLVs that may be included in the LSP Object are described in the following sections. Other optional TLVs, that are not defined in this document, MAY also be included within the LSP Object body.

7.3.1. LSP-IDENTIFIERS TLVs

The LSP-IDENTIFIERS TLV MUST be included in the LSP object in PCRpt messages for RSVP-signaled LSPs. If the TLV is missing, the PCE will generate an error with error-type 6 (mandatory object missing) and error-value 11 (LSP-IDENTIFIERS TLV missing) and close the session. The LSP-IDENTIFIERS TLV MAY be included in the LSP object in PCUpd messages for RSVP-signaled LSPs. The special value of all zeros for this TLV is used to refer to all paths pertaining to a particular PLSP-ID. There are two LSP-IDENTIFIERS TLVs, one for IPv4 and one for IPv6.

It is the responsibility of the PCC to send to the PCE the identifiers for each RSVP incarnation of the tunnel. For example, in a make-before-break scenario, the PCC MUST send a separate PCRpt for the old and for the reoptimized paths, and explicitly report removal of any of these paths using the R bit in the LSP object.

The format of the IPV4-LSP-IDENTIFIERS TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=18             |           Length=16           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    IPv4 Tunnel Sender Address                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             LSP ID            |           Tunnel ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Extended Tunnel ID                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    IPv4 Tunnel Endpoint Address               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 12: IPV4-LSP-IDENTIFIERS TLV format

The type (16 bits) of the TLV is 18. The length field is 16 bit-long and has a fixed value of 16. The value contains the following fields:

IPv4 Tunnel Sender Address: contains the sender node's IPv4 address, as defined in [RFC3209], Section 4.6.2.1 for the LSP_TUNNEL_IPv4 Sender Template Object.

LSP ID: contains the 16-bit 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.1 for the LSP_TUNNEL_IPv4 Sender Template Object. A value of 0 MUST be used if the LSP is not yet signaled.

Tunnel ID: contains the 16-bit 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP_TUNNEL_IPv4 Session Object.

Extended Tunnel ID: contains the 32-bit 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP_TUNNEL_IPv4 Session Object.

IPv4 Tunnel Endpoint Address: contains the egress node's IPv4 address, as defined in [RFC3209], Section 4.6.1.1 for the LSP_TUNNEL_IPv4 Sender Template Object.

The format of the IPV6-LSP-IDENTIFIERS TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=19             |           Length=52           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                  IPv6 tunnel sender address                   |
+                          (16 octets)                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             LSP ID            |           Tunnel ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                       Extended Tunnel ID                      |
+                          (16 octets)                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                  IPv6 tunnel endpoint address                 |
+                          (16 octets)                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 13: IPV6-LSP-IDENTIFIERS TLV format

The type (16 bits) of the TLV is 19. The length field is 16 bit-long and has a fixed value of 52. The value contains the following fields:

IPv6 Tunnel Sender Address: contains the sender node's IPv6 address, as defined in [RFC3209], Section 4.6.2.2 for the LSP_TUNNEL_IPv6 Sender Template Object.

LSP ID: contains the 16-bit 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.2 for the LSP_TUNNEL_IPv6 Sender Template Object. A value of 0 MUST be used if the LSP is not yet signaled.

Tunnel ID: contains the 16-bit 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP_TUNNEL_IPv6 Session Object.

Extended Tunnel ID: contains the 128-bit 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP_TUNNEL_IPv6 Session Object.

IPv6 Tunnel Endpoint Address: contains the egress node's IPv6 address, as defined in [RFC3209], Section 4.6.1.2 for the LSP_TUNNEL_IPv6 Session Object.

The Tunnel ID remains constant over the life time of a tunnel.

7.3.2. Symbolic Path Name TLV

Each LSP MUST have a symbolic path name that is unique in the PCC. The symbolic path name is a human-readable string that identifies an LSP in the network. The symbolic path name MUST remain constant throughout an LSP's lifetime, which may span across multiple consecutive PCEP sessions and/or PCC restarts. The symbolic path name MAY be specified by an operator in a PCC's configuration. If the operator does not specify a unique symbolic name for an LSP, then the PCC MUST auto-generate one.

The PCE uses the symbolic path name as a stable identifier for the LSP. If the PCEP session restarts, or the PCC restarts, or the PCC re-delegates the LSP to a different PCE, the symbolic path name for the LSP remains constant and can be used to correlate across the PCEP session instances.

The other protocol identifiers for the LSP cannot reliably be used to identify the LSP across multiple PCEP sessions, for the following reasons.

- o The PLSP-ID is unique only within the scope of a single PCEP session.

- o The LSP-IDENTIFIERS TLV is only guaranteed to be present for LSPs that are signalled with RSVP-TE, and may change during the lifetime of the LSP.

The SYMBOLIC-PATH-NAME TLV MUST be included in the LSP object in the LSP State Report (PCRpt) message when during a given PCEP session an LSP is first reported to a PCE. A PCC sends to a PCE the first LSP State Report either during State Synchronization, or when a new LSP is configured at the PCC.

The initial PCRpt creates a binding between the symbolic path name and the PLSP-ID for the LSP which lasts for the duration of the PCEP session. The PCC MAY omit the symbolic path name from subsequent LSP

State Reports for that LSP on that PCEP session, and just use the PLSP-ID.

The format of the SYMBOLIC-PATH-NAME TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=17             |       Length (variable)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                      Symbolic Path Name                     //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 14: SYMBOLIC-PATH-NAME TLV format

Type (16 bits): The type is 17.

Length (16 bits): indicates the total length of the TLV in octets and MUST be greater than 0. The TLV MUST be zero-padded so that the TLV is 4-octet aligned.

Symbolic Path Name (variable): symbolic name for the LSP, unique in the PCC. It SHOULD be a string of printable ASCII characters, without a NULL terminator.

7.3.3. LSP Error Code TLV

The LSP Error code TLV is an optional TLV for use in the LSP object to convey error information. When an LSP Update Request fails, an LSP State Report MUST be sent to report the current state of the LSP, and SHOULD contain the LSP-ERROR-CODE TLV indicating the reason for the failure. Similarly, when a PCRpt is sent as a result of an LSP transitioning to non-operational state, the LSP-ERROR-CODE TLV SHOULD be included to indicate the reason for the transition.

The format of the LSP-ERROR-CODE TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=20             |            Length=4           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          LSP Error Code                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 15: LSP-ERROR-CODE TLV format

The type (16 bits) of the TLV is 20. The length field is 16 bit-long and has a fixed value of 4. The value contains an error code that indicates the cause of the failure.

The following LSP Error Codes are currently defined:

Value	Meaning
1	Unknown reason
2	Limit reached for PCE-controlled LSPs
3	Too many pending LSP update requests
4	Unacceptable parameters
5	Internal error
6	LSP administratively brought down
7	LSP preempted
8	RSVP signaling error

7.3.4. RSVP Error Spec TLV

The RSVP-ERROR-SPEC TLV is an optional TLV for use in the LSP object to carry RSVP error information. It includes the RSVP ERROR_SPEC or USER_ERROR_SPEC Object ([RFC2205] and [RFC5284]) which were returned to the PCC from a downstream node. If the set up of an LSP fails at a downstream node which returned an ERROR_SPEC to the PCC, the PCC SHOULD include in the PCRpt for this LSP the LSP-ERROR-CODE TLV with LSP Error Code = "RSVP signaling error" and the RSVP-ERROR-SPEC TLV with the relevant RSVP ERROR-SPEC or USER_ERROR_SPEC Object.

The format of the RSVP-ERROR-SPEC TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=21             |            Length (variable)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                RSVP ERROR_SPEC or USER_ERROR_SPEC Object      +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 16: RSVP-ERROR-SPEC TLV format

Type (16 bits): The type is 21.

Length (16 bits): indicates the total length of the TLV in octets. The TLV MUST be zero-padded so that the TLV is 4-octet aligned.

Value (variable): contains the RSVP ERROR_SPEC or USER_ERROR_SPEC Object: as specified in [RFC2205] and [RFC5284], including the object header.

8. IANA Considerations

This document requests IANA actions to allocate code points for the protocol elements defined in this document.

8.1. PCE Capabilities in IGP Advertisements

IANA is requested to confirm the early allocation of the following bits in the OSPF Parameters "PCE Capability Flags" registry, and to update the reference in the registry to point to this document, when it is an RFC:

Bit	Meaning	Reference
11	Active Stateful PCE capability	This document
12	Passive Stateful PCE capability	This document

8.2. PCEP Messages

IANA is requested to confirm the early allocation of the following message types within the "PCEP Messages" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:

Value	Meaning	Reference
10	Report	This document
11	Update	This document

8.3. PCEP Objects

IANA is requested to confirm the early allocation of the following object-class values and object types within the "PCEP Objects" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:.

Object-Class Value	Name	Reference
32	LSP Object-Type 1	This document
33	SRP Object-Type 1	This document

8.4. LSP Object

This document requests that a new sub-registry, named "LSP Object Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field of the LSP object. New values are to be assigned by Standards Action [RFC5226]. 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
0-4	Reserved	This document
5-7	Operational (3 bits)	This document
8	Administrative	This document
9	Remove	This document
10	SYNC	This document
11	Delegate	This document

8.5. PCEP-Error Object

IANA is requested to confirm the early allocation of the following Error Types and Error Values within the "PCEP-ERROR Object Error Types and Values" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:

```
 Error-Type  Meaning
    6        Mandatory Object missing

              Error-value=8:  LSP Object missing
              Error-value=9:  ERO Object missing
              Error-value=10: SRP Object missing
              Error-value=11: LSP-IDENTIFIERS TLV missing
    19       Invalid Operation

              Error-value=1:  Attempted LSP Update Request for a non-
                              delegated LSP.  The PCEP-ERROR Object
                              is followed by the LSP Object that
                              identifies the LSP.
              Error-value=2:  Attempted LSP Update Request if the
                              stateful PCE capability was not
                              advertised.
              Error-value=3:  Attempted LSP Update Request for an LSP
                              identified by an unknown PLSP-ID.
              Error-value=5:  Attempted LSP State Report if stateful
                              PCE capability was not advertised.
    20       LSP State synchronization error.

              Error-value=1:  A PCE indicates to a PCC that it can
                              not process (an otherwise valid) LSP
                              State Report.  The PCEP-ERROR Object is
                              followed by the LSP Object that
                              identifies the LSP.
              Error-value=5:  A PCC indicates to a PCE that it can
                              not complete the state synchronization,
```

8.6. Notification Object

IANA is requested to confirm the early allocation of the following Notification Types and Notification Values within the "Notification Object" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:

Notification-Type	Meaning
4	Stateful PCE resource limit exceeded

Notification-value=1:	Entering resource limit exceeded state

Note to IANA: the early allocation included an additional Notification value 2 for "Exiting resource limit exceeded state". This Notification value is no longer required.

8.7. PCEP TLV Type Indicators

IANA is requested to confirm the early allocation of the following TLV Type Indicator values within the "PCEP TLV Type Indicators" sub-registry of the PCEP Numbers registry, and to update the reference in the registry to point to this document, when it is an RFC:

Value	Meaning	Reference
16	STATEFUL-PCE-CAPABILITY	This document
17	SYMBOLIC-PATH-NAME	This document
18	IPV4-LSP-IDENTIFIERS	This document
19	IPV6-LSP-IDENTIFIERS	This document
20	LSP-ERROR-CODE	This document
21	RSVP-ERROR-SPEC	This document

8.8. STATEFUL-PCE-CAPABILITY TLV

This document requests that a new sub-registry, named "STATEFUL-PCE-CAPABILITY TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the STATEFUL-PCE-CAPABILITY TLV of the PCEP OPEN object (class = 1). New values are to be assigned by Standards Action [RFC5226]. 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
31	LSP-UPDATE-CAPABILITY	This document

8.9. LSP-ERROR-CODE TLV

This document requests that a new sub-registry, named "LSP-ERROR-CODE TLV Error Code Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the LSP Error code field of the LSP-ERROR-CODE TLV. This field specifies the reason for failure to update the LSP.

New values are to be assigned by Standards Action [RFC5226]. Each value should be tracked with the following qualities: value, description and defining RFC. The following values are defined in this document:

Value	Meaning
1	Unknown reason
2	Limit reached for PCE-controlled LSPs
3	Too many pending LSP update requests
4	Unacceptable parameters
5	Internal error
6	LSP administratively brought down
7	LSP preempted
8	RSVP signaling error

9. Manageability Considerations

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

9.1. Control Function and Policy

In addition to configuring specific PCEP session parameters, as specified in [RFC5440], Section 8.1, a PCE or PCC implementation MUST allow configuring the stateful PCEP capability and the LSP Update capability. A PCC implementation SHOULD allow the operator to specify multiple candidate PCEs for and a delegation preference for each candidate PCE. A PCC SHOULD allow the operator to specify an LSP delegation policy where LSPs are delegated to the most-preferred online PCE. A PCC MAY allow the operator to specify different LSP delegation policies.

A PCC implementation which allows concurrent connections to multiple PCEs SHOULD allow the operator to group the PCEs by administrative domains and it MUST NOT advertise LSP existence and state to a PCE if the LSP is delegated to a PCE in a different group.

A PCC implementation SHOULD allow the operator to specify whether the PCC will advertise LSP existence and state for LSPs that are not

controlled by any PCE (for example, LSPs that are statically configured at the PCC).

A PCC implementation SHOULD allow the operator to specify both the Redelegation Timeout Interval and the State Timeout Interval. The default value of the Redelegation Timeout Interval SHOULD be set to 30 seconds. An operator MAY also configure a policy that will dynamically adjust the Redelegation Timeout Interval, for example setting it to zero when the PCC has an established session to a backup PCE. The default value for the State Timeout Interval SHOULD be set to 60 seconds.

After the expiration of the State Timeout Interval, the LSP reverts to operator-defined default parameters. A PCC implementation MUST allow the operator to specify the default LSP parameters. To achieve a behavior where the LSP retains the parameters set by the PCE until such time that the PCC makes a change to them, a State Timeout Interval of infinity SHOULD be used. Any changes to LSP parameters SHOULD be done in make-before-break fashion.

LSP Delegation is controlled by operator-defined policies on a PCC. LSPs are delegated individually - different LSPs may be delegated to different PCEs. An LSP is delegated to at most one PCE at any given point in time. A PCC implementation SHOULD support the delegation policy, when all PCC’s LSPs are delegated to a single PCE at any given time. Conversely, the policy revoking the delegation for all PCC’s LSPs SHOULD also be supported.

A PCC implementation SHOULD allow the operator to specify delegation priority for PCEs. This effectively defines the primary PCE and one or more backup PCEs to which primary PCE’s LSPs can be delegated when the primary PCE fails.

Policies defined for stateful PCEs and PCCs should eventually fit in the Policy-Enabled Path Computation Framework defined in [RFC5394], and the framework should be extended to support Stateful PCEs.

9.2. Information and Data Models

The PCEP YANG module [I-D.ietf-pce-pcep-yang] should include

- o advertised stateful capabilities and synchronization status per PCEP session

- o the delegation status of each configured LSP.

The PCEP MIB [RFC7420] could also be updated to include this information.

9.3. Liveness Detection and Monitoring

PCEP extensions defined in this document do not require any new mechanisms beyond those already defined in [RFC5440], Section 8.3.

9.4. Verifying Correct Operation

Mechanisms defined in [RFC5440], Section 8.4 also apply to PCEP extensions defined in this document. In addition to monitoring parameters defined in [RFC5440], a stateful PCC-side PCEP implementation SHOULD provide the following parameters:

- o Total number of LSP updates
- o Number of successful LSP updates
- o Number of dropped LSP updates
- o Number of LSP updates where LSP setup failed

A PCC implementation SHOULD provide a command to show for each LSP whether it is delegated, and if so, to which PCE.

A PCC implementation SHOULD allow the operator to manually revoke LSP delegation.

9.5. Requirements on Other Protocols and Functional Components

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

9.6. Impact on Network Operation

Mechanisms defined in [RFC5440], Section 8.6 also apply to PCEP extensions defined in this document.

Additionally, a PCEP implementation SHOULD allow a limit to be placed on the number of LSPs delegated to the PCE and on the rate of PCUpd and PCRpt messages sent by a PCEP speaker and processed from a peer. It SHOULD also allow sending a notification when a rate threshold is reached.

A PCC implementation SHOULD allow a limit to be placed on the rate of LSP Updates to the same LSP to avoid signaling overload discussed in Section 10.3.

10. Security Considerations

10.1. Vulnerability

This document defines extensions to PCEP to enable stateful PCEs. The nature of these extensions and the delegation of path control to PCEs results in more information being available for a hypothetical adversary and a number of additional attack surfaces which must be protected.

The security provisions described in [RFC5440] remain applicable to these extensions. However, because the protocol modifications outlined in this document allow the PCE to control path computation timing and sequence, the PCE defense mechanisms described in [RFC5440] section 7.2 are also now applicable to PCC security.

As a general precaution, 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) [I-D.ietf-pce-pceps], as per the recommendations and best current practices in [RFC7525].

The following sections identify specific security concerns that may result from the PCEP extensions outlined in this document along with recommended mechanisms to protect PCEP infrastructure against related attacks.

10.2. LSP State Snooping

The stateful nature of this extension explicitly requires LSP status updates to be sent from PCC to PCE. While this gives the PCE the ability to provide more optimal computations to the PCC, it also provides an adversary with the opportunity to eavesdrop on decisions made by network systems external to PCE. This is especially true if the PCC delegates LSPs to multiple PCEs simultaneously.

Adversaries may gain access to this information by eavesdropping on unsecured PCEP sessions, and might then use this information in various ways to target or optimize attacks on network infrastructure. For example by flexibly countering anti-DDoS measures being taken to protect the network, or by determining choke points in the network where the greatest harm might be caused.

PCC implementations which allow concurrent connections to multiple PCEs SHOULD allow the operator to group the PCEs by administrative domains and they MUST NOT advertise LSP existence and state to a PCE if the LSP is delegated to a PCE in a different group.

10.3. Malicious PCE

The LSP delegation mechanism described in this document allows a PCC to grant effective control of an LSP to the PCE for the duration of a PCEP session. While this enables PCE control of the timing and sequence of path computations within and across PCEP sessions, it also introduces a new attack vector: an attacker may flood the PCC with PCUpd messages at a rate which exceeds either the PCC's ability to process them or the network's ability to signal the changes, either by spoofing messages or by compromising the PCE itself.

A PCC is free to revoke an LSP delegation at any time without needing any justification. A defending PCC can do this by enqueueing the appropriate PCRpt message. As soon as that message is enqueued in the session, the PCC is free to drop any incoming PCUpd messages without additional processing.

10.4. Malicious PCC

A stateful session also results in an increased attack surface by placing a requirement for the PCE to keep an LSP state replica for each PCC. It is RECOMMENDED that PCE implementations provide a limit on resources a single PCC can occupy. A PCE implementing such a limit MUST send a PCNtf message with notification-type 4 (Stateful PCE resource limit exceeded) and notification-value 1 (Entering resource limit exceeded state) upon receiving an LSP state report causing it to exceed this threshold.

Delegation of LSPs can create further strain on PCE resources and a PCE implementation MAY preemptively give back delegations if it finds itself lacking the resources needed to effectively manage the delegation. Since the delegation state is ultimately controlled by the PCC, PCE implementations SHOULD provide throttling mechanisms to prevent strain created by flaps of either a PCEP session or an LSP delegation.

11. Contributing Authors

Xian Zhang
Huawei Technology
F3-5-B R&D Center
Huawei Industrial Base, Bantian, Longgang District
Shenzhen, Guangdong 518129
P.R.China
EMail: zhang.xian@huawei.com

Dhruv Dhody
Huawei Technology

Leela Palace
Bangalore, Karnataka 560008
INDIA
EMail: dhruv.dhody@huawei.com

Siva Sivabalan
Cisco Systems, Inc.
2000 Innovation Drive
Kanata, Ontario K2K 3E8
Canada
EMail: msiva@cisco.com

12. Acknowledgements

We would like to thank Adrian Farrel, Cyril Margaria and Ramon Casellas for their contributions to this document.

We would like to thank Shane Amante, Julien Meuric, Kohei Shiomoto, Paul Schultz and Raveendra Torvi for their comments and suggestions. Thanks also to Jon Hardwick, Oscar Gonzales de Dios, Tomas Janciga, Stefan Kobza, Kexin Tang, Matej Spanik, Jon Parker, Marek Zavodsky, Ambrose Kwong, Ashwin Sampath, Calvin Ying, Mustapha Aissaoui, Stephane Litkowski and Olivier Dugeon for helpful comments and discussions.

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, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2205] Braden, R., Ed., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, DOI 10.17487/RFC2205, September 1997, <http://www.rfc-editor.org/info/rfc2205>.

[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, <http://www.rfc-editor.org/info/rfc3209>.

[RFC5088] Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R. Zhang, "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5088, DOI 10.17487/RFC5088, January 2008, <http://www.rfc-editor.org/info/rfc5088>.

[RFC5089] Le Roux, JL., Ed., Vasseur, JP., Ed., Ikejiri, Y., and R. Zhang, "IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5089, DOI 10.17487/RFC5089, January 2008, <http://www.rfc-editor.org/info/rfc5089>.

[RFC5284] Swallow, G. and A. Farrel, "User-Defined Errors for RSVP", RFC 5284, DOI 10.17487/RFC5284, August 2008, <http://www.rfc-editor.org/info/rfc5284>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <http://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, <http://www.rfc-editor.org/info/rfc5511>.

[RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <http://www.rfc-editor.org/info/rfc8051>.

13.2. Informative References

[I-D.ietf-pce-gmpls-pcep-extensions]
Margaria, C., Dios, O., and F. Zhang, "PCEP extensions for GMPLS", draft-ietf-pce-gmpls-pcep-extensions-11 (work in progress), October 2015.

[I-D.ietf-pce-pce-initiated-lsp]
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-09 (work in progress), March 2017.

[I-D.ietf-pce-pcep-yang]
Dhody, D., Hardwick, J., Beeram, V., and j. jefftant@gmail.com, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-02 (work in progress), March 2017.

[I-D.ietf-pce-pceps]
Lopez, D., Dios, O., Wu, Q., and D. Dhody, "Secure Transport for PCEP", draft-ietf-pce-pceps-14 (work in progress), May 2017.

[I-D.ietf-pce-stateful-sync-optimizations]
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", draft-ietf-pce-stateful-sync-optimizations-10 (work in progress), March 2017.

[MPLS-PC] Chaieb, I., Le Roux, JL., and B. Cousin, "Improved MPLS-TE LSP Path Computation using Preemption", Global Information Infrastructure Symposium, July 2007.

[MXMN-TE] Danna, E., Mandal, S., and A. Singh, "Practical linear programming algorithm for balancing the max-min fairness and throughput objectives in traffic engineering", INFOCOM, 2012 Proceedings IEEE Page(s): 846-854, 2012.

[RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, DOI 10.17487/RFC2702, September 1999, <http://www.rfc-editor.org/info/rfc2702>.

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <http://www.rfc-editor.org/info/rfc3031>.

[RFC3346] Boyle, J., Gill, V., Hannan, A., Cooper, D., Awduche, D., Christian, B., and W. Lai, "Applicability Statement for Traffic Engineering with MPLS", RFC 3346, DOI 10.17487/RFC3346, August 2002, <http://www.rfc-editor.org/info/rfc3346>.

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <http://www.rfc-editor.org/info/rfc3630>.

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

[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, <http://www.rfc-editor.org/info/rfc4657>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <http://www.rfc-editor.org/info/rfc5305>.

[RFC5394] Bryskin, I., Papadimitriou, D., Berger, L., and J. Ash, "Policy-Enabled Path Computation Framework", RFC 5394, DOI 10.17487/RFC5394, December 2008, <http://www.rfc-editor.org/info/rfc5394>.

[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, <http://www.rfc-editor.org/info/rfc7420>.

[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, <http://www.rfc-editor.org/info/rfc7525>.

Authors' Addresses

Edward Crabbe
Oracle
1501 4th Ave, suite 1800
Seattle, WA 98101
US

Email: edward.crabbe@oracle.com

Ina Minei
Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: inaminei@google.com

Jan Medved
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134
US

Email: jmedved@cisco.com

Robert Varga
Pantheon Technologies SRO
Mlynske Nivy 56
Bratislava 821 05
Slovakia

Email: robert.varga@pantheon.tech

Path Computation Element
Internet-Draft
Intended status: Experimental
Expires: August 18, 2014

D. Lopez
O. Gonzalez de Dios
Telefonica I+D
Q. Wu
D. Dhody
Huawei

February 14, 2014

Secure Transport for PCEP
draft-lopez-pce-pceps-02

Abstract

The Path Computation Element Communication Protocol (PCEP) defines the mechanisms for the communication between a Path Computation Client (PCC) and a Path Computation Element (PCE), or among PCEs. This document describe the usage of Transport Layer Security (TLS) to enhance PCEP security, hence the PCEPS acronym proposed for it. The additional security mechanisms are provided by the transport protocol supporting PCEP, and therefore they do not affect its flexibility and extensibility.

