PCE Working Group
Internet-Draft
Updates: 8231 (if approved)
Intended status: Standards Track
Expires: September 2, 2018

D. Dhody, Ed.
Huawei Technologies
S. Litkowski
Orange
March 1, 2018

Extension for Stateful PCE to allow Optional Processing of PCEP Objects. draft-dhody-pce-stateful-pce-optional-00

Abstract

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

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 September 2, 2018.

Copyright Notice

Copyright (c) 2018 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

[RFC5440] describes the Path Computation Element communication Protocol (PCEP) which enables the communication between a Path Computation Client (PCC) and a Path Control Element (PCE), or between two PCEs based on the PCE architecture [RFC4655].

PCEP Extensions for Stateful PCE Model [RFC8231] describes a set of extensions to PCEP to enable active control of Multiprotocol Label Switching Traffic Engineering (MPLS-TE) and Generalzied MPLS (GMPLS) tunnels. [RFC8281] describes the setup and teardown of PCE-initiated LSPs under the active stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic network.

[RFC5440] defined P flag (Processing-Rule) as part of Common Object Header to allow a PCC to specify in a PCReq message sent to a PCE whether the object must be taken into account by the PCE during path

computation or is just optional. The I flag (Ignore) is used by a PCE in a PCRep message to indicate to a PCC whether or not an optional object was processed. Stateful PCE [RFC8231] specified that P and I flags of the PCEP objects defined in [RFC8231] is to be set to 0 on transmission and ignored on receipt since they are exclusively related to path computation requests. The behavior for P and I flag in other objects defined in [RFC5440] and other extension was not specified. This document clarifies how the P and I flag could be used in the stateful PCE model to identify optional objects in the Path Computation State Report (PCRpt), the Path Computation Update Request (PCUpd) and the LSP Initiate Request (PCInitiate) message.

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

1.1. Requirements Language

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

2. Overview

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

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

2.1. Usage Example

The PCRpt message is used to report the current state of an LSP. As part of the message both the <intended-attribute-list> and <actual-attribute-list> is encoded. The <intended-attribute-list> could include the METRIC object to indicate a limiting constraint (B flag set) for the Path Delay Variation metric [RFC8233]. In some scenarios it would be useful to state that this limiting constraint can be relaxed by the PCE, in case it cannot find a path. Similarly in case of an association groups [I-D.ietf-pce-association-group]

such as Disjoint Association [I-D.ietf-pce-association-diversity], the PCE may need to completely relax the disjointness constraint in order to provide a path to all the LSPs that are part of the association. In these case it would be useful mark the objects as optional and could be ignored by the PCEP peer. Also it would be used for the PCEP speaker to learn if the PCEP peer has relaxed the constraint and ignored the processing of the PCEP object.

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

3. PCEP Extension

3.1. STATEFUL-PCE-CAPABILITY TLV

A PCEP speaker indicates its ability to support for handling P and I flag during the stateful PCEP message exchanges 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, as defined in [RFC8231]. A new flag, the R (RELAX) flag, is introduced to this TLV to indicate support for relaxing the processing of some objects via the use of P and I flag in PCEP common object header.

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

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

3.2. Handling of P flag

3.2.1. The PCRpt message

The P flag in the PCRpt message [RFC8231] allows a PCC to specify to a PCE whether the object must be taken into account by the PCE (during path computation or re-optimization) or is just optional. When the P flag is set, the object MUST be taken into account by the PCE. Conversely, when the P flag is cleared, the object is optional and the PCE is free to ignore it. The P flag for the mandatory objects LSP and ERO (intended path) MUST be set in the PCRpt message.

If the mandatory object is received with the P flag set incorrectly according to the rules stated above, the receiving peer MUST send a PCErr message with Error-Type=10 (Reception of an invalid object) and Error-value=1 (reception of an object with P flag not set). By default, the PCC SHOULD set the P flag, unless a local configuration or local policy indicates that some constraints (corresponding PCEP objects) can be marked as optional and could be ignored by the PCE.

3.2.2. The PCUpd message and the PCInitiate message

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

3.3. Handling of I flag

3.3.1. The PCUpd message

The I flag in the PCUpd message [RFC8231] allows a PCE to indicate to a PCC whether or not an optional object was processed. The PCE MAY include the ignored optional object in its update request and set the I flag to indicate that the optional object was ignored. When the I flag is cleared, the PCE indicates that the optional object was processed. Note that for the delegated LSPs, the PCE can update and mark some object as ignored even when the PCC had set the P flag during delegation.

3.3.2. The PCRpt message

The I flag in the PCRpt message [RFC8231] allows a PCC to indicate to a PCE whether or not an optional object was processed in response to an LSP Update Request. The PCC MAY include the ignored optional object in its report and set the I flag to indicate that the optional object was ignored at PCC. When the I flag is cleared, the PCC indicates that the optional object was processed. The I flag has no

meaning if the PCRpt message is not in response to a PCUpd or PCInitiate message.

3.3.3. The PCInitiate message

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

3.4. Unknown Object Handling

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

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

4. Security Considerations

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

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

5. IANA Considerations

5.1. STATEFUL-PCE-CAPABILITY TLV

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

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

6. Manageability Considerations

6.1. Control of Function and Policy

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

6.2. Information and Data Models

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

6.3. Liveness Detection and Monitoring

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

6.4. Verify Correct Operations

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

6.5. Requirements On Other Protocols

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

6.6. Impact On Network Operations

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

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/info/rfc5440>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

7.2. 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>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

[RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <https://www.rfc-editor.org/info/rfc6952>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <https://www.rfc-editor.org/info/rfc7525>.

[RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki, "Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September 2017, <https://www.rfc-editor.org/info/rfc8233>.

[RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <https://www.rfc-editor.org/info/rfc8253>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

[I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-06 (work in progress), January 2018.

[I-D.ietf-pce-association-diversity] Litkowski, S., Sivabalan, S., Barth, C., and D. Dhody, "Path Computation Element communication Protocol extension for signaling LSP diversity constraint", draft-ietf-pce-association-diversity-03 (work in progress), February 2018.

[I-D.ietf-pce-association-group] Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "PCEP Extensions for Establishing Relationships Between Sets of LSPs", draft-ietf-pce-association-group-04 (work in progress), August 2017.

Authors' Addresses

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

EMail: dhruv.ietf@gmail.com

Stephane Litkowski
Orange

EMail: stephane.litkowski@orange.com

PCE Working Group
Internet-Draft
Updates: 8231 (if approved)
Intended status: Standards Track
Expires: December 22, 2018

D. Dhody, Ed.
Huawei Technologies
S. Litkowski
Orange
June 20, 2018

Extension for Stateful PCE to allow Optional Processing of PCEP Objects. draft-dhody-pce-stateful-pce-optional-01

Abstract

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

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 December 22, 2018.

Copyright Notice

Copyright (c) 2018 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

[RFC5440] describes the Path Computation Element communication Protocol (PCEP) which enables the communication between a Path Computation Client (PCC) and a Path Control Element (PCE), or between two PCEs based on the PCE architecture [RFC4655].

PCEP Extensions for Stateful PCE Model [RFC8231] describes a set of extensions to PCEP to enable active control of Multiprotocol Label Switching Traffic Engineering (MPLS-TE) and Generalzied MPLS (GMPLS) tunnels. [RFC8281] describes the setup and teardown of PCE-initiated LSPs under the active stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic network.

[RFC5440] defined P flag (Processing-Rule) as part of Common Object Header to allow a PCC to specify in a PCReq message sent to a PCE whether the object must be taken into account by the PCE during path computation or is just optional. The I flag (Ignore) is used by a PCE in a PCRep message to indicate to a PCC whether or not an optional object was processed. Stateful PCE [RFC8231] specified that P and I flags of the PCEP objects defined in [RFC8231] is to be set to 0 on transmission and ignored on receipt since they are exclusively related to path computation requests. The behavior for P and I flag in other objects defined in [RFC5440] and other extension was not specified. This document clarifies how the P and I flag could be used in the stateful PCE model to identify optional objects in the Path Computation State Report (PCRpt), the Path Computation Update Request (PCUpd) and the LSP Initiate Request (PCInitiate) message.

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

1.1. Requirements Language

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

2. Overview

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

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

2.1. Usage Example

The PCRpt message is used to report the current state of an LSP. As part of the message both the <intended-attribute-list> and <actual-attribute-list> is encoded. The <intended-attribute-list> could include the METRIC object to indicate a limiting constraint (B flag set) for the Path Delay Variation metric [RFC8233]. In some

scenarios it would be useful to state that this limiting constraint can be relaxed by the PCE, in case it cannot find a path. Similarly in case of an association groups [I-D.ietf-pce-association-group] such as Disjoint Association [I-D.ietf-pce-association-diversity], the PCE may need to completely relax the disjointness constraint in order to provide a path to all the LSPs that are part of the association. In these case it would be useful to mark the objects as 'optional' and it could be ignored by the PCEP peer. Also it would be used for the PCEP speaker to learn if the PCEP peer has relaxed the constraint and ignored the processing of the PCEP object.

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

3. PCEP Extension

3.1. STATEFUL-PCE-CAPABILITY TLV

A PCEP speaker indicates its ability to support for handling P and I flag during the stateful PCEP message exchanges 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, as defined in [RFC8231]. A new flag, the R (RELAX) flag, is introduced to this TLV to indicate support for relaxing the processing of some objects via the use of P and I flag in PCEP common object header.

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

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

3.2. Handling of P flag

3.2.1. The PCRpt message

The P flag in the PCRpt message [RFC8231] allows a PCC to specify to a PCE whether the object must be taken into account by the PCE (during path computation or re-optimization) or is just optional. When the P flag is set in PCRpt message, the object MUST be taken

into account by the PCE. Conversely, when the P flag is cleared, the object is optional and the PCE is free to ignore it. The P flag for the mandatory objects LSP and ERO (intended path) MUST be set in the PCRpt message. If the mandatory object is received with the P flag set incorrectly according to the rules stated above, the receiving peer MUST send a PCErr message with Error-Type=10 (Reception of an invalid object) and Error-value=1 (reception of an object with P flag not set). By default, the PCC SHOULD set the P flag, unless a local configuration or local policy indicates that some constraints (corresponding PCEP objects) can be marked as optional and could be ignored by the PCE.

3.2.2. The PCUpd message and the PCInitiate message

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

3.3. Handling of I flag

3.3.1. The PCUpd message

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

3.3.2. The PCRpt message

The I flag in the PCRpt message [RFC8231] allows a PCC to indicate to a PCE whether or not an optional object was processed in response to an LSP Update Request or LSP Initiate Request. The PCC MAY include the ignored optional object in its report and set the I flag to indicate that the optional object was ignored at PCC. When the I

flag is cleared, the PCC indicates that the optional object was processed. The I flag has no meaning if the PCRpt message is not in response to a PCUpd or PCInitiate message (i.e. without the SRP object in PCRpt message).

3.3.3. The PCInitiate message

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

3.4. Delegation

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

The PCC MUST confirm this by sending the PCRpt message with the P flag set as per the PCE expectation for the corresponding object. In case PCC cannot except this, it would reach as per the processing rules in [RFC8231].

3.5. Unknown Object Handling

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

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

4. Security Considerations

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

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

5. IANA Considerations

5.1. STATEFUL-PCE-CAPABILITY TLV

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

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

6. Manageability Considerations

6.1. Control of Function and Policy

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

6.2. Information and Data Models

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

6.3. Liveness Detection and Monitoring

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

6.4. Verify Correct Operations

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

6.5. Requirements On Other Protocols

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

6.6. Impact On Network Operations

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

7. Acknowledgments

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

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/info/rfc5440>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

8.2. Informative References

[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>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

[RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <https://www.rfc-editor.org/info/rfc6952>.

[RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <https://www.rfc-editor.org/info/rfc7525>.

[RFC8051] Zhang, X., Ed. and I. Minei, Ed., "Applicability of a Stateful Path Computation Element (PCE)", RFC 8051, DOI 10.17487/RFC8051, January 2017, <https://www.rfc-editor.org/info/rfc8051>.

[RFC8233] Dhody, D., Wu, Q., Manral, V., Ali, Z., and K. Kumaki, "Extensions to the Path Computation Element Communication Protocol (PCEP) to Compute Service-Aware Label Switched Paths (LSPs)", RFC 8233, DOI 10.17487/RFC8233, September 2017, <https://www.rfc-editor.org/info/rfc8233>.

[RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <https://www.rfc-editor.org/info/rfc8253>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

[I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-07 (work in progress), March 2018.

[I-D.ietf-pce-association-group]
Minei, I., Crabbe, E., Sivabalan, S., Ananthakrishnan, H., Dhody, D., and Y. Tanaka, "PCEP Extensions for Establishing Relationships Between Sets of LSPs", draft-ietf-pce-association-group-06 (work in progress), June 2018.

[I-D.ietf-pce-association-diversity]
Litkowski, S., Sivabalan, S., Barth, C., and D. Dhody, "Path Computation Element communication Protocol extension for signaling LSP diversity constraint", draft-ietf-pce-association-diversity-04 (work in progress), June 2018.

Authors' Addresses

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

EMail: dhruv.ietf@gmail.com

Stephane Litkowski
Orange

EMail: stephane.litkowski@orange.com

Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

T. Eckert
Huawei

Mar 5, 2018

Framework for Traffic Engineering with BIER-TE forwarding (Bit Index Explicit Replication with Traffic Engineering) draft-eckert-teas-bier-te-framework-00

Abstract

BIER-TE is an application-state free, (loose) source routed multicast forwarding method where every hop and destination is identified via bits in a bitstring of the data packets. It is described in [I-D.ietf-bier-te-arch]. BIER-TE is a variant of [RFC8279] in support of such explicit path engineering.

This document described the traffic engineering control framework for use with the BIER-TE forwarding plane: How to enable the ability to calculate paths and integrate this forwarding plane into an overall TE solution.

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 September 6, 2018.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction and Overview

This document proposes a framework and abstract data model for the control plane of BIER-TE as defined in [I-D.ietf-bier-te-arch] (BIER-TE-ARCH). That document primarily defines the forwarding plane and provides some example scenarios how to use it.

BIER-TE is a forwarding plane derived from BIER ([RFC8279]) in which the destinations of packets are bits in a bitstring. Every bit indicates a destination (BFER - BIER Forwarding Exit Router) and an IGP is used to flood those "bit addresses" so hops along the path from sender (BFIR - BIER Forwarding Ingres Router) through intermediate nodes (BFR) can calculate the shortest path for each destination (bit) and simply copy the received packet to every interface to one or more bits set in the packet.

In BIER-TE, shortest path calculation is replaced by bits of the bitstring indicating intermediate hops and pre-populated forwarding tables (BIFT - Bit Index Forwarding Tables) on every BFR indicating

those bits. In the simplest case, every interface on a BFR has a unique bit assigned to it, and the BIFT of only that BFR will have in its BIFT for this bit an adjacency entry indicating that interface. This ultimately allows to indicate any sub-graph of the network topology as a bitstring and hop-by-hop perform the necessary forwarding/replication for a packet with such a bitstring. More complex semantics of bits are used to help saving bits. A typical bitstring size supportable is 256 bits, the original BIER specification allows up to 1024 bits. BIER-TE may be specifically interesting for typically smaller topologies such as often encountered in DetNet scenarios, or else through intelligent allocating and saving of bits for larger topologies, some of which is exemplified in BIER-TE-ARCH.

One can compare BIER-TE in function to Segment Routing in so far that it attempts to be as much as possible a per-packet "source-routed" (for lack of better term) forwarding paradigm without per-application/flow state in the network. Whereas SR primarily supports simple sequential paths indicated as a sequence of SIDs, in BIER-TE, the bitstring indicate a directed and acyclic graphs (DAG) - with replications. BIER-TE can also be combined with SR and then bits in the bitstring are only required for the nodes (BFR) where replication is desired, and the paths between any two such replication nodes could be SIDs or stack of SIDs that are selected by assigning bits to them (routed adjacencies in the BIER-TE terminology).

In BIER-TE-ARCH, the control plane is not considered. In its place, a theoretical BIER-TE Controller Host uses unspecified signaling to control the setup of the BIER-TE forwarding-plane end to end (all bits/adjacencies in all BFR BFITs) and during the lifecycle of network device install through the determination of paths for specific traffic and changes to the topology. This document expands and refines this simplistic model and intends to serve as the framework for follow-up protocol and data model specification work.

The core forwarding documents relevant to this document are as follows:

- [RFC8279] (BIER-ARCH): as summarized above.
- [RFC8296] (BIER-ENCAP): The encapsulation for BIER packets using MPLS or non-MPLS networks underneath.
- [I-D.ietf-bier-te-arch] (BIER-TE-ARCH): as summarized above.
- [I-D.thubert-bier-replication-elimination] (BIER-EF-OAM): Extends the BIER-TE forwarding from BIER-TE-ARCH to support the Elimination Function (EF) and an OAM function. The Elimination

Function is a term from DetNets resilience architecture: Multiple copies of traffic flows are carried across disjoint path, merged in a BFR running the EF and duplicates are eliminated on that BFR based on recognizing duplicate sequence numbers. Engineered multiple transmission paths are a key reason to leverage BIER-TE.

- [I-D.huang-bier-te-encapsulation] (BIER-TE-ENCAP): Proposed encapsulation based on an extension of BIER-ENCAP. Identifies whether the packet expects to use a BIER or BIER-TE BIFT. Also adds a control-word in support of (optional) elimination function (EF) and interprets the pre-existing BFIR-ID and entropy fields as a flow-id.

- [I-D.eckert-bier-te-frr] (BIER-TE-FRR): This document describes protections methods applicable to BIER-TE. 1:1 / end-to-end path protection is referenced in this document in the context of DetNet style PREF path protection. The options not discucced yet (TBD) in this document are link protection tunnels (such as used in RSVP-TE as well) and the novel BIER-TE specific protection method, in which nodes modify the bitstring upon local discovery of a failure.

The relevant routing underlay documents are as follows:

- [I-D.ietf-bier-isis-extensions] (BIER-ISIS), [I-D.ietf-bier-ospf-bier-extensions] (BIER-OSPF): The BIER-ISIS and BIER-OSPF documents describe extensions to those two IGPs in support of BIER. Effictively, every BFR announces the <SD,SI-range> BIFTs it is configured for, the MT-ID (IGP Multitopology-ID) they are using, and the BFR-ID it has in each SD (none if it does not need to operate as a BFER). For MPLS encapsulation, the base label for every SD is announced as well as the SI-range (one label per <SD,SI> is used).

- There is currently no document describing IGP extensions for BIER-TE, but the goal is to define those based, using the proposals made in this framework, and as feasible re-using and/or amending those existing BIER IGP extensions.

- [I-D.ietf-bier-bier-yang] (BIER-YANG): This document describes the YANG data model to provision on every BFR BIER. It also provides OAM functions. There is currently no model expanding this to support BIER-TE. This framework document defines elements that should be included in a BIER-TE YANG model.

- TBD: incomplete list ?.

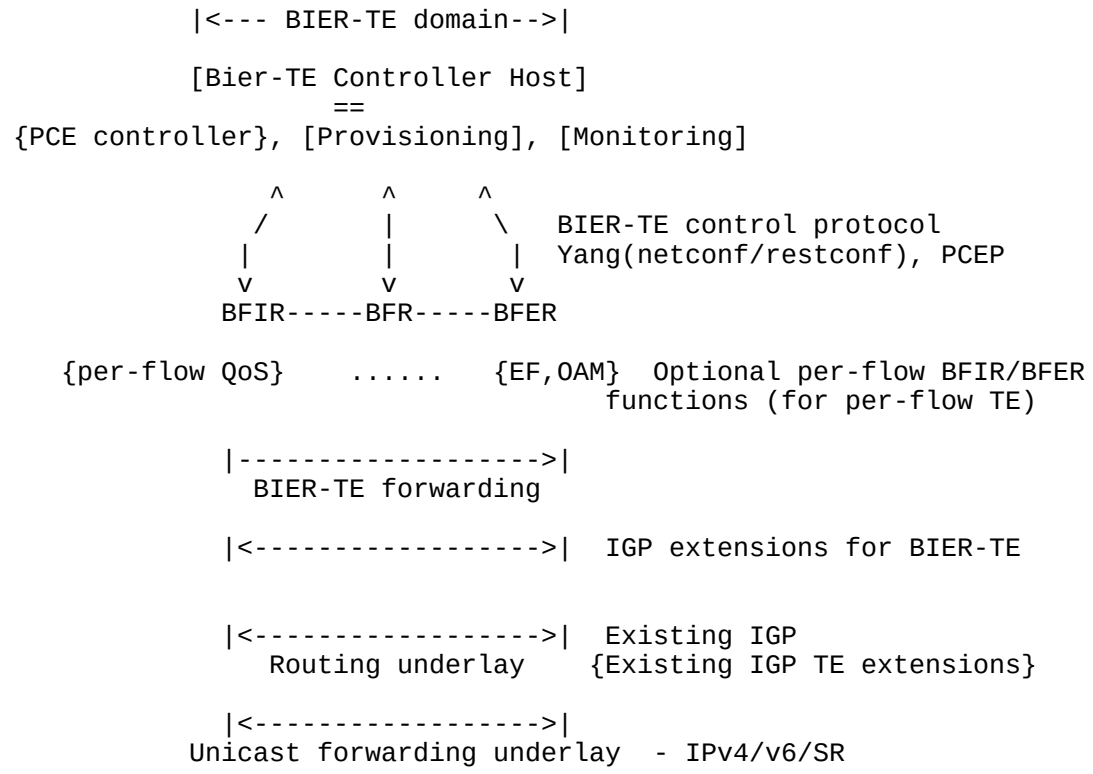


Figure 1: BIER-TE signaling architecture

The above picture is a modified version of Picture 2 from BIER-TE-ARCH reduced by the elements not considered in this document, and refined with those that are intended to be described by this document.

In comparison with BIER-TE-ARCH, Picture 2, this picture and this document do not include considerations for specific multicast flow overlay elements. Instead, it adds description of optional BFIR/BFER elements for per-flow QoS/EF (Elimination Function) and OAM, which are optional parts of an overall BIER-TE traffic engineering architecture. See BIER-EF-OAM for more background.

The routing underlay is refined in this document to consider a unicast forwarding underlay of IPv4/IPv6 and/or unicast SR (Segment Routing) for BIER-TE "forward_routed" adjacencies. It also assumes an existing IGP, such as ISIS or OSPF as the routing underlay. This may include (TBD) extensions already supporting TE aspects (like those IGP extensions done for RSVP-TE).

This framework intends to support a wide range of options to instantiate it:

In one extreme (PCEC only), there is no IGP in the network that BIER-TE depends on, but all BIER-TE operations is managed in an SDN-style fashion from centralized components called "BIER-TE Controller Host" in BIER-TE-ARCH. This central packend can be further subdivided into a Configuration/Provisioning component to install the BIER-TE topology into the network and a PCEC (Pat Computation Engine Controller) and (TBD) monitoring components. After BIER-TE is operational, the PCEC calculates BIER-TE bitstrings for BFIR when they need to send traffic flow to

In the other extreme (IGP only), there is no need for a PCEC or NMS. The initial setup of the BIER-TE topology can be performed manually, using configuration options to support automatic consistency checking and partial auto-configuration to simplify this work. BIER-TE extensions of the IGP are used for consistency checking and autoconfiguration and finally to provide the whole BIER-TE topology to BFIR that can then autonomously calculate BIER-TE bitstrings without the help of a PCEC.