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 http://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 18, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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. Requirements Language 3
3. Applying PCEPS 4
3.1. TCP ports 4
3.2. TLS Connection Establishment 4
3.3. Peer Identity 6
3.4. Connection Establishment Failure 7
4. Discovery Mechanisms 7
4.1. DANE Applicability 8
5. Backward Compatibility 8
6. IANA Considerations 8
7. Security Considerations 8
8. Acknowledgements 9
9. References 9
9.1. Normative References 9
9.2. Informative References 10
Authors' Addresses 11

1. Introduction

PCEP [RFC5440] defines the mechanisms for the communication between a Path Computation Client (PCC) and a Path Computation Element (PCE), or between two PCEs. These interactions include requests and replies that can be critical for a sustainable network operation and adequate resource allocation, and therefore appropriate security becomes a key element in the PCE infrastructure. As the applications of the PCE framework evolves, and more complex service patterns emerge, the definition of a secure mode of operation becomes more relevant.

[RFC5440] analyzes in its section on security considerations the potential threats to PCEP and their consequences, and discusses several mechanisms for protecting PCEP against security attacks, without making a specific recommendation on a particular one or defining their application in depth. Moreover, [RFC6952] remarks the importance of ensuring PCEP communication privacy, especially when PCEP communication endpoints do not reside in the same AS, as the interception of PCEP messages could leak sensitive information related to computed paths and resources.

Among the possible solutions mentioned in these documents, Transport Layer Security (TLS) [RFC5246] provides support for peer authentication, and message encryption and integrity. TLS supports the usage of well-know mechanisms to support key configuration and exchange, and means to perform security checks on the results of PCE discovery procedures via IGP ([RFC5088] and [RFC5089]).

This document describes a security container for the transport of PCEP requests and replies, and therefore it will not interfere with the protocol flexibility and extensibility.

This document describes how to apply TLS in securing PCE interactions, including the TLS handshake mechanisms, the TLS methods for peer authentication, the applicable TLS ciphersuites for data exchange, and the handling of errors in the security checks. In the rest of the document we will refer to this usage of TLS to provide a secure transport for PCEP as "PCEPS".

2. 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 [RFC2119].

3. Applying PCEPS

3.1. TCP ports

Since PCEP can operate either with or without TLS, it is necessary for the PCEP speaker to indicate whether it wants to set up a TLS connection or not. There are two main ways of achieving this:

- o One option is to use a different port number for TLS connections (for example, the port 443 used for HTTPS)

- o The other is to use the regular port number and have the PCEP speaker request that the PCE switch the connection to TLS using a protocol-specific mechanism (for example, the STARTTLS for mail and news protocols)

To avoid requiring a specific PCEP extension to request TLS, this document proposes the usage of the former solution to implement PCEPS.

The default destination port number for PCEPS is TCP/XXXX.

NOTE: This port has to be agreed and registered as PCEPS with IANA.

3.2. TLS Connection Establishment

PCEPS has no notion of negotiating TLS in an established connection. PCEP peers MAY either discover that the other PCEP endpoint supports PCEPS or can be preconfigured to use PCEPS for a given peer (see section Section 4 for more details). The connection establishment SHALL follow the following steps:

1. After completing the TCP handshake, immediately negotiate TLS sessions according to [RFC5246]. The following restrictions apply:

    * Support for TLS v1.2 [RFC5246] or later is REQUIRED.

    * Support for certificate-based mutual authentication is REQUIRED.

    * Negotiation of mutual authentication is REQUIRED.

    * Negotiation of a ciphersuite providing for integrity protection is REQUIRED.

    * Negotiation of a ciphersuite providing for confidentiality is RECOMMENDED.

* Support for and negotiation of compression is OPTIONAL.

* PCEPS implementations MUST, at a minimum, support negotiation of the TLS_RSA_WITH_3DES_EDE_CBC_SHA, and SHOULD support TLS_RSA_WITH_RC4_128_SHA and TLS_RSA_WITH_AES_128_CBC_SHA as well. In addition, PCEPS implementations MUST support negotiation of the mandatory-to-implement ciphersuites required by the versions of TLS that they support.

2. Peer authentication can be performed in any of the following two REQUIRED operation models:

   * TLS with X.509 certificates using PKIX trust models:

     + Implementations MUST allow the configuration of a list of trusted Certification Authorities (CAs) for incoming connections.

     + Certificate validation MUST include the verification rules as per [RFC5280].

     + Implementations SHOULD indicate their trusted CAs. For TLS 1.2, this is done using [RFC5246], Section 7.4.4, "certificate_authorities" (server side) and [RFC6066], Section 6 "Trusted CA Indication" (client side).

     + Peer validation always SHOULD include a check on whether the locally configured expected DNS name or IP address of the peer that is contacted matches its presented certificate. DNS names and IP addresses can be contained in the Common Name (CN) or subjectAltName entries. For verification, only one of these entries is to be considered. The following precedence applies: for DNS name validation, subjectAltName:DNS has precedence over CN; for IP address validation, subjectAltName:iPAddr has precedence over CN.

     + NOTE: Consider here whether peer validation MAY be extended by means of the DANE procedures, including its specs as informative references.

     + Implementations MAY allow the configuration of a set of additional properties of the certificate to check for a peer’s authorization to communicate (e.g., a set of allowed values in subjectAltName:URI or a set of allowed X509v3 Certificate Policies)

* TLS with X.509 certificates using certificate fingerprints: Implementations MUST allow the configuration of a list of trusted certificates, identified via fingerprint of the Distinguished Encoding Rules (DER) encoded certificate octets. Implementations MUST support SHA-256 as the hash algorithm for the fingerprint.

3. Start exchanging PCEP messages.

To support TLS re-negotiation both peers MUST support the mechanism described in [RFC5746]. Any attempt of initiate a TLS handshake to establish new cryptographic parameters not aligned with [RFC5746] SHALL be considered a TLS negotiation failure.

3.3. Peer Identity

Depending on the peer authentication method in use, PCEPS supports different operation modes to establish peer’s identity and whether it is entitled to perform requests or can be considered authoritative in its replies. PCEPS implementations SHOULD provide mechanisms for associating peer identities with different levels of access and/or authoritativeness, and they MUST provide a mechanism for establish a default level for properly identified peers. Any connection established with a peer that cannot be properly identified SHALL be terminated before any PCEP exchange takes place.

In TLS-X.509 mode using fingerprints, a peer is uniquely identified by the fingerprint of the presented client certificate.

There are numerous trust models in Public-Key Infrastructure (PKI) environments, and it is beyond the scope of this document to define how a particular deployment determines whether a client is trustworthy. Implementations that want to support a wide variety of trust models should expose as many details of the presented certificate to the administrator as possible so that the trust model can be implemented by the administrator. As a suggestion, at least the following parameters of the X.509 client certificate should be exposed:

- Peer’s IP address
- Peer’s fully qualified domain name (FQDN)
- Certificate Fingerprint
- Issuer

- o Subject
- o All X509v3 Extended Key Usage
- o All X509v3 Subject Alternative Name
- o All X509v3 Certificate Policies

In addition, a PCC MAY apply the procedures described in [RFC6698] (DANE) to verify its peer identity when using DNS discovery. See section Section 4.1 for further details.

3.4. Connection Establishment Failure

In case the initial TLS negotiation or the peer identity check fail according to the procedures listed in this document, the peer MUST immediately terminate the session. It SHOULD follow the procedure listed in [RFC5440] to retry session setup along with an exponential back-off session establishment retry procedure.

4. Discovery Mechanisms

A PCE can advertise its capability to support PCEPS using the IGP advertisement and discovery mechanism. The PCE-CAP-FLAGS sub-TLV is an optional sub-TLV used to advertise PCE capabilities. It MAY be present within the PCED sub-TLV carried by OSPF or IS-IS. [RFC5088] and [RFC5089] provide the description and processing rules for this sub-TLV when carried within OSPF and IS-IS, respectively. PCE capability bits are defined in [RFC5088]. A new capability flag bit for the PCE-CAP-FLAGS sub-TLV that can be announced as attribute to distribute PCEP security support information is proposed in [I-D.wu-pce-discovery-pceps-support]

NOTE: A new bit must be added here to advertise the PCEPS capability.

When DNS is used by a PCC (or a PCE acting as a client, for the rest of the section, PCC refers to both) willing to use PCEPS to locate an appropriate PCE [I-D.wu-pce-dns-pce-discovery], the PCC as initiating entity chooses at least one of the returned FQDNs to resolve, which it does by performing DNS "A" or "AAAA" lookups on the FDQN. This will eventually result in an IPv4 or IPv6 address. The PCC SHALL use the IP address(es) from the successfully resolved FDQN (with the corresponding port number returned by the DNS SRV lookup) as the connection address(es) for the receiving entity.

If the PCC fails to connect using an IP address but the "A" or "AAAA" lookups returned more than one IP address, then the PCC SHOULD use

the next resolved IP address for that FDQN as the connection address. If the PCC fails to connect using all resolved IP addresses for a given FDQN, then it SHOULD repeat the process of resolution and connection for the next FQDN returned by the SRV lookup based on the priority and weight.

If the PCC receives a response to its SRV query but it is not able to establish a PCEPS connection using the data received in the response, as initiating entity it MAY fall back to lookup a PCE that uses TCP as transport.

4.1. DANE Applicability

DANE [RFC6698] defines a secure method to associate the certificate that is obtained from a TLS server with a domain name using DNS, i.e., using the TLSA DNS resource record (RR) to associate a TLS server certificate or public key with the domain name where the record is found, thus forming a "TLSA certificate association". The DNS information needs to be protected by DNSSEC. A PCC willing to apply DANE to verify server identity MUST conform to the rules defined in section 4 of [RFC6698].

5. Backward Compatibility

Since the procedure described in this document describes a security container for the transport of PCEP requests and replies carried on a newly allocated TCP port there will be no impact on the base PCEP and/or any further extensions.

6. IANA Considerations

NOTE: PCEPS has to be registered as TCP port XXXX.

No new PCEP messages or other objects are defined.

7. Security Considerations

While the application of TLS satisfies the requirement on privacy as well as fine-grained, policy-based peer authentication, there are security threats that it cannot address. It is advisable to apply additional protection measures, in particular in what relates to attacks specifically addressed to forging the TCP connection underpinning TLS. TCP-AO (TCP Authentication Option [RFC5925]) is fully compatible with and deemed as complementary to TLS, so its usage is to be considered as a security enhancement whenever any of

the PCEPS peers require it, especially in the case of long-lived connections. The mechanisms to configure the requirements to use TCP-AO and other lower-layer protection measures, as well as the association of the required crypto material (MKT in the case of TCP-AO) with a particular peer are outside the scope of this document. [I-D.chunduri-karp-using-ikev2-with-tcp-ao] defines a method to perform such association.

Since computational resources required by TLS handshake and ciphersuite are higher than unencrypted TCP, clients connecting to a PCEPS server can more easily create high load conditions and a malicious client might create a Denial-of-Service attack more easily.

Some TLS ciphersuites only provide integrity validation of their payload, and provide no encryption. This specification does not forbid the use of such ciphersuites, but administrators must weight carefully the risk of relevant internal data leakage that can occur in such a case, as explicitly stated by [RFC6952].

When using certificate fingerprints to identify PCEPS peers, any two certificates that produce the same hash value will be considered the same peer. Therefore, it is important to make sure that the hash function used is cryptographically uncompromised so that attackers are very unlikely to be able to produce a hash collision with a certificate of their choice. This document mandates support for SHA-256, but a later revision may demand support for stronger functions if suitable attacks on it are known.

8. Acknowledgements

This specification relies on the analysis and profiling of TLS included in [RFC6614].

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5088] Le Roux, JL., Vasseur, JP., Ikejiri, Y., and R. Zhang, "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5088, January 2008.

[RFC5089] Le Roux, JL., Vasseur, JP., Ikejiri, Y., and R. Zhang, "IS-IS Protocol Extensions for Path Computation Element

(PCE) Discovery", RFC 5089, January 2008.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5746] Rescorla, E., Ray, M., Dispensa, S., and N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension", RFC 5746, February 2010.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, June 2010.

[RFC6066] Eastlake, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, January 2011.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, August 2012.

9.2. Informative References

[I-D.chunduri-karp-using-ikev2-with-tcp-ao]
Chunduri, U., Tian, A., and J. Touch, "A framework for RPs to use IKEv2 KMP", draft-chunduri-karp-using-ikev2-with-tcp-ao-06 (work in progress), February 2014.

[I-D.wu-pce-discovery-pceps-support]
Lopez, D., Wu, Q., Dhody, D., and D. King, "IGP extension for PCEP security capability support in the PCE discovery", draft-wu-pce-discovery-pceps-support-00 (work in progress), February 2014.

[I-D.wu-pce-dns-pce-discovery]
Wu, W., Dhody, D., King, D., and D. Lopez, "Path Computation Element (PCE) Discovery using Domain Name System(DNS)", draft-wu-pce-dns-pce-discovery-04 (work in progress), January 2014.

[RFC6614] Winter, S., McCauley, M., Venaas, S., and K. Wierenga, "Transport Layer Security (TLS) Encryption for RADIUS", RFC 6614, May 2012.

[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, May 2013.

Authors' Addresses

Diego R. Lopez
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid, 28006
Spain

Phone: +34 913 129 041
Email: diego@tid.es

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz, 82
Madrid, 28006
Spain

Phone: +34 913 129 041
Email: ogondio@tid.es

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: sunseawq@huawei.com

Dhruv Dhody
Huawei
Leela Palace
Bangalore, KA 560008
India

Email: dhruv.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: June 5, 2014

E. Crabbe
Google, Inc.
J. Medved
Cisco Systems, Inc.
I. Minei
Juniper Networks, Inc.
R. Varga
Pantheon Technologies SRO
X. Zhang
D. Dhody
Huawei Technologies
December 2, 2013

Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE
draft-minei-pce-stateful-sync-optimizations-01

Abstract

A stateful Path Computation Element (PCE) has access to not only the information disseminated by the network's Interior Gateway Protocol (IGP), but also the set of active paths and their reserved resources for its computation. The additional Label Switched Path (LSP) state information allows the PCE to compute constrained paths while considering individual LSPs and their interactions. This requires a reliable state synchronization mechanism between the PCE and the network, PCE and path computation clients (PCCs), and between cooperating PCEs. The basic mechanism for state synchronization is part of the stateful PCE specification. This draft presents motivations for optimizations to the base state synchronization procedure and specifies the required Path Computation Element Communication Protocol (PCEP) extensions.

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 [RFC2119].

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 http://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 June 5, 2014.

Copyright Notice

Copyright (c) 2013 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 (http://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 4
2. Terminology 4
3. State Synchronization Avoidance 4
3.1. Motivation 4
3.2. State Synchronization Avoidance Procedure 5
3.3. PCEP Extensions 9
3.3.1. LSP State Database Version Number TLV 9
3.3.2. Speaker Entity Identifier TLV 10
4. PCE-triggered State Synchronization 11
4.1. Motivation 11
4.2. PCE-triggered State Synchronization Procedure 11
5. Incremental State Synchronization 11
5.1. Motivation 12
5.2. Incremental Synchronization Procedure 13
6. Advertising Support of Synchronization Optimizations 15
7. IANA Considerations 16
7.1. PCEP-Error Object 16
7.2. PCEP TLV Type Indicators 17
7.3. STATEFUL-PCE-CAPABILITY TLV 17
8. Security Considerations 17
9. Acknowledgements 17
10. Contributors 17
11. References 18
11.1. Normative References 18
11.2. Informative References 18
Authors’ Addresses 18

1. Introduction

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

[I-D.ietf-pce-stateful-pce] 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. This requires a reliable state synchronization mechanism between the PCE and the network, PCE and PCC, and between cooperating PCEs. [I-D.ietf-pce-stateful-pce] describes the basic mechanism for state synchronization. This draft specifies optimizations for state synchronization and the correspoding PCEP extensions.

2. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [I-D.ietf-pce-stateful-pce] : Delegation, Redelegation Timeout Interval, LSP State Report, LSP Update Request, LSP State Database.

Within this document, when describing PCE-PCE communications, the requesting PCE fills the role of a PCC. This provides a saving in documentation without loss of function.

The message formats in this document are specified using Routing Backus-Naur Format (RBNF) encoding as specified in [RFC5511].

3. State Synchronization Avoidance

3.1. Motivation

The purpose of state synchronization is to provide a checkpoint-in-time state replica of a PCC's LSP state in a stateful PCE. State synchronization is performed immediately after the initialization phase ([RFC5440]). [I-D.ietf-pce-stateful-pce] describes the basic mechanism for state synchronization.

State synchronization is not always necessary following a PCEP

session restart. If the state of both PCEP peers did not change, the synchronization phase may be skipped. This can result in significant savings in both control-plane data exchanges and the time it takes for the stateful PCE to become fully operational.

3.2. State Synchronization Avoidance Procedure

State synchronization MAY be skipped following a PCEP session restart if the state of both PCEP peers did not change during the period prior to session re-initialization. To be able to make this determination, state must be exchanged and maintained by both PCE and PCC during normal operation. This is accomplished by keeping track of the changes to the LSP state database, using a version tracking field called the LSP State Database Version Number.

The LSP State Database Version Number, carried in LSP-DB-VERSION TLV (see Section 3.3.1), is owned by a PCC and it MUST be incremented by 1 for each successive change in the PCC's LSP state database. The LSP State Database Version Number MUST start at 1 and may wrap around. Values 0 and 0xFFFFFFFFFFFFFFFF are reserved. If either of the two values are used during LSP state (re)-synchronization, the PCE speaker receiving this node should send back a PCErr with Error-type 20 Error-value 6 'Received an invalid LSP DB Version Number', and close the PCEP session. Operations that trigger a change to the local LSP state database include a change in the LSP operational state, delegation of an LSP, removal or setup of an LSP or change in any of the LSP attributes that would trigger a report to the PCE.

State synchronization avoidance is advertised on a PCEP session during session startup using the INCLUDE-DB-VERSION (IDB) bit in the capabilities TLV (see Section 6). The peer may move in the network, either physically or logically, which may cause its connectivity details and transport-level identity (such as IP address) to change. To ensure that a PCEP peer can recognize a previously connected peer even in face of such mobility, each PCEP peer includes the SPEAKER-ENTITY-ID TLV described in Section 3.3.2 in the OPEN message.

If both PCEP speakers set the IDB flag in the OPEN object's STATEFUL-PCE-CAPABILITY TLV to 1, the PCC MUST include the LSP-DB-VERSION TLV in each LSP object of the PCRpt message. If the LSP-DB-VERSION TLV is missing in a PCRpt message, the PCE will generate an error with error-type 6 (mandatory object missing) and Error Value 12 (LSP-DB-VERSION TLV missing) and close the session. If state synchronization avoidance has not been enabled on a PCEP session, the PCC SHOULD NOT include the LSP-DB-VERSION TLV in the LSP Object and the PCE SHOULD ignore it were to receive one.

If a PCE's LSP state database survived the restart of a PCEP session,

the PCE will include the LSP-DB-VERSION TLV in its OPEN object, and the TLV will contain the last LSP State Database Version Number received on an LSP State Report from the PCC in the previous PCEP session. If a PCC's LSP State Database survived the restart of a PCEP session, the PCC will include the LSP-DB-VERSION TLV in its OPEN object and the TLV will contain the latest LSP State Database Version Number. If a PCEP speaker's LSP state database did not survive the restart of a PCEP session, the PCEP speaker MUST NOT include the LSP-DB-VERSION TLV in the OPEN object.

If both PCEP speakers include the LSP-DB-VERSION TLV in the OPEN Object and the TLV values match, the PCC MAY skip state synchronization. Otherwise, the PCC MUST perform state synchronization to the stateful PCE. If the PCC attempts to skip state synchronization (i.e., the SYNC Flag = 0 on the first LSP State Report from the PCC as per [I-D.ietf-pce-stateful-pce]), the PCE MUST send back a PCErr with Error-type 20 Error-value 2 'LSP Database version mismatch', and close the PCEP session.

If state synchronization is required, then prior to completing the initialization phase, the PCE MUST mark any LSPs in the LSP database that were previously reported by the PCC as stale. When the PCC reports an LSP during state synchronization, if the LSP already exists in the LSP database, the PCE MUST update the LSP database and clear the stale marker from the LSP. When it has finished state synchronization, the PCC MUST immediately send an end of synchronization marker. The end of synchronization marker is a Path Computation State Report (PCRpt) message with an LSP object containing a PLSP-ID of 0 and with the SYNC flag set to 0 ([I-D.ietf-pce-stateful-pce]). The LSP-DB-VERSION TLV MUST be included in this PCRpt message. On receiving this state report, the PCE MUST purge any LSPs from the LSP database that are still marked as stale.

Note that a PCE/PCC MAY force state synchronization by not including the LSP-DB-VERSION TLV in its OPEN object.

Since a PCE does not make changes to the LSP State Database Version Number, a PCC should never encounter this TLV in a message from the PCE (other than the OPEN message). A PCC SHOULD ignore the LSP-DB-VERSION TLV, were it to receive one from a PCE.

If state synchronization avoidance is enabled, a PCC MUST increment its LSP State Database Version Number when the 'Redelegation Timeout Interval' timer expires (see [I-D.ietf-pce-stateful-pce]) for the use of the Redelegation Timeout Interval).

Figure 1 shows an example sequence where the state synchronization is

skipped.

```
                   +-+-+                    +-+-+
                   |PCC|                    |PCE|
                   +-+-+                    +-+-+
                     |                        |
                     |--Open--,               |
                     |  DBv=42 \    ,---Open--|
                     |   IDB=1  \  /   DBv=42 |
                     |           \/     IDB=1 |
                     |           /\           |
                     |          /   '-------->| (OK to skip sync)
         (Skip sync) |<--------'              |
                     |            .           |
                     |            .           |
                     |            .           |
                     |                        |
                     |--PCRpt,DBv=43,SYNC=0-->| (Regular
                     |                        |  LSP State Report)
                     |--PCRpt,DBv=44,SYNC=0-->| (Regular
                     |                        |  LSP State Report)
                     |--PCRpt,DBv=45,SYNC=0-->|
                     |                        |
```

Figure 1: State Synchronization Skipped

Figure 2 shows an example sequence where the state synchronization is performed due to LSP state database version mismatch during the PCEP session setup. Note that the same state synchronization sequence would happen if either the PCC or the PCE would not include the LSP-DB-VERSION TLV in their respective Open messages.

```
                    +-+-+                    +-+-+
                    |PCC|                    |PCE|
                    +-+-+                    +-+-+
                      |                        |
                      |--Open--,               |
                      |  DBv=46 \    ,---Open--|
                      |   IDB=1  \  /   DBv=42 |
                      |           \/     IDB=1 |
                      |           /\           |
                      |          /   '-------->| (Expect sync)
            (Do sync) |<--------'              |
                      |                        |
                      |--PCRpt,DBv=46,SYNC=1-->| (Sync start)
                      |            .           |
                      |            .           |
                      |            .           |
                      |--PCRpt,DBv=46,SYNC=1-->| (Sync done)
                      |            .           |(Purge LSP State
                      |            .           | if applicable)
                      |            .           |
                      |--PCRpt,DBv=47,SYNC=0-->| (Regular
                      |                        |  LSP State Report)
                      |--PCRpt,DBv=48,SYNC=0-->| (Regular
                      |                        |  LSP State Report)
                      |--PCRpt,DBv=49,SYNC=0-->|
                      |                        |
```

Figure 2: State Synchronization Performed

Figure 3 shows an example sequence where the state synchronization is skipped, but because one or both PCEP speakers set the IDB Flag to 0, the PCC does not send LSP-DB-VERSION TLVs in subsequent PCRpt messages to the PCE. If the current PCEP session restarts, the PCEP speakers will have to perform state synchronization, since the PCE does not know the PCC's latest LSP State Database Version Number information.

```
                    +-+-+                    +-+-+
                    |PCC|                    |PCE|
                    +-+-+                    +-+-+
                      |                        |
                      |--Open--,               |
                      |  DBv=42 \    ,---Open--|
                      |   IDB=0  \  /   DBv=42 |
                      |           \/     IDB=0 |
                      |           /\           |
                      |          /   '-------->| (OK to skip sync)
          (Skip sync) |<--------'              |
                      |            .           |
                      |            .           |
                      |            .           |
                      |------PCRpt,SYNC=0----->| (Regular
                      |                        |  LSP State Report)
                      |------PCRpt,SYNC=0----->| (Regular
                      |                        |  LSP State Report)
                      |------PCRpt,SYNC=0----->|
                      |                        |
```

Figure 3: State Synchronization Skipped, no LSP-DB-VERSION TLVs sent from PCC

3.3. PCEP Extensions

3.3.1. LSP State Database Version Number TLV

The LSP State Database Version Number (LSP-DB-VERSION) TLV is an optional TLV that MAY be included in the OPEN object and the LSP object.