2. BIER-TE Topology management

2.1. Operational model

When a network is installed, BIER-TE is added as a service or later when it is meant to change, BFR need to be (re)provisioned. This involves a planning phase which physical adjacencies (links) should be used in the BIER-TE topology, and which virtual adjacencies (routed adjacencies) should be created and assigned bits. Ultimately this means the definition of the BIER-TE topology.

When the physical topology if the network is smaller than the possible bitstring size (e.g.: 256 bits), then this can be a simple, fully automated process. Likewise, if multiple disjoined services for BIER-TE each require active subsets of the network topology smaller than the network topology, it likewise can be simple to create a different SD (subdomain) BIER-TE topologies for each such service.

When the required network topology for a BIER-TE service exceeds the supportable bitstring size, bit-saving mechanisms can be employed as described in BIER-ARCH. Some of them such as p2p link bits or lan-bits are easily automatically calculated. Creation of virtual adjacencies (routed adjacencies) may likely best be done with operator defined policies applied to a system a system calculating the bits for the BIER-TE topology.

Ultimately, if the set of required destinations plus transit hops exceeds the size of available bitstrings after optimization, multiple BIFT == bitstrings need to be allocated to support this case. These multiple BIFT will likely need to be engineered to minimize duplicate traffic load on the network and minimize bit use. One example shown in BIER-TE-ARCH is to allocate different <SD,SI> BIFT to different areas of a network, therefore having to create one BIER-TE packet copy per required destination region, but in result having only one packet copy in each of those regions.

Provisioning / initial setup can be done manually in simpler networks or through a provisioning system. A PCEP may equally perform this function. If a PCEP is not used to perform this function, but a PCEP is used later for Flow Management, then the PCEP does of course need to also learn the BIER-TE topologies created by the provisioning system.

Unless a PCEC is used for provisioning/initial setup, YANG is likely the prefered model to install the BIER-TE topology information into the BFR. If a PCEC is used, YANG or PCEC seem to be valid choices.

When the network topology expands, bit assignements for the new parts of the topology need to be made. If expansion was not factored into the initial bit assignment plans, this can lead to the need to reassign bits for existing parts of the topology. Support for such processes could be simplified through additional topology information, for example to enable seamless switching of traffic flows from bits in one SD over to bits in another SD. This is currently not considered in this document.

2.2. BIER-TE topology model

```
<BFR> BIFT information:
  Instance: "configured", "operational",
            "learned-configured", "learned-operational" (pce, igp)
    BIFT-ID: <SD subdomain,BSL bitstring length,SI Set Identifier>
    BIFT-Name: string (optional)
    BFR-ID: 16 bit (BIER-TE ID of the <bfr> in this BIFT
                    or undefined if not BFER in this BIFT)
    Ingres-groups: (list of) string (1..16 bytes)
                   (that <bfr> is a member of)
    EF:  <TBD> (optional, parameters for EF Function on this BIFT)
    OAM: <TBD> (optional, parameter for OAM Function on this BIFT)
    Bits: (#BSL - BitStringLength)
      BitIndex: 1...BSL
      BitType(/Tag): "unassigned",
                     (if unassigned, must have no adjacencies)
                     "unique", "p2p", "lan", "leaf", "node", "flood",
                     "group"
                     (more BitTypes defined in text below)
      Names: (list of 0 or more) string (1..16 bytes)
             (for BitTypes that require it)
      List of 0 or more adjacencies:
        (The following is the list of possible types of adjacencies,
         as defined in BIER-TE-ARCH with parameters)
        local_decap:
          VRFcontext: string (TBD)
        forward_connected:
          destination-id: ip-addr (4/16 bytes, router-id/link-local)
          link-id: ifIndex Value (connecting to destination)
          boolean: DNR (Do Not Reset)
        forward_routed:
          destination-id: 20 bit (SID), 4 or 16 bytes (router-id)
          TBD: path/encap information (e.g: SR SID stack)
        ECMP:
          list of 2 or more forward_connect and/or
          forward_routed adjacencies
```

Figure 2: BIER-TE topology information

The above picture shows informally the data model for BIER-TE topology information. <BFR> is a domain-wide unique identifier of a BFR, for example the router-id of the IGP (if an IGP is used). Every <BFR> has a "configured" instance of the BIFT information for every BIFT configured on it. This configuration could be created from legacy models, a YANG model, PCEP, or other means.

Every <BFR> also has an "operational" instance of the BIFT information. If the BFR has nor "learned-configured" / "learned-operational" information, then the "operational" instance is just a

copy of the "configuration" instance, but would take additional local information into account. For example, if resource limits do not allow to activate configured BIFT. Or when bits in the BIFT point to interfaces/adjacencies that are down, this could potentially also be reflected in the operational instance. While the "configuration" instance is read/write, the operational instance is read-only (from NMS or PCEC).

To calculate paths/bitstrings through the topology without the help of a PCEC, a BIFT would need to know the network wide BIER-TE topology. This topology consists of the "operational" BIFT informations of the BFR itself plus the "learned-operational" BIFT information from all other BIER-TE nodes in the network plus the underlay routing topology information, for example from an IGP. When an IGP is used, the "learned-operational" information of another BFR is simply learned because the BFRs are flooding this information as IGP information.

In the absence of any IGP, or the desire not to use it to distribute BIER-TE topology information, an NMS or PCEC could collect the "operational" BIER-TE topology information from BFRs and distribute it to BFIR to enable them to calculate BIER-TE bitstrings autonomously.

The operational instance of the topology information can depend on the presence of an IGP. If the adjacency of a bit in the BIFT is configured to use a nexthop identifier that has to be learned from an IGP, such as a Segment Routing SID or a router-ID, then the operational instance (as well as distributed learned-operational ones) would indicate that such an adjacency is non-operational if the BFR could not resolve this nexthop information. Forward_connected adjacencies do not require a routing underlay, but just link-local connectivity.

Some information elements in the BIER-TE topology information is metadata to support automatic consistency checking of learned topology information which permit to prohibit use of adjacencies that would not lead to working paths or worst case could create loops. The same information can also be used to auto-configure some adjacencies, specifically routed adjacencies, allowing to minimize operator work in case BIFT topology information is not auto-created from an NMS/PCEP but through manual mechanisms, but also to automatically discover mis-wirings and avoid them to be used.

The semantic of BitType and Names are described in conjunction with consistency checking and autoconfiguration in the following sections.

2.3. Consistency checking

The BitType and associated Name or Names for the bit are intended to support automated consistency checking and different reactions. an NMS can for example discover misconfiguration or miscablings and alert the operator. BFIR can likewise discover misconfiguration when the "configured" and "operational" instances of BFR are distributed via the IGP and are therefore available as "learned-configured" and "learned-operational" on the BFIR. The BFIR can then fr example stop using those misconfigured bits in any bitstrings it calculates and further escalate (e.g.: overlay signaling) unreachability of any BFER (or inability to calculate paths supporting required TE features).

"Unique" bits doe not require a name, but the <SD,SI> bit in question must only have an adjacency on one BFR. If it shows up with adjacencies on more than one BFR, this is an inconsistency.

"p2p" bits need to be the same bit on both BFR connected to each other via a subnet, and must be pointing to each other via "forward_connected" adjacencies. A "p2p" bit needs to have one Name parameter unique in the domain - for example constructed from concatenating the IfIndex of both sides. Note that the actual subnet does not need to be p2p, a BFR can have multiple bits across a multiaccess subnet, one for each neighbor.

Not listed in the above picture, but a "remote-p2p" could be a BitType when a bidirectional adjacency between two remode BFR using forward_routed adjacencies.

A "leaf" bit is the one shared bit in a <SD,SI> bitstring assigned to the "local_decap" adjacency on all leaf BFER. Leaf BFER do not need a separate bit. See BIER-TE-ARCH. If more then one "lead" bits are used in an <SD,SI> across the domain that is an inconsistency - waste of bits.

A "node" bit is associated with a Name that follows a standardized form to identify a node - e.g.: its router-id. On a non-leaf BFER, this bit can only have one local_decap adjacency on the node indicated itself. On a leaf BFER, the "node" bit must be assigned to adjacencies on one or BFR that connect to the indicated BFER. Other configurations (or wirings) are a misconfiguration.

A "lan" bit indicates a bit for a LAN, as discussed in BIER-TE-ARCH. It must have one domain wide unique name. It must only be used by BFR connecting to the same subnet with a set of forward_connected adjacencies pointing to the other BFR on that subnet. Disabling the use of a "lan" bit either on a BFIR when sending packets, or even more son on the actual BFR connecting to a subnet and recognizing

inconsistent BIER-TE topolocy configuraiton for it - is the most important automatic function to avoid mis-routing of BIER-TE packets. The looping will be also stopped because bits are reset when packets traverse the paths, or ultimately by TTL, but neither mechanism can provide as specifica OAM information about what went wrong than recognizing inconsistencies via the IGP.

TBD: flood bit, DNR (like lan bit, but more complex.

Consistency checking may happen directly during configuration as well as later during rewiring/remot changes of topology.

In general, the operational instance of the BIER-TE topology are relevant to topology consistency checking (as hey are for path calculations). For example, future extensions may actually introduce some form of node/BFR redundancy where different BFR are configured for the same bits, but only one at a time is actively using a bit, and therefore announcing it in the operational instance of the BIER-TE topology.

2.4. Auto-configuration

For subnets, the actual adjacency to the neighbor on a link may not actually be configured explicitly, but only the interface. Discovery of the neighbor via the IGP would result in a complete working adjacency for a bit, and that adjacency would show then in the operational instance - while the configured instance would only show an incomplete adjacency and the bit that was configured for the adjacency. The Name parameter can be used in configuration to lock in the BFR that is expected to be on the other side of a subnet interface. If that node is not the one actually connected, the adjacency in the operational instance would not be completed.

When a "p2p" BitType is used, but the bit is configured inconsistently on both sides of a p2p link, an autoconfiguration mechanism may be specified to select which of the two bits should be used (e.g.: bit number configured on the higher router-id peer). This could help to auto-correct a configuration mistake, but it does of course not recover the inconsistently configured bit directly, it just ignores it.

When a "lan" or "flood" BitType is configured, likewise auto-configuration can be done to overcome misconfigurations. TBD: more details.

Most importantly, configuration of routed adjacencies can create most need for network-wide consistent configuration. This can be automated with the proposed "group" bittype.

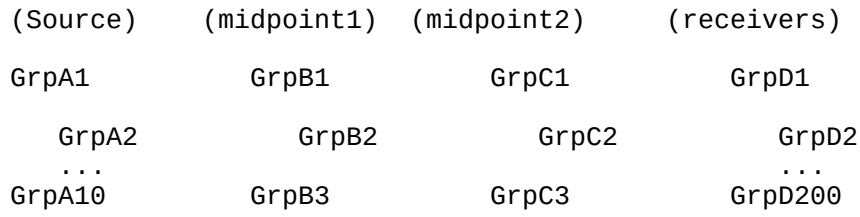


Figure 3: Group BitType use

The typical set of forward_routed adjacency is to allow steering of BIER-TE packets through a sequence of one or more members of a hop-group, load-balancing across them for TE reasons. In the above picture, those paths would start from a BFIR in GrpA and go via one (or more) nodes in GrpB, then GrpC and then BFER (GrpD).

To half-automate the setup of such loose hops, each member of GrpC would for example be configured with one unique bit of BitType "group" and the Name parameter would be set to "GrpB". Each midpoint1 BFR would "GrpB" in the list of strings for the BIFT Ingres-Group parameter. When such a BFR discovers (e.g.: via the IGP) a BFR "learned-operational" bit of BitType group with a name "GrpB" (and no adjacency!), then that midpoint1 BFR would create an adjacency in its "operational" instance, pointing to the announcing BFR with a "forward_routed" adjacency.

The saving through such group BitTypes is therefore that the bit had only to be configured on one node (the receiver side of the forward_routed adjacency), but would be configured on any number of ingres BFR for the adjacency. In the above picture, the benefit would be biggest if forward_routed adjacencies where used from Source to midpoint1, because the number of Sources is potentially largest (e.g: as shown in the picture 10 BFIR in Source group).

3. Flow Management

3.1. Operational / Architectural Models

Once a BIER-topology is active in a network, it can be used to pass BIER-TE packets. Typically this also requires the provisioning of some routing overlay because today, all applications defined for BIER today are classical SP PE-PE application where some customer traffic is mapped to SP traffic via PE-PE "overlay" signaling.

Applications in future environments such as industrial control or IoT may result in different overlay signaling. Even native end-to-end BIER-TE from application stacks is possible, but has so far not been defined.

Overlay signaling is currently out of scope of this document.

3.1.1. Overprovisioning

In the "overprovisioning flow management" model, the network operator is responsible to engineer the available network resources, BIER-TE Topology and applications generating BIER-TE flows such that the required resources can be guaranteed without contention - and potentially without the help of either PCEP or IGP, but simply using provisioning to configure BFIR and overlay signaling to determine active destinations.

Overprovisioning is the most control/signaling lightweight approach and currently the standard approach in most enterprises and service provider for IP multicast traffic.

For example: An ISP with a ++40Gbps network and a comparable small amount of high-value muticast traffic requiring in aggregate less than 5 Gbps can easily carry all of that multicast traffic across any available path. This is especially easy when the mayority of traffic is best effort traffic (such as Internet traffic). In that case, the multicast traffic would be carried in a traffic class that is overprovisioned, for example with 6 Gbps guaranteed on every link. Calculated BIER-TE bitstrings would for example be used to reduce cost of multicast distribution (e.g.: steiner tree calculation), use disjoint paths (in conjunction with EF), or simply load-balance across all available non-ECMP paths. Overprovisioning flow management is traditional in most SP networks (core/edge/access) for IP multicast traffic and requires no additional signaling.

The overprovisioning flow management model is one that likely would request for (only) a YANG model to provision the BIER-TE topology.

3.1.2. PCEC

In the PCEC based flow management model, a PCEP determines (calculates) the (flow-id,<SD,SI>,bitstring) for a traffic flows and signals this to the BFIR sourcing the flow (its BFR-ID is part of the flow-id). If the flow was not statically defined, then this step would be preceeded with the BFIR requesting the resources for the, indicating the requested resources as well as the set of destinations. The destinations could be indicated as BFR-ID or (likely easier for the BFIR) by their unique identifiers in unicast routing (e.g.: router-ID). The bitstring returned by the PCEP would include not only engineered paths to all these destinations, but those paths could also be disjoint paths, carrying the traffic twice towards each destination and merging them via the EF function. The BFIR could be fully agnostic to these PCEP choices.

One of the core benefits of using BIER-TE forwarding is the ability to change the bitstring on a per-packet basis to re-route traffic by setting different transit bits, or to quickly add/delete destinations. When the BFIR should be empowered to perform any of these functions without the need for help by the PCEP, then the PCEP needs to provide additiona information back to the BFIR.

If a BFIR has for example an OAM capability to determine without the help of a controller that a path has failed (too much packet loss on destination, signalled back to BFIR), and dual-transmission is not desired (due to double resource usage), then the PCP and BFIR could co-operate on a path-protection scheme in which the PCEP provides for flows not one, but two bitstrings, one being the backup path which is used by the BFIR when it discovers via OAM loss on the currently used path. This approach can extremely reduce the need to rely on controller help during failures.

When the destinations for a particular flow can potentially change over time, this can often be faster and more efficiently signalled directly via the overlay signaling to the BFIR instead of going through the PCEP. To support this mode of operations, the BFIR could request from the PCEP not simply the current set of destinations for a flow, but instead the maximum superset of receivers and request per-destination information. The PCEP would then return not just one bitstring, but one bitstring per destination (BFER). The BFIR would simply OR the bitstrings for all required destinations for each packet to create the final bitstring for that packet. Note that this description of of course on a per-<SD,SI> (aka: per BIFT) basis. Destinations using different BIFTs require always different BIER-TE packets to be sent by the BFIR.

3.1.2.1. per-flow QoS - policer/shaper/EF

In the PCEP based resource management model, it is up to the PCEP to determine how explicit resource reservations should be managed, e.g.: whether or how it tracks resource consumption. The BIER-TE forwarding plane itself does not support per-flow state with the exception of EF, which would usually be a function enabled on BFER.

Likewise, per-flow policer and/or shaper state may be a useful optional feature that the PCEP should be able to request to be enabled on a BFIR to ensure that the traffic passed by the BFIR into the BIER-TE domain does not overrun resources available. In the simplest case, such a shaper/policer could simply reflect the resources indicated by the BFIR in its request to the PCEP.

Per-flow policer/shaper or EF may need to be explicitly instantiated by BFIR/BFER. Instantiation of the Policer/Shaper on the BFIR can

happen as a function of the PCEP signaling to the BFIR, but instantiation of the EF would also require signaling of the PCEP to the BFER(s) for flows. Note that EF could also be instantiated on any midpoint BFR, so the PCEP would need to know the BIER-TE topology including where EF is considered and manage it through appropriate signaling.

Note that it is unclear yet, if EF implemenations could or should be implemented with or wihthout the need for explicit instantiation, the BIER-TE-EF-OAM document allows both options. Even in the absence of explicit signaling, per-flow Policer/Shaper and EF are limited resources and PCEP should keep track of how much of these resources are allocated and available for future flows. Like other path resources, exhaustion may require PCEP failure to allocate responses or other mitigating options.

3.1.2.2. DiffServ QoS

The only resource management that could be expected to exist in the BIER-TE domain hop-by-hop would be DiffServ QoS. As outlined in the above overprovisioning resource management model, it can serve as an easy method for lightweight resource management, and as soon as the network intends to use more than one such DiffServ codepoint across different BIER-TE flows, the PCEP should likely be able to understand and manage the DiffServ assignments of BIER-TE flows and signal the selected codepoint back to the BFIR.

3.2. BIER-TE flow model

```
BIER-TE traffic flow (change) request (from BFIR):
  Flow-control-ID: <identifier>
  Ingres BFIR of flow: (IGP router-id ?!)
  Destination-ID: set of BFER identifiers (IGP router-id ?!)
  extended-reply-required (boolean)
  Requirements:
    TSPEC (bandwidth, burst size,...)
    resilience: dual-transmission with EF
    shared-group: name

BIER-TE traffic flow reply/command (to BFIR):
  Flow-control-ID: <identifier>
  Ingres Policer/Shaper parameters (applies to each BIFT)
  Set of 1 or more BIFT:
    <SD, SI, BSL>
    BFIR-ID, entropy  (form together flow-ID)
    Bitstring
    QoS, TTL,

BIER-TE traffic flow extended reply/command (to BFIR):
  Flow-control-ID: <identifier>
  Ingres Policer/Shaper parameters (applies to each BIFT)
  Set of 1 or more BIFT:
    <SD, SI, BSL>
    BFIR-ID, entropy  (form together flow-ID)
    QoS, TTL
    List of 1 or more destinations
      Destination-ID, Bitstring

BIER-TE traffic flow command (to BFER):
  Flow-control-ID: <identifier>
  Ingres BFIR of flow: BFIR-ID (in BIER-TE packet header)
  Set of 1 or more BIFT:
    <SD, SI, BSL>
    BFIR-ID, entropy  (form together flow-ID)
    EF parameter (window size etc..)
```

Figure 4: Flow request/reply/commands

The above picture shows an initial abstract representation of the data models for the different type of request/replies discussed in the previous section between PCEC and BFIR (and in one case BFER).

The Flow-conrol-ID identifies the managed object itself: a flow to be sent from one BFIR to a set of BFER with some TE requirements, which ultimately may require BIER-TE packets for one or more BIFT.

BFIR and BFER need to be identified in the request in a form not specific to the bits of BIFT, so the PCEP can select the appropriate BIFT(s) to use. The above picture assumes the router-id of BFIR and BFER are appropriate.

The request includes TE requirements, including (something like a) TSPEC for bandwidth, burst-size or the like, whether or not dual-transmsision via PREF is required, and if the resource used are to be shared across multiple flows, then the name of a shared group. One example of sharing would for example be a video-conference where the speaker transmits video, every speaker requests/allocates a BIER-TE flow from the PCEP, but the resources for those flows are of course shared (only one flow active at a time).

The reply from the PCEP lists the BIFTS/packets that must be sent by the BFIR to reach the desired destinations as well as any other BIER-TE packet header fields relevant <SD,SI,BSL>, BFIR-ID, entropy, QoS, TTL. Beside the BIER-TE packet header, the parameters for the policer and/or shaper to be used by the BFIR are signalled back.

The extended reply does not provide simply the bitstring to use for each BIFT, but instead lists the bitstrings required for each destination so that (as described above), the BFIR can simply add/ delete destinations on a packet-by-packet basis OR’ing those bitstrings.

Finally, a command to BFER is required to instruct the creation of EF state in case this can not be done automatically.

4. Security Considerations

TBD.

5. IANA Considerations

This document requests no action by IANA.

6. Acknowledgements

TBD.

7. Change log [RFC Editor: Please remove]

00: Initial version.

8. References

[I-D.eckert-bier-te-frr]
Eckert, T., Cauchie, G., Braun, W., and M. Menth, "Protection Methods for BIER-TE", draft-eckert-bier-te-frr-02 (work in progress), June 2017.

[I-D.huang-bier-te-encapsulation]
Huang, R., Eckert, T., Wei, N., and P. Thubert, "Encapsulation for BIER-TE", draft-huang-bier-te-encapsulation-00 (work in progress), March 2018.

[I-D.ietf-bier-bier-yang]
Chen, R., hu, f., Zhang, Z., dai.xianxian@zte.com.cn, d., and M. Sivakumar, "YANG Data Model for BIER Protocol", draft-ietf-bier-bier-yang-03 (work in progress), February 2018.

[I-D.ietf-bier-isis-extensions]
Ginsberg, L., Przygienda, T., Aldrin, S., and Z. Zhang, "BIER support via ISIS", draft-ietf-bier-isis-extensions-09 (work in progress), February 2018.

[I-D.ietf-bier-ospf-bier-extensions]
Psenak, P., Kumar, N., Wijnands, I., Dolganow, A., Przygienda, T., Zhang, Z., and S. Aldrin, "OSPF Extensions for BIER", draft-ietf-bier-ospf-bier-extensions-15 (work in progress), February 2018.

[I-D.ietf-bier-te-arch]
Eckert, T., Cauchie, G., Braun, W., and M. Menth, "Traffic Engineering for Bit Index Explicit Replication (BIER-TE)", draft-ietf-bier-te-arch-00 (work in progress), January 2018.

[I-D.thubert-bier-replication-elimination]
Thubert, P., Eckert, T., Brodard, Z., and H. Jiang, "BIER-TE extensions for Packet Replication and Elimination Function (PREF) and OAM", draft-thubert-bier-replication-elimination-03 (work in progress), March 2018.

[RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Przygienda, T., and S. Aldrin, "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8296] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A., Tantsura, J., Aldrin, S., and I. Meilik, "Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks", RFC 8296, DOI 10.17487/RFC8296, January 2018, <https://www.rfc-editor.org/info/rfc8296>.

Author's Address

Toerless Eckert
Futurewei Technologies Inc.
2330 Central Expy
Santa Clara 95050
USA

Email: tte+ietf@cs.fau.de

PCE Working Group
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

D. Dhody
Y. Lee
Huawei Technologies
D. Ceccarelli
Ericsson

March 5, 2018

Applicability of Path Computation Element (PCE) for Abstraction and Control of TE Networks (ACTN)
draft-ietf-pce-applicability-actn-05

Abstract

Abstraction and Control of TE Networks (ACTN) refers to the set of virtual network (VN) operations needed to orchestrate, control and manage large-scale multi-domain TE networks so as to facilitate network programmability, automation, efficient resource sharing, and end-to-end virtual service aware connectivity and network function virtualization services.

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.

This document examines the applicability of PCE to the ACTN framework.

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 September 6, 2018.

Copyright Notice

Copyright (c) 2018 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

1.1. Path Computation Element (PCE)

The Path Computation Element Communication Protocol (PCEP) [RFC5440] provides mechanisms for Path Computation Elements (PCEs) [RFC4655] to perform path computations in response to Path Computation Clients (PCCs) requests.

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 motivation for PCE development.

A stateful PCE [RFC8231] is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

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

[RFC8231] also describes the active stateful PCE. The active PCE functionality allows a PCE to reroute an existing LSP or make changes to the attributes of an existing LSP, or a PCC to delegate control of specific LSPs to a new PCE.

1.1.1. Role of PCE in SDN

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

1.1.2. PCE in multi-domain and multi-layer deployments

Computing paths across large multi-domain environments require special computational components and cooperation between entities in different domains capable of complex path computation. The PCE provides an architecture and a set of functional components to address this problem space. A PCE may be used to compute end-to-end paths across multi-domain environments using a per-domain path computation technique [RFC5152]. The Backward recursive PCE based path computation (BRPC) mechanism [RFC5441] defines a PCE-based path computation procedure to compute inter-domain constrained MPLS and GMPLS TE networks. 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.

[RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs) when the domain sequence is not known. Within the Hierarchical PCE (H-PCE) architecture, the Parent PCE (P-PCE) is used to compute a multi-domain path based on the domain connectivity information. A Child PCE (C-PCE) may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its domain topology information.

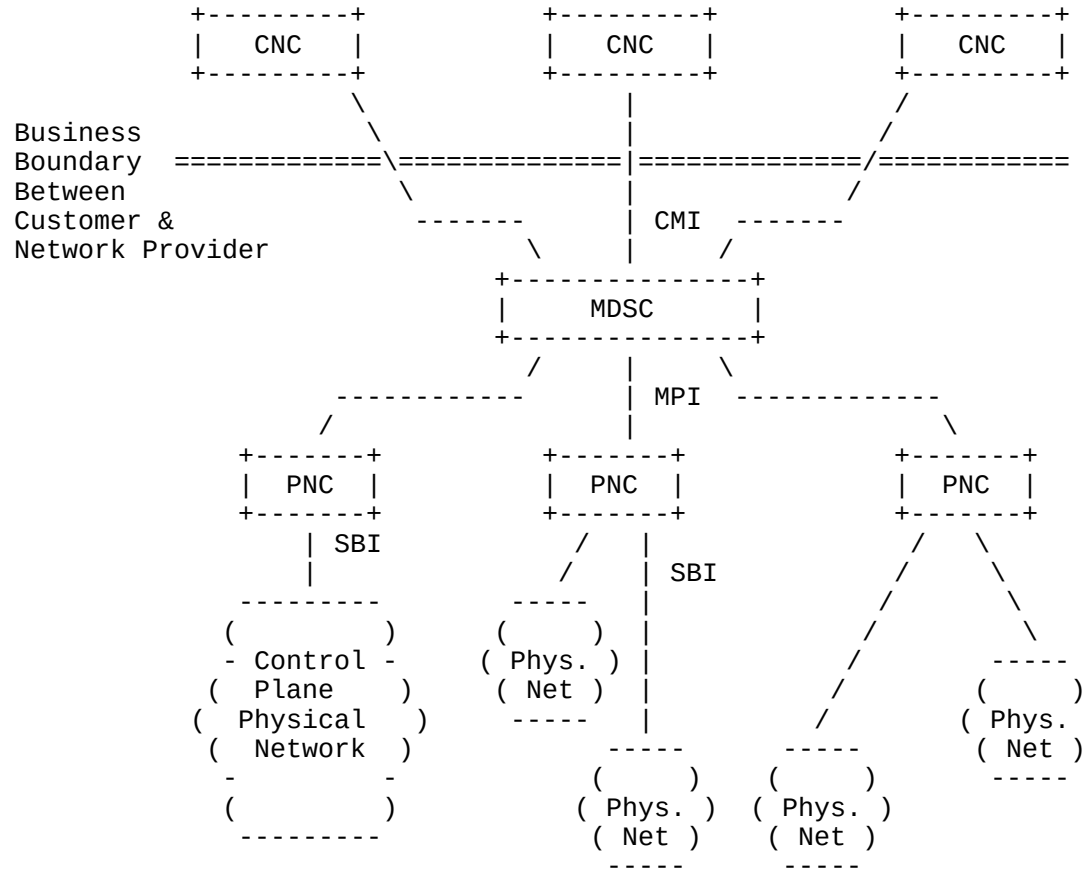
[I-D.ietf-pce-stateful-hpce] state the considerations for stateful PCE(s) in hierarchical PCE architecture. In particular, the behavior changes and additions to the existing stateful PCE mechanisms (including PCE- initiated LSP setup and active PCE usage) in the context of networks using the H-PCE architecture.

[RFC5623] describes a framework for applying the PCE-based architecture to inter-layer to (G)MPLS TE. It provides suggestions for the deployment of PCE in support of multi-layer networks. It also describes the relationship between PCE and a functional component in charge of the control and management of the VNT, called the Virtual Network Topology Manager (VNTM).

1.2. Abstraction and Control of TE Networks (ACTN)

[I-D.ietf-teas-actn-requirements] describes the high-level ACTN requirements. [I-D.ietf-teas-actn-framework] describes the architecture model for ACTN including the entities (Customer Network Controller(CNC), Multi-domain Service Coordinator(MDSC), and Provisioning Network Controller (PNC) and their interfaces.

The ACTN reference architecture identified a three-tier control hierarchy as depicted in Figure 1:



CMI - (CNC-MDSC Interface)
MPI - (MDSC-PNC Interface)

Figure 1: ACTN Hierarchy

The two interfaces with respect to the MDSC, one north of the MDSC (CMI CNC-MDSC Interface) and one south (MPI MDSC-PNC Interface). A hierarchy of MDSC is possible with a recursive MPI interface.

[I-D.ietf-teas-actn-info-model] provides an information model for ACTN interfaces.

1.3. PCE and ACTN

This document examines the PCE and ACTN architecture and describes how the PCE architecture is applicable to ACTN. It also lists the PCEP extensions that are needed to use PCEP as an ACTN interface. This document also identifies any gaps in PCEP, that exist at the time of publication of this document.

2. Architectural Considerations

ACTN [I-D.ietf-teas-actn-framework] architecture is based on hierarchy and recursiveness of controllers. It defines three types of controllers (depending on the functionalities they implement). The main functionalities are -

- Multi domain coordination function
- Virtualization/Abstraction function
- Customer mapping/translation function
- Virtual service coordination function

Section 3 of [I-D.ietf-teas-actn-framework] describes these functions.

It should be noted that, this document lists all possible ways in which PCEP could be used for each of the above functions, but all functions are not required to be implemented via PCEP. Operator may choose to use the PCEP for multi domain coordination via stateful H-PCE but use RESTCONF [RFC8040] or BGP-LS [RFC7752] to get the topology and support virtualization/abstraction function.

2.1. Multi domain coordination via Hierarchy

With the definition of domain being "everything that is under the control of the single logical controller", as per [I-D.ietf-teas-actn-framework], it is needed to have a control entity that oversees the specific aspects of the different domains and to build a single abstracted end-to-end network topology in order to coordinate end-to-end path computation and path/service provisioning.

The MDSC in ACTN framework realizes this function by coordinating the per-domain PNCs in a hierarchy of controllers. It also needs to detach from the underlying network technology and express customer concerns by business needs.

[RFC6805] and [I-D.ietf-pce-stateful-hpce] describes a hierarchy of PCE with Parent PCE coordinating multi-domain path computation function between Child PCE(s). It is easy to see how these principles align, and thus how stateful H-PCE architecture can be used to realize ACTN.

The Per domain stitched LSP in the Hierarchical stateful PCE architecture, described in Section 3.3.1 of [I-D.ietf-pce-stateful-hpce] is well suited for multi-domain coordination function. This includes domain sequence selection; E2E path computation; Controller (PCE) initiated path setup and reporting. This is also applicable to multi-layer coordination in case of IP+optical networks.

[I-D.litkowski-pce-state-sync]" describes the procedures to allow a stateful communication between PCEs for various use-cases. The procedures and extensions are also applicable to Child and Parent PCE communication and thus useful for ACTN as well.

2.2. Virtualization/Abstraction function

To realize ACTN, an abstracted view of the underlying network resources needs to be built. This includes global network-wide abstracted topology based on the underlying network resources of each domain. This also include abstract topology created as per the customer service connectivity requests and represented as a network slice allocated to each customer.

In order to compute and provide optimal paths, PCEs require an accurate and timely Traffic Engineering Database (TED). Traditionally this TED has been obtained from a link state (LS) routing protocol supporting traffic engineering extensions. PCE may construct its TED by participating in the IGP ([RFC3630] and [RFC5305] for MPLS-TE; [RFC4203] and [RFC5307] for GMPLS). An alternative is offered by BGP-LS [RFC7752].

In case of H-PCE [RFC6805], the parent PCE needs to build the domain topology map of the child domains and their interconnectivity. [RFC6805] and [I-D.ietf-pce-inter-area-as-applicability] suggest that BGP-LS could be used as a "northbound" TE advertisement from the child PCE to the parent PCE.

[I-D.dhodylee-pce-pcep-ls] proposes another approaches for learning and maintaining the Link-State and TE information as an alternative to IGPs and BGP flooding, using PCEP itself. The child PCE can use this mechanism to transport Link-State and TE information from child PCE to a Parent PCE using PCEP.

In ACTN, there is a need to control the level of abstraction based on the deployment scenario and business relationship between the controllers. The mechanism used to disseminate information from PNC (child PCE) to MDSC (parent PCE) should support abstraction. [I-D.lee-teas-actn-abstraction] describes a few alternative approaches of abstraction. The resulting abstracted topology can be encoded using the PCEP-LS mechanisms [I-D.dhodylee-pce-pcep-ls] and its optical network extension [I-D.lee-pce-pcep-ls-optical]. PCEP-LS is an attractive option when the operator would wish to have a single control plane protocol (PCEP) to achieve ACTN functions.

[I-D.ietf-teas-actn-framework] discusses two ways to build abstract topology from an MDSC standpoint with interaction with PNCs. The primary method is called authomatic generation of abstract topology by configuration. with this method, automatic generation is based on the abstraction/summarization of the whole domain by the PNC and its advertisement on the MPI. The seconday method is called on-demand generation of supplementary topology via Path Compute Request/Reply. This method may be needed to obtain further complementary information such as potential connectivity from child PCEs in order to facilitate an end-to-end path provisioning. PCEP is well suited to support both methods.

2.3. Customer mapping function

In ACTN, there is a need to map customer virtual network (VN) requirements into network provisioning request to the PNC. That is, the customer requests/commands are mapped into network provisioning requests that can be sent to the PNC. Specifically, it provides mapping and translation of a customer’s service request into a set of parameters that are specific to a network type and technology such that network configuration process is made possible.

[RFC8281] 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. To instantiate or delete an LSP, the PCE sends the Path Computation LSP Initiate Request (PCInitiate) message to the PCC. As described in [I-D.ietf-pce-stateful-hpce], for inter-domain LSP in Hierarchical PCE architecture, the initiation operations can be carried out at the parent PCE. In which case after parent PCE finishes the E2E path computation, it can send the PCInitiate message to the child PCE, the child PCE further propagates the initiate request to the LSR. The customer request is received by the MDSC (parent PCE) and based on the business logic, global abstracted topology, network conditions and local policy, the MDSC (parent PCE) translates this into per

domain LSP initiation request that a PNC (child PCE) can understand and act on. This can be done via the PCInitiate message.

PCEP extensions for associating opaque policy between PCEP peer [I-D.ietf-pce-association-policy] can be used.

2.4. Virtual Network Operations

Virtual service coordination function in ACTN incorporates customer service-related information into the virtual network service operations in order to seamlessly operate virtual networks while meeting customer’s service requirements.

[I-D.leedhody-pce-vn-association] describes the need for associating a set of LSPs with a VN "construct" to facilitate VN operations in PCE architecture. This association allows the PCEs to identify which LSPs belong to a certain VN.

This association based on VN is useful for various optimizations at the VN level which can be applied to all the LSPs that are part of the VN slice. During path computation, the impact of a path for an LSP is compared against the paths of other LSPs in the VN. This is to make sure that the overall optimization and SLA of the VN rather than of a single LSP. Similarly, during re-optimization, advanced path computation algorithm and optimization technique can be considered for all the LSPs belonging to a VN/customer and optimize them all together.

3. Interface Considerations

As per [I-D.ietf-teas-actn-framework], to allow virtualization and multi domain coordination, the network has to provide open, programmable interfaces, in which customer applications can create, replace and modify virtual network resources and services in an interactive, flexible and dynamic fashion while having no impact on other customers. The 3 ACTN interfaces are -

- The CNC-MDSC Interface (CMI) is an interface between a Customer Network Controller and a Multi Domain Service Coordinator. It requests the creation of the network resources, topology or services for the applications. The MDSC may also report potential network topology availability if queried for current capability from the Customer Network Controller.

- The MDSC-PNC Interface (MPI) is an interface between a Multi Domain Service Coordinator and a Provisioning Network Controller. It communicates the creation request, if required, of new connectivity of bandwidth changes in the physical network, via the

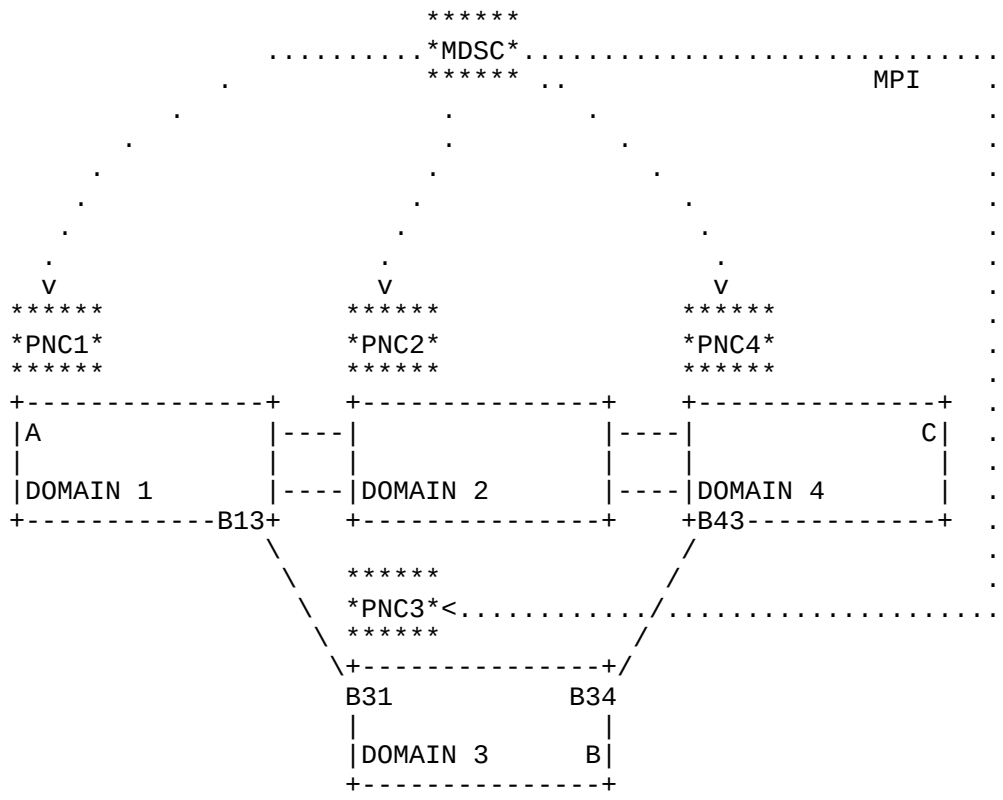
PNC. In multi-domain environments, the MDSC needs to establish multiple MPIs, one for each PNC, as there are multiple PNCs responsible for its domain control.

- In case of hierarchy of MDSC, the MPI is applied recursively. From an abstraction point of view, the top level MDSC which interfaces the CNC operates on a higher level of abstraction (i.e., less granular level) than the lower level MSDCs.

PCEP is especially suitable on the MPI as it meets the requirement and the functions as set out in the ACTN framework [I-D.ietf-teas-actn-framework]. Its recursive nature is well suited via the multi-level hierarchy of PCE. PCEP can also be applied to the CMI as the CNC can be a path computation client while the MDSC can be a path computation server. The Section 4 describe how PCE and PCEP could help realize ACTN on the MPI.

4. Realizing ACTN with PCE (and PCEP)

As per the example in the Figure 2, there are 4 domains, each with its own PNC and a MDSC at top. The PNC and MDSC need PCE as a important function. The PNC (or child PCE) already uses PCEP to communicate to the network device. It can utilize the PCEP as the MPI to communicate between controllers too.



MDSC -> Parent PCE
PNC -> Child PCE
MPI -> PCEP

Figure 2: ACTN with PCE

o Building Domain Topology at MDSC: PNC (or child PCE) needs to have the TED to compute path in its domain. As described in Section 2.2, it can learn the topology via IGP or BGP-LS. PCEP-LS is also a proposed mechanism to carry link state and traffic engineering information within PCEP. A mechanism to carry abstracted topology while hiding technology specific information between PNC and MDSC is described in [I-D.dhodylee-pce-pcep-ls]. At the end of this step the MDSC (or parent PCE) has the abstracted topology from each of its PNC (or child PCE). This could be as simple as a domain topology map as described in [RFC6805] or it can have full topology information of all domains. The latter is not scalable and thus an abstracted topology of each

domain interconnected by inter-domain links is the most common case.

* Topology Change: When the PNC learns of any topology change, the PNC needs to decide if the change needs to be notified to the MDSC. This is dependent on the level of abstraction between the MDSC and the PNC.

o VN Instantiate: MDSC is requested to instantiate a VN, the minimal information that is required would be a VN identifier and a set of end points. Various path computation, setup constraints and objective functions may also be provided. In PCE terms, a VN Instantiate can be considered as a set of paths belonging to the same VN. As described in Section 2.4 and [I-D.leedhody-pce-vn-association] the VN association can help in identifying the set of paths that belong to a VN. The rest of the information like the endpoints, constraints and objective function is already defined in PCEP in terms of a single path.

* Path Computation: As per the example in the Figure 2, the VN instantiate requires two end to end paths between (A in Domain 1 to B in Domain 3) and (A in Domain 1 to C in Domain 4). The MDSC (or parent PCE) triggers the end to end path computation for these two paths. MDSC can do path computation based on the abstracted domain topology that it already has or it may use the H-PCE procedures (Section 2.1) using the PCReq and PCRep messages to get the end to end path with the help of the child PCEs (PNC). Either way, the resulted E2E paths may be broken into per-domain paths.

* A-B: (A-B13,B13-B31,B31-B)

* A-C: (A-B13,B13-B31,B34-B43,B43-C)

* Per Domain Path Instantiation: Based on the above path computation, MDSC can issue the path instantiation request to each PNC via PCInitiate message (see [I-D.ietf-pce-stateful-hpce] and [I-D.leedhody-pce-vn-association]). A suitable stitching mechanism would be used to stitch these per domain LSPs. One such mechanism is described in [I-D.lee-pce-lsp-stitching-hpce], where PCEP is extended to support stitching in stateful H-PCE context.

* Per Domain Path Report: Each PNC should report the status of the per-domain LSP to the MDSC via PCRpt message, as per the Hierarchy of stateful PCE ([I-D.ietf-pce-stateful-hpce]). The

status of the end to end LSP (A-B and A-C) is made up when all the per domain LSP are reported up by the PNCs.

* Delegation: It is suggested that the per domain LSPs are delegated to respective PNC, so that they can control the path and attributes based on each domain network conditions.

* State Synchronization: The state needs to be synchronized between the parent PCE and child PCE. The mechanism described in [I-D.litkowski-pce-state-sync] can be used.

- VN Modify: MDSC is requested to modify a VN, for example the bandwidth for VN is increased. This may trigger path computation at MDSC as described in the previous step and can trigger an update to existing per-intra-domain path (via PCUpd message) or creation (or deletion) of a per-domain path (via PCInitiate message). As described in [I-D.ietf-pce-stateful-hpce], this should be done in make-before-break fashion.

- VN Delete: MDSC is requested to delete a VN, in this case, based on the E2E paths and the resulting per-domain paths need to be removed (via PCInitiate message).

- VN Update (based on network changes): Any change in the per-domain LSP are reported to the MDSC (via PCRpt message) as per [I-D.ietf-pce-stateful-hpce]. This may result in changes in the E2E path or VN status. This may also trigger a re-optimization leading to a new per-domain path, update to existing path, or deletion of the path.

- VN Protection: The VN protection/restoration requirements, need to applied to each E2E path as well as each per domain path. The MDSC needs to play a crucial role in coordinating the right protection/restoration policy across each PNC. The existing protection/restoration mechanism of PCEP can be applied on each path.

- In case PNC generates an abstract topology to the MDSC, the PCInitiate/PCUpd messages from the MDSC to a PNC will contain a path with abstract nodes and links. PNC would need to take that as an input for path computation to get a path with physical nodes and links. Similarly PNC would convert the path received from the device (with physical nodes and links) into abstract path (based on the abstract topology generated before with abstract nodes and links) and reported to the MDSC.

5. Relationship to PCE based central control

[RFC8283] introduces the architecture for PCE as a central controller (PCECC), it further examines the motivations and applicability for PCEP as a southbound interface, and introduces the implications for the protocol. The section 2.1.3 of [RFC8283] describe an hierarchy of PCE-based controller as per the Hierarchy of PCE framework defined in [RFC6805]. Both ACTN and PCECC is based on the same basic framework and thus compatible with each other.

6. IANA Considerations

This is an informational document and thus does not have any IANA allocations to be made.

7. Security Considerations

The ACTN framework described in [I-D.ietf-teas-actn-framework] defines key components and interfaces for managed traffic engineered networks. It also list various security considerations such as request and control of resources, confidentially of the information, and availability of function which should be taken into consideration.

When PCEP is used on the MPI, this interface needs to be secured, use of [RFC8253] is RECOMENDED. Each PCEP extension listed in this document, presents its individual security considerations, which continue to apply.

8. Acknowledgments

The authors would like to thank Jonathan Hardwick for the inspiration behind this document. Further thanks to Avantika for her comments with suggested text.

9. References

9.1. Normative References

[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>.

9.2. Informative References

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <https://www.rfc-editor.org/info/rfc3630>.

[RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, DOI 10.17487/RFC4203, October 2005, <https://www.rfc-editor.org/info/rfc4203>.

[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>.

[RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, DOI 10.17487/RFC5152, February 2008, <https://www.rfc-editor.org/info/rfc5152>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <https://www.rfc-editor.org/info/rfc5307>.

[RFC5441] Vasseur, JP., Ed., 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, DOI 10.17487/RFC5441, April 2009, <https://www.rfc-editor.org/info/rfc5441>.

[RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 5623, DOI 10.17487/RFC5623, September 2009, <https://www.rfc-editor.org/info/rfc5623>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <https://www.rfc-editor.org/info/rfc6805>.

[RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <https://www.rfc-editor.org/info/rfc7399>.

[RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <https://www.rfc-editor.org/info/rfc7491>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[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>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <https://www.rfc-editor.org/info/rfc8253>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

[RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <https://www.rfc-editor.org/info/rfc8283>.

[I-D.ietf-pce-stateful-hpce]
Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and O. Dios, "Hierarchical Stateful Path Computation Element (PCE).", draft-ietf-pce-stateful-hpce-04 (work in progress), March 2018.

[I-D.ietf-teas-actn-requirements]
Lee, Y., Ceccarelli, D., Miyasaka, T., Shin, J., and K. Lee, "Requirements for Abstraction and Control of TE Networks", draft-ietf-teas-actn-requirements-09 (work in progress), March 2018.

[I-D.ietf-teas-actn-framework]
Ceccarelli, D. and Y. Lee, "Framework for Abstraction and Control of Traffic Engineered Networks", draft-ietf-teas-actn-framework-11 (work in progress), October 2017.

[I-D.ietf-teas-actn-info-model]
Lee, Y., Belotti, S., Dhody, D., Ceccarelli, D., and B. Yoon, "Information Model for Abstraction and Control of TE Networks (ACTN)", draft-ietf-teas-actn-info-model-07 (work in progress), February 2018.

[I-D.ietf-pce-inter-area-as-applicability]
King, D., Meuric, J., Dugeon, O., Zhao, Q., Dhody, D., and O. Dios, "Applicability of the Path Computation Element to Inter-Area and Inter-AS MPLS and GMPLS Traffic Engineering", draft-ietf-pce-inter-area-as-applicability-06 (work in progress), July 2016.

[I-D.dhodylee-pce-pcep-ls]
Dhody, D., Lee, Y., and D. Ceccarelli, "PCEP Extension for Distribution of Link-State and TE Information.", draft-dhodylee-pce-pcep-ls-10 (work in progress), March 2018.

[I-D.lee-pce-pcep-ls-optical]
Lee, Y., zhenghaomian@huawei.com, z., Ceccarelli, D., weiw@bupt.edu.cn, w., Park, P., and B. Yoon, "PCEP Extension for Distribution of Link-State and TE information for Optical Networks", draft-lee-pce-pcep-ls-optical-04 (work in progress), February 2018.

[I-D.leedhody-pce-vn-association]
Lee, Y., Dhody, D., Zhang, X., and D. Ceccarelli, "PCEP Extensions for Establishing Relationships Between Sets of LSPs and Virtual Networks", draft-leedhody-pce-vn-association-04 (work in progress), February 2018.

[I-D.litkowski-pce-state-sync]
Litkowski, S., Sivabalan, S., and D. Dhody, "Inter Stateful Path Computation Element communication procedures", draft-litkowski-pce-state-sync-02 (work in progress), August 2017.

[I-D.ietf-pce-association-policy]
Dhody, D., Sivabalan, S., Litkowski, S., Tantsura, J., and J. Hardwick, "Path Computation Element communication Protocol extension for associating Policies and LSPs", draft-ietf-pce-association-policy-02 (work in progress), February 2018.

[I-D.lee-teas-actn-abstraction]
Lee, Y., Dhody, D., Ceccarelli, D., and O. Dios, "Abstraction and Control of TE Networks (ACTN) Abstraction Methods", draft-lee-teas-actn-abstraction-02 (work in progress), June 2017.

[I-D.lee-pce-lsp-stitching-hpce]
Lee, Y., Dhody, D., and D. Ceccarelli, "PCEP Extensions for Stitching LSPs in Hierarchical Stateful PCE Model", draft-lee-pce-lsp-stitching-hpce-01 (work in progress), December 2017.

Authors' Addresses

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

EMail: dhruv.ietf@gmail.com

Young Lee
Huawei Technologies
5340 Legacy Drive, Building 3
Plano, TX 75023
USA

EMail: leeyoung@huawei.com

Daniele Ceccarelli
Ericsson
Torshamnsgatan,48
Stockholm
Sweden

EMail: daniele.ceccarelli@ericsson.com

PCE Working Group
Internet-Draft
Intended status: Informational
Expires: December 19, 2018

D. Dhody
Y. Lee
Huawei Technologies
D. Ceccarelli
Ericsson

June 17, 2018

Applicability of Path Computation Element (PCE) for Abstraction and Control of TE Networks (ACTN) draft-ietf-pce-applicability-actn-06

Abstract

Abstraction and Control of TE Networks (ACTN) refers to the set of virtual network (VN) operations needed to orchestrate, control and manage large-scale multi-domain TE networks so as to facilitate network programmability, automation, efficient resource sharing, and end-to-end virtual service aware connectivity and network function virtualization services.

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.

This document examines the applicability of PCE to the ACTN framework.

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 December 19, 2018.

Copyright Notice

Copyright (c) 2018 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

1.1. Path Computation Element (PCE)

The Path Computation Element Communication Protocol (PCEP) [RFC5440] provides mechanisms for Path Computation Elements (PCEs) [RFC4655] to perform path computations in response to Path Computation Clients (PCCs) requests.

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 motivation for PCE development.

A stateful PCE [RFC8231] is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

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

[RFC8231] also describes the active stateful PCE. The active PCE functionality allows a PCE to reroute an existing LSP or make changes to the attributes of an existing LSP, or a PCC to delegate control of specific LSPs to a new PCE.

1.1.1. Role of PCE in SDN

Software-Defined Networking (SDN) [RFC7149] refers to a separation between the control elements and the forwarding components so that software running in a centralized system called a controller, can act to program the devices in the network to behave in specific ways. A required element in an SDN architecture is a component that plans how the network resources will be used and how the devices will be programmed. It is possible to view this component as performing specific computations to place flows within the network given knowledge of the availability of network resources, how other forwarding devices are programmed, and the way that other flows are routed. It is concluded in [RFC7399], that this is the same function that a PCE might offer in a network operated using a dynamic control plane. This is the function and purpose of a PCE, and the way that a PCE integrates into a wider network control system including SDN is presented in Application-Based Network Operation (ABNO) [RFC7491].

1.1.2. PCE in multi-domain and multi-layer deployments

Computing paths across large multi-domain environments require special computational components and cooperation between entities in different domains capable of complex path computation. The PCE provides an architecture and a set of functional components to address this problem space. A PCE may be used to compute end-to-end paths across multi-domain environments using a per-domain path computation technique [RFC5152]. The Backward recursive PCE based path computation (BRPC) mechanism [RFC5441] defines a PCE-based path computation procedure to compute inter-domain constrained MPLS and GMPLS TE networks. 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.

[RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs) when the domain sequence is not known. Within the Hierarchical PCE (H-PCE) architecture, the Parent PCE (P-PCE) is used to compute a multi-domain path based on the domain connectivity information. A Child PCE (C-PCE) may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its domain topology information.

[I-D.ietf-pce-stateful-hpce] state the considerations for stateful PCE(s) in hierarchical PCE architecture. In particular, the behavior changes and additions to the existing stateful PCE mechanisms (including PCE- initiated LSP setup and active PCE usage) in the context of networks using the H-PCE architecture.

[RFC5623] describes a framework for applying the PCE-based architecture to inter-layer to (G)MPLS TE. It provides suggestions for the deployment of PCE in support of multi-layer networks. It also describes the relationship between PCE and a functional component in charge of the control and management of the Virtual Network Topology (VNT) [RFC5212], called the VNT Manager (VNTM).

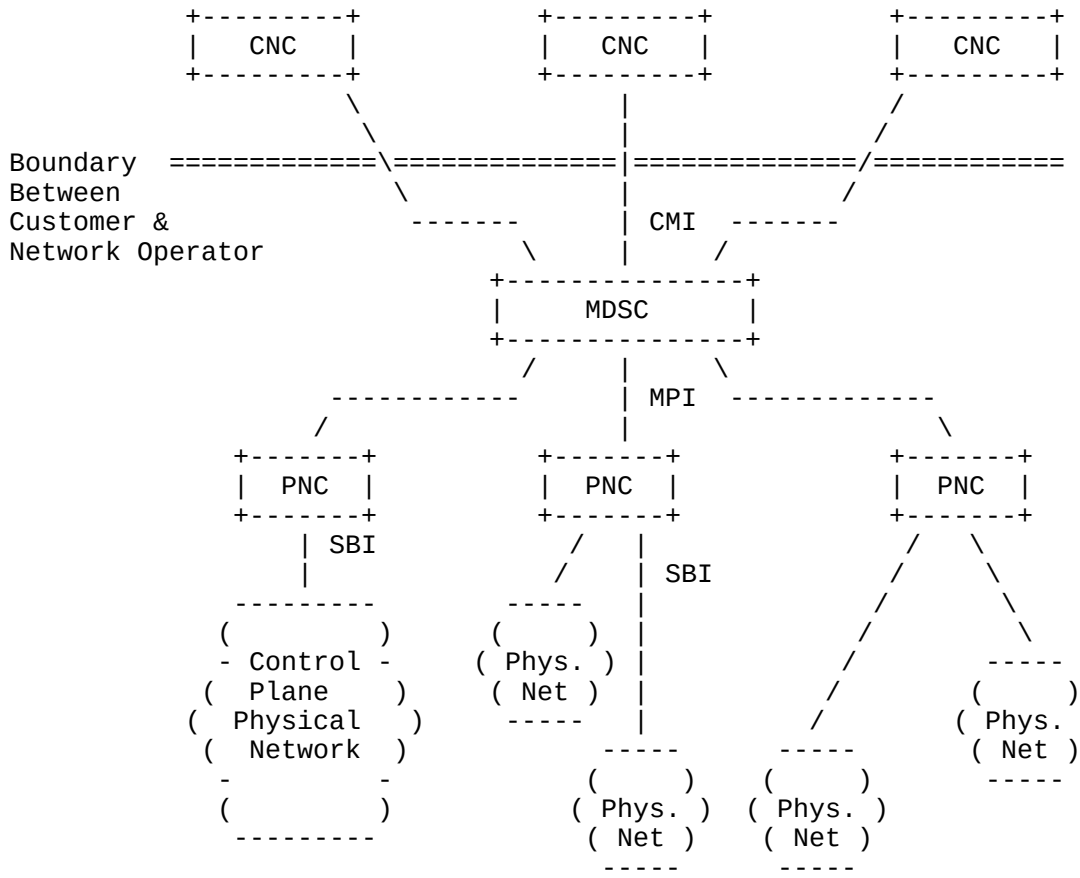
1.1.3. Relationship to PCE based central control

[RFC8283] introduces the architecture for PCE as a central controller (PCECC), it further examines the motivations and applicability for PCEP as a southbound interface, and introduces the implications for the protocol. The section 2.1.3 of [RFC8283] describe an hierarchy of PCE-based controller as per the Hierarchy of PCE framework defined in [RFC6805].

1.2. Abstraction and Control of TE Networks (ACTN)

[I-D.ietf-teas-actn-requirements] describes the high-level ACTN requirements. [I-D.ietf-teas-actn-framework] describes the architecture model for ACTN including the entities (Customer Network Controller(CNC), Multi-domain Service Coordinator(MDSC), and Provisioning Network Controller (PNC) and their interfaces.

The ACTN reference architecture identified a three-tier control hierarchy as depicted in Figure 1:



CMI - (CNC-MDSC Interface)
MPI - (MDSC-PNC Interface)

Figure 1: ACTN Hierarchy

The two interfaces with respect to the MDSC, one north of the MDSC (CMI CNC-MDSC Interface) and one south (MPI MDSC-PNC Interface). A hierarchy of MDSC is possible with a recursive MPI interface.

[I-D.ietf-teas-actn-info-model] provides an information model for ACTN interfaces.

1.3. PCE and ACTN

This document examines the PCE and ACTN architecture and describes how the PCE architecture is applicable to ACTN. It also lists the PCEP extensions that are needed to use PCEP as an ACTN interface. This document also identifies any gaps in PCEP, that exist at the time of publication of this document.

Further, ACTN, Stateful H-PCE, and PCECC are based on the same basic hierarchy framework and thus compatible with each other.

2. Architectural Considerations

ACTN [I-D.ietf-teas-actn-framework] architecture is based on hierarchy and recursiveness of controllers. It defines three types of controllers (depending on the functionalities they implement). The main functionalities are -

- Multi domain coordination function
- Abstraction function
- Customer mapping/translation function
- Virtual service coordination function

Section 3 of [I-D.ietf-teas-actn-framework] describes these functions.

It should be noted that, this document lists all possible ways in which PCEP could be used for each of the above functions, but all functions are not required to be implemented via PCEP. Operator may choose to use the PCEP for multi domain coordination via stateful H-PCE but use RESTCONF [RFC8040] or BGP-LS [RFC7752] to get the topology and support abstraction function.

2.1. Multi domain coordination via Hierarchy

With the definition of domain being "everything that is under the control of the single logical controller", as per [I-D.ietf-teas-actn-framework], it is needed to have a control entity that oversees the specific aspects of the different domains and to build a single abstracted end-to-end network topology in order to coordinate end-to-end path computation and path/service provisioning.

The MDSC in ACTN framework realizes this function by coordinating the per-domain PNCs in a hierarchy of controllers. It also needs to

detach from the underlying network technology and express customer concerns by business needs.

[RFC6805] and [I-D.ietf-pce-stateful-hpce] describes a hierarchy of PCE with Parent PCE coordinating multi-domain path computation function between Child PCE(s). It is easy to see how these principles align, and thus how stateful H-PCE architecture can be used to realize ACTN.

The Per domain stitched LSP in the Hierarchical stateful PCE architecture, described in Section 3.3.1 of [I-D.ietf-pce-stateful-hpce] is well suited for multi-domain coordination function. This includes domain sequence selection; E2E path computation; Controller (PCE) initiated path setup and reporting. This is also applicable to multi-layer coordination in case of IP+optical networks.

[I-D.litkowski-pce-state-sync]" describes the procedures to allow a stateful communication between PCEs for various use-cases. The procedures and extensions are also applicable to Child and Parent PCE communication and thus useful for ACTN as well.

2.2. Abstraction function

To realize ACTN, an abstracted view of the underlying network resources needs to be built. This includes global network-wide abstracted topology based on the underlying network resources of each domain. This also include abstract topology created as per the customer service connectivity requests and represented as a network slice allocated to each customer.

In order to compute and provide optimal paths, PCEs require an accurate and timely Traffic Engineering Database (TED). Traditionally this TED has been obtained from a link state (LS) routing protocol supporting traffic engineering extensions. PCE may construct its TED by participating in the IGP ([RFC3630] and [RFC5305] for MPLS-TE; [RFC4203] and [RFC5307] for GMPLS). An alternative is offered by BGP-LS [RFC7752].

In case of H-PCE [RFC6805], the parent PCE needs to build the domain topology map of the child domains and their interconnectivity. [RFC6805] and [I-D.ietf-pce-inter-area-as-applicability] suggest that BGP-LS could be used as a "northbound" TE advertisement from the child PCE to the parent PCE.

[I-D.dhodylee-pce-pcep-ls] proposes another approaches for learning and maintaining the Link-State and TE information as an alternative to IGPs and BGP flooding, using PCEP itself. The child PCE can use

this mechanism to transport Link-State and TE information from child PCE to a Parent PCE using PCEP.

In ACTN, there is a need to control the level of abstraction based on the deployment scenario and business relationship between the controllers. The mechanism used to disseminate information from PNC (child PCE) to MDSC (parent PCE) should support abstraction. [I-D.ietf-teas-actn-framework] describes a few alternative approaches of abstraction. The resulting abstracted topology can be encoded using the PCEP-LS mechanisms [I-D.dhodylee-pce-pcep-ls] and its optical network extension [I-D.lee-pce-pcep-ls-optical]. PCEP-LS is an attractive option when the operator would wish to have a single control plane protocol (PCEP) to achieve ACTN functions.

[I-D.ietf-teas-actn-framework] discusses two ways to build abstract topology from an MDSC standpoint with interaction with PNCs. The primary method is called automatic generation of abstract topology by configuration. With this method, automatic generation is based on the abstraction/summarization of the whole domain by the PNC and its advertisement on the MPI. The secondary method is called on-demand generation of supplementary topology via Path Compute Request/Reply. This method may be needed to obtain further complementary information such as potential connectivity from child PCEs in order to facilitate an end-to-end path provisioning. PCEP is well suited to support both methods.

2.3. Customer mapping function

In ACTN, there is a need to map customer virtual network (VN) requirements into network provisioning request to the PNC. That is, the customer requests/commands are mapped into network provisioning requests that can be sent to the PNC. Specifically, it provides mapping and translation of a customer’s service request into a set of parameters that are specific to a network type and technology such that network configuration process is made possible.

[RFC8281] 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. To instantiate or delete an LSP, the PCE sends the Path Computation LSP Initiate Request (PCInitiate) message to the PCC. As described in [I-D.ietf-pce-stateful-hpce], for inter-domain LSP in Hierarchical PCE architecture, the initiation operations can be carried out at the parent PCE. In which case after parent PCE finishes the E2E path computation, it can send the PCInitiate message to the child PCE, the child PCE further propagates the initiate request to the LSR. The customer request is received by the MDSC (parent PCE) and based on

the business logic, global abstracted topology, network conditions and local policy, the MDSC (parent PCE) translates this into per domain LSP initiation request that a PNC (child PCE) can understand and act on. This can be done via the PCInitiate message.

PCEP extensions for associating opaque policy between PCEP peer [I-D.ietf-pce-association-policy] can be used.

2.4. Virtual Service Coordination

Virtual service coordination function in ACTN incorporates customer service-related information into the virtual network service operations in order to seamlessly operate virtual networks while meeting customer’s service requirements.

[I-D.leedhody-pce-vn-association] describes the need for associating a set of LSPs with a VN "construct" to facilitate VN operations in PCE architecture. This association allows the PCEs to identify which LSPs belong to a certain VN.

This association based on VN is useful for various optimizations at the VN level which can be applied to all the LSPs that are part of the VN slice. During path computation, the impact of a path for an LSP is compared against the paths of other LSPs in the VN. This is to make sure that the overall optimization and SLA of the VN rather than of a single LSP. Similarly, during re-optimization, advanced path computation algorithm and optimization technique can be considered for all the LSPs belonging to a VN/customer and optimize them all together.

3. Interface Considerations

As per [I-D.ietf-teas-actn-framework], to allow virtualization and multi domain coordination, the network has to provide open, programmable interfaces, in which customer applications can create, replace and modify virtual network resources and services in an interactive, flexible and dynamic fashion while having no impact on other customers. The two ACTN interfaces are -

- The CNC-MDSC Interface (CMI) is an interface between a Customer Network Controller and a Multi Domain Service Coordinator. It requests the creation of the network resources, topology or services for the applications. The MDSC may also report potential network topology availability if queried for current capability from the Customer Network Controller.

- The MDSC-PNC Interface (MPI) is an interface between a Multi Domain Service Coordinator and a Provisioning Network Controller.

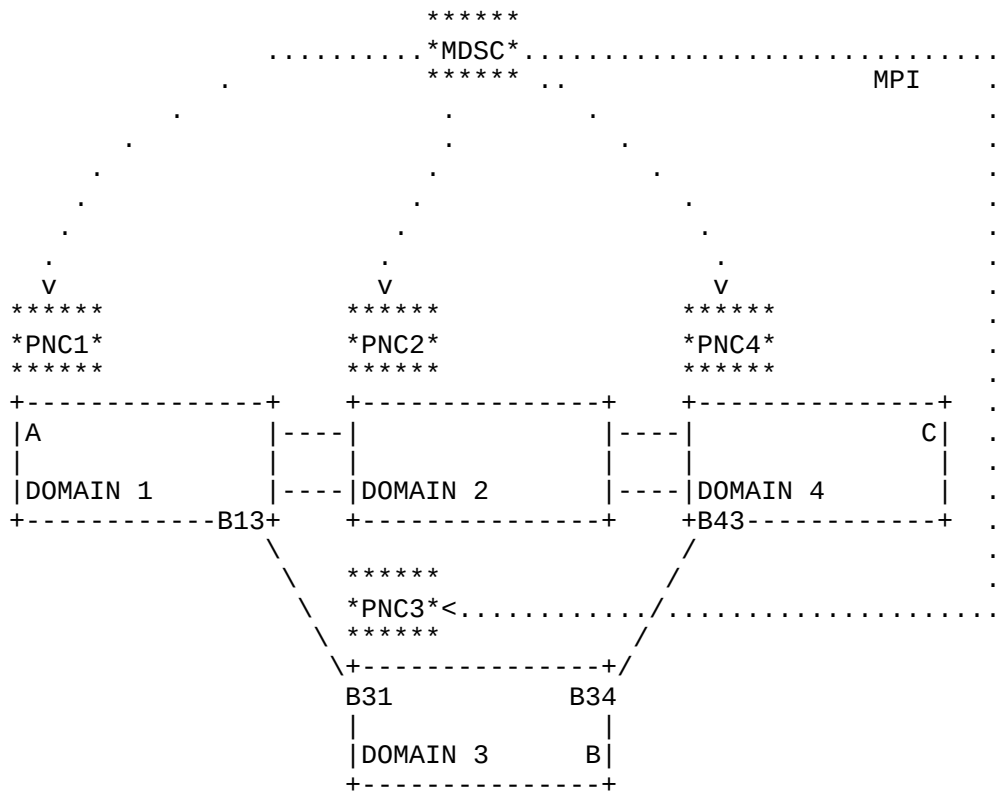
It communicates the creation request, if required, of new connectivity of bandwidth changes in the physical network, via the PNC. In multi-domain environments, the MDSC needs to establish multiple MPIs, one for each PNC, as there are multiple PNCs responsible for its domain control.

- In case of hierarchy of MDSC, the MPI is applied recursively. From an abstraction point of view, the top level MDSC which interfaces the CNC operates on a higher level of abstraction (i.e., less granular level) than the lower level MSDCs.

PCEP is especially suitable on the MPI as it meets the requirement and the functions as set out in the ACTN framework [I-D.ietf-teas-actn-framework]. Its recursive nature is well suited via the multi-level hierarchy of PCE. PCEP can also be applied to the CMI as the CNC can be a path computation client while the MDSC can be a path computation server. The Section 4 describe how PCE and PCEP could help realize ACTN on the MPI.

4. Realizing ACTN with PCE (and PCEP)

As per the example in the Figure 2, there are 4 domains, each with its own PNC and a MDSC at top. The PNC and MDSC need PCE as a important function. The PNC (or child PCE) already uses PCEP to communicate to the network device. It can utilize the PCEP as the MPI to communicate between controllers too.



MDSC -> Parent PCE
PNC -> Child PCE
MPI -> PCEP

Figure 2: ACTN with PCE

o Building Domain Topology at MDSC: PNC (or child PCE) needs to have the TED to compute path in its domain. As described in Section 2.2, it can learn the topology via IGP or BGP-LS. PCEP-LS is also a proposed mechanism to carry link state and traffic engineering information within PCEP. A mechanism to carry abstracted topology while hiding technology specific information between PNC and MDSC is described in [I-D.dhodylee-pce-pcep-ls]. At the end of this step the MDSC (or parent PCE) has the abstracted topology from each of its PNC (or child PCE). This could be as simple as a domain topology map as described in [RFC6805] or it can have full topology information of all domains. The latter is not scalable and thus an abstracted topology of each

domain interconnected by inter-domain links is the most common case.