The format of the LSP-DB-VERSION TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBD]          |            Length=8           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 LSP State DB Version Number                   |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: LSP-DB-VERSION TLV format

The type of the TLV is [TBD] and it has a fixed length of 8 octets. The value contains a 64-bit unsigned integer, representing the LSP

State DB Version Number.

3.3.2. Speaker Entity Identifier TLV

The Speaker Entity Identifier TLV (SPEAKER-ENTITY-ID) is an optional TLV that MAY be included in the OPEN Object when a PCEP speaker wishes to determine if state synchronization can be skipped when a PCEP session is restarted. It contains a unique identifier for the node that does not change during the lifetime of the PCEP speaker. It identifies the PCEP speaker to its peers even if the speaker's IP address is changed.

The format of the SPEAKER-ENTITY-ID TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBD]          |       Length (variable)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                 Speaker Entity Identifier                   //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: SPEAKER-ENTITY-ID TLV format

The type of the TLV is [TBD] and it has a a variable length, which MUST be greater than 0 and padded to 4-octet alignment (, and padding is not included in the Length field). The value contains the entity identifier of the speaker transmitting this TLV. This identifier is required to be unique within its scope of visibility, which is usually limited to a single domain. It MAY be configured by the operator. Alternatively, it can be derived automatically from a suitably-stable unique identifier, such as a MAC address, serial number, Traffic Engineering Router ID, or similar. In the case of inter-domain connections, the speaker SHOULD prefix its usual identifier with the domain identifier of its residence, such as Autonomous System number, IGP area identifier, or similar.

The relationship between this identifier and entities in the Traffic Engineering database is intentionally left undefined.

From a manageability point of view, a PCE or PCC implementation SHOULD allow the operator to configure this Speaker Entity Identifier.

4. PCE-triggered State Synchronization

4.1. Motivation

The accuracy of the computations performed by the PCE is tied to the accuracy of the view the PCE has on the state of the LSPs. Therefore, it can be beneficial to be able to resynchronize this state even after the session has been established. The PCE may use this approach to continuously sanity check its state against the network, or to recover from error conditions without having to tear down sessions.

4.2. PCE-triggered State Synchronization Procedure

Support of PCE-triggered state synchronization is advertised during session startup using the TRIGGERED-SYNC (T) bit in the STATEFUL-PCE-CAPABILITY TLV (see Section 6). The PCE can choose to resynchronize its entire LSP database or a single LSP.

To trigger resynchronization for an LSP, the PCE MUST first mark the LSP as stale and then send a Path Computation State Update (PCUpd) for it, with the SYNC flag in the LSP object set to 1. The PCE SHOULD NOT include any parameter updates for the LSP, and the PCC SHOULD ignore such updates if the SYNC flag is set. The PCC MUST respond with a PCRpt message and SHOULD include the SRP-ID-number of the PCUpd that triggered the resynchronization.

The PCE can also trigger resynchronization of the entire LSP database. The PCE MUST first mark all LSPs in the LSP database that were previously reported by the PCC as stale and then send a PCUpd with an LSP object containing a PLSP-ID of 0 and with the SYNC flag set to 1. This PCUpd message is the trigger for the PCC to enter the synchronization phase as described in [I-D.ietf-pce-stateful-pce] and start sending PCRpt messages. After the receipt of the end-of-synchronization marker, the PCE will purge LSPs which were not refreshed. The SRP-ID-number of the PCUpd that triggered the resynchronization SHOULD be included in each of the PCRpt messages.

If the TRIGGERED-SYNC capability is not advertised and the PCC receives a PCUpd with the SYNC flag set to 1, it MUST send a PCErr with the SRP-ID-number of the PCUpd, error-type 20 and error-value 4.(see Section 7.1)

5. Incremental State Synchronization

[I-D.ietf-pce-stateful-pce] describes the LSP state synchronization mechanism between PCCs and stateful PCEs. During the state

synchronization, a PCC sends the information of all its LSPs (full LSP-DB) to the stateful PCE. In order to save the state synchronization overhead when there is a small number of LSP state change in the network between PCEP session restart as well as avoiding overloading a PCE during state (re-)synchronization phase, this section proposes a mechanism for incremental (Delta) LSP Database (LSP-DB) synchronization as well as allowing PCE to control the timing of the LSP-DB synchronization process during incremental syncronization.

5.1. Motivation

According to [I-D.ietf-pce-stateful-pce] , if a PCE restarts and its LSP-DB survived, PCCs with mismatched LSP State Database Version Number will send all their LSPs information (full LSP-DB) to the stateful PCE, even if only a small number of LSPs underwent state change. It can take a long time and consume large communication channel bandwidth. Moreover, the stateful PCE can get overloaded with all the PCC performing full synchronization with it at the same time.

Figure 6 shows an example of LSP state synchronization.

```
                                  +-----+
                                  | PCE |
                                  +-----+
                                 /
                                /
                               /
                              /
                       +------+            +------+
                       | PCC1 |------------| PCC2 |
                       +------+            +------+
                          |                   |
                          |                   |
                       +------+            +------+
                       | PCC3 |------------| PCC4 |
                       +------+            +------+
```

Figure 6: Topology Example

Assuming there are 320 LSPs in the network, with each PCC having 80 LSPs. During the time when the PCEP session is down, 20 LSPs of each PCC (i.e., 80 LSPs in total), are changed. Hence when PCEP session restarts, the stateful PCE needs to synchronize 320 LSPs with all PCCs. But actually, 240 LSPs stay the same. If performing full LSP state synchronization, it can take a long time to carry out the synchronization of all LSPs. It is especially true when only a low

bandwidth communication channel is available and there is a substantial number of LSPs in the network. Another disadvantage of full LSP synchronization is that it is a waste of communication bandwidth to perform full LSP synchronization given the fact that the number of LSP changes can be small during the time when PCEP session is down.

An incremental (Delta) LSP Database (LSP-DB) state synchronization is described in this section, where only the LSPs underwent state change are synchronized between the session restart. This may include new/modified/deleted LSPs. Furthermore, to avoid overloading the PCE, the proposed method enable a stateful PCE to trigger the LSP synchronization (similar to Section 4).

PCEP extensions for stateful PCEs to perform LSP synchronization SHOULD allow:

- o Incremental LSP state synchronization between session restarts. Note this does not exclude the need for a stateful PCE to request a full LSP DB synchronization.

- o A stateful PCE to control the timing of PCC synchronizing its LSP state with the PCE during incremental synchronisation.

5.2. Incremental Synchronization Procedure

[I-D.ietf-pce-stateful-pce] describes state synchronization and Section 3 describes state synchronization avoidance by using LSP-DB-VERSION TLV in its OPEN object. This section extends this idea to only synchronize the delta (changes) in case of version mismatch as well as to allow a stateful PCE to control the timing of this process.

If both PCEP speakers include the LSP-DB-VERSION TLV in the OPEN object and the LSP-DB-VERSION TLV values match, the PCC MAY skip state synchronization. Otherwise, the PCC MUST perform state synchronization. Instead of dumping full LSP-DB to PCE again, the PCC synchronizes the delta (changes) as described in Figure 7 when DELTA-LSP-SYNC-CAPABILITY (D flag) is set to 1 by both PCC and PCE (see Section 6). Other combinations of D flag setting by PCC and PCE result in full LSP-DB synchronization procedure as described in [I-D.ietf-pce-stateful-pce]. If a PCC has to force full LSP DB synchronization due to reasons including but not limited: (1) local policy configured at the PCC; (2) no sufficient LSP state caches for incremental update, the PCC can set the D flag to 0.

```
                        +-+-+                    +-+-+
                        |PCC|                    |PCE|
                        +-+-+                    +-+-+
                          |                        |
                          |--Open--,               |
                          |  DBv=46 \    ,---Open--|
                          |   IDB=1  \  /   DBv=42 |
                          |   D=1     \/     IDB=1 |
                          |   T=1     /\      D=1  |
                          |          /  \     T=1  |
                          |         /    '-------->| (Expect Delta sync)
                 (Do sync)|<--------'              | (DONOT Purge LSP
                 (Delta)  |                        | State)
     (Wait for PCE to     |                        |
      trigger LSP state   |                        |
      sync)               |                        |
                          |<----PCUpd, SYNC=1------| (ask for LSP Sync,
                          |                        |  PLSP-ID =0)
     (Delta Sync starts)  |--PCRpt,DBv=43,SYNC=1-->|
                          |            .           |
                          |            .           |
                          |            .           |
                          |            .           |
                          |--PCRpt,DBv=46,SYNC=0-->| (Sync done,
                          |                        | PLSP-ID=0)
                          |                        |
                          |--PCRpt,DBv=47,SYNC=0-->| (Regular
                          |                        |  LSP State Report)
                          |--PCRpt,DBv=48,SYNC=0-->| (Regular
                          |                        |  LSP State Report)
                          |--PCRpt,DBv=49,SYNC=0-->|
                          |                        |
```

Figure 7: Incremental Synchronization Procedure

A stateful PCE MAY choose to control the LSP-DB synchronization process. To allow PCE to do so, PCEP speakers MUST set T bit to 1 to indicate this (as described in Section 4). If the LSP-DB Version is mis-matched, it can send a PCUpd message with PLSP-ID = 0 and SYNC = 1 in order to trigger the LSP-DB synchronization process. In this way, the PCE can control the sequence of LSP synchronization among all the PCCs that are re-establishing PCEP sessions with it. When the capability of PCE control is enabled, only after a PCC receives this message, it will start sending information that PCE does not possess, which is inferred from the LSP-DB version information exchanged in the OPEN message. Note that the PCE should not mark the existing LSPs as stale for incremental state synchronisation

procedure.

As per Section 3, the LSP State Database Version Number is incremented each time a change is made to the PCC's local LSP State Database. Each LSP is associated with the DB version at the time of its state change. This is needed to determine which LSP and what information needs to be synchronized in incremental state synchronization.

It is not necessary for a PCC to store a complete history of LSP Database change, but rather remember the LSP state changes (including LSP modification, setup and deletion) that happend between the PCEP session(s) restart in order to carry out incremental state synchronization. After the synchronization procedure finishes, the PCC can dump this history information. In the example shown in Figure 7, the PCC needs to store the LSP state changes that happend between DB Version 43 to 46 and synchronizes these changes only when performing incremental LSP state update. So a PCC needs to remember the LSP state changes that happened when an existing PCEP session to a stateful PCE goes down in the hope of doing incremental synchronisation when the session is re-established.

If a PCC finds out it does not have sufficient information to complete incremental synchronisation after advertising incremental LSP state synchronization capability, it MUST send a PCErr with error-type 20 and error-value 5(see Section 7.1) and terminate the session.

6. Advertising Support of Synchronization Optimizations

Support for each of the optimizations described in this document requires advertising the corresponding capabilities during session establishment time.

New flags are defined for the STATEFUL-PCE-CAPABILITY TLV defined in [I-D.ietf-pce-stateful-pce]. Its format is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Type            |             Length=4          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                   |D|T|I|S|U|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 8: STATEFUL-PCE-CAPABILITY TLV Format

The value comprises a single field - Flags (32 bits):

U (LSP-UPDATE-CAPABILITY - 1 bit): defined in [I-D.ietf-pce-stateful-pce].

S (INCLUDE-DB-VERSION - 1 bit): if set to 1 by both PCEP Speakers, the PCC will include the LSP-DB-VERSION TLV in each LSP Object.

I (LSP-INSTANTIATION-CAPABILITY - 1 bit): defined in [I-D.crabbe-pce-pce-initiated-lsp].

T (TRIGGERED-SYNC - 1 bit): if set to 1 by both PCEP Speakers, the PCE can trigger synchronization of LSPs at any point in the life of the session.

D (DELTA-LSP-SYNC-CAPABILITY - 1 bit): if set to 1 by a PCEP speaker, it indicates that the PCEP speaker allows incremental state synchronization.

7. IANA Considerations

This document requests IANA actions to allocate code points for the protocol elements defined in this document. Values shown here are suggested for use by IANA.

7.1. PCEP-Error Object

This document defines new Error-Value values for the LSP state synchronization error defined in [I-D.ietf-pce-stateful-pce].

```
 Error-Type  Meaning
    6        Mandatory Object missing
              Error-value=12: LSP-DB-VERSION TLV missing
    20       LSP State synchronization error
              Error-value=2:  LSP Database version mismatch.
              Error-value=3:  The LSP-DB-VERSION TLV Missing when
                              state synchronization avoidance is
                              enabled.
              Error-value=4:  Attempt to trigger a synchronization
                              when the TRIGGERED-SYNC capability has
                              not been advertised.
              Error-value=5:  No sufficient LSP change information
                              for incremental LSP state
                              synchronization.
```

Error-value=6: Received an invalid LSP DB Version Number

7.2. PCEP TLV Type Indicators

This document defines the following new PCEP TLVs:

Value	Meaning	Reference
23	LSP-DB-VERSION	This document
24	SPEAKER-ENTITY-ID	This document

7.3. STATEFUL-PCE-CAPABILITY TLV

The following values are defined in this document for the Flags field in the STATEFUL-PCE-CAPABILITY-TLV in the OPEN object:

Bit	Description	Reference
28	DELTA-LSP-SYNC-CAPABILITY	This document
29	TRIGGERED-SYNC	This document
30	INCLUDE-DB-VERSION	This document

8. Security Considerations

The security considerations listed in [I-D.ietf-pce-stateful-pce] apply to this document as well.

9. Acknowledgements

We would like to thank Young Lee for his contributions.

10. Contributors

Gang Xie
Huawei Technologies
F3-5-B R&D Center, Huawei Industrial Base, Bantian, Longgang District
Shenzhen, Guangdong, 518129
P.R. China
Email: xiegang09@huawei.com

11. References

11.1. Normative References

[I-D.ietf-pce-stateful-pce]
Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5511] Farrel, A., "Routing Backus-Naur Form (RBNF): A Syntax Used to Form Encoding Rules in Various Routing Protocol Specifications", RFC 5511, April 2009.

11.2. Informative References

[I-D.crabbe-pce-pce-initiated-lsp]
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-crabbe-pce-pce-initiated-lsp-03 (work in progress), October 2013.

Authors' Addresses

Edward Crabbe
Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: edc@google.com

Jan Medved
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134
US

Email: jmedved@cisco.com

Ina Minei
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: ina@juniper.net

Robert Varga
Pantheon Technologies SRO
Mlynske Nivy 56
Bratislava 821 05
Slovakia

Email: robert.varga@pantheon.sk

Xian Zhang
Huawei Technologies
F3-5-B R&D Center, Huawei Industrial Base, Bantian, Longgang District
Shenzhen, Guangdong 518129
P.R.China

Email: zhang.xian@huawei.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

Email: dhruv.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 18, 2014

U. Palle
D. Dhody
Huawei Technologies
Y. Tanaka
Y. Kamite
NTT Communications

February 14, 2014

PCEP Extensions for PCE-initiated Point-to-Multipoint LSP Setup in a Stateful PCE Model

draft-palle-pce-stateful-pce-initiated-p2mp-lsp-01

Abstract

The Path Computation Element (PCE) has been identified as an appropriate technology for the determination of the paths of point-to-multipoint (P2MP) TE LSPs. The extensions described in [I-D.ietf-pce-stateful-pce] provide stateful control of Multiprotocol Label Switching (MPLS) Traffic Engineering Label Switched Paths (TE LSP) via PCE communication Protocol (PCEP), for a model where the Path Computation Client (PCC) delegates control over one or more locally configured LSPs to the PCE. Further [I-D.ietf-pce-pce-initiated-lsp] describes the creation and deletion of PCE-initiated LSPs under the stateful PCE model. This document provides extensions required for PCEP so as to enable the usage of a stateful PCE initiation capability in supporting point-to-multipoint (P2MP) TE LSP instantiation.

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 http://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 18, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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 3
2. Terminology 4
3. Architectural Overview 4
  3.1. Motivation 4
  3.2. Operation Overview 4
4. Support of PCE Initiated P2MP TE LSPs 5
5. PCE-initiated P2MP TE LSP Operations 5
  5.1. The PCInitiate message 5
  5.2. P2MP TE LSP Instantiation 6
  5.3. P2MP TE LSP Deletion 7
  5.4. Adding and Pruning Leaves for the P2MP TE LSP 7
  5.5. P2MP TE LSP Delegation and Cleanup 7
6. PCIntiate Message Fragmentation 7
  6.1. PCIntiate Fragmentation Procedure 8
7. Non-Support of P2MP TE LSP Instantiation for Stateful PCE . . 8
8. Security Considerations 8
9. Manageability Considerations 8
  9.1. Control of Function and Policy 8
  9.2. Information and Data Models 8
  9.3. Liveness Detection and Monitoring 8
  9.4. Verify Correct Operations 9
  9.5. Requirements On Other Protocols 9
  9.6. Impact On Network Operations 9
10. IANA Considerations 9
11. Acknowledgments 9
12. References 9
  12.1. Normative References 9
  12.2. Informative References 10

1. Introduction

As per [RFC4655], the Path Computation Element (PCE) is an entity that is capable of computing a network path or route based on a network graph, and applying computational constraints. A Path Computation Client (PCC) may make requests to a PCE for paths to be computed.

[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 PCEP is designed as a communication protocol between PCCs and PCEs for point-to-point (P2P) path computations and is defined in [RFC5440]. The extensions of PCEP to request path computation for P2MP TE LSPs are described in [RFC6006].

Stateful PCEs are shown to be helpful in many application scenarios, in both MPLS and GMPLS networks, as illustrated in [I-D.ietf-pce-stateful-pce-app]. These scenarios apply equally to P2P and P2MP TE LSPs. [I-D.ietf-pce-stateful-pce]provides the fundamental extensions needed for stateful PCE to support general functionality for P2P TE LSP. Further [I-D.palle-pce-stateful-pce-p2mp]focuses on the extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs. It includes mechanisms to effect P2MP LSP state synchronization between PCCs and PCEs, delegation of control of P2MP LSPs to PCEs, and PCE control of timing and sequence of P2MP path computations within and across PCEP sessions and focuses on a model where P2MP LSPs are configured on the PCC and control over them is delegated to the PCE.

[I-D.ietf-pce-pce-initiated-lsp]provides the fundamental extensions needed for stateful PCE-initiated P2P TE LSP instantiation.

This document describes the setup, maintenance and teardown of PCE-initiated P2MP LSPs under the stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic network that is centrally controlled and deployed.

1.1. 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 [RFC2119].

2. Terminology

Terminology used in this document is same as terminology used in [I-D.ietf-pce-stateful-pce], [I-D.ietf-pce-pce-initiated-lsp]and [RFC6006].

3. Architectural Overview

3.1. Motivation

[I-D.palle-pce-stateful-pce-p2mp] provides stateful control over P2MP TE LSPs that are locally configured on the PCC. This model relies on the Ingress taking an active role in delegating locally configured P2MP TE LSPs to the PCE, and is well suited in environments where the P2MP TE LSP placement is fairly static. However, in environments where the P2MP TE LSP placement needs to change in response to application demands, it is useful to support dynamic creation and tear down of P2MP TE LSPs. The ability for a PCE to trigger the creation of P2MP TE LSPs on demand can be seamlessly integrated into a controller-based network architecture, where intelligence in the controller can determine when and where to set up paths.

Section 3 of [I-D.ietf-pce-pce-initiated-lsp]further describes the motivation behind the PCE-Initiation capability, which are equally applicable for P2MP TE LSPs.

3.2. Operation Overview

A PCC or PCE indicates its ability to support PCE provisioned dynamic LSPs and P2MP operations during the PCEP Initialization Phase via mechanism described in Section 4.

As per section 5.1 of [I-D.ietf-pce-pce-initiated-lsp], the PCE sends a Path Computation LSP Initiate Request (PCInitiate) message to the PCC to instantiate or delete a P2P TE LSP. This document extends the PCInitiate message to support P2MP TE LSP Instantiation (see details in Section 5.1).

P2MP TE LSP instantiation and deletion operations are same as P2P LSP Instantiation as described in section 5.3 and 5.4 of [I-D.ietf-pce-pce-initiated-lsp]. This document focuses on extensions needed for further handling of P2MP TE LSP Instantiation (see details in Section 5.2).

4. Support of PCE Initiated P2MP TE LSPs

As per Section 3.1 of [RFC6006], PCE advertises P2MP capability via IGP discovery or a P2MP capable TLV in open message. To support instantiation of PCE-initiated P2MP TE LSPs, this document extends the advertisement of P2MP capable TLV in open message by all PCEP speakers. As per Section 4 of [I-D.ietf-pce-pce-initiated-lsp], PCC or PCE advertises capability for instantiation of PCE-initiated LSPs via Stateful PCE Capability TLV (LSP-INSTANTIATION-CAPABILITY bit) in open message. These mechanism when used together indicates the instantiation capability for P2MP TE LSPs by the PCEP speaker.

5. PCE-initiated P2MP TE LSP Operations

5.1. The PCInitiate message

As defined in section 5.1 of [I-D.ietf-pce-pce-initiated-lsp], PCE sends a PCInitiate message to a PCC to instantiate a P2P TE LSP, this document extends the format of PCInitiate message for the creation of P2MP TE LSPs but the creation and deletion operations of P2MP TE LSP are same to the P2P TE LSP.