* Topology Change: When the PNC learns of any topology change, the PNC needs to decide if the change needs to be notified to the MDSC. This is dependent on the level of abstraction between the MDSC and the PNC.

o VN Instantiate: MDSC is requested to instantiate a VN, the minimal information that is required would be a VN identifier and a set of end points. Various path computation, setup constraints and objective functions may also be provided. In PCE terms, a VN Instantiate can be considered as a set of paths belonging to the same VN. As described in Section 2.4 and [I-D.leedhody-pce-vn-association] the VN association can help in identifying the set of paths that belong to a VN. The rest of the information like the endpoints, constraints and objective function (OF) is already defined in PCEP in terms of a single path.

* Path Computation: As per the example in the Figure 2, the VN instantiate requires two end to end paths between (A in Domain 1 to B in Domain 3) and (A in Domain 1 to C in Domain 4). The MDSC (or parent PCE) triggers the end to end path computation for these two paths. MDSC can do path computation based on the abstracted domain topology that it already has or it may use the H-PCE procedures (Section 2.1) using the PCReq and PCRep messages to get the end to end path with the help of the child PCEs (PNC). Either way, the resulted E2E paths may be broken into per-domain paths.

* A-B: (A-B13,B13-B31,B31-B)

* A-C: (A-B13,B13-B31,B34-B43,B43-C)

* Per Domain Path Instantiation: Based on the above path computation, MDSC can issue the path instantiation request to each PNC via PCInitiate message (see [I-D.ietf-pce-stateful-hpce] and [I-D.leedhody-pce-vn-association]). A suitable stitching mechanism would be used to stitch these per domain LSPs. One such mechanism is described in [I-D.lee-pce-lsp-stitching-hpce], where PCEP is extended to support stitching in stateful H-PCE context.

* Per Domain Path Report: Each PNC should report the status of the per-domain LSP to the MDSC via PCRpt message, as per the Hierarchy of stateful PCE ([I-D.ietf-pce-stateful-hpce]). The

status of the end to end LSP (A-B and A-C) is made up when all the per domain LSP are reported up by the PNCs.

* Delegation: It is suggested that the per domain LSPs are delegated to respective PNC, so that they can control the path and attributes based on each domain network conditions.

* State Synchronization: The state needs to be synchronized between the parent PCE and child PCE. The mechanism described in [I-D.litkowski-pce-state-sync] can be used.

- VN Modify: MDSC is requested to modify a VN, for example the bandwidth for VN is increased. This may trigger path computation at MDSC as described in the previous step and can trigger an update to existing per-intra-domain path (via PCUpd message) or creation (or deletion) of a per-domain path (via PCInitiate message). As described in [I-D.ietf-pce-stateful-hpce], this should be done in make-before-break fashion.

- VN Delete: MDSC is requested to delete a VN, in this case, based on the E2E paths and the resulting per-domain paths need to be removed (via PCInitiate message).

- VN Update (based on network changes): Any change in the per-domain LSP are reported to the MDSC (via PCRpt message) as per [I-D.ietf-pce-stateful-hpce]. This may result in changes in the E2E path or VN status. This may also trigger a re-optimization leading to a new per-domain path, update to existing path, or deletion of the path.

- VN Protection: The VN protection/restoration requirements, need to applied to each E2E path as well as each per domain path. The MDSC needs to play a crucial role in coordinating the right protection/restoration policy across each PNC. The existing protection/restoration mechanism of PCEP can be applied on each path.

- In case PNC generates an abstract topology to the MDSC, the PCInitiate/PCUpd messages from the MDSC to a PNC will contain a path with abstract nodes and links. PNC would need to take that as an input for path computation to get a path with physical nodes and links. Similarly PNC would convert the path received from the device (with physical nodes and links) into abstract path (based on the abstract topology generated before with abstract nodes and links) and reported to the MDSC.

5. IANA Considerations

This is an informational document and thus does not have any IANA allocations to be made.

6. Security Considerations

The ACTN framework described in [I-D.ietf-teas-actn-framework] defines key components and interfaces for managed traffic engineered networks. It also list various security considerations such as request and control of resources, confidentially of the information, and availability of function which should be taken into consideration.

When PCEP is used on the MPI, this interface needs to be secured, use of [RFC8253] is RECOMENDED. Each PCEP extension listed in this document, presents its individual security considerations, which continue to apply.

7. Acknowledgments

The authors would like to thank Jonathan Hardwick for the inspiration behind this document. Further thanks to Avantika for her comments with suggested text.

8. References

8.1. Normative References

[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>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <https://www.rfc-editor.org/info/rfc6805>.

[I-D.ietf-teas-actn-framework] Ceccarelli, D. and Y. Lee, "Framework for Abstraction and Control of Traffic Engineered Networks", draft-ietf-teas-actn-framework-15 (work in progress), May 2018.

8.2. Informative References

[RFC3630] Katz, D., Kompella, K., and D. Yeung, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, DOI 10.17487/RFC3630, September 2003, <https://www.rfc-editor.org/info/rfc3630>.

[RFC4203] Kompella, K., Ed. and Y. Rekhter, Ed., "OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 4203, DOI 10.17487/RFC4203, October 2005, <https://www.rfc-editor.org/info/rfc4203>.

[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>.

[RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, DOI 10.17487/RFC5152, February 2008, <https://www.rfc-editor.org/info/rfc5152>.

[RFC5212] Shiomoto, K., Papadimitriou, D., Le Roux, JL., Vigoureux, M., and D. Brungard, "Requirements for GMPLS-Based Multi-Region and Multi-Layer Networks (MRN/MLN)", RFC 5212, DOI 10.17487/RFC5212, July 2008, <https://www.rfc-editor.org/info/rfc5212>.

[RFC5305] Li, T. and H. Smit, "IS-IS Extensions for Traffic Engineering", RFC 5305, DOI 10.17487/RFC5305, October 2008, <https://www.rfc-editor.org/info/rfc5305>.

[RFC5307] Kompella, K., Ed. and Y. Rekhter, Ed., "IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)", RFC 5307, DOI 10.17487/RFC5307, October 2008, <https://www.rfc-editor.org/info/rfc5307>.

[RFC5441] Vasseur, JP., Ed., 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, DOI 10.17487/RFC5441, April 2009, <https://www.rfc-editor.org/info/rfc5441>.

[RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 5623, DOI 10.17487/RFC5623, September 2009, <https://www.rfc-editor.org/info/rfc5623>.

[RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Perspective from within a Service Provider Environment", RFC 7149, DOI 10.17487/RFC7149, March 2014, <https://www.rfc-editor.org/info/rfc7149>.

[RFC7399] Farrel, A. and D. King, "Unanswered Questions in the Path Computation Element Architecture", RFC 7399, DOI 10.17487/RFC7399, October 2014, <https://www.rfc-editor.org/info/rfc7399>.

[RFC7491] King, D. and A. Farrel, "A PCE-Based Architecture for Application-Based Network Operations", RFC 7491, DOI 10.17487/RFC7491, March 2015, <https://www.rfc-editor.org/info/rfc7491>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[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>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <https://www.rfc-editor.org/info/rfc8253>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

[RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., and C. Zhou, "An Architecture for Use of PCE and the PCE Communication Protocol (PCEP) in a Network with Central Control", RFC 8283, DOI 10.17487/RFC8283, December 2017, <https://www.rfc-editor.org/info/rfc8283>.

[I-D.ietf-pce-stateful-hpce] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and O. Dios, "Hierarchical Stateful Path Computation Element (PCE).", draft-ietf-pce-stateful-hpce-04 (work in progress), March 2018.

[I-D.ietf-teas-actn-requirements] Lee, Y., Ceccarelli, D., Miyasaka, T., Shin, J., and K. Lee, "Requirements for Abstraction and Control of TE Networks", draft-ietf-teas-actn-requirements-09 (work in progress), March 2018.

[I-D.ietf-teas-actn-info-model] Lee, Y., Belotti, S., Dhody, D., Ceccarelli, D., and B. Yoon, "Information Model for Abstraction and Control of TE Networks (ACTN)", draft-ietf-teas-actn-info-model-09 (work in progress), June 2018.

[I-D.ietf-pce-inter-area-as-applicability] King, D., Meuric, J., Dugeon, O., Zhao, Q., Dhody, D., and O. Dios, "Applicability of the Path Computation Element to Inter-Area and Inter-AS MPLS and GMPLS Traffic Engineering", draft-ietf-pce-inter-area-as-applicability-06 (work in progress), July 2016.

[I-D.dhodylee-pce-pcep-ls] Dhody, D., Lee, Y., and D. Ceccarelli, "PCEP Extension for Distribution of Link-State and TE Information.", draft-dhodylee-pce-pcep-ls-10 (work in progress), March 2018.

[I-D.lee-pce-pcep-ls-optical] Lee, Y., zhenghaomian@huawei.com, z., Ceccarelli, D., weiw@bupt.edu.cn, w., Park, P., and B. Yoon, "PCEP Extension for Distribution of Link-State and TE information for Optical Networks", draft-lee-pce-pcep-ls-optical-04 (work in progress), February 2018.

[I-D.leedhody-pce-vn-association]
Lee, Y., Dhody, D., Zhang, X., and D. Ceccarelli, "PCEP Extensions for Establishing Relationships Between Sets of LSPs and Virtual Networks", draft-leedhody-pce-vn-association-04 (work in progress), February 2018.

[I-D.litkowski-pce-state-sync]
Litkowski, S., Sivabalan, S., and D. Dhody, "Inter Stateful Path Computation Element communication procedures", draft-litkowski-pce-state-sync-03 (work in progress), April 2018.

[I-D.ietf-pce-association-policy]
Dhody, D., Sivabalan, S., Litkowski, S., Tantsura, J., and J. Hardwick, "Path Computation Element communication Protocol extension for associating Policies and LSPs", draft-ietf-pce-association-policy-02 (work in progress), February 2018.

[I-D.lee-pce-lsp-stitching-hpce]
Lee, Y., Dhody, D., and D. Ceccarelli, "PCEP Extensions for Stitching LSPs in Hierarchical Stateful PCE Model", draft-lee-pce-lsp-stitching-hpce-01 (work in progress), December 2017.

Authors' Addresses

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

EMail: dhruv.ietf@gmail.com

Young Lee
Huawei Technologies
5340 Legacy Drive, Building 3
Plano, TX 75023
USA

EMail: leeyoung@huawei.com

Daniele Ceccarelli
Ericsson
Torshamnsgatan,48
Stockholm
Sweden

EMail: daniele.ceccarelli@ericsson.com

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 6, 2018

F. Zhang
Q. Zhao
Huawei
O. Gonzalez de Dios
Telefonica I+D
R. Casellas
CTTC
D. King
Old Dog Consulting
March 5, 2018

Extensions to Path Computation Element Communication Protocol (PCEP) for Hierarchical Path Computation Elements (PCE)
draft-ietf-pce-hierarchy-extensions-04

Abstract

The Hierarchical Path Computation Element (H-PCE) architecture RFC 6805, provides a mechanism to allow the optimum sequence of domains to be selected, and the optimum end-to-end path to be derived through the use of a hierarchical relationship between domains.

This document defines the Path Computation Element Protocol (PCEP) extensions for the purpose of implementing necessary Hierarchical PCE procedures and protocol extensions.

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 September 6, 2018.

Copyright Notice

Copyright (c) 2018 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

[RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs).

Within the hierarchical PCE architecture, the parent PCE is used to compute a multi-domain path based on the domain connectivity information . A child PCE may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its own domain topology information.

The H-PCE end-to-end domain path computation procedure is described below:

- A path computation client (PCC) sends the inter-domain path computation requests to the child PCE responsible for its domain;
- The child PCE forwards the request to the parent PCE;
- The parent PCE computes the likely domain paths from the ingress domain to the egress domain;
- The parent PCE sends the intra-domain path computation requests (between the domain border nodes) to the child PCEs which are responsible for the domains along the domain path;
- The child PCEs return the intra-domain paths to the parent PCE;
- The parent PCE constructs the end-to-end inter-domain path based on the intra-domain paths;
- The parent PCE returns the inter-domain path to the child PCE;
- The child PCE forwards the inter-domain path to the PCC.

In addition, the parent PCE may be requested to provide only the sequence of domains to a child PCE so that alternative inter-domain path computation procedures, including Per Domain (PD) [RFC5152] and Backwards Recursive Path Computation (BRPC) [RFC5441] may be used.

This document defines the PCEP extensions for the purpose of implementing Hierarchical PCE procedures, which are described in [RFC6805].

1.1. Scope

The following functions are out of scope of this document.

- Determination of Destination Domain (section 4.5 of [RFC6805])
  - via collection of reachability information from child domain;
  - via requests to the child PCEs to discover if they contain the destination node;
  - or any other methods.
- Parent Traffic Engineering Database (TED) methods (section 4.4 of [RFC6805])
- Learning of Domain connectivity and boundary nodes (BN) addresses.

1.2. Terminology

This document uses the terminology defined in [RFC4655], [RFC5440] and the additional terms defined in section 1.4 of [RFC6805].

1.3. Requirements Language

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

2. Requirements for H-PCE

This section compiles the set of requirements of the PCEP protocol to support the H-PCE architecture and procedures.

[RFC6805] identifies high-level requirements of PCEP extensions required to support the hierarchical PCE model.

2.1. Path Computation Request

The Path Computation Request (PCReq) messages are used by a PCC or PCE to make a path computation request to a PCE. In order to achieve the full functionality of the H-PCE procedures, the PCReq message needs to include:

- Qualification of PCE Requests;
- Multi-domain Objective Functions (OF);
- Multi-domain Metrics.

2.1.1. Qualification of PCEP Requests

As described in section 4.8.1 of [RFC6805], the H-PCE architecture introduces new request qualifications, which are:

- It MUST be possible for a child PCE to indicate that a request it sends to a parent PCE should be satisfied by a domain sequence only, that is, not by a full end-to-end path. This allows the child PCE to initiate a per-domain (PD) [RFC5152] or a backward recursive path computation (BRPC) [RFC5441].
- As stated in [RFC6805], section 4.5, if a PCC knows the egress domain, it can supply this information as the path computation

request. It SHOULD be possible to specify the destination domain information in a PCEP request, if it is known.

- It MAY be possible to indicate that the inter domain path computed by parent PCE should disallow domain re-entry.

2.1.2. Multi-domain Objective Functions

For inter-domain path computation, there is one new objective Function which is defined in section 1.3.1 and 4.1 of [RFC6805]:

- Minimize the number of domains crossed. A domain can be either an Autonomous System (AS) or an Internal Gateway Protocol (IGP) area depending on the type of multi-domain network hierarchical PCE is applied to.

Another objective Function to minimize the number of border nodes is also defined in this document.

During the PCEP session establishment procedure, the parent PCE needs to be capable of indicating the Objective Functions (OF) [RFC5541] capability in the Open message. This capability information may then be announced by child PCEs, and used for selecting the PCE when a PCC wants a path that satisfies one or multiple inter-domain objective functions.

When a PCC requests a PCE to compute an inter-domain path, the PCC needs also to be capable of indicating the new objective functions for inter-domain path. Note that a given child PCE may also act as a parent PCE.

For the reasons described previously, new OF codes need to be defined for the new inter-domain objective functions. Then the PCE can notify its new inter-domain objective functions to the PCC by carrying them in the OF-list TLV which is carried in the OPEN object. The PCC can specify which objective function code to use, which is carried in the OF object when requesting a PCE to compute an inter-domain path.

A parent PCE MUST be capable of ensuring homogeneity, across domains, when applying OF codes for strict OF intra-domain requests.

2.1.3. Multi-domain Metrics

For inter-domain path computation, there are several path metrics of interest.

- Domain count (number of domains crossed);

o Border Node count.

A PCC may be able to limit the number of domains crossed by applying a limit on these metrics. Details in section 3.3.

2.2. Parent PCE Capability Advertisement

Parent and child PCE relationships are likely to be configured. However, as mentioned in [RFC6805], it would assist network operators if the child and parent PCEs could indicate their H-PCE capabilities.

During the PCEP session establishment procedure, the child PCE needs to be capable of indicating to the parent PCE whether it requests the parent PCE capability or not. Also, during the PCEP session establishment procedure, the parent PCE needs to be capable of indicating whether its parent capability can be provided or not.

A PCEP Speaker (Parent PCE or Child PCE or PCC) includes the "H-PCE Capability" TLV, described in Section 3.1.1, in the OPEN Object to advertise its support for PCEP extensions for H-PCE Capability.

2.3. PCE Domain Discovery

A PCE domain is a single domain with an associated PCE. Although it is possible for a PCE to manage multiple domains. The PCE domain may be an IGP area or AS.

The PCE domain identifiers may be provided during the PCEP session establishment procedure.

2.4. Domain Diversity

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.

The main motivation behind domain diversity is to avoid fate sharing, but it can also be because of some geo-political reasons and commercial relationships that would require domain diversity. for example, a pair of paths should choose different transit Autonomous System (AS) because of some policy considerations.

In case when full domain diversity could not be achieved, it is helpful to minimize the common shared domains. Also it is interesting to note that other scope of diversity (node, link, SRLG etc) can still be applied inside the common shared domains.

3. PCEP Extensions

This section defines PCEP extensions to ([RFC5440]) so as to support the H-PCE procedures.

3.1. OPEN object

Two new TLVs are defined in this document to be carried within an OPEN object. This way, during PCEP session establishment, the H-PCE capability and Domain information can be advertised.

3.1.1. H-PCE capability TLV

The H-PCE-CAPABILITY TLV is an optional TLV associated with the OPEN Object [RFC5440] to exchange H-PCE capability of PCEP speakers.

Its format is shown in the following figure:

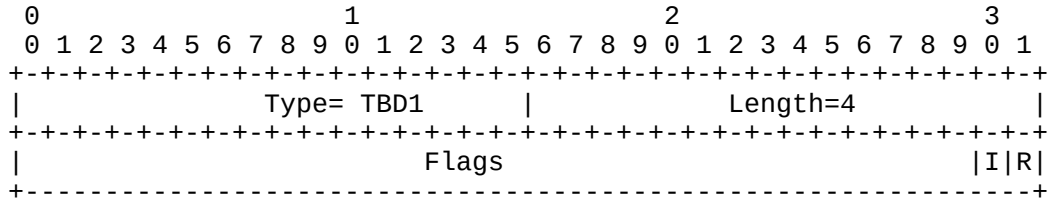


Figure 1: H-PCE-CAPABILITY TLV format

The type of the TLV is TBD1 (to be assigned by IANA) and it has a fixed length of 4 octets.

The value comprises a single field - Flags (32 bits):

R (Parent PCE Request bit): if set, will signal that the child PCE wishes to use the peer PCE as a parent PCE.

I (Parent PCE Indication bit): if set, will signal that the PCE can be used as a parent PCE by the peer PCE.

The inclusion of this TLV in an OPEN object indicate that the H-PCE extensions are supported by the PCEP speaker. The PCC MAY include this TLV to indicate that it understands the H-PCE extensions. The Child PCE MUST include this TLV and set the R flag (and unset the I

flag) on the PCEP session towards the Parent PCE. The Parent PCE MUST include this TLV and set the I flag and unset the R flag on the PCEP session towards the child PCE. The parent-child PCEP session is set to be established only when this capability is advertised.

If such capability is not exchanged and the parent PCE receive a "H-PCE path computation request", it MUST send a PCErr message with Error-Type=TBD8 (H-PCE error) and Error-Value=1 (Parent PCE Capability not advertised).

3.1.2. Domain-ID TLV

The Domain-ID TLV when used in OPEN object identify the domain(s) served by the PCE. The child PCE uses this mechanism to inform the domain information to the parent PCE.

The Domain-ID TLV is defined below:

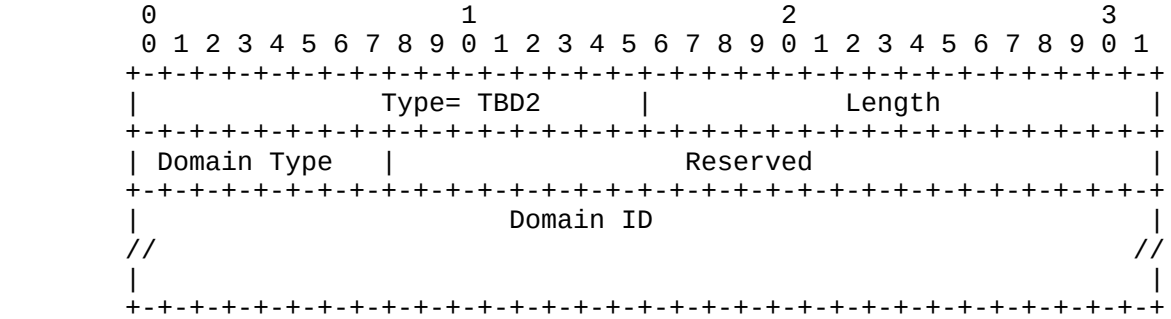


Figure 2: Domain-ID TLV format

The type of the TLV is TBD2 (to be assigned by IANA) and it has a variable Length of the value portion. The value part comprises of -

Domain Type (8 bits): Indicates the domain type. Four types of domain are currently defined:

* Type=1: the Domain ID field carries a 2-byte AS number. Padded with trailing zeroes to a 4-byte boundary.

* Type=2: the Domain ID field carries a 4-byte AS number.

* Type=3: the Domain ID field carries an 4-byte OSPF area ID.

* Type=4: the Domain ID field carries [2-byte Area-Len, variable length IS-IS area ID]. Padded with trailing zeroes to a 4-byte boundary.

Reserved: Zero at transmission; ignored at receipt.

Domain ID (variable): Indicates an IGP Area ID or AS number. It can be 2 bytes, 4 bytes or variable length depending on the domain identifier used. It is padded with trailing zeroes to a 4-byte boundary.

In case a PCE serves more than one domain, multiple Domain-ID TLV is included for each domain it serves.

3.2. RP object

3.2.1. H-PCE-FLAG TLV

The H-PCE-FLAG TLV is an optional TLV associated with the RP Object [RFC5440] to indicate the H-PCE path computation request and options.

Its format is shown in the following figure:

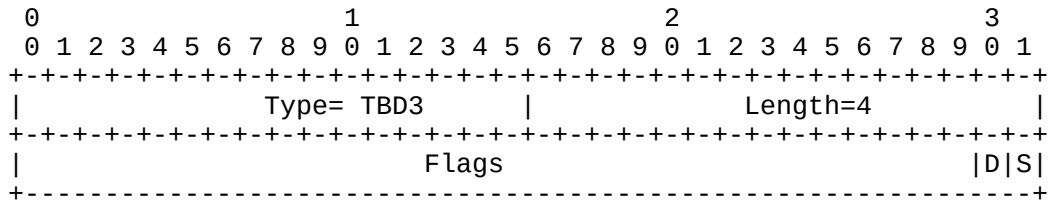


Figure 3: H-PCE-FLAG TLV format

The type of the TLV is TBD3 (to be assigned by IANA) and it has a fixed length of 4 octets.

The value comprises a single field - Flags (32 bits):

S (Domain Sequence bit): if set, will signal that the child PCE wishes to get only the domain sequence in the path computation reply. Refer section 3.7 of [RFC7897] for details.

D (Disallow Domain Re-entry bit): if set, will signal that the computed path does not enter a domain more than once.

3.2.2. Domain-ID TLV

The usage of Domain-ID TLV carried in an OPEN object is used to indicate a (list of) managed domains and is described in section 3.1.2. This TLV when carried in a RP object, indicates the destination domain ID. If a PCC knows the egress domain, it can

supply this information in the PCReq message. The format of this TLV is defined in Section 3.1.2.