The format of PCInitiate message is 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-point-path-pair-list>
                                      [<attribute-list>]

<PCE-initiated-lsp-deletion> ::= <SRP>
                                 <LSP>

Where:

<end-point-path-pair-list>::=
                   [<END-POINTS>]
                   <path>
                   [<end-point-path-pair-list>]

<path> ::= (<ERO>|<SERO>)
           [<path>]

<attribute-list> is defined in [RFC5440] and extended
by PCEP extensions.
```

The PCInitiate message with an LSP object with N bit (P2MP) set is used to convey operation on a P2MP TE LSP. The SRP object is used to correlate between initiation requests sent by the PCE and the error reports and state reports sent by the PCC as described in [I-D.ietf-pce-stateful-pce].

5.2. P2MP TE LSP Instantiation

The Instantiation operation of P2MP TE LSP is same as defined in section 5.3 of [I-D.ietf-pce-pce-initiated-lsp] including handling of PLSP-ID, SYMBOLIC-PATH-NAME etc. Rules of processing and error codes remains unchanged. Further, as defined in section 6.1 of [I-D.palle-pce-stateful-pce-p2mp], N bit MUST be set in LSP object in PCInitiate message by PCE to specify the instantiation is for P2MP TE LSP and the PCC or PCE MUST follow the mechanism defined in

[I-D.palle-pce-stateful-pce-p2mp] for delegation and updation of P2MP TE LSPs.

Though N bit is set in the LSP object, P2MP-LSP-IDENTIFIER TLV defined in section 6.2 of [I-D.palle-pce-stateful-pce-p2mp] MUST NOT be included in the LSP object in PCIntiitate message as it SHOULD be generated by PCC and carried in PCRpt message.

5.3. P2MP TE LSP Deletion

The deletion operation of P2MP TE LSP is same as defined in section 5.4 of [I-D.ietf-pce-pce-initiated-lsp] by sending an LSP Initiate Message with an LSP object carrying the PLSP-ID of the LSP to be removed and an SRP object with the R flag set (LSP-REMOVE as per section 5.2 of [I-D.ietf-pce-pce-initiated-lsp]). Rules of processing and error codes remains unchanged.

5.4. Adding and Pruning Leaves for the P2MP TE LSP

Adding of new leaves and Pruning of old Leaves for the PCE initiated P2MP TE LSP MUST be carried in PCUpd message and SHOULD refer [I-D.palle-pce-stateful-pce-p2mp] for P2MP TE LSP extensions. As defined in [RFC6006], leaf type = 1 for adding of new leaves, leaf type = 2 for pruning of old leaves of P2MP END-POINTS Object are used in PCUpd message.

5.5. P2MP TE LSP Delegation and Cleanup

P2MP TE LSP delegation and cleanup operations are same as defined in section 6 of [I-D.ietf-pce-pce-initiated-lsp]. Rules of processing and error codes remains unchanged.

6. PCIntiate Message Fragmentation

The total PCEP message length, including the common header, is 16 bytes. In certain scenarios the P2MP LSP Initiate may not fit into a single PCEP message (initial PCInitiate message). The F-bit is used in the LSP object to signal that the initial PCInitiate was too large to fit into a single message and will be fragmented into multiple messages.

Fragmentation procedure described below for PCInitiate message is similar to [RFC6006] which describes request and response message fragmentation.

6.1. PCIntiate Fragmentation Procedure

Once the PCE initiates to set up the P2MP TE LSP, a PCInitiate message is sent to the PCC. If the PCInitiate is too large to fit into a single PCInitiate message, the PCE will split the PCInitiate over multiple messages. Each PCInitiate message sent by the PCE, except the last one, will have the F-bit set in the LSP object to signify that the PCInitiate has been fragmented into multiple messages. In order to identify that a series of PCInitiate messages represents a single Initiate, each message will use the same PLSP-ID (in this case 0) and SRP-ID-number.

[Editor Note: P2MP message fragmentation errors associated with a P2MP path initiation will be defined in future version].

7. Non-Support of P2MP TE LSP Instantiation for Stateful PCE

The PCEP protocol extensions described in this document for PCC or PCE with instantiation capability for P2MP TE LSPs MUST NOT be used if PCC or PCE has not advertised its stateful capability with Instantiation and P2MP capability as per Section 4. If this is not the case and Stateful operations on P2MP TE LSPs are attempted, then a PCErr with error-type 19 (Invalid Operation) and error-value TBD needs to be generated.

[Editor Note: more information on exact error value is needed]

8. Security Considerations

TBD

9. Manageability Considerations

9.1. Control of Function and Policy

TBD.

9.2. Information and Data Models

TBD.

9.3. Liveness Detection and Monitoring

TBD.

9.4. Verify Correct Operations

TBD.

9.5. Requirements On Other Protocols

TBD.

9.6. Impact On Network Operations

TBD.

10. IANA Considerations

TBD

11. Acknowledgments

TBD

12. References

12.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[I-D.ietf-pce-stateful-pce] Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

[I-D.ietf-pce-pce-initiated-lsp] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-00 (work in progress), December 2013.

[I-D.palle-pce-stateful-pce-p2mp] Palle, U. and D. Dhody, "Path Computation Element (PCE) Protocol Extensions for Stateful PCE usage for Point-to-Multipoint Traffic Engineering Label Switched Paths", draft-palle-pce-stateful-pce-p2mp-01 (work in progress), January 2014.

12.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC4857] Fogelstroem, E., Jonsson, A., and C. Perkins, "Mobile IPv4 Regional Registration", RFC 4857, June 2007.

[RFC5671] Yasukawa, S. and A. Farrel, "Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)", RFC 5671, October 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

[I-D.ietf-pce-stateful-pce-app] Zhang, X. and I. Minei, "Applicability of Stateful Path Computation Element (PCE)", draft-ietf-pce-stateful-pce-app-01 (work in progress), September 2013.

Authors' Addresses

Udayasree Palle
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: udayasree.palle@huawei.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.ietf@gmail.com

Yosuke Tanaka
NTT Communications Corporation
Granpark Tower
3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

EMail: yosuke.tanaka@ntt.com

Yuji Kamite
NTT Communications Corporation
Granpark Tower
3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

EMail: y.kamite@ntt.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 18, 2014

U. Palle
D. Dhody
Huawei Technologies
Y. Tanaka
Y. Kamite
NTT Communications

February 14, 2014

Path Computation Element (PCE) Protocol Extensions for Stateful PCE usage for Point-to-Multipoint Traffic Engineering Label Switched Paths

draft-palle-pce-stateful-pce-p2mp-02

Abstract

The Path Computation Element (PCE) has been identified as an appropriate technology for the determination of the paths of point-to-multipoint (P2MP) TE LSPs. [I-D.ietf-pce-stateful-pce-app] presents several use cases, demonstrating scenarios that benefit from the deployment of a stateful PCE. [I-D.ietf-pce-stateful-pce] provides the fundamental PCE communication Protocol (PCEP) extensions needed to support stateful PCE functions. This memo provides extensions required for PCEP so as to enable the usage of a stateful PCE capability in supporting point-to-multipoint (P2MP) TE LSPs.

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 http://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 18, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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 3
2. Terminology 4
3. Supporting P2MP TE LSP for Stateful PCE 4
  3.1. Motivation 4
  3.2. Objectives 4
4. Functions to Support P2MP TE LSPs for Stateful PCEs 4
5. Architectural Overview of Protocol Extensions 5
  5.1. Extension of PCEP Messages 5
  5.2. Capability Advertisement 5
  5.3. State Synchronization 6
  5.4. LSP Delegation 6
  5.5. LSP Operations 6
    5.5.1. Passive Stateful PCE 6
    5.5.2. Active Stateful PCE 6
6. PCEP Object Extensions 6
  6.1. Extension of LSP Object 6
  6.2. P2MP-LSP-IDENTIFIER TLV 7
  6.3. S2L Object 10
7. PCEP Message Extensions 10
  7.1. The PCRpt Message 11
  7.2. The PCUpd Message 12
  7.3. Example 13
    7.3.1. P2MP TE LSP Update Request 13
    7.3.2. P2MP TE LSP Report 14
  7.4. Report and Update Message Fragmentation 15
    7.4.1. Report Fragmentation Procedure 15
    7.4.2. Update Fragmentation Procedure 15
8. Non-Support of P2MP TE LSPs for Stateful PCE 16
9. Security Considerations 16
10. Manageability Considerations 16
  10.1. Control of Function and Policy 16
  10.2. Information and Data Models 16
  10.3. Liveness Detection and Monitoring 16
  10.4. Verify Correct Operations 16
  10.5. Requirements On Other Protocols 16

10.6. Impact On Network Operations 17
11. IANA Considerations 17
11.1. Extension of LSP Object 17
11.2. Extension of PCEP-Error Object 17
11.3. PCEP TLV Type Indicators 18
12. Acknowledgments 18
13. References 18
13.1. Normative References 18
13.2. Informative References 18

1. Introduction

As per [RFC4655], the Path Computation Element (PCE) is an entity that is capable of computing a network path or route based on a network graph, and applying computational constraints. A Path Computation Client (PCC) may make requests to a PCE for paths to be computed.

[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 PCEP is designed as a communication protocol between PCCs and PCEs for point-to-point (P2P) path computations and is defined in [RFC5440]. The extensions of PCEP to request path computation for P2MP TE LSPs are described in [RFC6006].

Stateful PCEs are shown to be helpful in many application scenarios, in both MPLS and GMPLS networks, as illustrated in [I-D.ietf-pce-stateful-pce-app]. These scenarios apply equally to P2P and P2MP TE LSPs. [I-D.ietf-pce-stateful-pce]provides the fundamental extensions needed for stateful PCE to support general functionality for P2P TE LSP. Complementarily, this document focuses on the extensions that are necessary in order for the deployment of stateful PCEs to support P2MP TE LSPs.

1.1. 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 [RFC2119].

2. Terminology

Terminology used in this document is same as terminology used in [I-D.ietf-pce-stateful-pce]and [RFC6006].

3. Supporting P2MP TE LSP for Stateful PCE

3.1. Motivation

[I-D.ietf-pce-stateful-pce-app]presents several use cases, demonstrating scenarios that benefit from the deployment of a stateful PCE including optimization, recovery, etc. [I-D.ietf-pce-stateful-pce]defines the extensions to PCEP for P2P TE LSPs in applying these scenarios. But these scenarios apply equally to P2MP TE LSPs as well.

In addition to that, the stateful nature of a PCE simplifies the information conveyed in PCEP messages since it is possible to refer to the LSPs via PLSP-ID. For P2MP this is an added advantage, where the size of message is much larger. Incase of stateless PCE, a modification of P2MP tree requires encoding of all leaves along with the paths in PCReq message, but using a stateful PCE with P2MP capability, the PCEP message can be used to convey only the modifications (the other information can be retrieved from the P2MP LSP identifier).

3.2. Objectives

The objectives for the protocol extensions to support P2MP TE LSP for stateful PCE are same as the objectives described in section 3.2 of [I-D.ietf-pce-stateful-pce].

4. Functions to Support P2MP TE LSPs for Stateful PCEs

[I-D.ietf-pce-stateful-pce]specifies new functions to support a stateful PCE. It also specifies that a function can be initiated either from a PCC towards a PCE (C-E) or from a PCE towards a PCC (E-C).

This document extends these functions to support P2MP TE LSPs.

Capability Advertisement (E-C,C-E): both the PCC and the PCE must announce during PCEP session establishment that they support PCEP Stateful PCE extensions for P2MP using mechanisms defined in [I-D.ietf-pce-stateful-pce]and [RFC6006].

LSP State Synchronization (C-E): after the session between the PCC and a stateful PCE with P2MP capability is initialized, the PCE

must learn the state of a PCC's P2MP TE LSPs before it can perform path computations or update LSP attributes in a PCC.

LSP Update Request (E-C): a stateful PCE with P2MP capability requests modification of attributes on a PCC's P2MP TE LSP.

LSP State Report (C-E): a PCC sends an LSP state report to a PCE whenever the state of a P2MP TE LSP changes.

LSP Control Delegation (C-E,E-C): a PCC grants to a PCE the right to update LSP attributes on one or more P2MP TE LSPs; the PCE becomes the authoritative source of the LSP's attributes as long as the delegation is in effect (See Section 5.5 of [I-D.ietf-pce-stateful-pce]); the PCC may withdraw the delegation or the PCE may give up the delegation at any time.

[I-D.sivabalan-pce-disco-stateful]and [RFC6006]defines the IGP extensions needed to support autodiscovery of stateful PCEs with P2MP capability.

5. Architectural Overview of Protocol Extensions

5.1. Extension of PCEP Messages

New PCEP messages are defined in [I-D.ietf-pce-stateful-pce]to support stateful PCE for P2P TE LSPs. In this document these messages are extended to support P2MP TE LSPs.

Path Computation State Report (PCRpt): Each P2MP TE LSP State Report in a PCRpt message can contain actual P2MP TE LSP path attributes, LSP status, etc. An LSP State Report carried on a PCRpt message is also used in delegation or revocation of control of a P2MP TE LSP to/from a PCE. The extension of PCRpt message is described in Section 7.1.

Path Computation Update Request (PCUpd): Each P2MP TE LSP Update Request in a PCUpd message MUST contain all LSP parameters that a PCE wishes to set for a given P2MP TE LSP. An LSP Update Request carried on a PCUpd message is also used to return LSP delegations if at any point PCE no longer desires control of a P2MP TE LSP. The PCUpd message is described in Section 7.2.

5.2. Capability Advertisement

During PCEP Initialization Phase, as per Section 7.1.1 of [I-D.ietf-pce-stateful-pce], PCEP speakers advertises Stateful capability via Stateful PCE Capability TLV in open message and as per Section 3.1 of [RFC6006], PCE advertises P2MP capability via IGP

discovery or a P2MP capable TLV in open message. These mechanism when used together indicates a stateful PCE with P2MP capability.

5.3. State Synchronization

State Synchronization operations described in Section 5.4 of [I-D.ietf-pce-stateful-pce]are applicable for P2MP TE LSPs as well.

5.4. LSP Delegation

LSP delegation operations described in Section 5.5 of [I-D.ietf-pce-stateful-pce]are applicable for P2MP TE LSPs as well.

5.5. LSP Operations

5.5.1. Passive Stateful PCE

LSP operations for passive stateful PCE described in Section 5.6.1 of [I-D.ietf-pce-stateful-pce]are applicable for P2MP TE LSPs as well.

The Path Computation Request and Response message format for P2MP TE LSPs is as per Section 3.4 and Section 3.5 of [RFC6006]respectively.

[Editor’s Note: The Request and Response message should support LSP object, so that it is possible to refer to a LSP with a unique identifier and simplify the PCEP message exchange. for example, incase of modification of one leaf in a P2MP tree, there should be no need to carry the full P2MP tree in PCReq message.]

5.5.2. Active Stateful PCE

LSP operations for active stateful PCE described in Section 5.6.2 of [I-D.ietf-pce-stateful-pce]are applicable for P2MP TE LSPs as well.

6. PCEP Object Extensions

The PCEP TLV defined in this document is compliant with the PCEP TLV format defined in [RFC5440].

6.1. Extension of LSP Object

LSP Object is defined in Section 7.3 of [I-D.ietf-pce-stateful-pce]. It specifies PLSP-ID to uniquely identify an LSP that is constant for the life time of a PCEP session. Similarly for P2MP tunnel, PLSP-ID identify a P2MP TE LSP uniquely. This document adds the following flags to the LSP Object:

N (P2MP bit): If the bit is set to 1, it specifies the message is for P2MP TE LSP which MUST be set in PCRpt or PCUpd message for a P2MP TE LSP.

F (Fragmentation bit): If the bit is set to 1, it specifies the message is fragmented.

If P2MP bit is set, the following P2MP-LSP-IDENTIFIER TLV MUST be present in LSP object.

6.2. P2MP-LSP-IDENTIFIER TLV

The P2MP LSP Identifier TLV MUST be included in the LSP object in PCRpt message for RSVP-signaled P2MP TE LSPs. If the TLV is missing, the PCE will generate an error with error-type 6 (mandatory object missing) and error-value TBD (12) (P2MP-LSP-IDENTIFIERS TLV missing) and close the PCEP session.

The P2MP LSP Identifier TLV MAY be included in the LSP object in PCUpd message for RSVP-signaled P2MP TE LSPs. The special value of all zeros for this TLV is used to refer to all paths pertaining to a particular PLSP-ID.

There are two P2MP LSP Identifier TLVs, one for IPv4 and one for IPv6.

The format of the IPV4-P2MP-LSP-IDENTIFIER TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBD]          |           Length=16           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   IPv4 Tunnel Sender Address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             LSP ID            |           Tunnel ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Extended Tunnel ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             P2MP ID                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: IPV4-P2MP-LSP-IDENTIFIER TLV format

The type of the TLV is [TBD] and it has a fixed length of 12 octets. The value contains the following fields:

IPv4 Tunnel Sender Address: contains the sender node's IPv4 address, as defined in [RFC3209], Section 4.6.2.1 for the LSP_TUNNEL_IPv4 Sender Template Object.

LSP ID: contains the 16-bit 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.1 for the LSP_TUNNEL_IPv4 Sender Template Object.

Tunnel ID: contains the 16-bit 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP_TUNNEL_IPv4 Session Object. Tunnel ID remains constant over the life time of a tunnel.

Extended Tunnel ID: contains the 32-bit 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.1 for the LSP_TUNNEL_IPv4 Session Object.

P2MP ID: contains the 32-bit 'P2MP ID' identifier defined in Section 19.1.1 of [RFC4875]for the P2MP LSP Tunnel IPv4 SESSION Object.

The format of the IPV6-P2MP-LSP-IDENTIFIER TLV is shown in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBD]          |           Length=40           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                  IPv6 tunnel sender address                   |
+                          (16 octets)                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             LSP ID            |           Tunnel ID           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                                                               +
|                       Extended Tunnel ID                      |
+                          (16 octets)                          +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             P2MP ID                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: IPV6-P2MP-LSP-IDENTIFIER TLV format

The type of the TLV is [TBD] and it has a fixed length of 24 octets. The value contains the following fields:

IPv6 Tunnel Sender Address: contains the sender node's IPv6 address, as defined in [RFC3209], Section 4.6.2.2 for the LSP_TUNNEL_IPv6 Sender Template Object.

LSP ID: contains the 16-bit 'LSP ID' identifier defined in [RFC3209], Section 4.6.2.2 for the LSP_TUNNEL_IPv6 Sender Template Object.

Tunnel ID: contains the 16-bit 'Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP_TUNNEL_IPv6 Session Object. Tunnel ID remains constant over the life time of a tunnel.

Extended Tunnel ID: contains the 128-bit 'Extended Tunnel ID' identifier defined in [RFC3209], Section 4.6.1.2 for the LSP_TUNNEL_IPv6 Session Object.

P2MP ID: As defined above in IPV4-P2MP-LSP-IDENTIFIERS TLV.

6.3. S2L Object

The S2L (Source-to-Leaf) Object is used to report RSVP state of one or more destiantions(leaves) encoded within the END-POINTS object for a P2MP TE LSP. It MUST be carried in PCRpt message along with END-POINTS object when N bit is set in LSP object.

The format of the S2L object is shown in the following figure:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Flags                         |    O|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                      Optional TLVs                          //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: S2L object format

Flags(32 bits):

O(Operational - 3 bits) the O Field represents the operational status of the group of destinations. The values are as per Operational field in LSP object defined in Section 7.3 of [I-D.ietf-pce-stateful-pce].

When N bit is set in LSP object then the O field in LSP object represents the operational status of the full P2MP TE LSP and the O field in S2L object represents the operational status of a group of destinations encoded within the END-POINTS object.

Optional TLVs that may be included in the S2L Object.

[Editor’s Note: Further discussions are needed to find the need to encode Sub-Group Originator ID and Sub-Group ID in a TLVs.]

7. PCEP Message Extensions

7.1. The PCRpt Message

As per Section 6.1 of [I-D.ietf-pce-stateful-pce], PCRpt message is used to report the current state of a P2P TE LSP. This document extends the PCRpt message in reporting the status of P2MP TE LSP.

The format of PCRpt message is as follows:

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

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

<state-report> ::= [<SRP>]
                    <LSP>
                    <end-point-path-pair-list>
                    <attribute-list>
Where:

<end-point-path-pair-list>::=
                   [<END-POINTS>]
                   <S2L>
                   <path>
                   [<end-point-path-pair-list>]

<path> ::= (<ERO>|<SERO>)
           [<RRO>]
           [<path>]
```

<attribute-list> is defined in [RFC5440] and extended by PCEP extensions.

The P2MP END-POINTS object defined in [RFC6006]is mandatory for specifying address of P2MP leaves grouped based on leaf types.

- o New leaves to add (leaf type = 1)
- o Old leaves to remove (leaf type = 2)
- o Old leaves whose path can be modified/reoptimized (leaf type = 3)
- o Old leaves whose path must be left unchanged (leaf type = 4)

When reporting the status of a P2MP TE LSP, the destinations are grouped in END-POINTS object based on the operational status (O field in S2L object) and leaf type (in END-POINTS). This way the leaves that share the same operational status are grouped together. For reporing the status of delegated P2MP TE LSP, leaf-type = 3, where as for non-delegated P2MP TE LSP, leaf-type = 4 is used.

For delegated P2MP TE LSP configuration changes are reported via PCRpt message. For example, adding of new leaves END-POINTS (leaf-type = 1) is used where as removing of old leaves (leaf-type = 2) is used.

Note that we preserve compatibility with the [I-D.ietf-pce-stateful-pce]definition of <state-report>. At least one instance of <END-POINTS> MUST be present in this message.

[Editor Note: suggest to add <END-POINTS> object mandatory in [I-D.ietf-pce-stateful-pce]document for <state-report>].

During state synchronization, the PCRpt message must report the status of the full P2MP TE LSP.

7.2. The PCUpd Message

As per Section 6.2 of [I-D.ietf-pce-stateful-pce], PCUpd message is used to update P2P TE LSP attributes. This document extends the PCUpd message in updating the attributes of P2MP TE LSP.

The format of a PCUpd message is as follows:

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

Where:

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

<update-request> ::= <SRP>
                     <LSP>
                     <end-point-path-pair-list>

<attribute-list>
   Where:

   <end-point-path-pair-list>::=
                  [<END-POINTS>]
                  <path>
                  [<end-point-path-pair-list>]

   <path> ::= (<ERO>|<SERO>)
              [<path>]

   <attribute-list> is defined in [RFC5440] and
   extended by PCEP extensions.
```

Note that we preserve compatibility with the [I-D.ietf-pce-stateful-pce]definition of <update-request>.

7.3. Example

7.3.1. P2MP TE LSP Update Request

LSP Update Request message is sent by an active stateful PCE to update the P2MP TE LSP parameters or attributes. An example of a PCUpd message for P2MP TE LSP is described below:

```
Common Header
SRP
LSP with P2MP flag set
END-POINTS for leaf type 3
  ERO list
```

In this example, a stateful PCE request updation of path taken by some of the leaves in a P2MP tree. The update request uses the END-POINT type 3 (modified/reoptimized). The ERO list represents the S2L

path after modification. The update message does not need to encode the full P2MP tree in this case.

7.3.2. P2MP TE LSP Report

LSP State Report message is sent by a PCC to report or delegate the P2MP TE LSP. An example of a PCRpt message for a delegated P2MP TE LSP is described below to add new leaves to an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 1
  S2L (O=DOWN)
  ERO list (empty)
```

An example of a PCRpt message for P2MP TE LSP is described below to prune leaves from an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 2
  S2L (O=UP)
  ERO list
```

An example of a PCRpt message for a delegated P2MP TE LSP is described below to report status of leaves in an existing P2MP TE LSP:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 3
  S2L (O=UP)
  ERO list
END-POINTS for leaf type 3
  S2L (O=DOWN)
  ERO list
```

An example of a PCRpt message for a non-delegated P2MP TE LSP is described below to report status of leaves:

```
Common Header
LSP with P2MP flag set
END-POINTS for leaf type 4
  S2L (O=ACTIVE)
  ERO list
END-POINTS for leaf type 4
  S2L (O=DOWN)
  ERO list
```

7.4. Report and Update Message Fragmentation

The total PCEP message length, including the common header, is 16 bytes. In certain scenarios the P2MP report and update request may not fit into a single PCEP message (initial report or update). The F-bit is used in the LSP object to signal that the initial report or update was too large to fit into a single message and will be fragmented into multiple messages. In order to identify the single report or update, each message will use the same PLSP-ID.

Fragmentation procedure described below for report or update message is similar to [RFC6006]which describes request and response message fragmentation.

7.4.1. Report Fragmentation Procedure

If the initial report is too large to fit into a single report message, the PCC will split the report over multiple messages. Each message sent to the PCE, except the last one, will have the F-bit set in the LSP object to signify that the report has been fragmented into multiple messages. In order to identify that a series of report messages represents a single report, each message will use the same PLSP-ID.

7.4.2. Update Fragmentation Procedure

Once the PCE computes and updates a path for some or all leaves in a P2MP TE LSP, an update message is sent to the PCC. If the update is too large to fit into a single update message, the PCE will split the update over multiple messages. Each update message sent by the PCE, except the last one, will have the F-bit set in the LSP object to signify that the update has been fragmented into multiple messages. In order to identify that a series of update messages represents a single update, each message will use the same PLSP-ID and SRP-ID-number.

[Editor Note: P2MP message fragmentation errors associated with a P2MP path report and update will be defined in future version].

8. Non-Support of P2MP TE LSPs for Stateful PCE

The PCEP protocol extensions described in this document for stateful PCEs with P2MP capability MUST NOT be used if PCE has not advertised its stateful capability with P2MP as per Section 5.2. If this is not the case and Stateful operations on P2MP TE LSPs are attempted, then a PCErr with error-type 19 (Invalid Operation) and error-value TBD needs to be generated.

If a Stateful PCE receives a P2MP TE LSP report message and it understands the P2MP flag in the LSP object, but the stateful PCE is not capable of P2MP computation, the PCE MUST send a PCErr message with error-type 19 (Invalid Operation) and error-value TBD.

If a Stateful PCE receives a P2MP TE LSP report message and the PCE does not understand the P2MP flag in the LSP object, and therefore the PCEP extensions described in this document, then the PCE SHOULD reject the request.

[Editor Note: more information on exact error value is needed]

9. Security Considerations

TBD

10. Manageability Considerations

10.1. Control of Function and Policy

TBD.

10.2. Information and Data Models

TBD.

10.3. Liveness Detection and Monitoring

TBD.

10.4. Verify Correct Operations

TBD.

10.5. Requirements On Other Protocols

TBD.

10.6. Impact On Network Operations

TBD.

11. IANA Considerations

This document requests IANA actions to allocate code points for the protocol elements defined in this document. Values shown here are suggested for use by IANA.

11.1. Extension of LSP Object

This document requests that a registry is created to manage the Flags field of the LSP object. New values are to be assigned by Standards Action [RFC5226]. 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
24	P2MP	This.I-D
23	Fragmentation	This.I-D

11.2. Extension of PCEP-Error Object

A new error types 6 and 19 defined in section 8.4 of [I-D.ietf-pce-stateful-pce]. This document extend the new Error-Values for those error types for the following error conditions:

```
Error-Type  Meaning
   6        Mandatory Object missing
              Error-value=12: P2MP-LSP-IDENTIFIER TLV missing
   19       Invalid Operation
              Error-value= TBD.
```

11.3. PCEP TLV Type Indicators

This document defines the following new PCEP TLVs:

Value	Meaning	Reference
22	P2MP-IPV4-LSP-IDENTIFIERS	This.I-D
23	P2MP-IPV6-LSP-IDENTIFIERS	This.I-D

12. Acknowledgments

Thanks to Quintin Zhao for his comments.

13. References

13.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[I-D.ietf-pce-stateful-pce] Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

13.2. 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, December 2001.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC4857] Fogelstroem, E., Jonsson, A., and C. Perkins, "Mobile IPv4 Regional Registration", RFC 4857, June 2007.

[RFC4875] Aggarwal, R., Papadimitriou, D., and S. Yasukawa, "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, May 2007.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5671] Yasukawa, S. and A. Farrel, "Applicability of the Path Computation Element (PCE) to Point-to-Multipoint (P2MP) MPLS and GMPLS Traffic Engineering (TE)", RFC 5671, October 2009.