3.3. Objective Functions

3.3.1. OF Codes

[RFC5541] defines a mechanism to specify an objective function that is used by a PCE when it computes a path. Two new objective functions are defined for the H-PCE experiment.

- MTD
  - Name: Minimize the number of Transit Domains (MTD)
  - Objective Function Code - TBD4 (to be assigned by IANA)
  - Description: Find a path P such that it passes through the least number of transit domains.
  - Objective functions are formulated using the following terminology:
    - A network comprises a set of N domains {Di, (i=1...N)}.
    - A path P passes through K domains {Dpi,(i=1...K)}.
    - Find a path P such that the value of K is minimized.
- MBN
  - Name: Minimize the number of border nodes.
  - Objective Function Code - TBD5 (to be assigned by IANA)
  - Description: Find a path P such that it passes through the least number of border nodes.
  - Objective functions are formulated using the following terminology:
    - A network comprises a set of N nodes {Ni, (i=1...N)}.
    - A path P is a list of K nodes {Npi,(i=1...K)}.
    - B(N) if a function that determine if the node is a border node. B(Ni) = 1 if Ni is border node; B(Nk) = 0 if Nk is not a border node.

+ The number of border node in a path P is denoted by B(P), where B(P) = sum{B(Npi),(i=1...K)}.

+ Find a path P such that B(P) is minimized.

MCTD

- Name: Minimize the number of Common Transit Domains.

- Objective Function Code: TBD13

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

3.3.2. OF Object

The OF (Objective Function) object [RFC5541] is carried within a PCReq message so as to indicate the desired/required objective function to be applied by the PCE during path computation. As per section 3.2 of [RFC5541] a single OF object may be included in a path computation request.

The new OF code described in section 3.3.1 are applicable at the inter-domain level (parent), it is also necessary to specify the OF code that may be applied at the intra-domain (child) path computation level. To accommodate this, the OF-List TLV (described in section 2.1. of [RFC5541]) is included in the OF object as an optional TLV.

OF-List TLV allow encoding of multiple OF codes. When this TLV is included inside the OF object, only the first OF-code in the OF-LIST TLV is considered. The parent PCE would use this OF code in the OF object when sending the intra domain path computation request to the child PCE.

If the objective functions defined in this document are unknown/ unsupported by a PCE, then the procedure as defined in [RFC5541] is followed.

3.4. Metric Object

The METRIC object is defined in section 7.8 of [RFC5440], comprising metric-value, metric-type (T field) and flags. This document defines the following types for the METRIC object for H-PCE:

- T=TBD6: Domain count metric (number of domains crossed);

- T=TBD7: Border Node count metric (number of border nodes crossed).

The domain count metric type of the METRIC object encodes the number of domain crossed in the path. The border node count metric type of the METRIC object encodes the number of border nodes in the path.

A PCC or child PCE MAY use these metric in PCReq message an inter-domain path meeting the number of domain or border nodes requirement. In this case, the B bit MUST be set to suggest a bound (a maximum) for the metric that must not be exceeded for the PCC to consider the computed path as acceptable.

A PCC or child PCE MAY also use this metric to ask the PCE to optimize the metric during inter-domain path computation. In this case, the B flag MUST be cleared.

The Parent PCE MAY use these metric in a PCRep message along with a NO-PATH object in the case where the PCE cannot compute a path meeting this constraint. A PCE MAY also use this metric to send the computed end to end metric in a reply message.

3.5. 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.

Following new bit is added in the Flags field:

- O (Domain diverse) bit - TBD12 : 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.

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

3.6. PCEP-ERROR object

3.6.1. Hierarchy PCE Error-Type

A new PCEP Error-Type is used for this H-PCE experiment and is defined below:

Error-Type	Meaning
TBD8	H-PCE error Error-value=1: parent PCE capability was not advertised Error-value=2: parent PCE capability cannot be provided

Figure 4: H-PCE error

3.7. NO-PATH Object

To communicate the reason(s) for not being able to find a multi-domain path or domain sequence, the NO-PATH object can be used in the PCRep message. [RFC5440] defines the format of the NO-PATH object. The object may contain a NO-PATH-VECTOR TLV to provide additional information about why a path computation has failed.

Three new bit flags are defined to be carried in the Flags field in the NO-PATH-VECTOR TLV carried in the NO-PATH Object.

- Bit number TBD9: When set, the parent PCE indicates that destination domain unknown;
- Bit number TBD10: When set, the parent PCE indicates unresponsive child PCE(s);
- Bit number TBD11: When set, the parent PCE indicates no available resource available in one or more domain(s).

4. H-PCE Procedures

4.1. OPEN Procedure between Child PCE and Parent PCE

If a child PCE wants to use the peer PCE as a parent, it can set the R (parent PCE request flag) in the H-PCE-CAPABILITY TLV inside the OPEN object carried in the Open message during the PCEP session creation procedure.

If the parent PCE can provide the parent function to the peer PCE, it may set the I (parent PCE indication flag) in the H-PCE-CAPABILITY TLV inside the OPEN object carried in the Open message during the PCEP session creation procedure.

The PCE may also report its list of domain IDs to the peer PCE by specifying them in the Domain-ID TLVs in the OPEN object carried in the Open message during the PCEP session creation procedure.

The OF codes defined in this document can be carried in the OF-list TLV of the OPEN object. If the OF-list TLV carries the OF codes, it means that the PCE is capable of implementing the corresponding objective functions. This information can be used for selecting a proper parent PCE when a child PCE wants to get a path that satisfies a certain objective function.

When a specific child PCE sends a PCReq to a peer PCE that requires parental activity and H-PCE capability flags were not set in the session establishment procedure as described above, the peer PCE should send a PCErr message to the child PCE and specify the error-type=TBD (H-PCE error) and error-value=1 (parent PCE capability was not advertised) in the PCEP-ERROR object.

When a specific child PCE sends a PCReq to a peer PCE that requires parental activity and the peer PCE does not want to act as the parent for it, the peer PCE should send a PCErr message to the child PCE and specify the error-type=TBD (H-PCE error) and error-value=2 (parent PCE capability cannot be provided) in the PCEP-ERROR object.

4.2. Procedure to obtain Domain Sequence

If a child PCE only wants to get the domain sequence for a multi-domain path computation from a parent PCE, it can set the Domain Path Request bit in the H-PCE-FLAG TLV in the RP object carried in a PCReq message. The parent PCE which receives the PCReq message tries to compute a domain sequence for it (instead for E2E path). If the domain path computation succeeds the parent PCE sends a PCRep message which carries the domain sequence in the ERO to the child PCE. Refer [RFC7897] for more details about domain sub-objects in the ERO. Otherwise it sends a PCReq message which carries the NO-PATH object to the child PCE.

5. Error Handling

A PCE that is capable of acting as a parent PCE might not be configured or willing to act as the parent for a specific child PCE.

This fact could be determined when the child sends a PCReq that requires parental activity, and could result in a negative response in a PCEP Error (PCErr) message and indicate the hierarchy PCE error-type=TBD8 (H-PCE error) and suitable error-value. (section 3.5.1)

Additionally, the parent PCE may fail to find the multi-domain path or domain sequence due to one or more of the following reasons:

- o A child PCE cannot find a suitable path to the egress;
- o The parent PCE do not hear from a child PCE for a specified time;
- o The objective functions specified in the path request cannot be met.

In this case, the parent PCE MAY need to send a negative path computation reply specifying the reason. This can be achieved by including NO-PATH object in the PCRep message. Extension to NO-PATH object is needed to include the aforementioned reasons described in section 3.6.

6. Manageability Considerations

General PCE and PCEP management considerations are discussed in [RFC4655] and [RFC5440]. There are additional management considerations for H-PCE which are described in [RFC6805], and repeated in this section.

The administrative entity responsible for the management of the parent PCEs must be determined for the following cases:

- o multi-domains (e.g., IGP areas or multiple ASes) within a single service provider network, the management responsibility for the parent PCE would most likely be handled by the service provider,
- o multiple ASes within different service provider networks, 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.

6.1. Control of Function and Policy

Control and function will need to be carefully managed in a H-PCE network. A child PCE will need to be configured with the address of its parent PCE. It is expected that there will only be one or two parents of any child.

The parent PCE also needs to be aware of the child PCEs for all child domains that it can see. This information is most likely to be configured (as part of the administrative definition of each domain).

Discovery of the relationships between parent PCEs and child PCEs does not form part of the hierarchical PCE architecture. Mechanisms

that rely on advertising or querying PCE locations across domain or provider boundaries are undesirable for security, scaling, commercial, and confidentiality reasons. Specific behavior of the child and parent PCE are described in the following sub-sections.

6.1.1. Child PCE

Support of the hierarchical procedure will be controlled by the management organization responsible for each child PCE. A child PCE must be configured with the address of its parent PCE in order for it to interact with its parent PCE. The child PCE must also be authorized to peer with the parent PCE.

6.1.2. Parent PCE

The parent PCE must only accept path computation requests from authorized child PCEs. If a parent PCE receives requests from an unauthorized child PCE, the request should be dropped. This means that a parent PCE must be configured with the identities and security credentials of all of its child PCEs, or there must be some form of shared secret that allows an unknown child PCE to be authorized by the parent PCE.

6.1.3. Policy Control

It may be necessary to maintain a policy module on the parent PCE [RFC5394]. This would allow the parent PCE to apply commercially relevant constraints such as SLAs, security, peering preferences, and monetary costs.

It may also be necessary for the parent PCE to limit end-to-end path selection by including or excluding specific domains based on commercial relationships, security implications, and reliability.

6.2. Information and Data Models

A MIB module for PCEP was published as RFC 7420 [RFC7420] that describes managed objects for modeling of PCEP communication. A YANG module for PCEP has also been proposed [I-D.ietf-pce-pcep-yang].

A H-PCE MIB module, or additional data model, will be required to report parent PCE and child PCE information, including:

- parent PCE configuration and status,
- child PCE configuration and information,

- o notifications to indicate session changes between parent PCEs and child PCEs, and

- o notification of parent PCE TED updates and changes.

6.3. Liveness Detection and Monitoring

The hierarchical procedure requires interaction with multiple PCEs. Once a child PCE requests an end-to-end path, a sequence of events occurs that requires interaction between the parent PCE and each child PCE. If a child PCE is not operational, and an alternate transit domain is not available, then a failure must be reported.

6.4. Verify Correct Operations

Verifying the correct operation of a parent PCE can be performed by monitoring a set of parameters. The parent PCE implementation should provide the following parameters monitored by the parent PCE:

- o number of child PCE requests,

- o number of successful hierarchical PCE procedures completions on a per-PCE-peer basis,

- o number of hierarchical PCE procedure completion failures on a per-PCE-peer basis, and

- o number of hierarchical PCE procedure requests from unauthorized child PCEs.

6.5. Requirements On Other Protocols

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

6.6. Impact On Network Operations

The hierarchical PCE procedure is a multiple-PCE path computation scheme. Subsequent requests to and from the child and parent PCEs do not differ from other path computation requests and should not have any significant impact on network operations.

7. IANA Considerations

7.1. PCEP TLV Type Indicators

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.

This document defines three new PCEP TLVs. IANA is requested to make the following allocation:

Type	TLV name	References
TBD1	H-PCE-CAPABILITY TLV	This I-D
TBD2	Domain-ID TLV	This I-D
TBD3	H-PCE-FLAG TLV	This I-D

7.2. H-PCE-CAPABILITY TLV Flags

This document requests that a new sub-registry, named " H-PCE-CAPABILITY TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the H-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:

- Bit number (counting from bit 0 as the most significant bit)
- Capability description
- Defining RFC

The following values are defined in this document:

Bit	Description	Reference
31	R (Parent PCE Request bit)	This I.D.
30	I (Parent PCE Indication bit)	This I.D.

7.3. Domain-ID TLV Domain type

This document requests that a new sub-registry, named " Domain-ID TLV Domain type", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Domain-Type field of the Domain-ID TLV.

Value	Meaning
1	2-byte AS number
2	4-byte AS number
3	4-byte OSPF area ID
4	Variable length IS-IS area ID

7.4. H-PCE-FLAG TLV Flags

This document requests that a new sub-registry, named "H-PCE-FLAGS TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the H-PCE-FLAGS 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:

- Bit number (counting from bit 0 as the most significant bit)
- Capability description
- Defining RFC

The following values are defined in this document:

Bit	Description	Reference
31	S (Domain Sequence bit)	This I.D.
30	D (Disallow Domain Re-entry bit)	This I.D.

7.5. OF Codes

IANA maintains registry of Objective Function (described in [RFC5541]) at the sub-registry "Objective Function". Two new Objective Functions have been defined in this document.

IANA is requested to make the following allocations:

Code Point	Name	Reference
TBD4	Minimum number of Transit Domains (MTD)	This I.D.
TBD5	Minimize number of Border Nodes (MBN)	This I.D.
TBD13	Minimize the number of Common Transit Domains. (MCTD)	This I.D.

7.6. METRIC Types

IANA maintains one sub-registry for "METRIC object T field". Two new metric types are defined in this document for the METRIC object (specified in [RFC5440]).

IANA is requested to make the following allocations:

Value	Description	Reference
TBD6	Domain Count metric	This I.D.
TBD7	Border Node Count metric	This I.D.

7.7. New PCEP Error-Types and Values

IANA maintains a registry of Error-Types and Error-values for use in PCEP messages. This is maintained as the "PCEP-ERROR Object Error Types and Values" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry.

IANA is requested to make the following allocations:

Error-Type	Meaning and error values	Reference
TBD8	H-PCE Error	This I.D.
	Error-value=1 Parent PCE Capability not advertised	
	Error-value=2 Parent PCE Capability not supported	

7.8. New NO-PATH-VECTOR TLV Bit Flag

IANA maintains a registry of bit flags carried in the PCEP NO-PATH-VECTOR TLV in the PCEP NO-PATH object as defined in [RFC5440]. IANA Is requested to assign three new bit flag as follows:

Bit Number	Name Flag	Reference
TBD9	Destination Domain unknown	This I.D.
TBD10	Unresponsive child PCE(s)	This I.D.
TBD11	No available resource in one or more domain	This I.D.

7.9. SVEC Flag

IANA maintains a registry of bit flags carried in the PCEP SVEC object as defined in [RFC5440]. IANA Is requested to assign one new bit flag as follows:

Bit Number	Name Flag	Reference
TBD13	Domain Diverse	This I.D.

8. Security Considerations

The hierarchical PCE procedure relies on PCEP and inherits the security requirements defined in [RFC5440]. As PCEP operates over TCP, it may also make use of TCP security mechanisms, such as TCP-AO or [RFC8253].

H-PCE operation also relies on information used to build the TED. Attacks on a parent or child PCE may be achieved by falsifying or impeding this flow of information. If the child PCE listens to the IGP or BGP-LS for populating the TED, then normal IGP or BGP-LS security measures may be applied, and it should be noted that an IGP routing system is generally assumed to be a trusted domain such that router subversion is not a risk. The parent PCE TED is constructed as described in this document and may involve:

- multiple parent-child relationships using PCEP
- the parent PCE listening to child domain IGPs (with the same security features as a child PCE listening to its IGP)
- an external mechanism (such as [RFC7752]), which will need to be authorized and secured.

Any multi-domain operation necessarily involves the exchange of information across domain boundaries. This is bound to represent a significant security and confidentiality risk especially when the child domains are controlled by different commercial concerns. PCEP allows individual PCEs to maintain confidentiality of their domain path information using path-keys [RFC5520], and the H-PCE architecture is specifically designed to enable as much isolation of domain topology and capabilities information as is possible.

For further considerations of the security issues related to inter-AS path computation, see [RFC5376].

9. Implementation Status

The H-PCE architecture and protocol procedures describe in this I-D were implemented and tested for a variety of optical research applications.

9.1. Inter-layer traffic engineering with H-PCE

This work was led by:

- Ramon Casellas [ramon.casellas@cttc.es]
- Centre Tecnologic de Telecomunicacions de Catalunya (CTTC)

The H-PCE instances (parent and child) were multi-threaded asynchronous processes. Implemented in C++11, using C++ Boost Libraries. The targeted system used to deploy and run H-PCE applications was a POSIX system (Debian GNU/Linux operating system).

Some parts of the software may require a Linux Kernel, the availability of a Routing Controller running collocated in the same host and the usage of libnetfilter / libipq and GNU/Linux firewalling capabilities. Most of the functionality, including algorithms is done by means of plugins (e.g., as shared libraries or .so files in Unix systems).

The CTTC PCE supports the H-PCE architecture, but also supports stateful PCE with active capabilities, as an OpenFlow controller, and has dedicated plugins to support monitoring, BRPC, P2MP, path keys, back end PCEs. Management of the H-PCE entities was supported via HTTP and CLI via Telnet.

Further details of the H-PCE prototyping and experimentation can be found in the following scientific papers:

R. Casellas, R. Martinez, R. Munoz, L. Liu, T. Tsuritani, I. Morita, "Inter-layer traffic engineering with hierarchical-PCE in MPLS-TP over wavelength switched optical networks" , Optics Express, Vol. 20, No. 28, December 2012.

R. Casellas, R. Martinez, R. Munoz, L. Liu, T. Tsuritani, I. Morita, M. Msurusawa, "Dynamic virtual link mesh topology aggregation in multi-domain translucent WSON with hierarchical-PCE", Optics Express Journal, Vol. 19, No. 26, December 2011.

R. Casellas, R. Munoz, R. Martinez, R. Vilalta, L. Liu, T. Tsuritani, I. Morita, V. Lopez, O. Gonzalez de Dios, J. P. Fernandez-Palacios, "SDN based Provisioning Orchestration of OpenFlow/GMPLS Flexi-grid Networks with a Stateful Hierarchical PCE", in Proceedings of Optical Fiber Communication Conference and Exposition (OFC), 9-13 March, 2014, San Francisco (EEUU). Extended Version to appear in Journal Of Optical Communications and Networking January 2015

F. Paolucci, O. Gonzalez de Dios, R. Casellas, S. Duhovnikov, P. Castoldi, R. Munoz, R. Martinez, "Experimenting Hierarchical PCE Architecture in a Distributed Multi-Platform Control Plane Testbed" , in Proceedings of Optical Fiber Communication Conference and Exposition (OFC) and The National Fiber Optic Engineers Conference (NFOEC), 4-8 March, 2012, Los Angeles, California (USA).

R. Casellas, R. Martinez, R. Munoz, L. Liu, T. Tsuritani, I. Morita, M. Tsurusawa, "Dynamic Virtual Link Mesh Topology Aggregation in Multi-Domain Translucent WSON with Hierarchical-PCE", in Proceedings of 37th European Conference and Exhibition on Optical Communication (ECOC 2011), 18-22 September 2011, Geneve (Switzerland).

R. Casellas, R. Munoz, R. Martinez, "Lab Trial of Multi-Domain Path Computation in GMPLS Controlled WSON Using a Hierarchical PCE", in Proceedings of OFC/NFOEC Conference (OFC2011), 10 March 2011, Los Angeles (USA).

9.2. Telefonica Netphony (Open Source PCE)

The Telefonica Netphony PCE is an open source Java-based implementation of a Path Computation Element, with several flavours, and a Path Computation Client. The PCE follows a modular architecture and allows to add customized algorithms. The PCE has also stateful and remote initiation capabilities. In current version, three components can be built, a domain PCE (aka child PCE),

a parent PCE (ready for the H-PCE architecture) and a PCC (path computation client).

This work was led by:

- Oscar Gonzalez de Dios [oscar.gonzalezdedios@telefonica.com]
- Victor Lopez Alvarez [victor.lopezalvarez@telefonica.com]
- Telefonica I+D, Madrid, Spain

The PCE code is publicly available in a GitHub repository:

- https://github.com/telefonicaid/netphony-pce

The PCEP protocol encodings are located in the following repository:

- https://github.com/telefonicaid/netphony-network protocols

The traffic engineering database and a BGP-LS speaker to fill the database is located in:

- https://github.com/telefonicaid/netphony-topology

The parent and child PCE are multi-threaded java applications. The path computation uses the jgrapht free Java class library (0.9.1) that provides mathematical graph-theory objects and algorithms. Current version of netphony PCE runs on java 1.7 and 1.8, and has been tested in GNU/Linux, Mac OS-X and Windows environments. The management of the parent and domain PCEs is supported though CLI via Telnet, and configured via XML files.

Further details of the netphony H-PCE prototyping and experimentation can be found in the following research papers:

- O. Gonzalez de Dios, R. Casellas, F. Paolucci, A. Napoli, L. Gifre, A. Dupas, E, Hugues-Salas, R. Morro, S. Belotti, G. Meloni, T. Rahman, V.P Lopez, R. Martinez, F. Fresi, M. Bohn, S. Yan, L. Velasco, . Layec and J. P. Fernandez-Palacios: Experimental Demonstration of Multivendor and Multidomain EON With Data and Control Interoperability Over a Pan-European Test Bed, in Journal of Lightwave Technology, Dec. 2016, Vol. 34, Issue 7, pp. 1610-1617.
- O. Gonzalez de Dios, R. Casellas, R. Morro, F. Paolucci, V. Lopez, R. Martinez, R. Munoz, R. Villalta, P. Castoldi: "Multi-partner Demonstration of BGP-LS enabled multi-domain EON,

in Journal of Optical Communications and Networking, Dec. 2015, Vol. 7, Issue 12, pp. B153-B162.

- F. Paolucci, O. Gonzalez de Dios, R. Casellas, S. Duhovnikov, P. Castoldi, R. Munoz, R. Martinez, "Experimenting Hierarchical PCE Architecture in a Distributed Multi-Platform Control Plane Testbed" , in Proceedings of Optical Fiber Communication Conference and Exposition (OFC) and The National Fiber Optic Engineers Conference (NFOEC), 4-8 March, 2012, Los Angeles, California (USA).

9.3. Implementation 3: H-PCE Proof of Concept developed by Huawei

Huawei developed this H-PCE on the Huawei Versatile Routing Platform (VRP) to experiment with the hierarchy of PCE. Both end to end path computation as well as computation for domain-sequence are supported.

This work was led by:

- Udayasree Pallee [udayasreereddy@gmail.com]
- Dhruv Dhody [dhruv.ietf@gmail.com]
- Huawei Technologies, Bangalore, India

Further work on stateful H-PCE [I-D.ietf-pce-stateful-hpce] is being carried out on ONOS.

10. Contributing Authors

Xian Zhang
Huawei
EMail: zhang.xian@huawei.com

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

EMail: dhruv.ietf@gmail.com

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, DOI 10.17487/RFC5152, February 2008, <https://www.rfc-editor.org/info/rfc5152>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/info/rfc5440>.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, DOI 10.17487/RFC5541, June 2009, <https://www.rfc-editor.org/info/rfc5541>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

11.2. Informative References

[RFC4655] Farrel, A., Vasseur, J., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, DOI 10.17487/RFC4655, August 2006, <https://www.rfc-editor.org/info/rfc4655>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <https://www.rfc-editor.org/info/rfc5226>.

[RFC5376] Bitar, N., Zhang, R., and K. Kumaki, "Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)", RFC 5376, DOI 10.17487/RFC5376, November 2008, <https://www.rfc-editor.org/info/rfc5376>.

[RFC5394] Bryskin, I., Papadimitriou, D., Berger, L., and J. Ash, "Policy-Enabled Path Computation Framework", RFC 5394, DOI 10.17487/RFC5394, December 2008, <https://www.rfc-editor.org/info/rfc5394>.