[RFC6006] Zhao, Q., King, D., Verhaeghe, F., Takeda, T., Ali, Z., and J. Meuric, "Extensions to the Path Computation Element Communication Protocol (PCEP) for Point-to-Multipoint Traffic Engineering Label Switched Paths", RFC 6006, September 2010.

[I-D.ietf-pce-stateful-pce-app]
Zhang, X. and I. Minei, "Applicability of Stateful Path Computation Element (PCE)", draft-ietf-pce-stateful-pce-app-01 (work in progress), September 2013.

[I-D.sivabalan-pce-disco-stateful]
Sivabalan, S., Medved, J., and X. Zhang, "IGP Extensions for Stateful PCE Discovery", draft-sivabalan-pce-disco-stateful-03 (work in progress), January 2014.

Authors' Addresses

Udayasree Palle
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: udayasree.palle@huawei.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.ietf@gmail.com

Yosuke Tanaka
NTT Communications Corporation
Granpark Tower
3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

EMail: yosuke.tanaka@ntt.com

Yuji Kamite
NTT Communications Corporation
Granpark Tower
3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

EMail: y.kamite@ntt.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 17, 2014

Y. Tanaka
Y. Kamite
NTT Communications
I. Minei
Google
D. Dhody
Huawei Technologies
Feb 13, 2014

Stateful PCE Extensions for Data Plane Switchover and Balancing
draft-tanaka-pce-stateful-pce-data-ctrl-02

Abstract

Stateful Path Computation Element (PCE) and its corresponding protocol extensions provide a mechanism that enables PCE to do stateful control of Multiprotocol Label Switching (MPLS) Traffic Engineering Label Switched Paths (TE LSP). One application that stateful PCE can realize is data traffic reoptimization among the LSPs. Data traffic traversed in a LSP can be switched to another PCE-initiated LSP. Moreover, data traffic can be balanced to multiple PCE-initiated LSPs and may also be policed based on a signaling bandwidth of a PCE-Initiated LSP using stateful PCE.

This document specifies the extensions to Path Computation Element Protocol (PCEP) that allow a stateful PCE to do switchover, balancing and policing of data traffic with PCE-initiated LSPs. This document also specifies the extensions, usage and handling of stateful PCEP messages and the expected behavior of PCC as the RSVP-TE headend.

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 http://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 17, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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. Terminology 4
4. PCEP Operation for Data Switchover and Balancing 4
5. TLVs in LSP Objects 6
  5.1. ASSOCIATION-GROUP TLV in LSP Objects 6
  5.2. DATA-CONTROL TLV in LSP Objects 8
  5.3. DATA-REPORT TLV in LSP Objects 10
  5.4. Advertising Support of Data Switchover and Balancing . . . 11
6. Operation Examples 11
  6.1. Data switchover operation (100:0 => 0:100) 11
  6.2. Load balancing operation (100:0 => 50:50) 13
  6.3. Load balancing operation (100:0 => 66:33) 14
7. Redundant stateful PCEs 15
8. Security Considerations 16
  8.1. Malicious PCE 16
  8.2. Malicious PCC 16
9. IANA Considerations 16
  9.1. PCEP TLV Indicators 17
  9.2. PCEP Error Objects 17
10. Acknowledgments 17
11. References 17
  11.1. Normative References 17
  11.2. Informative References 18
Authors' Addresses 18

1. Introduction

[I-D.ietf-pce-stateful-pce] describes the stateful Path Computation Elements (PCE) procedures and defines the extensions to PCEP to enable stateful control of LSPs between and across PCEP sessions, further it also describes mechanisms to effect LSP state synchronization between PCCs and PCEs, and PCE control of timing and sequence of path computations within and across PCEP sessions. A PCE can update LSP settings (such as bandwidth, priority, path) using an update message (called PCUpd).

[I-D.ietf-pce-pce-initiated-lsp] defines the extensions to PCEP to allow a PCE to instantiate new LSPs (called PCE-Initiated LSPs). Before these extensions, the LSP ingress point had to be preconfigured at the head end Label Edge Router (LER), the LSP control automatically delegated to the initiating stateful PCE and then its parameters (e.g., bandwidth, priority, path) could be modified via a PCUpd message. The extensions for PCE-initiated LSPs eliminate the need for preconfiguration, and allow more flexible operations on the network. Stateful-PCE with LSP instantiation is attracting attention as an enabler for Software Defined Networking (SDN) operation of MPLS networks.

In SDN, it is highly expected to support intelligent and interactive control of the amount of network traffic by means of a logically-centralized controller. Optimizing the path and bandwidth of MPLS-TE LSP by using stateful PCE is a leading use case of SDN applications. A PCE is able to calculate an optimized route from the topology and bandwidth information in the Traffic Engineering Database (TED) and the LSP state database (LSPDB) and it can integrate with a controller that takes into account additional information such as historical trends and service orders to trigger some PCE actions. For example, when data traffic on a LSP increases the bandwidth utilization and if there is no capacity left in the currently signaled path (i.e., no remaining bandwidth of links), a PCE is able to update the existing LSP’s parameters (PCE-updated LSP) or create a totally new LSP (PCE-initiated LSP).

The former method is oriented for keeping the existing instance of LSP tunnel. Meanwhile, the latter method is oriented for adding a new instance of a LSP tunnel.

Specifically regarding the latter method, PCE-initiated LSP, there are some operational scenarios in the network: one is that PCE instantiate a new LSP that have alternate route with increased-bandwidth LSP and performs switchover from old LSP. Another is that PCE creates one or more additional LSPs and performs load balancing of data traffic. Today, however, there is no detailed procedure

specified as to how to control data traffic switching from an old LSP to new PCE-Initiated LSP(s).

For another example, when data traffic on a LSP increases its bandwidth utilization and if there is strict traffic contract, a PCE is able to force a PCC not to exceed the contract bandwidth.

This document specifies the procedures that a stateful PCE can use to control data traffic switchover, load balancing with multiple PCE-Initiated LSPs and policing activation/deactivation. This document also specifies the usage and handling of stateful PCEP messages and the expected behavior of PCC as an RSVP-TE headend.

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 RFC2119[RFC2119].

3. Terminology

This document uses the following terms defined in [RFC5440]: PCC, PCE, PCEP Peer.

This document uses the following terms defined in [I-D.ietf-pce-stateful-pce]: Stateful PCE, LSP State Request, LSP Update Request.

This document uses the following terms defined in [I-D.ietf-pce-pce-initiated-lsp]: LSP Initiate Message.

4. PCEP Operation for Data Switchover and Balancing

There are two typical operations for explaining the functionality of data switchover and balancing.

- o Whole data switchover, where a PCC switches all data traffic from one LSP tunnel to another.
- o Load balancing of multi-instance LSP tunnels with different paths, where a PCC (headend) balances data traffic among two or more tunnels (ex fifty percent each, for two instances).

Both operational cases are completed by the messaging over a single protocol, PCEP, keeping this as a simple and straightforward solution for MPLS networks.

A PCEP speaker indicates its ability to support PCE control over the data switchover and balancing during the PCEP Initialization phase. The Open Object in the Open message contains the "Stateful PCE Capability" TLV, defined in [I-D.ietf-pce-stateful-pce]. A new flag, the W (LSP-DATASWITCHOVER-BALANCE-CAPABILITY) flag is introduced. A PCE can control the data switchover and loadbalancing only for PCCs that advertised this capability and a PCC will follow the procedures described in this document only on sessions where the PCE advertised the W flag. (Refer Section 5.4)

Data switchover and balancing for an MPLS-TE LSP is available once a PCEP session is established and then a PCC delegates its LSPs to a PCE.

First step is LSP instantiation. In this step, a PCE sends as many PCInitiate messages as PCE-Initiated LSP as per demand. Once the PCC receives them and successfully establishes PCE-Initiated LSPs, it sends PCRpt messages in reply to the PCInitiate messages and delegates the newly established LSP to the PCE. Message formats and behaviors of the PCC and the PCE are described in detail in [I-D.ietf-pce-pce-initiated-lsp].

Second step is LSP association. After the PCE-Initiated LSP successfully established and delegated the PCE sends a PCUpd message that contains the ASSOCIATION-GROUP TLV in the LSP Object in order to assemble the members of an association group of LSPs to take over the traffic. Once a PCC receives the PCUpd message with ASSOCIATION-GROUP TLV, the PCC sends back a PCRpt message that contains the ASSOCIATION-GROUP TLV with current operational status.

[Editor’s Note: The option of specifying the association at LSP instantiation time (as part of the PCInitiate message) will be evaluated in a future version of this document.]

Third step is executing the data switchover and/or load balancing. In this step, the PCE sends a single PCUpd message which updates the operational status of the LSP from "up and carrying traffic" to just "up". This Update request message for data plane switchover/ balancing execution MUST contain DATA-CONTROL TLV in LSP Object. The associated group of traffic origin and that of target to take over the traffic are listed in the DATA-CONTROL TLV. The PCC (LSP headend) load-balances between LSPs in the same association group based on their respective bandwidths. The switchover case is supported since there will be an association of a single LSP, so that LSP will get hundred percent of data traffic.

The PCC MUST send a PCRpt message to the PCE in order to notify of the result of the data switchover/balancing. The PCRpt message MUST

have the DATA-CONTROL TLV that indicates the actual assigned percentages of each member of association group after the execution of the data switchover/balancing operation. The LSP object in the PCRpt will have the reserved PLSP-ID of 0.

The final step is the deletion of old LSP. It is OPTIONAL to carry out this step. The PCE sends PCInitiate message requesting deletion of the LSP that does not carry data traffic anymore after data switchover/balancing execution. Once the PCC tears down the LSP, a PCRpt message MUST be sent from the PCC to the PCE in order to notify that the LSP is no longer used.

Note that, both RSVP-TE [RFC3209] Tunnel-ID and LSP-ID for PCE-Initiated LSP signaling is not allocated by a PCE. A PCC locally assigns those IDs that are related to RSVP-TE parameters. Therefore, the operations of data switchover and balancing specified in this document is the traffic control procedure across multiple RSVP-TE Tunnels (i.e., different Tunnel instances). Data switchover method across LSPs within a single RSVP-TE Tunnel, which is the switchover in the middle of make-before-break reoptimization, is covered by [I-D.tanaka-pce-stateful-pce-mbb].

5. TLVs in LSP Objects

5.1. ASSOCIATION-GROUP TLV in LSP Objects

This section defines ASSOCIATION-GROUP TLV in LSP Objects. An ASSOCIATION-GROUP TLV is used in the LSP Object in PCUpd messages when a PCE creates an association group of LSPs on a PCC. Further it is used in a LSP object in a PCRpt message to confirm the association.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=TBD            |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Association Group ID              |     Flags     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
 0                   1                   2                   3
```

```
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=TBD            |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Association Group ID              |     Flags     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: ASSOCIATION-GROUP TLV format

Flags and fields

Association Group ID - 24 bits: This field specifies a identifier of association group of LSPs. The IDs are assigned by a PCE. 0x000000 and 0xFFFFFF is reserved for special use.
Flags - 8 bit: None defined. MUST be set to zero.

An association group is a group of LSPs that is referenced by a single identifier, by both the PCE and PCC. This number is significant in the context of a single PCEP session. An association group may have one or more LSPs. Association groups with zero members are removed and the id can be reused. The PCE is the entity managing association, and this is considered PCE’s state that will be cleaned up when the State Timeout Interval expires.

The status of the association group is active when the group is up and carrying data traffic. Otherwise, it is inactive when the group does not carry any data traffic. An LSP is able to associate with up to two association groups, unless both association groups are active at any given point in time. This is done to allow a new LSP to be instantiated and assigned with a new inactive association group, the existing LSP is also associated with this group. This allows switching to the new group.

To create a new association group on a PCC, a PCE sends a PCUpd message which contains the LSP Object(e.g. PLSP-ID=100) and ASSOCIATION-GROUP TLV (Association Group ID=10) in the LSP object. Next, a PCE sends the another PCUpd message with another LSP Object(e.g. PLSP-ID=200) and ASSOIATION-GROUP TLV(Association Group ID=10). As a result, the PCC and PCE both recognize that Association Group ID 10 represents PLSP-ID=100 and 200.

To remove a specific PLSP-ID from the association group, a PCE sends PCUpd message which contains the LSP Object(PLSP-ID=100) and ASSOCIATION-GROUP TLV (Association Group ID=0x0000). Then a PCC removes the PLSP-ID 100 from any inactive association group on the

PCC.

To flush all association groups on a PCC, a PCE sends a PCUpd message which contains the LSP Object(PLSP-ID=0x0000) and ASSOCIATION-GROUP TLV(Association Group ID=0x0000). Then a PCC flushes all association groups. A traffic handling behavior of a PCC when it flushes the active association group is left for a future version of this document.

To associate a PLSP-ID with the existing inactive association group, A PCE sends a PCUpd message which contains the PLSP-ID and the existing Association Group ID. A PCE is not allowed to add any PLSP-ID to the active association group in order to avoid rebalancing traffic without data-ctrl requests. If the PCUpd message contains a PLSP-ID and the active Association Group ID, the PCC MUST send out a PCErr with error value TBD to indicate an invalid operation.

When the LSP of the active association group is torn down by a reason of either network failure or administrative down-request from the PCE, a PCC MUST remove the PLSP-ID from the group and rebalance the traffic based on the respective bandwidths of the rest of LSPs. After rebalancing, The PCC MUST report the actual percentage to the PCE using PCRpt with DATA-REPORT TLV (Section 5.3).

Note that a PCE is able to associate not only PCE-Initiated LSP but also existing LSP(i.e., PCE-updated LSP) with any association group on a PCC.

The definition of PCRpt messages when a PCC creates/removes/flushes an association group will be clarified in the future version of this draft. Redundant stateful PCE section needs the PCRpt in order to sync the association group IDs and actual percentages of balancing.

5.2. DATA-CONTROL TLV in LSP Objects

This section defines DATA-CONTROL TLV in LSP Objects. A DATA-CONTROL TLV is used in the LSP Object in PCUpd messages when a PCE makes a PCC to execute traffic switchover or load balancing. It is also mandatory in a LSP object in a PCRpt message with DATA-REPORT TLV to notify the results of execution.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=TBD            |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Origin Association Group ID            | Flags   |  O  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Target Association Group ID            | Flags |P|  O  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: DATA-CONTROL TLV format

Flags and fields

Origin Association Group ID - 24 bits: data traffic origin
Target Association Group ID - 24 bits: for taking over whole data traffic from origin.
P (Policing - 1 bit: This flag is used when a PCE makes a PCC apply traffic policer. If this flag is set 1, traffic exceeding the bandwidth of the LSP is discarded on the PCC after traffic switchover execution. Otherwise, the PCC does not apply any traffic policer and traffic on a target association group will not be discarded.
O (Operational - 3 bits): This flag represents the requested operational status for each Origin Association Group ID and Target Association Group ID by a PCE when this TLV is used in a PCUpd message. It is also used as a status report in a PCRpt message. The meanings of the values are defined in [I-D.ietf-pce-stateful-pce].

An LSP Object in a PCUpd message MUST have DATA-CONTROL TLV when a PCE operates data switchover and balancing on a PCC. DATA-CONTROL TLV is sub-TLV of an LSP Object and is used in both a PCUpd and a PCRpt message.

An operation of data switchover/balancing is the action of transferring traffic from an origin association group to a target association group. A PCUpd message with reserved LSP Object (PLSP-ID=0x00000) and DATA-CONTROL TLV (a set of an origin and a target association group) MUST triggers data switchover/balancing execution.

Traffic policer is able to be applied in both traffic switchover case and load-balancing case.

5.3. DATA-REPORT TLV in LSP Objects

This section defines DATA-REPORT TLV in LSP Objects. A DATA-REPORT TLV is used in the LSP Object in PCRpt message to notify the results of execution with the DATA-CONTROL TLV.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=TBD            |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Member 1 (PLSP-ID )              |  Flags  |  Percentage |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Member 2 (PLSP-ID )              |  Flags  |  Percentage |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
//                                                             //
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Member N (PLSP-ID )              |  Flags  |  Percentage |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: DATA-REPORT TLV format

Flags and fields

Member(PLSP-ID) - 20 bit: This TLV is only used in a PCRpt message and represents actual percentages of load balancing per respective PLSP-ID after load balancing execution. Member field fills PLSP-ID that is member of target association group. As per [I-D.ietf-pce-stateful-pce].

Flags - 5 bit: None defined. MUST be set to zero.

Percentage - 7 bits: This field specifies actual percentage of load balancing as a closest integer, with 100% as the max allowed value.

A PCC replies to a PCE a PCRpt message as an acknowledgment of data switchover/balancing result. The PCRpt message MUST have reserved LSP Object(PLSP-ID=0x00000) and DATA-CONTROL TLV with DATA-REPORT TLV inside.

The PCC load-balances between LSPs in the same association group based on their respective bandwidths.If one of the LSPs goes down by network failure, the traffic would load-balance correctly over the others. If a PCE updates the bandwidth of the LSP, the traffic would

rebalance after a PCC completes the signaling. If one of the LSPs is signaled with zero bandwidth, no traffic would be transferred to the LSP. If all LSPs of the association group are signaled with zero bandwidth, the traffic would load-balance equally. In switchover case, the hundred percent traffic will be transferred to the LSP even if the LSP is zero bandwidth.

The traffic on the existing LSP is able to load-balance over both the existing LSP itself and new PCE-Initiated LSPs, by means of that the existing LSP belongs to both the origin association group and that of target.

5.4. Advertising Support of Data Switchover and Balancing

New flags are defined for the STATEFUL-PCE-CAPABILITY TLV defined in [I-D.ietf-pce-stateful-pce].

W (LSP-DATASWITCHOVER-BALANCE-CAPABILITY - 1 bit): if set to 1 by a PCEP speaker, it indicates that the PCEP speaker allows data switchover and balancing.

6. Operation Examples

For easy understanding this section introduces typical operation examples of data switchover/balancing.

6.1. Data switchover operation (100:0 => 0:100)

A PCE instructs a PCC to switchover 100% traffic from association group ID 1 to association group ID 2. A PCE sends single PCUpd message containing the mandatory LSP Objects with DATA-CONTROL TLV.

Expected PCUpd, PCRpt messages to create association group and to trigger data switchover follow.

```
    PCE                          PCC(Ingress)     Egress
   [LSP Association for existing LSP]
     |                           |                |
     | --PCUpd ----------------->|                |
     |   LSP Obj: PLSP-ID=1      |                |
     |   + ASSOC-G: Assoc-G-ID 10|                |
     |                           |                |
     |<--PCRpt ----------------- |                |
     |   LSP Obj: PLSP-ID=1      |                |
     |   + ASSOC-G: Assoc-G-ID 10|                |

   [LSP Creation]
     |                           |                |
     | --PCInitiate ------------>|                |
     |                           | --Path ------->|
     |                           |<------- Resv-- | Establish a new
     |<--PCRpt ----------------- |                | PCE-Initiated LSP
     |   LSP Obj: PLSP-ID=2      |                |
     |                           |                |

   [LSP Association for PCE-Initiated LSP]
     |                           |                |
     | --PCUpd ----------------->|                |
     |   LSP Obj: PLSP-ID=2      |                |
     |   + ASSOC-G: Assoc-G-ID 20|                |
     |                           |                |
     |<--PCRpt ----------------- |                |
     |   LSP Obj: PLSP-ID=2      |                |
     |   + ASSOC-G: Assoc-G-ID 20|                |
     |                           |                |

   [Switchover Execution]
     |                           |                |
     | --PCUpd ----------------->|                |
     |   LSP Obj: PLSP-ID=0x0000 |                |
     |   + D-CTRL:               |        :       |
     |    Origin Assoc-G-ID 10(O=up)      :       |
     |    Target Assoc-G-ID 20(O=active)  :       |
     |                           |))))))))))))))))| Switchover
     |                           |}}}}}}}}}}}}}}}}| Execution
     |<--PCRpt------------------ |        :       |
     |   LSP Obj: PLSP-ID=0x0000 |        :       |
     |   + D-CTRL:               |        :       |
     |    Origin Assoc-G-ID 10(O=up)              |
     |    Target Assoc-G-ID 20(O=active)          |
     |   + D-REPORT:             |                |
     |    PLSP-ID 2, 100%        |                |
     |                           |                |
```

Figure 4: Switchover Operation Example

6.2. Load balancing operation (100:0 => 50:50)

The scenario is one where the starting state is a single LSP (of bandwidth 100 M) is carrying the traffic. To enable better bin-packing, the PCE may want to create two smaller LSPs instead, each of 50M, and load balance the traffic over them. To accomplish this, two association groups are used, the first (say association group ID 10) contains the LSP carrying the traffic, and the second (say association group ID 30) contains the two new smaller LSPs. Expected PCUpd, PCRpt messages to create association group and to trigger load-balance follow (The instantiation of the original LSP of bandwidth 100M and its association into group ID 10 is not shown)

```
 PCE                          PCC(Ingress)     Egress

[LSP Creation]
  |                           |                |
  | --PCInitiate x2---------->|                |
  |      BW: 50M              | --Path x2----->|
  |                           |<-----Resv x2-- | Establish two new
  |<--PCRpt ----------------- |                | PCE-Initiated LSP
  |   LSP Obj: PLSP-ID=3      |                |
  |<--PCRpt ----------------- |                |
  |   LSP Obj: PLSP-ID=4      |                |
  |                           |                |


[LSP Association for PCE-Initiated LSPs]
  |                           |                |
  | --PCUpd ----------------->|                | Create new
  |   LSP Obj: PLSP-ID=3      |                | Association Group
  |   + ASSOC-G: Assoc-G-ID 30|                | for PCE-Initiated
  |                           |                | LSP
  |<--PCRpt ----------------- |                |
  |   LSP Obj: PLSP-ID=3      |                |
  |   + ASSOC-G: Assoc-G-ID 30|                |
  |                           |                |
  | --PCUpd ----------------->|                | Add a new LSP
  |   LSP Obj: PLSP-ID=4      |                | to Association Group
  |   + ASSOC-G: Assoc-G-ID 30|                |
  |                           |                |
  |<--PCRpt ----------------- |                |
  |   LSP Obj: PLSP-ID=4      |                |
  |   + ASSOC-G: Assoc-G-ID 30|                |
```

```
   [Load Balancing Execution]
     | --PCUpd------------------>|                |
     |   LSP Obj: PLSP-ID=0x0000 |                |
     |   + D-CTRL:               |        :       |
     |    Origin Assoc-G-ID 10(O=up)      :       |
     |    Target Assoc-G-ID 30(O=active)  :       |
     |                           |))))))))))))))))| Balancing
     |                           |)})})})})})})})})| Execution
     |                           |        :       |
     |<--PCRpt------------------ |        :       |
     |   LSP Obj: PLSP-ID=0x0000 |        :       |
     |   + D-CTRL:               |        :       |
     |    Origin Assoc-G-ID 10(O=up)              |
     |    Target Assoc-G-ID 30(O=active)          |
     |   + D-REPORT:             |                |
     |    PLSP-ID 3, 50%         |                |
     |    PLSP-ID 4, 50%         |                |
     |                           |                |
```

Figure 5: Load-Balance Operation Example

6.3. Load balancing operation (100:0 => 66:33)

The scenario is one where the starting state is a single LSP (of bandwidth 100 M) is carrying the traffic. But as the data traffic load increases another 50 M is required. The PCE may want to create another LSP of 50 M, and load balance the traffic over the existing and new LSP. To accomplish this, two association groups are used, the first (say association group ID 10) contains the LSP carrying the traffic, and the second (say association group ID 40) contains the new initiated LSP as well as the original LSP. Expected PCUpd, PCRpt messages to create association group and to trigger load-balance follow (The instantiation of the original LSP of bandwidth 100M and its association into group ID 10 is not shown)

```
    PCE                         PCC(Ingress)     Egress

   [LSP Creation]
     |                           |                |
     | --PCInitiate ------------>|                |
     |      BW: 50M              | --Path ------->|
     |                           |<-----Resv ---- | Establish new
     |<--PCRpt ----------------- |                | PCE-Initiated LSP
     |   LSP Obj: PLSP-ID=5      |                |
     |                           |                |
```

```
   [LSP Association for PCE-Initiated LSPs]
     |                            |                 |
     | --PCUpd ----------------->|                 | Create new
     |   LSP Obj: PLSP-ID=5      |                 | Association Group
     |   + ASSOC-G: Assoc-G-ID 40|                 | for PCE-Initiated
     |                           |                 | LSP
     |<--PCRpt ----------------- |                 |
     |   LSP Obj: PLSP-ID=5      |                 |
     |   + ASSOC-G: Assoc-G-ID 40|                 |
     |                           |                 |
     | --PCUpd ----------------->|                 | Add the old LSP
     |   LSP Obj: PLSP-ID=1      |                 | to the Association
     |   + ASSOC-G: Assoc-G-ID 40|                 | Group
     |                           |                 |
     |<--PCRpt ----------------- |                 |
     |   LSP Obj: PLSP-ID=1      |                 |
     |   + ASSOC-G: Assoc-G-ID 40|                 |


   [Load Balancing Execution]
     | --PCUpd------------------>|                 |
     |   LSP Obj: PLSP-ID=0x0000 |                 |
     |   + D-CTRL:               |        :        |
     |    Origin Assoc-G-ID 10(O=up)      :        |
     |    Target Assoc-G-ID 40(O=active)  :        |
     |                           |))))))))))))))))| Balancing
     |                           |)})})})})})})})})| Execution
     |                           |        :        |
     |<--PCRpt------------------ |        :        |
     |   LSP Obj: PLSP-ID=0x0000 |        :        |
     |   + D-CTRL:               |        :        |
     |    Origin Assoc-G-ID 10(O=up)               |
     |    Target Assoc-G-ID 40(O=active)           |
     |   + D-REPORT:             |                 |
     |    PLSP-ID 1, 66%         |                 |
     |    PLSP-ID 5, 33%         |                 |
     |                           |                 |
```

Figure 6: Load-Balance Operation Example

7. Redundant stateful PCEs

Association group IDs are unique within a PCEP session across the primary PCE and the PCC. A backup PCE has to synchronize the

association group IDs, PCE that created the association group and balancing percentages in advance of the failure on the primary PCE. One practical method to synchronize is a PCC replicates each PCRpt message for the backup PCEP session. A backup PCE is able to receive the association group IDs from ASSOCIATION-GROUP TLV and the result of balancing percentages from DATA-REPORT TLV.

8. Security Considerations

This document defines extensions to PCEP to control load balancing of traffic across multiple LSPs or to completely switch traffic from one LSP to another. The nature of these extensions results in more information being available for a hypothetical adversary and a number of additional attack surfaces which must be protected. As a general precaution, 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

In addition to the security considerations and recommendations described in [I-D.ietf-pce-stateful-pce], the following also apply.

8.1. Malicious PCE

A malicious PCE may flap the traffic between several LSPs, creating shifting patterns in the network and excessive load on the PCC. A PCC may protect itself from such an attack by enforcing a limit on the number of data-control requests per unit of time and MAY take additional steps ranging from delegation revocation to closing the PCEP session.

8.2. Malicious PCC

Because the PCE keeps state regarding LSP associations for all the PCCs, it is RECOMMENDED that the PCE have a bound on the amount of state each PCC can occupy, and in the context of this draft, the number of associations on a PCC and the number of associations each LSP may be part of. Otherwise, a malicious PCC may create an unbounded number of associations. Additionally, a malicious PCC may purposely fail data-control messages in order to force the PCE to continuously resend them and create artificial load on the PCE. The PCE may protect itself from these situations by placing a limit on the number of failures and closing the PCEP session.

9. IANA Considerations

9.1. PCEP TLV Indicators

This document defines the following new PCEP TLVs:

Value	Meaning	Reference
TBD	DATA-CONTROL	This document
TBD	DATA-REPORT	This document

9.2. PCEP Error Objects

This document defines new Error-Type and Error-Value for the following new error conditions:

Error-Type	Meaning	
6	Mandatory Object missing	
	Error-value=TBD:	DATA-CONTROL TLV missing.
	Error-value=TBD:	DATA-REPORT TLV missing.
19	Invalid operation	
	Error-value=TBD:	No association group existing.
	Error-value=TBD:	No association group specified.
	Error-value=TBD:	No PLSP can be added to the active association group.

10. Acknowledgments

Many thanks to Adrian Farrel for their ideas and suggestions.

11. References

11.1. Normative References

[I-D.ietf-pce-pce-initiated-lsp]
Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-ietf-pce-pce-initiated-lsp-00 (work in progress), December 2013.

[I-D.ietf-pce-stateful-pce]
Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4872] Lang, J., Rekhter, Y., and D. Papadimitriou, "RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery", RFC 4872, May 2007.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

11.2. Informative References

[I-D.tanaka-pce-stateful-pce-mbb]
Tanaka, Y. and Y. Kamite, "Make-Before-Break MPLS-TE LSP restoration and reoptimization procedure using Stateful PCE", draft-tanaka-pce-stateful-pce-mbb-02 (work in progress), October 2013.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

Authors' Addresses

Yosuke Tanaka
NTT Communications Corporation
Granpark Tower
3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

Email: yosuke.tanaka@ntt.com

Yuji Kamite
NTT Communications Corporation
Granpark Tower
3-4-1 Shibaura, Minato-ku
Tokyo 108-8118
Japan

Email: y.kamite@ntt.com

Ina Minei
Google
US

Email: inaminei@google.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

Email: dhruv.ietf@gmail.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: July 30, 2014

Q. Wu
D. Dhody
Huawei
D. King
Old Dog Consulting
D. Lopez
Telefonica I+D

January 26, 2014

Path Computation Element (PCE) Discovery using Domain Name System(DNS)
draft-wu-pce-dns-pce-discovery-04

Abstract

Discovery of the Path Computation Element (PCE) within an IGP area or routing domain is possible using OSPF [RFC5088] and IS-IS [RFC5089]. However, it has been established that in certain deployment scenarios PCEs may not wish, or be able to participate within the IGP process. In those scenarios, it is beneficial for the Path Computation Client (PCC) (or other PCE) to discover PCEs via an alternative mechanism to those proposed in [RFC5088] and [RFC5089].

This document specifies the requirements, use cases, procedures and extensions to support PCE type and capability discovery via DNS.

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 http://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 July 30, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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. Terminology 3
  1.2. Requirements 3
2. Conventions used in this document 5
3. Motivation 6
  3.1. Outside the Routing Domain 6
  3.2. Discovery Mechanisms 7
    3.2.1. Query-Response versus Advertisement 7
  3.3. Network Address Translation Gateway 7
4. Additional Capabilities 8
  4.1. Load Sharing of Path Computation Requests 8
5. Extended Naming Authority Pointer (NAPTR)Service Field Format 9
  5.1. IETF Standards Track PCE Applications 10
6. Backwards Compatibility 11
7. Discovering a Path Computation Element 12
  7.1. Determining the PCE Service and transport protocol 13
  7.2. Determining the IP Address of the PCE 13
    7.2.1. Examples 15
  7.3. Determining the PCE domains and Neighbor PCE domains . . . 16
8. IANA Considerations 17
  8.1. IETF PCE Application Service Tags 17
  8.2. PCE Application Protocol Tags 17
9. Security Considerations 18
10. Acknowledgements 19
11. References 20
  11.1. Normative References 20
  11.2. Informative References 21
Authors' Addresses 23

1. Introduction

The Path Computation Element Communication Protocol (PCEP) is a transaction-based protocol carried over TCP [RFC4655]. In order to be able to direct path computation requests to the Path Computation Element (PCE), a Path Computation Client (PCC) (or other PCE) needs to know the location and capability of a PCE.

In a network where an IGP is used and where the PCE participates in the IGP, discovery mechanisms exist for PCC (or PCE) to learn the identity and capability of each PCE. [RFC5088] defines a PCE Discovery (PCED) TLV carried in an OSPF Router LSA. Similarly, [RFC5089] defines the PCED sub-TLV for use in PCE Discovery using IS-IS. Scope of the advertisement is limited to IGP area/level or Autonomous System (AS).

However in certain scenarios not all PCEs will participate in the IGP instance, section 3 (Motivation) outlines a number of use cases. In these cases, current PCE Discovery mechanisms are therefore not appropriate and another PCE discovery function would be required.

This document describes PCE discovery via DNS. The mechanism with which DNS comes to know about the PCE and its capability is out of scope of this document.

1.1. Terminology

The following terminology is used in this document.

PCE-Domain: As per [RFC4655], any collection of network elements within a common sphere of address management or path computational responsibility. Examples of domains include Interior Gateway Protocol (IGP) areas and Autonomous Systems (ASs).

Domain-Name: An identification string that defines a realm of administrative autonomy, authority, or control on the Internet. Any name registered in the DNS is a domain name. DNS Domain names are used in various networking contexts and application-specific naming and addressing purposes. In general, a domain name represents an Internet Protocol (IP) resource. Examples of DNS domain name is "www.example.com" or "example.com"[RFC1035].

1.2. Requirements

As described in [RFC4674], the PCE Discovery information should at least be composed of:

- o The PCE location: an IPv4 and/or IPv6 address that is used to reach the PCE. It is RECOMMENDED to use an address that is always reachable if there is any connectivity to the PCE;

- o The PCE path computation scope (i.e., inter-area, inter-AS, or inter-layer);

- o The set of one or more PCE-Domain(s) into which the PCE has visibility and for which the PCE can compute paths;

- o The set of zero, one, or more neighbor PCE-Domain(s) toward which the PCE can compute paths;

These PCE discovery information allows PCCs to select appropriate PCEs:

This document specifies the procedures and extension to facilitate DNS-based PCE information discovery for specific use cases, and to complement existing IGP discovery mechanism.

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 RFC2119 [RFC2119].

3. Motivation

This section discusses in more detail the motivation and use cases for an alternative DNS-based PCE discovery mechanism.

3.1. Outside the Routing Domain

When the PCE is a router participating in the IGP, or even a server participating passively in the IGP, with all PCEP speakers in the same routing domain, a simple and efficient way to announce PCEs consists of using IGP flooding.

It has been identified that the existing PCE discovery mechanisms do not work in following scenarios:

Inter-AS: Per domain path computation mechanism [RFC5152] or Backward recursive path computation (BRPC) [RFC5441] MAY be used by cooperating PCEs to compute inter-domain path. In which case these cooperating PCEs should be known to other PCEs. In case of inter-AS where the PCEs do not participate in a common IGP, the existing IGP discovery mechanism cannot be used to discover inter-AS PCE.

Hierarchy of PCE: The H-PCE [RFC6805] architecture does not require disclosure of internals of a child domain to the parent PCE. It may be necessary for a third party to manage the parent PCEs according to commercial and policy agreements from each of the participating service providers [PCE-QUESTION]. [RFC6805] specifies that a child PCE must be configured with the address of its parent PCE in order for it to interact with its parent PCE. However handling changes in parent PCE identities and coping with failure events would be an issue for a configured system. There is no scope for parent PCEs to advertise their presence to child PCEs when they are not a part of the same routing domain.

BGP: [BGP-LS] describes a mechanism by which links state and traffic engineering information can be collected from networks and shared with external components using the BGP routing protocol. An external PCE MAY use this mechanism to populate its TED and not take part in the same IGP routing domain.

NMS/OSS: PCE MAY gain the knowledge of Topology information from some management system (e.g.,NMS/OSS) and not take part in the same routing domain. Also note that in some case PCC may not be a router and instead be a management system like NMS and may not be able to discover PCE via IGP discovery.

3.2. Discovery Mechanisms

3.2.1. Query-Response versus Advertisement

Advertisement based PCE discovery using IGP methods [RFC5088] and [RFC5089] floods the PCE information to an area, a subset of areas or to a full routing domain. By the very nature of flooding and advertisements it generates unwanted traffic and may lead to unnecessary advertisement, especially when PCE information needs frequent changes.

DNS is a query-response based mechanism, a client (a PCC) can use DNS to discover a PCE only when it needs to compute a path and does not require any other node in the network to be involved.

In case of Intermittent PCEP session, where PCEP sessions are systematically open and closed for each PCEP request, a DNS-based query-response mechanism is more suitable. One may also utilize DNS-based load-balancing and recovery functions.

3.3. Network Address Translation Gateway

PCEP uses TCP as the transport mechanism between PCC and PCE, and PCE to PCE, communications [RFC5440]. To secure TCP connection that underlay PCEP sessions, Transport Layer Security (TLS) can be used besides using TCP-MD5 [RFC2385] and TCP-AUTH [RFC5295]. When PCC and PCE support TCP-MD5 or TCP-AUTH while NAT does not, TCP connection establishment fails. When NAT gateway is in presence, a TCP or TCP/TLS connection can be opened by Interactive Connectivity Establishment (ICE) [RFC5245] for the purpose of connectivity checks. However the TCP connection cannot be established in cases where one of the peers is behind a NAT with connection-dependent filtering properties [RFC5382]. Therefore IGP discovery is limited within an IGP domain and cannot be used in this case.

4. Additional Capabilities

4.1. Load Sharing of Path Computation Requests

Multiple PCEs can be present in a single network domain for redundancy. DNS supports inherent load balancing where multiple PCEs (with different IP addresses) are known in DNS for a single PCE server name and are hidden from the PCC.

In an IGP advertisement based PCE discovery, one learns of all the PCEs and it is the job of the PCC to do load-balancing.

A DNS-based load-balancing mechanism works well in case of Intermittent PCEP sessions and request are load-balanced among PCEs similar to HTTP request without any complexity at the client.

5. Extended Naming Authority Pointer (NAPTR)Service Field Format

The NAPTR service field format defined by the S-NAPTR DDDS application in [RFC3958] follows this Augmented Backus-Naur Form (ABNF) [RFC5234]:

```
service-parms = [ [app-service] *(":" app-protocol)]
app-service   = experimental-service  / iana-registered-service
app-protocol  = experimental-protocol / iana-registered-protocol
experimental-service      = "x-" 1*30ALPHANUMSYM
experimental-protocol     = "x-" 1*30ALPHANUMSYM
iana-registered-service   = ALPHA *31ALPHANUMSYM
iana-registered-protocol  = ALPHA *31ALPHANUMSYM
ALPHA         =  %x41-5A / %x61-7A   ; A-Z / a-z
DIGIT         =  %x30-39 ; 0-9
SYM           =  %x2B / %x2D / %x2E  ; "+" / "-" / "."
ALPHANUMSYM   =  ALPHA / DIGIT / SYM
; The app-service and app-protocol tags are limited to 32
; characters and must start with an alphabetic character.
; The service-parms are considered case-insensitive.
```

This specification refines the "iana-registered-service" tag definition for the discovery of PCE supporting a specific PCE application or multiple PCE applications as defined below.

```
iana-registered-service =/ pce-service
pce-service             = "pce" *("+" appln-name)
appln-name                = non-ws-string
non-ws-string  = 1*(%x21-FF)
```

The appln-name element is the Application Identifier used to identify a specific PCE application. The PCE Application Name are allocated by IANA as defined in section 8.1.

This specification also refines the "iana-registered-protocol" tag definition for the discovery of PCE supporting a specific transport protocol as defined below.

```
iana-registered-protocol =/ pce-protocol
pce-protocol             = "pce." pce-transport
pce-transport            = "tcp" / "tls.tcp"
```

Similar to application protocol tags defined in the [RFC6408],the S-NAPTR application protocol tags defined by this specification MUST NOT be parsed in any way by the querying application or Resolver. The delimiter (".") is present in the tag to improve readability and does not imply a structure or namespace of any kind. The choice of delimiter (".") for the application protocol tag follows the format

of existing S-NAPTR application protocol tag registry entries, but this does not imply that it shares semantics with any other specifications that create registry entries with the same format.

The S-NAPTR application service and application protocol tags defined by this specification are unrelated to the IANA "Service Name and Transport Protocol Port Number Registry" (see [RFC6335]).

The maximum length of the NAPTR service field is 256 octets, including a one-octet length field (see Section 4.1 of [RFC3403] and Section 3.3 of [RFC1035]).

5.1. IETF Standards Track PCE Applications

A PCE Client MUST be capable of using the extended S-NAPTR application service tag for dynamic discovery of a PCE supporting Standards Track applications. Therefore, every IETF Standards Track PCE application MUST be associated with a "PCE-service" tag formatted as defined in this specification and allocated in accordance with IANA policy (see Section 8).

For example, a NAPTR service field value of:

’PCE+gco:pce.tcp’

means that the PCE in the SRV or A/AAAA record supports the Global Concurrent Optimization Application (See section 8.1)and the Transport Control Protocol (TCP) as the transport protocol (See section 8.2).

6. Backwards Compatibility

Domain Name System (DNS) administrators SHOULD also provision legacy NAPTR records [RFC3403] in order to guarantee backwards compatibility with legacy PCE that only support S-NAPTR DDDS application in [RFC3958]. If the DNS administrator provisions both extended S-NAPTR records as defined in this specification and legacy NAPTR records defined in [RFC3403], then the extended S-NAPTR records MUST have higher priority(e.g., lower order and/or preference values) than legacy NAPTR records.

7. Discovering a Path Computation Element

The extended-format NAPTR records provide a mapping from a domain to the SRV record or A/AAAA record for contacting a server supporting a specific transport protocol and PCE application. The resource record will contain an empty regular expression and a replacement value, which is the SRV record or the A/AAAA record for that particular transport protocol.

The assumption for this mechanism to work is that the DNS administrator of the queried domain has first provisioned the DNS with extended-format NAPTR entries.

When the PCC or other PCEs performs a NAPTR query for a server in a particular realm, the PCC or other PCEs has to know in advance the search path of the resolver, i.e.,in which realm to look for a PCE, and in which Application Identifier it is interested.

The search path of the resolver can either be pre-configured, or discovered using Diameter, DHCP or other means. For example, the realm could be deduced from the Network Access Identifier (NAI) in the User-Name attribute-value pair (AVP) or extracted from the Destination-Realm AVP in Diameter [RFC6733].

When pre-configuration is used, PCE domain(e.g.,AS200)can be added as "subdomains" of the first-level domain of the underlying service (e.g., AS200.example.com), which allows a NAPTR query for a server in a PCE domain associated with DNS domain-name.

When DHCP is used, it SHOULD know the domain-name of that realm and use DHCP to discover IP address of the PCE in that realm that provides path computation service along with some PCE location information useful to a PCC (or other PCE) for a PCE selection, and contact it directly. In some instances, the discovery may result in a per protocol/application list of domain-names that are then used as starting points for the subsequent S-NAPTR lookups [RFC3958]. If neither the IP address nor other PCE location information can be discovered with the above procedure, the PCC (or other PCE) MAY request a domain search list, as described in [RFC3397] and[RFC3646], and use it as input to the DDDS application.

When the PCC (or other PCE) does not find valid domain-names using the mechanisms above, it MUST stop the attempt to discover any PCE.

The following procedures result in an IP address, PCE domain, neighboring PCE domain and PCE Computation Scope where the PCC (or other PCE) can contact the PCE that hosts the service it is looking for.

7.1. Determining the PCE Service and transport protocol

The PCC (or other PCE) should know the service identifier for the Path Computation service and associated transport protocol. The service identifier for the Path Computation service is defined as "PCE+apX" as specified in section 5, The PCE supporting "PCE" service MUST support TCP as transport, as described in [RFC5440].

The services relevant for the task of transport protocol selection are those with S-NAPTR service fields with values "PCE+apX:Y", where ’PCE+apX’ is the service identifier defined in the previous paragraph, and ’ Y’ is the letter that corresponds to a transport protocol supported by the PCE. This document also establishes an IANA registry for mappings of S-NAPTR service name to transport protocol.

These NAPTR [RFC3958] records provide a mapping from a domain to the SRV [RFC2782] record for contacting a PCE with the specific transport protocol in the S-NAPTR services field. The resource record MUST contain an empty regular expression and a replacement value, which indicates the domain name where the SRV record for that particular transport protocol can be found. As per [RFC3403], the client discards any records whose services fields are not applicable.

The PCC (or other PCE) MUST discard any service fields that identify a resolution service whose value is not valid. The S-NAPTR processing as described in [RFC3403] will result in the discovery of the most preferred PCE that is supported by the client, as well as an SRV record for the PCE.

7.2. Determining the IP Address of the PCE

If the returned NAPTR service fields contain entries formatted as "pce+apX:Y" where "X" indicates the Application Identifier and "Y" indicates the supported transport protocol(s), the target realm supports the extended format for NAPTR-based PCE discovery defined in this document.

- o If "X" contains the required Application Identifier and "Y" matches a supported transport protocol, the PCEP implementation resolves the "replacement" field entry to a target host using the lookup method appropriate for the "flags" field.

- o If "X" does not contain the required Application Identifier or "Y" does not match a supported transport protocol, the PCEP implementation abandons the peer discovery.

If the returned NAPTR service fields contain entries formatted as

"pce+apX" where "X" indicates the Application Identifier, the target realm supports the extended format for NAPTR-based PCE discovery defined in this document.

- o If "X" contains the required Application Identifier, the PCEP implementation resolves the "replacement" field entry to a target host using the lookup method appropriate for the "flags" field and attempts to connect using all supported transport protocols.

- o If "X" does not contain the required Application Identifier, the PCEP implementation abandons the PCE discovery.

If the returned NAPTR service fields contain entries formatted as "pce:X" where "X" indicates the supported transport protocol(s), the target realm supports PCEP but does not support the extended format for NAPTR-based PCE discovery defined in this document.

- o If "X" matches a supported transport protocol, the PCEP implementation resolves the "replacement" field entry to a target host using the lookup method appropriate for the "flags" field.

If the returned NAPTR service fields contain entries formatted as "pce", the target realm supports PCEP but does not support the extended format for NAPTR-based PCE discovery defined in this document. The PCEP implementation resolves the "replacement" field entry to a target host using the lookup method appropriate for the "flags" field and attempts to connect using TCP (in future it SHOULD attempt all supported transport Protocols) .

Note that the regexp field in the S-NAPTR example above is empty. The regexp field MUST NOT be used when discovering PCE, as its usage can be complex and error prone. Also, the discovery of the PCE does not require the flexibility provided by this field over a static target present in the TARGET field.

As the default behavior, the client is configured with the information about which transport protocol is used for a path computation service in a particular domain. The client can directly perform an SRV query for that specific transport using the service identifier of the path computation Service. For example, if the client knows that it should be using TCP for path computation service, it can perform a SRV query directly for_PCE._tcp.example.com.

Once the server providing the desired service and the transport protocol has been determined, the next step is to determine the IP address.

According to the specification of SRV RRs in [RFC2782], the TARGET field is a fully qualified domain-name (FQDN) that MUST have one or more address records; the FQDN must not be an alias, i.e., there MUST NOT be a CNAME or DNAME RR at this name. Unless the SRV DNS query already has reported a sufficient number of these address records in the Additional Data section of the DNS response (as recommended by [RFC2782]), the PCC needs to perform A and/or AAAA record lookup(s) of the domain-name, as appropriate. The result will be a list of IP addresses, each of which can be contacted using the transport protocol determined previously.

7.2.1. Examples

As an example, consider a client that wishes to find PCED service in the as100.example.com domain. The client performs a S-NAPTR query for that domain, and the following NAPTR records are returned:

```
  Order Pref Flags  Service     Regexp       Replacement
   IN NAPTR  50   50   "s"  "pce:pce.tls.tcp"    ""
     _PCE._tcp.as100.example.com
   IN NAPTR  90   50   "s"  "pce:pce.tcp"    ""
      _PCE._tcp.as100.example.com
```

This indicates that the domain does have a PCE providing Path Computation services over TCP, in that order of preference. If the client only supports TCP, TCP will be used, targeted to a host determined by an SRV lookup of _PCE._tcp.example.com. That lookup would return:

```
    ;;  Priority  Weight    Port        Target
  IN  SRV    0        1      XXXX   server1.as100.example.com
  IN  SRV    0        2      XXXX   server2.as100.example.com
```

where XXXX represents the port number at which the service is reachable.

As an alternative example, a client wishes to discover a PCE in the ex2.example.com realm that supports the GCO application over TCP. The client performs a NAPTR query for that domain, and the following NAPTR records are returned:

```
;;        order pref flags service   regexp replacement
IN NAPTR  150   50   "a"   "pce:pce.tcp"  ""
             server1.ex2.example.com
IN NAPTR  150   50   "a"   "pce:pce.tls.tcp"  ""
             server2.ex2.example.com
IN NAPTR  150   50   "a"   "pce+gco:pce.tcp"  ""
             server1.ex2.example.com
IN NAPTR  150   50   "a"   "pce+gco:pce.tls.tcp"  ""
             server2.ex2.example.com
```

This indicates that the server supports GCO(ID=1) over TCP and TLS/TCP via hosts server1.ex2.example.com and server2.ex2.example.com, respectively.

7.3. Determining the PCE domains and Neighbor PCE domains

DNS servers MAY use DNS TXT record to give additional information about PCE service and add such TXT record to the additional information section (See section 4.1 of [RFC1035]) that are relevant to the answer and have the same authenticity as the data (Generally this will be made up of A and SRV records)in the answer section. The additional information may include path computation capability, the PCE domains and Neighbor PCE domains associated with the PCE. If discovery of PCE supporting a specific PCE capability described in section 7.2 has already been performed, capability associated with the PCE does not need to be included in the additional information.

To store new types of information, the TXT record uses a structured format in its TXT-DATA field [RFC1035]. The format consists of the attribute name followed by the value of the attribute. The name and value are separated by an equals sign (=). The general syntax may follow one defined in section 2 of [RFC1464] as follows:

```
<owner> <class> <ttl> TXT "<attribute name>=<attribute value>"
```

For example, the following TXT records contain attributes specified in this fashion:

```
ex2.example.com    IN   TXT   "pce domain = as10"
ex2.example.com    IN   TXT   "neigh domain= as5"
ex2.example.com    IN   TXT   "cap=link constraint"
```

The client MAY inspect those Additional Information section in the DNS message and be capable of handling responses from nameservers that never fill in the Additional Information part of a response.