[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>.

[RFC5441] Vasseur, JP., Ed., 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, DOI 10.17487/RFC5441, April 2009, <https://www.rfc-editor.org/info/rfc5441>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <https://www.rfc-editor.org/info/rfc6805>.

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

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[RFC7897] Dhody, D., Palle, U., and R. Casellas, "Domain Subobjects for the Path Computation Element Communication Protocol (PCEP)", RFC 7897, DOI 10.17487/RFC7897, June 2016, <https://www.rfc-editor.org/info/rfc7897>.

[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>.

[I-D.ietf-pce-pcep-yang]
Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-07 (work in progress), March 2018.

[I-D.ietf-pce-stateful-hpce]
Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and O. Dios, "Hierarchical Stateful Path Computation Element (PCE).", draft-ietf-pce-stateful-hpce-04 (work in progress), March 2018.

Authors' Addresses

Fatai Zhang
Huawei
Huawei Base, Bantian, Longgang District
Shenzhen 518129
China

EMail: zhangfatai@huawei.com

Quintin Zhao
Huawei
125 Nagog Technology Park
Acton, MA 01719
USA

EMail: quintin.zhao@huawei.com

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid 28045
Spain

EMail: ogondio@tid.es

Ramon Casellas
CTTC
Av. Carl Friedrich Gauss n.7
Barcelona, Castelldefels
Spain

EMail: ramon.casellas@cttc.es

Daniel King
Old Dog Consulting
UK

EMail: daniel@olddog.co.uk

PCE Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 16, 2019

F. Zhang
Q. Zhao
Huawei
O. Gonzalez de Dios
Telefonica I+D
R. Casellas
CTTC
D. King
Old Dog Consulting
July 15, 2018

Extensions to Path Computation Element Communication Protocol (PCEP) for Hierarchical Path Computation Elements (PCE)
draft-ietf-pce-hierarchy-extensions-05

Abstract

The Hierarchical Path Computation Element (H-PCE) architecture RFC 6805, provides a mechanism to allow the optimum sequence of domains to be selected, and the optimum end-to-end path to be derived through the use of a hierarchical relationship between domains.

This document defines the Path Computation Element Protocol (PCEP) extensions for the purpose of implementing necessary Hierarchical PCE procedures and protocol extensions.

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 January 16, 2019.

Copyright Notice

Copyright (c) 2018 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

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 capability to compute the routes of end-to-end inter-domain MPLS Traffic Engineering (MPLS-TE) and GMPLS Label Switched Paths (LSPs) is expressed as requirements in [RFC4105] and [RFC4216]. This capability may be realized by a PCE [RFC4655]. The methods for establishing and controlling inter-domain MPLS-TE and GMPLS LSPs are documented in [RFC4726].

[RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs).

Within the hierarchical PCE architecture, the parent PCE is used to compute a multi-domain path based on the domain connectivity information . A child PCE may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its own domain topology information.

The H-PCE end-to-end domain path computation procedure is described below:

- A path computation client (PCC) sends the inter-domain path computation requests to the child PCE responsible for its domain;
- The child PCE forwards the request to the parent PCE;
- The parent PCE computes the likely domain paths from the ingress domain to the egress domain;
- The parent PCE sends the intra-domain path computation requests (between the domain border nodes) to the child PCEs which are responsible for the domains along the domain path;
- The child PCEs return the intra-domain paths to the parent PCE;
- The parent PCE constructs the end-to-end inter-domain path based on the intra-domain paths;
- The parent PCE returns the inter-domain path to the child PCE;
- The child PCE forwards the inter-domain path to the PCC.

In addition, the parent PCE may be requested to provide only the sequence of domains to a child PCE so that alternative inter-domain path computation procedures, including Per Domain (PD) [RFC5152] and Backwards Recursive Path Computation (BRPC) [RFC5441] may be used.

This document defines the PCEP extensions for the purpose of implementing Hierarchical PCE procedures, which are described in [RFC6805].

1.1. Scope

The following functions are out of scope of this document.

- Determination of Destination Domain (section 4.5 of [RFC6805])

  * via collection of reachability information from child domain;

  * via requests to the child PCEs to discover if they contain the destination node;

  * or any other methods.

- Parent Traffic Engineering Database (TED) methods (section 4.4 of [RFC6805])

- Learning of Domain connectivity and boundary nodes (BN) addresses.

- Stateful PCE Operations. (Refer [I-D.ietf-pce-stateful-hpce])

1.2. Terminology

This document uses the terminology defined in [RFC4655], [RFC5440] and the additional terms defined in section 1.4 of [RFC6805].

1.3. Requirements Language

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

2. Requirements for H-PCE

This section compiles the set of requirements of the PCEP protocol to support the H-PCE architecture and procedures.

[RFC6805] identifies high-level requirements of PCEP extensions required to support the hierarchical PCE model.

2.1. Path Computation Request

The Path Computation Request (PCReq) messages are used by a PCC or PCE to make a path computation request to a PCE. In order to achieve the full functionality of the H-PCE procedures, the PCReq message needs to include:

- Qualification of PCE Requests;

- Multi-domain Objective Functions (OF);

- Multi-domain Metrics.

2.1.1. Qualification of PCEP Requests

As described in section 4.8.1 of [RFC6805], the H-PCE architecture introduces new request qualifications, which are:

- It MUST be possible for a child PCE to indicate that a path computation request sent to a parent PCE should be satisfied by a domain sequence only, that is, not by a full end-to-end path. This allows the child PCE to initiate a per-domain (PD) [RFC5152] or a backward recursive path computation (BRPC) [RFC5441].

- As stated in [RFC6805], section 4.5, if a PCC knows the egress domain, it can supply this information as the path computation request. It SHOULD be possible to specify the destination domain information in a PCEP request, if it is known.

- It MAY be possible to indicate that the inter domain path computed by parent PCE should disallow domain re-entry.

2.1.2. Multi-domain Objective Functions

For inter-domain path computation, there is one new objective Function which is defined in section 1.3.1 and 4.1 of [RFC6805]:

- Minimize the number of domains crossed. A domain can be either an Autonomous System (AS) or an Internal Gateway Protocol (IGP) area depending on the type of multi-domain network hierarchical PCE is applied to.

Another objective Function to minimize the number of border nodes is also defined in this document.

During the PCEP session establishment procedure, the parent PCE needs to be capable of indicating the Objective Functions (OF) [RFC5541] capability in the Open message. This capability information may then be announced by child PCEs, and used for selecting the PCE when a PCC wants a path that satisfies one or multiple inter-domain objective functions.

When a PCC requests a PCE to compute an inter-domain path, the PCC needs to be capable of indicating the new objective functions for inter-domain path. Note that a given child PCE may also act as a parent PCE (for some other child PCE).

For the reasons described previously, new OF codes need to be defined for the new inter-domain objective functions. Then the PCE can notify its new inter-domain objective functions to the PCC by carrying them in the OF-list TLV which is carried in the OPEN object.

The PCC can specify which objective function code to use, which is carried in the OF object when requesting a PCE to compute an inter-domain path.

A parent PCE MUST be capable of ensuring homogeneity, across domains, when applying OF codes for strict OF intra-domain requests.

2.1.3. Multi-domain Metrics

For inter-domain path computation, there are several path metrics of interest.

- Domain count (number of domains crossed);
- Border Node count.

A PCC may be able to limit the number of domains crossed by applying a limit on these metrics. Details in Section 3.4.

2.2. Parent PCE Capability Advertisement

Parent and child PCE relationships are likely to be configured. However, as mentioned in [RFC6805], it would assist network operators if the child and parent PCEs could indicate their H-PCE capabilities.

During the PCEP session establishment procedure, the child PCE needs to be capable of indicating to the parent PCE whether it requests the parent PCE capability or not. Also, during the PCEP session establishment procedure, the parent PCE needs to be capable of indicating whether its parent capability can be provided or not.

A PCEP Speaker (Parent PCE or Child PCE or PCC) includes the "H-PCE Capability" TLV, described in Section 3.1.1, in the OPEN Object to advertise its support for PCEP extensions for H-PCE Capability.

2.3. PCE Domain Discovery

A PCE domain is a single domain with an associated PCE. Although it is possible for a PCE to manage multiple domains simultaneously. The PCE domain could be an IGP area or AS.

The PCE domain identifiers MAY be provided during the PCEP session establishment procedure.

2.4. Domain Diversity

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.

The main motivation behind domain diversity is to avoid fate sharing, but it can also be because of some geo-political reasons and commercial relationships that would require domain diversity. for example, a pair of paths should choose different transit Autonomous System (AS) because of some policy considerations.

In case when full domain diversity could not be achieved, it is helpful to minimize the common shared domains. Also it is interesting to note that other scope of diversity (node, link, SRLG etc) can still be applied inside the common shared domains.

3. PCEP Extensions

This section defines PCEP extensions to ([RFC5440]) so as to support the H-PCE procedures.

3.1. OPEN object

Two new TLVs are defined in this document to be carried within an OPEN object. This way, during PCEP session establishment, the H-PCE capability and Domain information can be advertised.

3.1.1. H-PCE capability TLV

The H-PCE-CAPABILITY TLV is an optional TLV associated with the OPEN Object [RFC5440] to exchange H-PCE capability of PCEP speakers.

Its format is shown in the following figure:

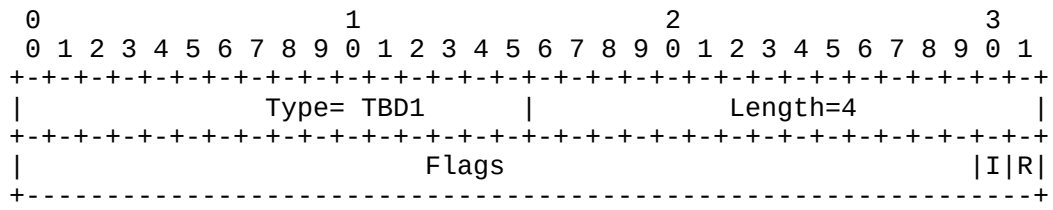


Figure 1: H-PCE-CAPABILITY TLV format

The type of the TLV is TBD1 (to be assigned by IANA) and it has a fixed length of 4 octets.

The value comprises a single field - Flags (32 bits):

R (Parent PCE Request bit): if set, will signal that the child PCE wishes to use the peer PCE as a parent PCE.

I (Parent PCE Indication bit): if set, will signal that the PCE can be used as a parent PCE by the peer PCE.

The inclusion of this TLV in an OPEN object indicate that the H-PCE extensions are supported by the PCEP speaker. The PCC MAY include this TLV to indicate that it understands the H-PCE extensions. The Child PCE MUST include this TLV and set the R flag (and unset the I flag) on the PCEP session towards the Parent PCE. The Parent PCE MUST include this TLV and set the I flag and unset the R flag on the PCEP session towards the child PCE. The parent-child PCEP session is set to be established only when this capability is advertised.

If such capability is not exchanged and the parent PCE receive a "H-PCE path computation request", it MUST send a PCErr message with Error-Type=TBD8 (H-PCE error) and Error-Value=1 (Parent PCE Capability not advertised).

3.1.2. Domain-ID TLV

The Domain-ID TLV when used in OPEN object identify the domain(s) served by the PCE. The child PCE uses this mechanism to inform the domain information to the parent PCE.

The Domain-ID TLV is defined below:

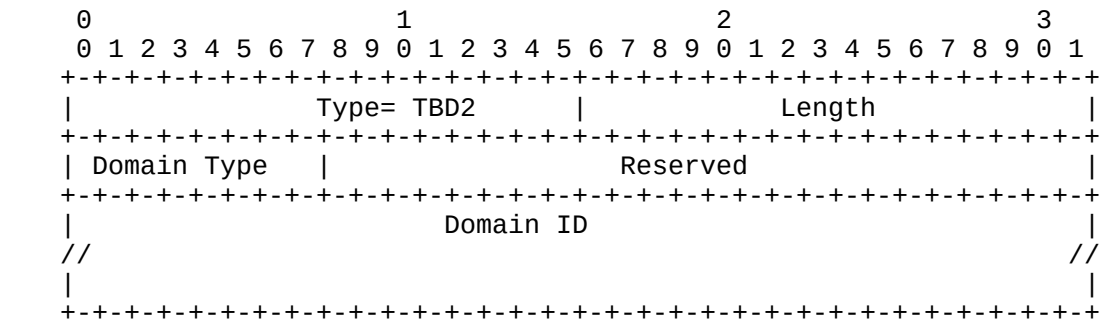


Figure 2: Domain-ID TLV format

The type of the TLV is TBD2 (to be assigned by IANA) and it has a variable Length of the value portion. The value part comprises of -

Domain Type (8 bits): Indicates the domain type. Four types of domain are currently defined:

* Type=1: the Domain ID field carries a 2-byte AS number. Padded with trailing zeros to a 4-byte boundary.

* Type=2: the Domain ID field carries a 4-byte AS number.

* Type=3: the Domain ID field carries an 4-byte OSPF area ID.

* Type=4: the Domain ID field carries (2-byte Area-Len, variable length IS-IS area ID). Padded with trailing zeros to a 4-byte boundary.

Reserved: Zero at transmission; ignored at receipt.

Domain ID (variable): Indicates an IGP Area ID or AS number. It can be 2 bytes, 4 bytes or variable length depending on the domain identifier used. It is padded with trailing zeros to a 4-byte boundary.

In case a PCE serves more than one domain, multiple Domain-ID TLV is included for each domain it serves.

3.2. RP object

3.2.1. H-PCE-FLAG TLV

The H-PCE-FLAG TLV is an optional TLV associated with the RP Object [RFC5440] to indicate the H-PCE path computation request and options.

Its format is shown in the following figure:

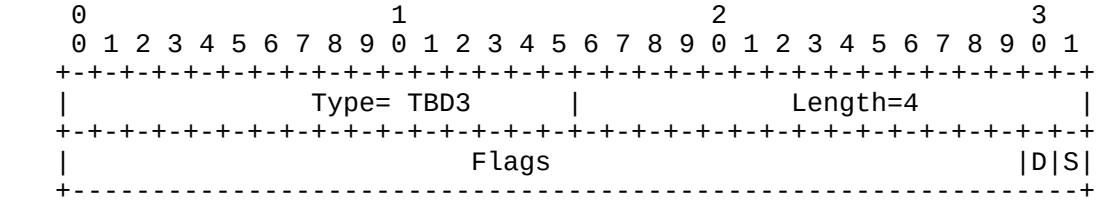


Figure 3: H-PCE-FLAG TLV format

The type of the TLV is TBD3 (to be assigned by IANA) and it has a fixed length of 4 octets.

The value comprises a single field - Flags (32 bits):

S (Domain Sequence bit): if set, will signal that the child PCE wishes to get only the domain sequence in the path computation reply. Refer section 3.7 of [RFC7897] for details.

D (Disallow Domain Re-entry bit): if set, will signal that the computed path does not enter a domain more than once.

3.2.2. Domain-ID TLV

The usage of Domain-ID TLV carried in an OPEN object is used to indicate a (list of) managed domains and is described in Section 3.1.2. This TLV when carried in a RP object, indicates the destination domain ID. If a PCC knows the egress domain, it can supply this information in the PCReq message. The format and procedure of this TLV is defined in Section 3.1.2.

3.3. Objective Functions

3.3.1. OF Codes

[RFC5541] defines a mechanism to specify an objective function that is used by a PCE when it computes a path. Two new objective functions are defined for the H-PCE experiment.

- MTD
  - Name: Minimize the number of Transit Domains (MTD)
  - Objective Function Code - TBD4 (to be assigned by IANA)

  * Description: Find a path P such that it passes through the least number of transit domains.

  * Objective functions are formulated using the following terminology:

    + A network comprises a set of N domains {Di, (i=1...N)}.

    + A path P passes through K domains {Dpi,(i=1...K)}.

    + Find a path P such that the value of K is minimized.

- MBN

  * Name: Minimize the number of border nodes.

  * Objective Function Code - TBD5 (to be assigned by IANA)

  * Description: Find a path P such that it passes through the least number of border nodes.

  * Objective functions are formulated using the following terminology:

    + A network comprises a set of N nodes {Ni, (i=1...N)}.

    + A path P is a list of K nodes {Npi,(i=1...K)}.

    + B(N) if a function that determine if the node is a border node. B(Ni) = 1 if Ni is border node; B(Nk) = 0 if Nk is not a border node.

    + The number of border node in a path P is denoted by B(P), where B(P) = sum{B(Npi),(i=1...K)}.

    + Find a path P such that B(P) is minimized.

MCTD

- Name: Minimize the number of Common Transit Domains.

- Objective Function Code: TBD13

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

3.3.2. OF Object

The OF (Objective Function) object [RFC5541] is carried within a PCReq message so as to indicate the desired/required objective function to be applied by the PCE during path computation. As per section 3.2 of [RFC5541] a single OF object may be included in a path computation request.

The new OF code described in Section 3.3.1 are applicable at the inter-domain level (parent), it is also necessary to specify the OF code that may be applied at the intra-domain (child) path computation level. To accommodate this, the OF-List TLV (described in section 2.1. of [RFC5541]) is included in the OF object as an optional TLV.

OF-List TLV allow encoding of multiple OF codes. When this TLV is included inside the OF object, only the first OF-code in the OF-LIST TLV is considered. The parent PCE MUST use this OF code in the OF object when sending the intra domain path computation request to the child PCE.

If the objective functions defined in this document are unknown/ unsupported by a PCE, then the procedure as defined in [RFC5541] is followed.

3.4. Metric Object

The METRIC object is defined in section 7.8 of [RFC5440], comprising metric-value, metric-type (T field) and flags. This document defines the following types for the METRIC object for H-PCE:

- T=TBD6: Domain count metric (number of domains crossed);
- T=TBD7: Border Node count metric (number of border nodes crossed).

The domain count metric type of the METRIC object encodes the number of domain crossed in the path. The border node count metric type of the METRIC object encodes the number of border nodes in the path.

A PCC or child PCE MAY use these metric in PCReq message an inter-domain path meeting the number of domain or border nodes requirement. As per [RFC5440], in this case, the B bit is set to suggest a bound (a maximum) for the metric that must not be exceeded for the PCC to consider the computed path as acceptable.

A PCC or child PCE MAY also use this metric to ask the PCE to optimize the metric during inter-domain path computation. In this case, the B flag is cleared.

The Parent PCE MAY use these metric in a PCRep message along with a NO-PATH object in the case where the PCE cannot compute a path meeting this constraint. A PCE MAY also use this metric to send the computed end to end metric in a reply message.

3.5. 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.

Following new bit is added in the Flags field:

- O (Domain diverse) bit - TBD12 : 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.

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

3.6. PCEP-ERROR object

3.6.1. Hierarchy PCE Error-Type

A new PCEP Error-Type [RFC5440] is used for the H-PCE extension as defined below:

Error-Type	Meaning
TBD8	H-PCE error Error-value=1: parent PCE capability was not advertised Error-value=2: parent PCE capability cannot be provided

Figure 4: H-PCE error

3.7. NO-PATH Object

To communicate the reason(s) for not being able to find a multi-domain path or domain sequence, the NO-PATH object can be used in the PCRep message. [RFC5440] defines the format of the NO-PATH object. The object may contain a NO-PATH-VECTOR TLV to provide additional information about why a path computation has failed.

Three new bit flags are defined to be carried in the Flags field in the NO-PATH-VECTOR TLV carried in the NO-PATH Object.

- Bit number TBD9: When set, the parent PCE indicates that destination domain unknown;
- Bit number TBD10: When set, the parent PCE indicates unresponsive child PCE(s);
- Bit number TBD11: When set, the parent PCE indicates no available resource available in one or more domain(s).

4. H-PCE Procedures

4.1. OPEN Procedure between Child PCE and Parent PCE

If a child PCE wants to use the peer PCE as a parent, it MUST set the R (parent PCE request flag) in the H-PCE-CAPABILITY TLV inside the OPEN object carried in the Open message during the PCEP session initialization procedure.

If the parent PCE can provide the parent function to the peer PCE, it MUST set the I (parent PCE indication flag) in the H-PCE-CAPABILITY TLV inside the OPEN object carried in the Open message during the PCEP session creation procedure.

The child PCE MAY also report its list of domain IDs to the parent PCE by specifying them in the Domain-ID TLVs in the OPEN object carried in the Open message during the PCEP session initialization procedure.

The OF codes defined in this document can be carried in the OF-list TLV of the OPEN object. If the OF-list TLV carries the OF codes, it means that the PCE is capable of implementing the corresponding objective functions. This information can be used for selecting a proper parent PCE when a child PCE wants to get a path that satisfies a certain objective function.

When a specific child PCE sends a PCReq to a peer PCE that requires parental activity and H-PCE capability flags were not set in the

session establishment procedure as described above, the peer PCE should send a PCErr message to the child PCE and specify the error-type=TBD (H-PCE error) and error-value=1 (parent PCE capability was not advertised) in the PCEP-ERROR object.

When a specific child PCE sends a PCReq to a peer PCE that requires parental activity and the peer PCE does not want to act as the parent for it, the peer PCE should send a PCErr message to the child PCE and specify the error-type=TBD (H-PCE error) and error-value=2 (parent PCE capability cannot be provided) in the PCEP-ERROR object.

4.2. Procedure to obtain Domain Sequence

If a child PCE only wants to get the domain sequence for a multi-domain path computation from a parent PCE, it can set the Domain Path Request bit in the H-PCE-FLAG TLV in the RP object carried in a PCReq message. The parent PCE which receives the PCReq message tries to compute a domain sequence for it (instead for E2E path). If the domain path computation succeeds the parent PCE sends a PCRep message which carries the domain sequence in the ERO to the child PCE. Refer [RFC7897] for more details about domain sub-objects in the ERO. Otherwise it sends a PCReq message which carries the NO-PATH object to the child PCE.

5. Error Handling

A PCE that is capable of acting as a parent PCE might not be configured or willing to act as the parent for a specific child PCE. This fact could be determined when the child sends a PCReq that requires parental activity, and could result in a negative response in a PCEP Error (PCErr) message and indicate the hierarchy PCE error-type=TBD8 (H-PCE error) and suitable error-value. (Section 3.6)

Additionally, the parent PCE may fail to find the multi-domain path or domain sequence due to one or more of the following reasons:

- A child PCE cannot find a suitable path to the egress;
- The parent PCE do not hear from a child PCE for a specified time;
- The objective functions specified in the path request cannot be met.

In this case, the parent PCE MAY need to send a negative path computation reply specifying the reason. This can be achieved by including NO-PATH object in the PCRep message. Extension to NO-PATH object is needed to include the aforementioned reasons described in Section 3.7.

6. Manageability Considerations

General PCE and PCEP management considerations are discussed in [RFC4655] and [RFC5440]. There are additional management considerations for H-PCE which are described in [RFC6805], and repeated in this section.

The administrative entity responsible for the management of the parent PCEs must be determined for the following cases:

- multi-domains (e.g., IGP areas or multiple ASes) within a single service provider network, the management responsibility for the parent PCE would most likely be handled by the service provider,
- multiple ASes within different service provider networks, 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.

6.1. Control of Function and Policy

Control and function will need to be carefully managed in a H-PCE network. A child PCE will need to be configured with the address of its parent PCE. It is expected that there will only be one or two parents of any child.

The parent PCE also needs to be aware of the child PCEs for all child domains that it can see. This information is most likely to be configured (as part of the administrative definition of each domain).