8. IANA Considerations

8.1. IETF PCE Application Service Tags

IANA specifies to create a new registry ’ S-NAPTR application service tags’ for existing IETF PCE applications.

Tag	PCE Application
pce+gco	GCO [RFC5557]
pce+p2mp	P2MP [RFC5671]
pce+stateful	Stateful [STATEFUL-PCE]
pce+gmpls	GMPLS [RFC7025]
pce+interas	Inter-AS[RFC5376]
pce+interarea	Inter-Area [RFC4927]
pce+interlayer	Inter-layer [RFC6457]

Future IETF PCE applications MUST reserve the S-NAPTR application service tag corresponding to the allocated PCE Application ID as defined in Section 3.

8.2. PCE Application Protocol Tags

IANA has reserved the following S-NAPTR Application Protocol Tags for the PCE transport protocols in the "S-NAPTR Application Protocol Tag" registry created by [RFC3958].

Tag	Protocol
pce.tcp	TCP

Future PCE versions that introduce new transport protocols MUST reserve an appropriate S-NAPTR Application Protocol Tag in the "S-NAPTR Application Protocol Tag" registry created by [RFC3958].

9. Security Considerations

This document specifies an enhancement to the NAPTR service field format. The enhancement and modifications are based on the S-NAPTR, which is actually a simplification of the NAPTR, and therefore the same security considerations described in [RFC3958] are applicable to this document.

For most of those identified threats, the DNS Security Extensions [RFC4033] does provide protection. It is therefore recommended to consider the usage of DNSSEC [RFC4033] and the aspects of DNSSEC Operational Practices [RFC6781] when deploying Path Computation Services.

In deployments where DNSSEC usage is not feasible, measures should be taken to protect against forged DNS responses and cache poisoning as much as possible. Efforts in this direction are documented in [RFC5452].

However a malicious host doing S-NAPTR queries learns applications supported by PCEs in a certain realm faster, which might help the malicious host to scan potential targets for an attack more efficiently when some applications have known vulnerabilities.

Where inputs to the procedure described in this document are fed via DHCP, DHCP vulnerabilities can also cause issues. For instance, the inability to authenticate DHCP discovery results may lead to the Path Computation service results also being incorrect, even if the DNS process was secured.

10. Acknowledgements

The author would like to thank Claire Bi,Ning Kong, Liang Xia, Stephane Bortzmeyer,Yi Yang, Ted Lemon, Adrian Farrel and Stuart Cheshire for their review and comments that help improvement to this document.

11. References

11.1. Normative References

[RFC1035] Mockapetris, P., "DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION", RFC 1035, November 1987.

[RFC1464] Rosenbaum, R., "Using the Domain Name System To Store Arbitrary String Attributes", RFC 1464, May 1993.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.

[RFC2782] Gulbrandsen, A., "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[RFC3397] Aboba, B., "Dynamic Host Configuration Protocol (DHCP) Domain Search Option", RFC 3397, November 2002.

[RFC3403] Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database", RFC 3403, October 2002.

[RFC3646] Droms, R., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, December 2003.

[RFC3958] Daigle, D. and A. Newton, "Domain-Based Application Service Location Using SRV RRs and the Dynamic Delegation Discovery Service (DDDS)", RFC 3958, January 2005.

[RFC4033] Arends, R., "DNS Security Introduction and Requirements", RFC 4033, March 2005.

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC4674] Droms, R., "Requirements for Path Computation Element (PCE) Discovery", RFC 4674, December 2003.

[RFC5440] Le Roux, JL., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, April 2007.

[RFC6733] Fajardo, V., "Diameter Base Protocol", RFC 6733, October 2012.

[RFC6781] Kolkman, O., Mekking, W., and R. Gieben, "DNSSEC Operational Practices, Version 2", RFC 6781,

December 2012.

[RFC6805] King, D. and A. Farrel, "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, November 2012.

11.2. Informative References

[ALTO] Kiesel, S., "ALTO Server Discovery", ID draft-ietf-alto-server-discovery-22, December 2013.

[BGP-LS] Gredler, H., "North-Bound Distribution of Link-State and TE Information using BGP", ID draft-ietf-idr-ls-distribution-04, November 2013.

[PCE-QUESTION] Farrel, A., "Unanswered Questions in the Path Computation Element Architecture", ID http://tools.ietf.org/html/draft-ietf-pce-questions-00, July 2013.

[RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, August 1998.

[RFC4927] Le Roux, JL., "Path Computation Element Communication Protocol (PCECP) Specific Requirements for Inter-Area MPLS and GMPLS Traffic Engineering", RFC 4927, June 2007.

[RFC5088] Le Roux, JL., "OSPF Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5088, January 2008.

[RFC5089] Le Roux, JL., "IS-IS Protocol Extensions for Path Computation Element (PCE) Discovery", RFC 5089, January 2008.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.

[RFC5295] Touch, J., "The TCP Authentication Option", RFC 5295, June 2010.

[RFC5376] Bitar, N., "Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)", RFC 5376, November 2008.

[RFC5382] Guha, S., "NAT Behavioral Requirements for TCP", RFC 5382, October 2008.

[RFC5452] Hubert, A., "Measures for Making DNS More Resilient against Forged Answers", RFC 5452, January 2009.

[RFC6457] Takeda, T., "PCC-PCE Communication and PCE Discovery Requirements for Inter-Layer Traffic Engineering", RFC 6457, June 2007.

[RFC7025] Otani, T., "Requirements for GMPLS Applications of PCE", RFC 7025, September 2013.

Authors' Addresses

Qin Wu
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: sunseawq@huawei.com

Dhruv Dhody
Huawei
Leela Palace
Bangalore, Karnataka 560008
INDIA

Email: dhruv.dhody@huawei.com

Daniel King
Old Dog Consulting
UK

Email: daniel@olddog.co.uk

Diego R. Lopez
Telefonica I+D

Email: diego@tid.es

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 18, 2014

Q. Wu
D. Dhody
Huawei
S. Previdi
Cisco Systems, Inc
October 15, 2013

Extensions to Path Computation Element Communication Protocol (PCEP) for handling Link Bandwidth Utilization
draft-wu-pce-pcep-link-bw-utilization-00

Abstract

The Path Computation Element Communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

Link bandwidth utilization considering the total bandwidth of a link in current use for the forwarding is an important factor to consider during path computation. This document describes extensions to PCEP to consider them as new constraints during path computation.

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 http://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 April 18, 2014.

Copyright Notice

Copyright (c) 2013 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 (http://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 3
2. Terminology 3
3. Link Bandwidth Utilization (LBU) 4
4. Link Reserved Bandwidth Utilization (LRBU) 4
5. PCEP Requirements 4
6. PCEP Extensions 5
  6.1. BU Object 5
    6.1.1. Elements of Procedure 6
  6.2. New Objective Functions 6
  6.3. PCEP Message 7
7. Other Considerations 9
  7.1. Reoptimization Consideration 9
  7.2. Inter-domain Consideration 9
  7.3. P2MP Consideration 9
  7.4. Stateful PCE 10
8. IANA Considerations 10
9. Security Considerations 10
10. Security Considerations 10
11. Manageability Considerations 10
  11.1. Control of Function and Policy 10
  11.2. Information and Data Models 10
  11.3. Liveness Detection and Monitoring 10
  11.4. Verify Correct Operations 10
  11.5. Requirements On Other Protocols 10
  11.6. Impact On Network Operations 11
12. Acknowledgments 11
13. References 11
  13.1. Normative References 11
  13.2. Informative References 11
Appendix A. Contributor Addresses 12

1. Introduction

Real time link bandwidth utilization is becoming critical in the path computation in some networks. It is important that link bandwidth utilization is factored in during path computation. PCC can request a PCE to provide a path such that it selects under-utilized links. This document extends PCEP [RFC5440] for this purpose.

Traffic Engineering Database (TED) as populated by Interior Gateway Protocol (IGP) contains Maximum bandwidth, Maximum reservable bandwidth and Unreserved bandwidth ([RFC3630] and [RFC3784]). [OSPF-TE-EXPRESS] and [ISIS-TE-EXPRESS] further populate Residual bandwidth and Available bandwidth. Further [ISIS-TE-EXPRESS] also define Bandwidth Utilization.

[Editors Note: [OSPF-TE-EXPRESS] should also be extended in future version for real time link bandwidth utilization]

The links in the path MAY be monitored for changes in the link bandwidth utilization, re-optimization of such path MAY be further requested.

1.1. 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 [RFC2119].

2. Terminology

The following terminology is used in this document.

IGP: Interior Gateway Protocol. Either of the two routing protocols, Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS).

PCC: Path Computation Client: any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. An entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints.

PCEP: Path Computation Element Protocol.

RSVP: Resource Reservation Protocol

TE LSP: Traffic Engineering Label Switched Path.

3. Link Bandwidth Utilization (LBU)

The bandwidth utilization on a link, forwarding adjacency, or bundled link is populated in the TED (Bandwidth Utilization in [ISIS-TE-EXPRESS]). For a link or forwarding adjacency, bandwidth utilization represent the actual utilization of the link (i.e.: as measured in the router). For a bundled link, bandwidth utilization is defined to be the sum of the component link bandwidth utilization. This includes traffic for both RSVP and non-RSVP.

LBU Percentage is described as the (LBU / Maximum bandwidth) * 100.

4. Link Reserved Bandwidth Utilization (LRBU)

The reserved bandwidth utilization on a link, forwarding adjacency, or bundled link can be calculated from the TED. This includes traffic for only RSVP-TE LSPs.

LRBU can be calculated by using the Residual bandwidth, available bandwidth and LBU. The actual bandwidth by non-RSVP TE traffic can be calculated by subtracting Available Bandwidth from Residual Bandwidth. Once we have the actual bandwidth for non-RSVP TE traffic, subtracting this from LBU would result in LRBU.

LRBU Percentage is described as the (LRBU / (current reserved bandwidth)) * 100; where the current reserved bandwidth can be calculated by subtracting Residual bandwidth from Maximum bandwidth.

5. PCEP Requirements

Following requirements associated with bandwidth utilization are identified for PCEP:

1. PCE supporting this draft MUST have the capability to compute end-to-end path with bandwidth utilization constraints. It MUST also support the combination of bandwidth utilization constraint with existing constraints (cost, hop-limit...).

2. PCC MUST be able to request for bandwidth utilization constraint in PCReq message as the boundary condition that should not be crossed for each link in the path.

3. PCC MUST be able to request for bandwidth utilization constraint in PCReq message as an Objective function (OF) [RFC5541] to be

optimized.

4. PCEs are not required to support bandwidth utilization constraint. Therefore, it MUST be possible for a PCE to reject a PCReq message with a reason code that indicates no support for bandwidth utilization constraint.

5. PCEP SHOULD provide mechanism to handle bandwidth utilization constraint in multi-domain (e.g., Inter-AS, Inter-Area or Multi-Layer) environment.

6. PCEP Extensions

This section defines extensions to PCEP [RFC5440] for requirements outlined in Section 5. The proposed solution is used to consider bandwidth utilization during path computation.

6.1. BU Object

The BU (Bandwidth Utilization) is used to indicate the upper limit of the acceptable link bandwidth utilization percentage.

The BU object may be carried within the PCReq message and PCRep messages.

BU Object-Class is TBD.

Two Object-Type values are defined for the BU object:

o Link Bandwidth Utilization (LBU): BU Object-Type is 1.

o Link Reserved Bandwidth Utilization (LRBU): BU Object-Type is 2.

The format of the BU object body is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Bandwidth Utilization                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

BU Object Body Format

Bandwidth utilization (32 bits): Represents the bandwidth utilization quantified as a percentage (as described in Section 3 and Section 4). The basic unit is 0.000000023%, with the maximum value 4,294,967,295 representing 98.784247785% (4,294,967,295 * 0.000000023%). This value is the maximum Bandwidth utilization

percentage that can be expressed.

The BU object body has a fixed length of 4 bytes.

6.1.1. Elements of Procedure

PCC SHOULD request the PCE to factor in the bandwidth utilization during path computation by including a BU object in the PCReq message.

Multiple BU objects MAY be inserted in a PCReq or a PCRep message for a given request but there MUST be at most one instance of the BU object for each object type. If, for a given request, two or more instances of a BU object with the same object type are present, only the first instance MUST be considered and other instances MUST be ignored.

BU object MAY be carried in a PCRep message in case of unsuccessful path computation along with a NO-PATH object to indicate the constraints that could not be satisfied.

If the P bit is clear in the object header and PCE does not understand or does not support bandwidth utilization during path computation it SHOULD simply ignore BU object.

If the P Bit is set in the object header and PCE receives BU object in path request and it understands the BU object, but the PCE is not capable of bandwidth utilization check during path computation, the PCE MUST send a PCErr message with a PCEP-ERROR Object Error-Type = 4 (Not supported object) [RFC5440]. The path computation request MUST then be cancelled.

If the PCE does not understand the BU object, then the PCE MUST send a PCErr message with a PCEP-ERROR Object Error-Type = 3 (Unknown object) [RFC5440].

6.2. New Objective Functions

This document defines two additional objective functions -- namely, MUP (Maximum Under-Utilized Path) and MRUP (Maximum Reserved Under-Utilized Path. Hence two new objective function codes have to be defined.

Objective functions are formulated using the following terminology:

- o A network comprises a set of N links {Li, (i=1...N)}.

- o A path P is a list of K links {Lpi,(i=1...K)}.
- o Bandwidth Utilization on link L is denoted u(L).
- o Reserved Bandwidth Utilization on link L is denoted ru(L).
- o Maximum bandwidth on link L is denoted M(L).
- o Current Reserved bandwidth on link L is denoted c(L).

The description of the two new objective functions is as follows.

Objective Function Code: TBD

Name: Maximum Under-Utilized Path (MUP)

Description: Find a path P such that (Min {(M(Lpi)- u(Lpi)) / M(Lpi), i=1...K }) is maximized.

Objective Function Code: TBD

Name: Maximum Reserved Under-Utilized Path (MRUP)

Description: Find a path P such that (Min {(c(Lpi)- ru(Lpi)) / c(Lpi), i=1...K }) is maximized.

These new objective function are used to optimize paths based on bandwidth utilization as the optimization criteria.

If the objective function defined in this document are unknown/ unsupported, the procedure as defined in [RFC5541] is followed.

6.3. PCEP Message

The new optional BU objects MAY be specified in the PCReq message. As per [RFC5541], an OF object specifying a new objective function MAY also be specified.

The format of the PCReq message (with [RFC5541] as a base) is updated as follows:

```
<PCReq Message> ::= <Common Header>
                     [<svec-list>]
                     <request-list>
where:
     <svec-list> ::= <SVEC>
                     [<OF>]
                     [<metric-list>]
                     [<svec-list>]

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

     <request> ::= <RP>
                   <END-POINTS>
                   [<LSPA>]
                   [<BANDWIDTH>]
                   [<bu-list>]
                   [<metric-list>]
                   [<OF>]
                   [<RRO>[<BANDWIDTH>]]
                   [<IRO>]
                   [<LOAD-BALANCING>]

and where:
     <bu-list>::=<BU>[<bu-list>]
     <metric-list> ::= <METRIC>[<metric-list>]
```

The BU objects MAY be specified in the PCRep message, in case of an unsuccessful path computation to indicate the bandwidth utilization as a reason for failure. The OF object MAY be carried within a PCRep message to indicate the objective function used by the PCE during path computation.

The format of the PCRep message (with [RFC5541] as a base) is updated as follows:

```
<PCRep Message> ::= <Common Header>
                    [<svec-list>]
                    <response-list>

where:

      <svec-list> ::= <SVEC>
                      [<OF>]
                      [<metric-list>]
                      [<svec-list>]

     <response-list> ::= <response> [<response-list>]

     <response> ::= <RP>
                    [<NO-PATH>]
                    [<attribute-list>]
                    [<path-list>]

     <path-list> ::= <path> [<path-list>]

     <path> ::= <ERO>
                <attribute-list>

and where:

     <attribute-list> ::= [<OF>]
                          [<LSPA>]
                          [<BANDWIDTH>]
                          [<bu-list>]
                          [<metric-list>]
                          [<IRO>]

         <bu-list>::=<BU>[<bu-list>]
     <metric-list> ::= <METRIC> [<metric-list>]
```

7. Other Considerations

7.1. Reoptimization Consideration

PCC can monitor the link bandwidth utilization of the setup LSPs and in case of drastic change, it MAY ask PCE for reoptimization as per [RFC5440].

7.2. Inter-domain Consideration

7.3. P2MP Consideration

7.4. Stateful PCE

8. IANA Considerations

TBD

9. Security Considerations

TBD

10. Security Considerations

This document defines a new BU object and OF codes which does not add any new security concerns beyond those discussed in [RFC5440].

11. Manageability Considerations

11.1. Control of Function and Policy

The only configurable item is the support of the new constraints on a PCE which MAY be controlled by a policy module. If the new constraints are not supported/allowed on a PCE, it MUST send a PCErr message as specified in Section 6.1.1.

11.2. Information and Data Models

[PCEP-MIB] describes the PCEP MIB, there are no new MIB Objects for this document.

11.3. 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].

11.4. Verify Correct Operations

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

11.5. Requirements On Other Protocols

PCE requires the TED to be populated with the bandwidth utilization. This mechanism is described in [OSPF-TE-EXPRESS] or [ISIS-TE-EXPRESS].

11.6. 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].

12. Acknowledgments

We would like to thank Alia Atlas, John E Drake, David Ward 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, March 1997.

13.2. Informative References

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, September 2003.

[RFC3784] Smit, H. and T. Li, "Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE)", RFC 3784, June 2004.

[RFC5440] Vasseur, JP. and JL. Le Roux, "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, June 2009.

[OSPF-TE-EXPRESS] Giacalone, S., Ward, D., Drake, J., Atlas, A., and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions [draft-ietf-ospf-te-metric-extensions]", June 2013.

[ISIS-TE-EXPRESS] Previdi, S., Giacalone, S., Ward, D., Drake, J., Atlas, A., Filsfils, C., and W. Qin, "IS-IS Traffic Engineering (TE) Metric Extensions [draft-ietf-isis-te-metric-extensions-01]", October 2013.

[PCEP-MIB] Kiran Koushik, A S., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "PCE communication protocol(PCEP) Management Information Base [draft-ietf-pce-pcep-mib]", Feb 2013.

Appendix A. Contributor Addresses

Udayasree Palle
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA
EMail: udayasree.palle@huawei.com

Authors' Addresses

Qin Wu
Huawei Technologies
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

EMail: sunseawq@huawei.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.ietf@gmail.com

Stefano Previdi
Cisco Systems, Inc
Via Del Serafico 200
Rome 00191
IT

EMail: sprevidi@cisco.com

CCAMP Working Group
Internet Draft
Category: Standards track

Xian Zhang
Haomian Zheng
Huawei
Oscar Gonzales de Dios
Victor Lopez
Telefonica I+D

Expires: August 14, 2014 February 14, 2014

Extensions to Path Computation Element Protocol (PCEP) to Support Resource Sharing-based Path Computation

draft-zhang-pce-resource-sharing-00.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on August 14, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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.

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 [RFC2119].

Abstract

Resource sharing in a network means two or more Label Switched Paths (LSPs) use common piece(s) of resource along their paths. This can help save network resource and useful in scenarios such as LSP recovery or two LSPs do not need to be active at the same time. A Path Computation Element (PCE) is a centralized entity, responsible for path calculation. Given this feature and its access to the network resource information and possibly active LSPs information, it can be used to support resource-sharing-based path computation with better efficiency.

This document extends the Path Computation Element Protocol (PCEP) in order to support resource sharing-based path computation.

Table of Contents

1. Introduction and Motivation..... 3
2. Motivation 4
   2.1. Use Case 1 4
   2.2. Use Case 2 5
3. Extensions to PCEP 7
   3.1. Resource Sharing Object..... 7
   3.2. Processing Rules..... 9
   3.3. Carrying RSO in a PCEP Message 10
4. Security Considerations..... 11
5. IANA Considerations 12
   5.1. New Object Type..... 12
6. References 13
   6.1. Normative References..... 13
   6.2. Informative References..... 13
7. Authors' Addresses 13

1. Introduction and Motivation

A Path Computation Element (PCE) provides an alternative way for providing path computation function, and it is especially useful in the scenarios where complex constraints and/or a demanding amount of computation resource are required [RFC4655]. The development of PCE standardization has evolved from stateless to stateful. A stateful PCE has access to the LSP database information of the network(s) it serves as a computation engine [Stateful-PCE]. Unless specified otherwise, this document assumes a PCE mentioned is a stateful PCE (either passive or active).

Resource sharing denotes that two or more Label Switched Paths (LSPs) share common piece(s) of resource, (such as a common time slot of a link in an Optical Transport Network (OTN)). This is usually useful in the scenario where only one LSP is active and the benefit herein is to save network resources. A simple example of this is dynamically calculating a LSP for an existing LSP undergoing a link failure. Note that the resource sharing can be worked out using a statelss PCE, but the mechanism may be complex and is out the scope of this draft.

This document considers the following requirement: resource sharing with one or multiple existing LSPs. In a single domain, this is a common requirement in the recovery cases especially in order to increase traffic resilience against failure while reducing the amount of network resource used for recovery purpose [RFC4428].

The current protocol supporting the communication between a PCE and a Path Computation Client (PCC), i.e. PCE Protocol (PCEP), allows for re-optimization of an existing LSP [RFC5440]. This is achieved by setting R bit in the Request Parameter (RP) object, together with some additional information if applicable, in the Path Computation Request (PCReq) message sent from a PCC to the PCE. To support this type of resource sharing, a PCC needs to ask a PCE to compute a new path with the constraints of sharing resource with one or multiple existing LSPs. Current PCEP specifications do not provide such function.

As mentioned in [stateful-PCE], the standardization of stateful PCEs also facilitates PCEP to meet this requirement since a LSP can be identified using a unique number. This simplifies configuration of PCCs by making it simpler to for a PCC to request resource sharing without having to determine all of the resources to be shared.

The resource sharing can also be required across layers. This is similar to the previous requirement. However, it is more complex and therefore deserves a more detailed explanation here.

In a multi-layer network, Label Switched Paths (LSPs) in a lower layer are used to carry higher-layer LSPs across the lower-layer network [RFC5623]. Therefore, the resource sharing constraints in the higher layer might actually relate to the resource sharing in the lower layer. Thus, it is useful to consider how this can be achieved and whether additional extensions are needed using the models defined in [RFC5623].

In the next sections, use cases are provided to show what information needs to be exchanged to fulfill these requirements. This memo then provides extensions to PCEP to enable this function.

2. Motivation

2.1. Use Case 1

Figure 1 shows a single domain network with a stateful PCE. Assume a working LSP (N1-N2-N3) exists in the network. When there is failure on the link N2-N3, it is desired to set up a restoration path for this working LSP. Suppose N1 serves as the PCC and sends a request to the stateful PCE for such an LSP. Before sending the request, N1 may need to check what policy is configured locally on N1. For example, it might value resource sharing more than effectiveness. Effectiveness here denotes whether the traffic can be diverted back to the working LSP immediately once the failure on the working LSP is repaired. In this case, it would prefer to share as much resource with the working LSP as possible and specify this in the PCReq message.

On the other hand, if N1 considers effectiveness more important, it would prefer to share as few resources as possible. Note this is different from path diversity, since diversity is a much stricter requirement and it would cause path computation failure if the diverse recovery path cannot be found. A simple illustration is provided below:

```
                 +--------------+
                 |              |
                 | Stateful PCE |
                 |              |
                 +--------------+



            +------+          +------+          +------+
            |  N1  +----------+  N2  +-----X---+  N3  |
            +--+---+          +---+--+          +---+--+
               |                  |                 |
               |                  +---------+       |
               |                            |       |
               |     +------+          +------+     |
               +-----+  N5  +----------+  N4  +-----+
                     +------+          +------+
```

Figure 1: A Single Domain Example

Available recovery paths computed by the stateful PCE:

LSP1: N1-N2-N4-N3
LSP2: N1-N5-N4-N3

If resource sharing is preferred, the stateful PCE will reply with LSP1 information. Instead, if effectiveness is valued higher, it will reply with LSP2 information.

Another piece of information that needs to be conveyed to the PCE is the information about the working path LSP. Note this simple use case assumes end-to-end recovery. But in order to be applicable to use cases such as shared mesh protection purpose, where the head-end and tail-end nodes may be different, this information is necessary in the message exchange between PCCs and PCEs, so that the stateful PCE knows which LSP the path computation request wants to share the resource with.

2.2. Use Case 2

Figure 2 shows a two-layer network example, with each layer managed by a PCE (referred as PCE Hi for higher layer and PCE Lo for lower layer later). As Discussed in Section 3 of [RFC5623], there are three models for inter-layer path computation. They are single PCE computation, multiple PCE with inter-PCE communication and multiple PCE without inter-PCE communication, respectively. For the single

PCE computation, the process would be similar to that of the use case in Section 2.1. Thus, this model is not discussed further.

```
                              ................................| LSR |
                            .:                                | H5  |
                          .:                                  /-----
                        .:                                   /   |
        -----    -----.:                       -----    -----/    |
       | LSR |--| LSR |......................| LSR |--| LSR |   /
       | H1  |  | H2  |                      | H3  |  | H4  |  /
        -----    -----\                      /-----    -----  /
                       \                    /                /
                        \                  /                /
                         \                /                /
                          \              /                /
                           \-----   -----/               /
                           | LSR |-| LSR |             /
                           | L1  | | L2  |            /
                            -----   -----\            /
                              |           \          /
                              |            \        /
                              |             \      /
                            -----            \-----/
                           | LSR |-----------| LSR |
                           | L3  |           | L4  |
                            -----             -----
```

Figure 2: A Two-layer Network Example

In this example, assume a LSP (LSP1: H2-H3) has been established already. A new request comes at H2 to establish a new LSP (LSP2: from H2 to H5), given the constraint it can share resource with LSP1. This requirement is possible if only one of the LSPs needs to be active and resource sharing is the target.

If multiple PCE with inter-PCE communication model is employed, the path computation request sent by H2 to PCE Hi will be passed to PCE Lo since there is no resource readily available in the upper layer. So it leaves to the PCE Lo to compute a path in the lower layer in order to support the upper layer request. In this case, PCE Lo is required to compute a path between H2 and H5 under the constraint that it can share the resource with that of the LSP1. Assume here LSP1 goes from H2, via L1-L2 to H3. So when PCE Lo computes the path for LSP2, it can view the resource used by LSP1 available. For example, PCE Lo may choose H2-L1-L2-L4-H5 as the computation result.

The issue to solve during this procedure is that PCE Hi can only use LSP1 information (such as its five-tuple LSP information) as the information, how PCE Lo can resolve this information to the actual resource usage in its own layer, i.e. lower layer. This could be solved by edge LSR L1 reporting this higher-lower layer LSP correlation to the Lo PCE as part of the LSP information during the LSP state synchronization process. If needed, it can be later updated when there is a change in this information. Alternatively, the PCE Lo can get this information from other sources, such as network management system, where this information should be stored.

If multiple PCE without inter-PCE communication model is employed, the path computation request in the lower layer will be initiated the border LSR node, i.e., L1. The process would be similar to that of the previous scenario. A point worth noting is that the border LSR node may be able to resolve the higher LSP information itself, such as mapping it to the corresponding LSP in the lower layer, thus PCE Lo do not need to perform this function. Otherwise, the method mentioned above can still be used.

3. Extensions to PCEP

This section provides PCEP extensions to allow a PCC to specify resource sharing when sending a PCReq message. It also details the processing rule and error codes needed.

3.1. Resource Sharing Object

The PCEP Resource Sharing Object (RSO) is optional. It MAY be carried within a PCRep message so as to indicate the desired resource sharing requirements to be applied by the stateful PCE during path computation.

The RSO object format is compliant with the PCEP object format defined in [RFC5440].

The RSO Object-Class is TBA.

The RSO Object-type is 1.

The format of the RSO object body is:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| RSO codes                 |R|D|       Reserved                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                        Optional TLVs                          ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: RSO Object Format

RSO codes (16 bits): the objective of the resource sharing. Currently, the following objectives are defined:

D (1 bit): sharing as little as possible.

R (1 bit): sharing as much as possible

If D and R are both set to 0, it denotes the requesting node only requires resource sharing without further constraint (i.e., the extent of resource sharing). The combination of D=1 and R=1 is not allowed.

Reserved (2 bytes): This field MUST be set to zero on transmission and MUST be ignored on receipt.

Optional TLVs may be needed to indicate the LSP with which the resource is shared. The LSP Info TLV is defined as follows, for IPv4 and IPv6 addresses respectively

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBD]          |           Length=20           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv4 tunnel end point address                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          0                    |     Tunnel ID                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Extended Tunnel ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    IPv4 tunnel sender address                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          0                    |            LSP ID             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: IPv4 LSP Info TLV

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type=[TBD]          |           Length=44           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address (cont.)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address (cont.)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address (cont.)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          0                    |     Tunnel ID                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Extended Tunnel ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   IPv6 tunnel sender address                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address (cont.)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address (cont.)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 IPv6 tunnel end point address (cont.)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          0                    |            LSP ID             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: IPv6 LSP Info TLV

3.2. Processing Rules

To request a path allowing sharing resource with one or multiple existing LSPs, a PCC includes a RSO object in the PCReq message.

On receipt of a PCReq message with a RSO object, a stateful PCE MUST proceed as follows:

- If the RSO object is unknown/unsupported, the PCE will follow procedures defined in [RFC5440]. That is, the PCE sends a PCErr message with error type 3 or 4 (Unknown / Not supported object) and error value 1 or 2 (unknown / unsupported object class / object type), and the related path computation request is discarded.

- If TLV(s) present in the RSO object are unknown/unsupported and the P bit is set, the PCE MUST send a PCErr message with error type 3 or 4 (Unknown / Not supported object) and error value 4 (Unrecognized/Unsupported parameter), and the related path computation request MUST be discarded as defined in [RFC5440].

- If the resource sharing information is extracted correctly, the PCE MUST apply the requested resource sharing requirement.

If the received RSO has D bit set, the PCE will find a path that shares as much resources as possible with the specified LSP(s). Otherwise, if S bit is set, the PCE will find a path that shares as little resources as possible with the specified LSP(s). The RSO codes may be locally configured on the requesting nodes via external entities, such as a network management system or the entity that impose the resource sharing requirement.

3.3. Carrying RSO in a PCEP Message

The RSO is applied to an individual path computation request and the format of the PCReq message is updated as follows:

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

                    [<svec-list>]

                    <request-list>
```

where:

```
     <svec-list> ::= <SVEC>

                     [<OF>]

                     [<metric-list>]

                     [<svec-list>]


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


    <request> ::= <RP>

                  <END-POINTS>
```

```
                      [<LSPA>]

                      [<BANDWIDTH>]

                      [<metric-list>]

                      [<OF>]

                      [<RRO>[<BANDWIDTH>]]

                      [<IRO>]

                      [<RSO>]

                      [<LOAD-BALANCING>]
```

and where:

```
<metric-list> ::= <METRIC>[<metric-list>]
```

4. Security Considerations

Security of PCEP is discussed in [RFC5440] and [RFC6952]. The extensions in this document do not change the fundamentals of security for PCEP.

However, the introduction of the RSO provides a vector that may be used to probe for information from a network. For example, a PCC that wants to discover the path of an LSP with which it is not involved, can issue a PCReq with an RSO and may be able to get back quite a lot of information about the path of the LSP through issuing multiple such requests for different endpoints and analyzing the received results. To protect against this, a PCE should be configured with access and authorization controls such that only authorized PCCs (for example, those within the network) can make computation requests, only specifically authorized PCCs can make requests using the RSO, and resource sharing requests relating to specific LSPs are further limited to a select few PCCs. How such access controls and authorization is managed is outside the scope of this document, but it will at the least include Access Control Lists.

Furthermore, a PCC must be aware that setting up an LSP that shares resources with another LSP may be a way of attacking the other LSP, for example by depriving it of the resources it needs to operate correctly. Thus it is important that, both in PCEP and the associated signaling protocols, only authorized resource sharing is allowed.

5. IANA Considerations

5.1. New Object Type

IANA manages the PCEP Objects code point registry (see [RFC5440]). This is maintained as the "PCEP Objects" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry.

This document defines a new PCEP object, the RSO object, to be carried in PCReq messages. IANA is requested to make the following allocation in the "PCEP Objects" sub-registry:

Object Class	Name	Object Type	Name	Reference
TBA	RSO		Resource Sharing	[this document]

5.2 New RSO TLVs

IANA is request to create and maintain a new sub-registry named "RSO TLVs" and include the following TLVs:

Value	Description	Reference
1	IPv4 LSP Info TLV	[this document]
2	IPv6 LSP Info TLV	[this document]

5.3 RSO codes

IANA is requested to create and maintain a new sub-registry named "RSO codes". The following codes are defined in this document:

Bit	Code Name	Meaning	Reference
0	D	sharing as much as possible	[this document]
1	R	sharing as little as possible	[this document]

6. References

6.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to indicate requirements levels", RFC 2119, March 1997.

[RFC4655] Farrel, A., Vasseur, J.-P., and Ash, J., "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC5440] Vasseur, J.-P., and Le Roux, JL., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[Stateful-PCE] Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

6.2. Informative References

[RFC4428] Papadimitriou, D., Mannie., E., "Analysis of Generalized Multi-Protocol Label Switching (GMPLS)-based Recovery Mechanisms (including Protection and Restoration)", RFC4428, March 2006.

[RFC5623] Oki., E., Takeda, T., Le Roux, JL., Farrel, A., "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC5623, September 2009.

[RFC6952] Jethanandani, M., Patel, K., Zheng, L., "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC6952, May 2013.

7. Authors' Addresses

Xian Zhang
Huawei Technologies

Email: zhang.xian@huawei.com

Haomian Zheng

Huawei Technologies

Email: zhenghaomian@huawei.com

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid 28045
Spain

EMail: ogondio@tid.es

Victor Lopez
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid 28045
Spain

EMail: vlopez@tid.es

PCE Working Group
Internet Draft
Category: Informational
Expires: August 14, 2014

. Haomian Zheng
Xian Zhang
Huawei Technologies
February 14, 2014

Path Computation Element to Support Software-Defined Transport Networks Control

draft-zheng-pce-for-sdn-transport-00.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on August 14, 2014.

Copyright Notice

Copyright (c) 2014 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 (http://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.

Abstract

This draft describes PCE architecture and protocol in SDN-based transport network. It is demonstrated that PCE can fit in the transport SDN architecture and complete corresponding requests. The PCE and its protocol can satisfy the functional requirement in several transport SDN applications.

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 [RFC2119].

Table of Contents

1. Introduction 2
2. Path Computation in Transport SDN..... 4
   2.1. Transport SDN Architecture..... 4
   2.2. Path computation and establishment..... 5
      2.2.1 Stateless PCE..... 5
      2.2.2 Stateful PCE..... 5
      2.2.3 PCE Initiation..... 5
3. PCE Applicability for Transport SDN..... 6
   3.1. Photonic Enterprise Networks..... 6
   3.2. Virtualized Transport Network..... 6
   3.3. Data center interconnection..... 8
   3.4. Packet Optical Integration..... 10
4. IANA Considerations 11
5. References 11
   5.1. Normative References..... 11
   5.2. Informative References..... 11
6. Authors' Addresses..... 12

1. Introduction

Software Defined Networking (SDN) is an emerging approach to networking and has great potential to shift paradigm in the field of network control and management. SDN greatly simplifies the network control and management by separating the control plane from data plane, which allows network operators to manage network services on an abstraction of the underlying physical networks. SDN requires some method for the control plane to communicate with the data

plane. OpenFlow is one of such mechanisms and is under development to provide standardized communication [OpenFlow].

Specifically, for transport network, the idea of SDN is a perfect choice for future development due to the natural separation of data and control planes. There are some emerging service features and requirements for transport networks that include services that are time scheduled, dynamic, elastic, and underpinned by a Pay As You Go billing model. These features require the transport controllers to provide services with large bandwidth in a short period. All of the above are the motivations for introducing the SDN idea into the transport network.

Path Computation Element (PCE) was firstly developed to solve the path computation problem for MPLS and GMPLS-controlled networks [RFC4655]. Given the demand to simplify the network management and a centralized PCE, the functional transport architecture is very close to the idea of Software-defined Networking (SDN). In the SDN architecture, PCE can be envisioned to be a core functional block in the SDN controller, responsible not only for path computation but for other functions such as provisioning and abstraction control.

This draft intends to discuss how the PCE architecture and PCEP protocol developed to date can support the transport SDN by analyzing the role PCE plays in some typical use cases.

Optical Enterprise network

PCE can provide transport service in small-scale enterprise network, which is under a totally centralized control environment.

Virtualized transport network

PCE can be used for virtual service provisioning. In this way clients can propose various network requests to operators.

Data-center Interconnection

Current PCE architecture and protocol can support transport services provisioning in data-center Interconnection Networks.

Packet-optical Integration

Packet traffic can be transported over optical transport network. The path computation in this case can be supported by coordinating between Packet PCE and Optical PCE.

This draft describes several classic scenarios in transport network, to demonstrate the current PCE can be extended beyond path computation functionality.

There is NO protocol extension (such as PCEP) in this draft. We only focus on how current PCE (including RFCs and some WG/I-D Draft) can satisfy the requirements.

2. Path Computation in Transport SDN

2.1. Transport SDN Architecture

A general architecture of transport SDN is shown as follow:

```
                                  +-------------+
                                  |   Client    |
                                  | Controller  |
                                  +-------------+
                                        /|\
                                         |
                                        \|/
                             +-------------------------+
                             |Transport   +--------+   |
                             | Network    |  PCE   |   |
                             |Controller  +--------+   |
                             +-------------------------+
                                        /|\
                                         |
                                        \|/
                                +---------------------+
                                |     Physical        |
                                |     Network         |
                                |   Infrastructure    |
                                +---------------------+
```

Fig. 1 Generic Architecture of Transport SDN

As shown in Fig. 1, transport network controller (TNC) is core block that connects with both clients and physical network infrastructure. PCE is one of the functional blocks in the TNC for path computation. In general, to request a transport service, client controller sends a request with path requirement to TNC. PCE will compute a path according to the request, and report the result to client.

For path establishment, the client will trigger the TNC and then TNC will operate on the physical network infrastructure, for example, network elements.

2.2. Path computation and establishment

2.2.1 Stateless PCE

The PCE Protocol (PCEP) is the protocol that enables the communication between Path Computation Clients (PCCs) and PCE. It was firstly developed to support a stateless PCE with in [RFC4655]. PCCs will send path computation requests via the Path Computation Request (PCReq) message to a Path Computation Elements (PCE). The PCE, upon receiving this request, will calculate a path or multiple paths and reply the result to the PCC via the Path Computation Reply (PCRep) message. During the computation, the PCE has access to the Traffic Engineering Database (TED). Network topology and resource usage information are stored in the TED. Specific path computation algorithms or policy-based routing schemes are out of scope for this draft. Other details for PCEP can be referred to [RFC5440].

2.2.2 Stateful PCE

In stateless PCE the network information is managed in TED. However, the state of active LSPs is not managed by PCE and as such there may some limitations for real-time, dynamic LSP operations with stateless PCE. With dynamic configuration and management request, there will be resource contention problem in PCE. To address this issue, stateful PCE is introduced in [draft-ietf-pce-stateful-pce-07], with a LSP Database (LSPD) in the PCE. The LSPD allows efficient LSP state synchronization between PCC and PCEs. A delegation mode is proposed, with allowing PCC to delegate control of its LSP to an active stateful PCE.

The stateful PCE can be applied in various scenarios, as presented in [draft-ietf-pce-stateful-pce-app-01], including online optimization, bandwidth scheduling, recovery and so on.

2.2.3 PCE Initiation

More dynamical management over LSP is proposed in [draft-ietf-pce-pce-initiated-lsp-00], named as PCE initiation. In this mechanism, PCE is able to trigger the creation of LSPs on demand, which is especially suitable for a controller-based network in service provisioning and path setup.

One of the typical applications for PCE initiation is the path computing in SDN. When the request is generated from application stratum, the PCE can compute the path and directly set it up, instead of responding and triggering the application to establish the connection. This new mechanism is more efficient than stateful PCE and can provide better real-time performance in a dynamic transport network.

3. PCE Applicability for Transport SDN

Several applications are proposed under the transport SDN arhictecture to demonstrate its ability to enhance the control and management of transport networks, such as virtual transport services, data center interconnection network and so on. For these applications, PCE is playing an important role. In the following sub-sections, we describe how the PCE and its protocol can support applications in transport SDN via some typical examples.

3.1. Photonic Enterprise Networks

```
                        +---------+
                        |   PCE   |
                        +---------+
                          /  |  \
                         /   |   \
                        /    |    \
                       /     |     \
                      /      |      \
                 +-----+  +-----+ +-----+
                 | NE1 |  | NE2 | | NE3 |
                 +-----+  +-----+ +-----+
```

Fig. 2: PCE Control over Network Element

The enterprise networks are usually a small-scale network with limited network elements. For simplicity, we assume in this use case one PCE is enough for path computation, i.e., no multi-domain or inter-PCE communication is involved. As shown in Fig. 2, NEs are directly connected with PCE.

NEs can be but not limited to data centers. NEs are considered as PCC to create request for path computation and establishment. PCEP can be used for communication between PCCs and PCE, through the interfaces between PCE and NEs. Stateful PCE is suited for this use case for dynamic service provisioning, since the path is setup by the PCC directly.

3.2. Virtualized Transport Network

Virtual transport service (VTS) is one of the advantages of transport SDN. The architecture of providing VTS is described in Fig. 3.

```
                        +------------+       +-----------+
                        |   Client   |       |   Client  |
                        | Controller | ..... | Controller|
                        +------------+       +-----------+
                              /|\                 /|\
                               |               |
                              \|/                 \|/
         +-------------------------------------------------+
         |                      |                   |       |
         |  Transport    +--------------------------------+ |
         |  Controller   |   Virtual Network Controller   | |
         |               +--------------------------------+ |
         |                              /|\                 |
         |  +-------------+              |Provider Interface|
         |  |    Other    |             \|/                 |
         |  |  Functional |       +-------------+           |
         |  |    Blocks   |       |     PCE     |           |
         |  +-------------+       +-------------+           |
         +-----------------------//-/---|-----\-------------+
                              //   /   | \
                            //    / +--|--+   \
                          /       / | NE  |    \
                         //      /  +-----+     \
                       //        /               \
                     //         /                \
               +-----/          /                 \
               | NE  |         /                 +-\---+
               +-----+         /                 | NE  |
                              /                  +-----+
                             /
                         +--/--+
                         | NE  |
                        +-----+
```

Fig. 3: Architecture for VTS Provisioning

In this use case, the network elements are totally infrastructure and we do not consider NE as PCC anymore. The path computation in this case can be divided into two types: virtual path computation and physical path computation.

The virtual path computation requests come from the client controller and are sent to the virtualized network controller (VNC) in the transport controller. Virtual network information such as topology and available resources are stored in the VNC to determine whether the requests can be satisfied or not and then respond the

result to client controller. In this procedure, the VNC can be treated as a PCE, and the interaction between client controller (PCC) and VNC (PCE) is achieved via corresponding interface between PCC and PCE.

The physical path computation is similar as described in section 3.1. The PCE is connected with NEs for path computation. VNC projected the virtual path request from client controller to a physical path request and send it to PCE. The request is from the VNC via a provider interface, which can be either considered as an internal interface in transport controller or an external interface between two blocks, depends on implementation policy. By receiving the request, PCE will check the availability of resources and respond to VNC, also via provider interface.

The path establishment can be completed by stateless PCE, stateful PCE or PCE initiation. Stateless and stateful PCE will allocate the physical resource by configuring the NEs after receiving the corresponding message from PCC. In PCE initiation, dynamic creation and teardown of LSPs are supported by PCE together with responding LSP information to PCCs.

3.3. Data center interconnection

The virtual network services in Data Center Interconnection Network are described in this section. As shown in Fig. 4, the DC controller is connected with network provider controller, while DCs are connected with NEs respectively.

In this use case it is assumed that Data center controller knows all information, including endpoints interfaces, resource, location information and any other application/user related information. For the DC interconnection application, the client controller is the DC controller, which can be an internal entity or an external entity with respect to the relationship with the service provider.

```
                              +------------+
                              | Data Center|
                              | Controller |
                              +------------+
                                   /|\
                                    |
                                   \|/
                              +------------+
                              |  Network   |
                              |  Provider  |
                              | Controller |
                              +------------+
                                   /|\
 +---------+                       |                        +----------+
 |  Data   |                       |                        |   Data   |
 |Center A |                       |                        | Center B |
 +---------+                      \|/                       +----------+
      \                         ------------                       /
       \                  //----            ----\\                 /
        \              ///                        \\\             /
         \          ///               +-----+        \\\        /
          \        /                  | NE  |           \      /
          \      //                   +-/---\\           \\   /
           \    |                      /      \\           | /
            \ |                       /         \\          /
            \|+-----/       +----+ /            \\       /|
            |\| NE  |-------| NE |/              +\\---+/  |
            | +---\-+       +-|--+               | NE  |   |
             |     \          |                  +-----+  |
             |      \\        |                           |
              |       \      |       Transport           |
               \\      \     |        Network          //
                 \      \+--/|-+                      /
                  \\\   /| NE  |                   ///
                     \\/ +-----+                ///
                      // \\----            ----//
                     /         ------------
              +-----/----+
              |   Data   |
              | Center C |
              +----------+
```

Fig. 4 Use case: Data Center Interconnection Network

In this use case the Network Provider Controller is playing as PCE and DC controller is corresponding PCC. The requests can be either from DC or the DC controller, with the DC controller have a full visibility of all DCs. PCE is used to respond the path computation request, to provide virtual network services. Stateless/Stateful PCE and PCE initiation are all applicable in this case, similar as the way described in Section 4.2. The communication between DC controller and Network Provider Controller can also be implemented via PCEP.

3.4. Packet Optical Integration

```
                            +------------+
                            |   Client   |
                            | Controller |
                            +------------+
                                 /|\
                                  |
                                 \|/
           +-----------------------------------------------+
           |                   Network                     |
           |                   Provider                    |
           |                  Controller                   |
           +-----------------------------------------------+
                   /|\                           /|\
                    |                             |
                   \|/                           \|/
            +---------------+             +---------------+
            |    Packet     |             |    Optical    |
            |    Network    |             |    Network    |
            |  Controller   |             |   Controller  |
            +---------------+             +---------------+
                   /|\                           /|\
                    |                             |
                   \|/                           \|/
               /----------\                 /----------\
           ////    IP      \\\\         ////   Optical  \\\\
          |      Packet        |       |      Transport     |
          |      Network       |       |       Network      |
           \\\\            ////         \\\\            ////
               \----------/                 \----------/
```

Fig. 5 Use Case: Packet Optical Integration

In this use case we describe packet traffic transported over an optical transport server network (potentially incorporating multiple layer networks), as shown in Fig. 5. The objective of this use case is for packet and optical topologies to be jointly optimized for greater efficiency, taking advantage of knowledge of topologies and status, as well as dynamic capabilities supported by the optical transport network.

There can be a few variations for path computation in this case, depends on where the PCE is located. In this draft we assume that there is one PCE located in Packet network controller and another PCE located in Optical Network controller, respectively. A joint optimization is applied by Network Provider controller, which is connected with PCEs, for better resource utilization. Moreover, client controller is also connected with network provider controller.

In this case the path computation request is from the client controller. All the three controllers has their respective TED and LSPD (only if stateful or PCE initiation) and there are some

synchronization mechanisms among them to guarantee the resource consistency. Once the path computation request arrives the network provider controller, it will be decomposed according to the network topology and sent to the IP-PCE and Optical PCE respectively. The IP-PCE and Optical PCE will then compute the path and respond.

The procedure of path establishment is similar as described in section 4.2, which is applicable via stateless PCE, stateful PCE and PCE initiation respectively. Communications among the controllers in Fig. 5 can be achieved by PCEP.

4. IANA Considerations

5. References

5.1. Normative References

[RFC4655] Farrel, A., Vasseur, J.-P., and Ash, J., "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006.

[RFC5440] Vasseur, J.-P., and Le Roux, JL., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009.

[draft-ietf-pce-pce-initiated-lsp-00] Crabbe, E., Minei, I., Sivabalan, S., Varga. R, PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model, draft-ietf-pce-pce-initiated-lsp-00, December 2013.

[draft-ietf-pce-stateful-pce-app-01] Zhang, X., Minei, I., Applicability of Stateful Path Computation Element (PCE), draft-ietf-pce-stateful-pce-app-01, September 2013.

[draft-ietf-pce-stateful-pce-07] Crabbe, E., Medved, J., Minei, I., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-07 (work in progress), October 2013.

5.2. Informative References

[Openflow] Openflow Switch Specification, https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.3.0.pdf

6. Authors' Addresses

Haomian Zheng
Huawei Technologies
F3 R&D Center, Huawei Industrial Base,
Bantian, Longgang District,
Shenzhen 518129 P.R.China
Email: zhenghaomian@huawei.com

Xian Zhang
Huawei Technologies
F3 R&D Center, Huawei Industrial Base,
Bantian, Longgang District,
Shenzhen 518129 P.R.China
Email: zhang.xian@huawei.com