Discovery of the relationships between parent PCEs and child PCEs does not form part of the hierarchical PCE architecture. Mechanisms that rely on advertising or querying PCE locations across domain or provider boundaries are undesirable for security, scaling, commercial, and confidentiality reasons. Specific behavior of the child and parent PCE are described in the following sub-sections.

6.1.1. Child PCE

Support of the hierarchical procedure will be controlled by the management organization responsible for each child PCE. A child PCE must be configured with the address of its parent PCE in order for it to interact with its parent PCE. The child PCE must also be authorized to peer with the parent PCE.

6.1.2. Parent PCE

The parent PCE must only accept path computation requests from authorized child PCEs. If a parent PCE receives requests from an unauthorized child PCE, the request should be dropped. This means that a parent PCE must be configured with the identities and security credentials of all of its child PCEs, or there must be some form of shared secret that allows an unknown child PCE to be authorized by the parent PCE.

6.1.3. Policy Control

It may be necessary to maintain a policy module on the parent PCE [RFC5394]. This would allow the parent PCE to apply commercially relevant constraints such as SLAs, security, peering preferences, and monetary costs.

It may also be necessary for the parent PCE to limit end-to-end path selection by including or excluding specific domains based on commercial relationships, security implications, and reliability.

6.2. Information and Data Models

A MIB module for PCEP was published as RFC 7420 [RFC7420] that describes managed objects for modeling of PCEP communication. A YANG module for PCEP has also been proposed [I-D.ietf-pce-pcep-yang].

A H-PCE MIB module, or additional data model, will be required to report parent PCE and child PCE information, including:

- parent PCE configuration and status,
- child PCE configuration and information,
- notifications to indicate session changes between parent PCEs and child PCEs, and
- notification of parent PCE TED updates and changes.

6.3. Liveness Detection and Monitoring

The hierarchical procedure requires interaction with multiple PCEs. Once a child PCE requests an end-to-end path, a sequence of events occurs that requires interaction between the parent PCE and each child PCE. If a child PCE is not operational, and an alternate transit domain is not available, then a failure must be reported.

6.4. Verify Correct Operations

Verifying the correct operation of a parent PCE can be performed by monitoring a set of parameters. The parent PCE implementation should provide the following parameters monitored by the parent PCE:

- number of child PCE requests,
- number of successful hierarchical PCE procedures completions on a per-PCE-peer basis,
- number of hierarchical PCE procedure completion failures on a per-PCE-peer basis, and
- number of hierarchical PCE procedure requests from unauthorized child PCEs.

6.5. Requirements On Other Protocols

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

6.6. Impact On Network Operations

The hierarchical PCE procedure is a multiple-PCE path computation scheme. Subsequent requests to and from the child and parent PCEs do not differ from other path computation requests and should not have any significant impact on network operations.

7. IANA Considerations

7.1. PCEP TLV Type Indicators

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.

This document defines three new PCEP TLVs. IANA is requested to make the following allocation:

Type	TLV name	References
TBD1	H-PCE-CAPABILITY TLV	This I-D
TBD2	Domain-ID TLV	This I-D
TBD3	H-PCE-FLAG TLV	This I-D

7.2. H-PCE-CAPABILITY TLV Flags

This document requests that a new sub-registry, named " H-PCE-CAPABILITY TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the H-PCE-CAPABILITY TLV of the PCEP OPEN object.

New values are to be assigned by Standards Action [RFC5226]. Each bit should be tracked with the following qualities:

- Bit number (counting from bit 0 as the most significant bit)
- Capability description
- Defining RFC

The following values are defined in this document:

Bit	Description	Reference
31	R (Parent PCE Request bit)	This I.D.
30	I (Parent PCE Indication bit)	This I.D.

7.3. Domain-ID TLV Domain type

This document requests that a new sub-registry, named " Domain-ID TLV Domain type", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Domain-Type field of the Domain-ID TLV.

Value	Meaning
1	2-byte AS number
2	4-byte AS number
3	4-byte OSPF area ID
4	Variable length IS-IS area ID

7.4. H-PCE-FLAG TLV Flags

This document requests that a new sub-registry, named "H-PCE-FLAGS TLV Flag Field", is created within the "Path Computation Element Protocol (PCEP) Numbers" registry to manage the Flag field in the H-PCE-FLAGS TLV of the PCEP RP object. New values are to be assigned by Standards Action [RFC5226]. Each bit should be tracked with the following qualities:

- 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	S (Domain Sequence bit)	This I.D.
30	D (Disallow Domain Re-entry bit)	This I.D.

7.5. OF Codes

IANA maintains registry of Objective Function (described in [RFC5541]) at the sub-registry "Objective Function". Two new Objective Functions have been defined in this document.

IANA is requested to make the following allocations:

Code Point	Name	Reference
TBD4	Minimum number of Transit Domains (MTD)	This I.D.
TBD5	Minimize number of Border Nodes (MBN)	This I.D.
TBD13	Minimize the number of Common Transit Domains. (MCTD)	This I.D.

7.6. METRIC Types

IANA maintains one sub-registry for "METRIC object T field". Two new metric types are defined in this document for the METRIC object (specified in [RFC5440]).

IANA is requested to make the following allocations:

Value	Description	Reference
TBD6	Domain Count metric	This I.D.
TBD7	Border Node Count metric	This I.D.

7.7. New PCEP Error-Types and Values

IANA maintains a registry of Error-Types and Error-values for use in PCEP messages. This is maintained as the "PCEP-ERROR Object Error Types and Values" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry.

IANA is requested to make the following allocations:

Error-Type	Meaning and error values	Reference
TBD8	H-PCE Error	This I.D.
	Error-value=1 Parent PCE Capability not advertised	
	Error-value=2 Parent PCE Capability not supported	

7.8. New NO-PATH-VECTOR TLV Bit Flag

IANA maintains a registry of bit flags carried in the PCEP NO-PATH-VECTOR TLV in the PCEP NO-PATH object as defined in [RFC5440]. IANA Is requested to assign three new bit flag as follows:

Bit Number	Name Flag	Reference
TBD9	Destination Domain unknown	This I.D.
TBD10	Unresponsive child PCE(s)	This I.D.
TBD11	No available resource in one or more domain	This I.D.

7.9. SVEC Flag

IANA maintains a registry of bit flags carried in the PCEP SVEC object as defined in [RFC5440]. IANA Is requested to assign one new bit flag as follows:

Bit Number	Name Flag	Reference
TBD13	Domain Diverse	This I.D.

8. Security Considerations

The hierarchical PCE procedure relies on PCEP and inherits the security requirements defined in [RFC5440]. As PCEP operates over TCP, it may also make use of TCP security mechanisms, such as TCP-AO or [RFC8253].

H-PCE operation also relies on information used to build the TED. Attacks on a parent or child PCE may be achieved by falsifying or impeding this flow of information. If the child PCE listens to the IGP or BGP-LS for populating the TED, then normal IGP or BGP-LS security measures may be applied, and it should be noted that an IGP routing system is generally assumed to be a trusted domain such that router subversion is not a risk. The parent PCE TED is constructed as described in this document and may involve:

- multiple parent-child relationships using PCEP
- the parent PCE listening to child domain IGPs (with the same security features as a child PCE listening to its IGP)
- an external mechanism (such as [RFC7752]), which will need to be authorized and secured.

Any multi-domain operation necessarily involves the exchange of information across domain boundaries. This is bound to represent a significant security and confidentiality risk especially when the child domains are controlled by different commercial concerns. PCEP allows individual PCEs to maintain confidentiality of their domain path information using path-keys [RFC5520], and the H-PCE architecture is specifically designed to enable as much isolation of domain topology and capabilities information as is possible.

For further considerations of the security issues related to inter-AS path computation, see [RFC5376].

9. Implementation Status

The H-PCE architecture and protocol procedures describe in this I-D were implemented and tested for a variety of optical research applications.

9.1. Inter-layer traffic engineering with H-PCE

This work was led by:

- Ramon Casellas [ramon.casellas@cttc.es]

o Centre Tecnologic de Telecomunicacions de Catalunya (CTTC)

The H-PCE instances (parent and child) were multi-threaded asynchronous processes. Implemented in C++11, using C++ Boost Libraries. The targeted system used to deploy and run H-PCE applications was a POSIX system (Debian GNU/Linux operating system).

Some parts of the software may require a Linux Kernel, the availability of a Routing Controller running collocated in the same host and the usage of libnetfilter / libipq and GNU/Linux firewalling capabilities. Most of the functionality, including algorithms is done by means of plugins (e.g., as shared libraries or .so files in Unix systems).

The CTTC PCE supports the H-PCE architecture, but also supports stateful PCE with active capabilities, as an OpenFlow controller, and has dedicated plugins to support monitoring, BRPC, P2MP, path keys, back end PCEs. Management of the H-PCE entities was supported via HTTP and CLI via Telnet.

Further details of the H-PCE prototyping and experimentation can be found in the following scientific papers:

R. Casellas, R. Martinez, R. Munoz, L. Liu, T. Tsuritani, I. Morita, "Inter-layer traffic engineering with hierarchical-PCE in MPLS-TP over wavelength switched optical networks" , Optics Express, Vol. 20, No. 28, December 2012.

R. Casellas, R. Martinez, R. Munoz, L. Liu, T. Tsuritani, I. Morita, M. Msurusawa, "Dynamic virtual link mesh topology aggregation in multi-domain translucent WSON with hierarchical-PCE", Optics Express Journal, Vol. 19, No. 26, December 2011.

R. Casellas, R. Munoz, R. Martinez, R. Vilalta, L. Liu, T. Tsuritani, I. Morita, V. Lopez, O. Gonzalez de Dios, J. P. Fernandez-Palacios, "SDN based Provisioning Orchestration of OpenFlow/GMPLS Flexi-grid Networks with a Stateful Hierarchical PCE", in Proceedings of Optical Fiber Communication Conference and Exposition (OFC), 9-13 March, 2014, San Francisco (EEUU). Extended Version to appear in Journal Of Optical Communications and Networking January 2015

F. Paolucci, O. Gonzalez de Dios, R. Casellas, S. Duhovnikov, P. Castoldi, R. Munoz, R. Martinez, "Experimenting Hierarchical PCE Architecture in a Distributed Multi-Platform Control Plane Testbed" , in Proceedings of Optical Fiber Communication Conference and Exposition (OFC) and The National Fiber Optic

Engineers Conference (NFOEC), 4-8 March, 2012, Los Angeles, California (USA).

R. Casellas, R. Martinez, R. Munoz, L. Liu, T. Tsuritani, I. Morita, M. Tsurusawa, "Dynamic Virtual Link Mesh Topology Aggregation in Multi-Domain Translucent WSON with Hierarchical-PCE", in Proceedings of 37th European Conference and Exhibition on Optical Communication (ECOC 2011), 18-22 September 2011, Geneve (Switzerland).

R. Casellas, R. Munoz, R. Martinez, "Lab Trial of Multi-Domain Path Computation in GMPLS Controlled WSON Using a Hierarchical PCE", in Proceedings of OFC/NFOEC Conference (OFC2011), 10 March 2011, Los Angeles (USA).

9.2. Telefonica Netphony (Open Source PCE)

The Telefonica Netphony PCE is an open source Java-based implementation of a Path Computation Element, with several flavours, and a Path Computation Client. The PCE follows a modular architecture and allows to add customized algorithms. The PCE has also stateful and remote initiation capabilities. In current version, three components can be built, a domain PCE (aka child PCE), a parent PCE (ready for the H-PCE architecture) and a PCC (path computation client).

This work was led by:

- Oscar Gonzalez de Dios [oscar.gonzalezdedios@telefonica.com]
- Victor Lopez Alvarez [victor.lopezalvarez@telefonica.com]
- Telefonica I+D, Madrid, Spain

The PCE code is publicly available in a GitHub repository:

- https://github.com/telefonicaid/netphony-pce

The PCEP protocol encodings are located in the following repository:

- https://github.com/telefonicaid/netphony-network protocols

The traffic engineering database and a BGP-LS speaker to fill the database is located in:

- https://github.com/telefonicaid/netphony-topology

The parent and child PCE are multi-threaded java applications. The path computation uses the jgrapht free Java class library (0.9.1) that provides mathematical graph-theory objects and algorithms. Current version of netphony PCE runs on java 1.7 and 1.8, and has been tested in GNU/Linux, Mac OS-X and Windows environments. The management of the parent and domain PCEs is supported though CLI via Telnet, and configured via XML files.

Further details of the netphony H-PCE prototyping and experimentation can be found in the following research papers:

- O. Gonzalez de Dios, R. Casellas, F. Paolucci, A. Napoli, L. Gifre, A. Dupas, E, Hugues-Salas, R. Morro, S. Belotti, G. Meloni, T. Rahman, V.P Lopez, R. Martinez, F. Fresi, M. Bohn, S. Yan, L. Velasco, . Layec and J. P. Fernandez-Palacios: Experimental Demonstration of Multivendor and Multidomain EON With Data and Control Interoperability Over a Pan-European Test Bed, in Journal of Lightwave Technology, Dec. 2016, Vol. 34, Issue 7, pp. 1610-1617.

- O. Gonzalez de Dios, R. Casellas, R. Morro, F. Paolucci, V. Lopez, R. Martinez, R. Munoz, R. Villalta, P. Castoldi: "Multi-partner Demonstration of BGP-LS enabled multi-domain EON, in Journal of Optical Communications and Networking, Dec. 2015, Vol. 7, Issue 12, pp. B153-B162.

- F. Paolucci, O. Gonzalez de Dios, R. Casellas, S. Duhovnikov, P. Castoldi, R. Munoz, R. Martinez, "Experimenting Hierarchical PCE Architecture in a Distributed Multi-Platform Control Plane Testbed" , in Proceedings of Optical Fiber Communication Conference and Exposition (OFC) and The National Fiber Optic Engineers Conference (NFOEC), 4-8 March, 2012, Los Angeles, California (USA).

9.3. Implementation 3: H-PCE Proof of Concept developed by Huawei

Huawei developed this H-PCE on the Huawei Versatile Routing Platform (VRP) to experiment with the hierarchy of PCE. Both end to end path computation as well as computation for domain-sequence are supported.

This work was led by:

- Udayasree Pallee [udayasreereddy@gmail.com]

- Dhruv Dhody [dhruv.ietf@gmail.com]

- Huawei Technologies, Bangalore, India

Further work on stateful H-PCE [I-D.ietf-pce-stateful-hpce] is being carried out on ONOS.

10. Contributing Authors

Xian Zhang
Huawei
EMail: zhang.xian@huawei.com

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

EMail: dhruv.ietf@gmail.com

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, DOI 10.17487/RFC5152, February 2008, <https://www.rfc-editor.org/info/rfc5152>.

[RFC5440] Vasseur, JP., Ed. and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/info/rfc5440>.

[RFC5541] Le Roux, JL., Vasseur, JP., and Y. Lee, "Encoding of Objective Functions in the Path Computation Element Communication Protocol (PCEP)", RFC 5541, DOI 10.17487/RFC5541, June 2009, <https://www.rfc-editor.org/info/rfc5541>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

11.2. Informative References

[RFC4105] Le Roux, J., Ed., Vasseur, J., Ed., and J. Boyle, Ed., "Requirements for Inter-Area MPLS Traffic Engineering", RFC 4105, DOI 10.17487/RFC4105, June 2005, <https://www.rfc-editor.org/info/rfc4105>.

[RFC4216] Zhang, R., Ed. and J. Vasseur, Ed., "MPLS Inter-Autonomous System (AS) Traffic Engineering (TE) Requirements", RFC 4216, DOI 10.17487/RFC4216, November 2005, <https://www.rfc-editor.org/info/rfc4216>.

[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>.

[RFC4726] Farrel, A., Vasseur, J., and A. Ayyangar, "A Framework for Inter-Domain Multiprotocol Label Switching Traffic Engineering", RFC 4726, DOI 10.17487/RFC4726, November 2006, <https://www.rfc-editor.org/info/rfc4726>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <https://www.rfc-editor.org/info/rfc5226>.

[RFC5376] Bitar, N., Zhang, R., and K. Kumaki, "Inter-AS Requirements for the Path Computation Element Communication Protocol (PCECP)", RFC 5376, DOI 10.17487/RFC5376, November 2008, <https://www.rfc-editor.org/info/rfc5376>.

[RFC5394] Bryskin, I., Papadimitriou, D., Berger, L., and J. Ash, "Policy-Enabled Path Computation Framework", RFC 5394, DOI 10.17487/RFC5394, December 2008, <https://www.rfc-editor.org/info/rfc5394>.

[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>.

[RFC5441] Vasseur, JP., Ed., 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, DOI 10.17487/RFC5441, April 2009, <https://www.rfc-editor.org/info/rfc5441>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <https://www.rfc-editor.org/info/rfc6805>.

[RFC7420] Koushik, A., Stephan, E., Zhao, Q., King, D., and J. Hardwick, "Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module", RFC 7420, DOI 10.17487/RFC7420, December 2014, <https://www.rfc-editor.org/info/rfc7420>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[RFC7897] Dhody, D., Palle, U., and R. Casellas, "Domain Subobjects for the Path Computation Element Communication Protocol (PCEP)", RFC 7897, DOI 10.17487/RFC7897, June 2016, <https://www.rfc-editor.org/info/rfc7897>.

[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>.

[I-D.ietf-pce-pcep-yang] Dhody, D., Hardwick, J., Beeram, V., and J. Tantsura, "A YANG Data Model for Path Computation Element Communications Protocol (PCEP)", draft-ietf-pce-pcep-yang-08 (work in progress), June 2018.

[I-D.ietf-pce-stateful-hpce] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and O. Dios, "Hierarchical Stateful Path Computation Element (PCE).", draft-ietf-pce-stateful-hpce-05 (work in progress), June 2018.

Authors' Addresses

Fatai Zhang
Huawei
Huawei Base, Bantian, Longgang District
Shenzhen 518129
China

EMail: zhangfatai@huawei.com

Quintin Zhao
Huawei
125 Nagog Technology Park
Acton, MA 01719
USA

EMail: quintin.zhao@huawei.com

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid 28045
Spain

EMail: ogondio@tid.es

Ramon Casellas
CTTC
Av. Carl Friedrich Gauss n.7
Barcelona, Castelldefels
Spain

EMail: ramon.casellas@cttc.es

Daniel King
Old Dog Consulting
UK

EMail: daniel@olddog.co.uk

PCE Working Group
Internet-Draft
Intended status: Standards Track

Xian Zhang
Young Lee (Editor)
Fatai Zhang
Huawei
Ramon Casellas
CTTC
Oscar Gonzalez de Dios
Telefonica I+D
Zafar Ali
Cisco Systems

Expires: August 26, 2018 February 26, 2018

Path Computation Element (PCE) Protocol Extensions for Stateful PCE Usage in GMPLS-controlled Networks

draft-ietf-pce-pcep-stateful-pce-gmpls-08.txt

Abstract

The Path Computation Element (PCE) facilitates Traffic Engineering (TE) based path calculation in large, multi-domain, multi-region, or multi-layer networks. The PCE communication Protocol (PCEP) has been extended to support stateful PCE functions where the PCE retains information about the paths already present in the network, but those extensions are technology-agnostic. This memo provides extensions required for PCEP so as to enable the usage of a stateful PCE capability in GMPLS-controlled networks.

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 26, 2018.

Copyright Notice

Copyright (c) 2018 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.

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

[RFC4655] presents the architecture of a Path Computation Element (PCE)-based model for computing Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) Traffic Engineering Label Switched Paths (TE LSPs). To perform such a constrained computation, a PCE stores the network topology (i.e., TE links and nodes) and resource information (i.e., TE attributes) in its TE Database (TED). Such a PCE is usually referred as a stateless PCE. To request path computation services to a PCE, [RFC5440] defines the PCE communication Protocol (PCEP) for interaction between a Path Computation Client (PCC) and a PCE, or between two PCEs. PCEP as specified in [RFC 5440] mainly focuses on MPLS networks and the PCEP extensions needed for GMPLS-controlled networks are provided in [PCEP-GMPLS].

Stateful PCEs are shown to be helpful in many application scenarios, in both MPLS and GMPLS networks, as illustrated in [RFC8051]. Further discussion of concept of a stateful PCE can be found in [RFC7399]. In order for these applications to able to exploit the capability of stateful PCEs, extensions to PCEP are required.

[RFC8051] describes how a stateful PCE can be applicable to solve various problems for MPLS-TE and GMPLS networks and the benefits it brings to such deployments.

[RFC8231] provides the fundamental extensions needed for stateful PCE to support general functionality, but leaves out the specification for technology-specific objects/TLVs. This document

focuses on the extensions that are necessary in order for the deployment of stateful PCEs in GMPLS-controlled networks.

2. Context of Stateful PCE and PCEP for GMPLS

This document is built on the basis of Stateful PCE [RFC8231] and PCEP for GMPLS [PCEP-GMPLS].

There are two types of LSP operation for Stateful PCE.

For Active Stateful PCE, PCUpd message is sent from PCE to PCC to update the LSP state for the LSP delegated to PCE. Any changes to the delegated LSPs generate a PCRpt message by the PCC to PCE to convey the changes of the LSP. Any modifications to the Objects/TLVs that are identified in this document to support GMPLS technology-specific attributes will be carried in the PCRpt and PCUpd messages.

For Passive Stateful PCE where PCReq/PCRep messages are used to convey path computation instruction. As GMPLS-technology specific Objects/TLVs are defined in [PCEP-GMPLS], this document just points to the work in [PCEP-GMPLS] and add only the stateful PCE aspect only if applicable. Passive Stateful PCE makes use of PCRpt messages when reporting LSP State changes sent by PCC to PCEs. Any modifications to the Objects/TLVs that are identified in this document to support GMPLS technology-specific attributes will be carried in the PCRpt message.

[PCEP-GMPLS] defines GMPLS-technology specific Objects/TLVs and this document makes use of these Objects/TLVs without modifications where applicable. Some of these Objects/TLVs may require modifications to incorporate stateful PCE element where applicable.

3. Main Requirements

This section notes the main functional requirements for PCEP extensions to support stateful PCE for use in GMPLS-controlled networks, based on the description in [RFC8051]. Many requirements are common across a variety of network types (e.g., MPLS-TE networks and GMPLS networks) and the protocol extensions to meet the requirements are already described in [RFC8231]. This document does not repeat the description of those protocol extensions. This document presents protocol extensions for a set of requirements which are specific to the use of a stateful PCE in a GMPLS-controlled network.

The basic requirements are as follows:

- Advertisement of the stateful PCE capability. This generic requirement is covered in Section 5.4. of [RFC8231]. This document assumes that STATEFUL-PCE-CAPABILITY TLV can be used for GMPLS Stateful PCE capability and therefore does not provide any further extensions.

- LSP delegation is already covered in Section 5.7. of [RFC8231]. Section 2.2. of this document does not provide any further extensions.

- Active LSP update is covered in Section 6.2 of [RFC8231]. Section 4.1. of this document provides extension for its application in GMPLS-controlled networks.

- LSP state synchronization and LSP state report. This is a generic requirement already covered in Section 5.6. of [RFC8231]. However, there are further extensions required specifically for GMPLS-controlled networks and discussed in Section 4.2.

4. PCEP Extensions

4.1. LSP Update in GMPLS-controlled Networks

[RFC8231] defines the Path Computation LSP Update Request (PCUpd) message to enable to update the attributes of an LSP. However, that document does not define technology-specific parameters.

A key element of the PCUpd message is the attribute-list construct defined in [RFC5440] and extended by many other PCEP specifications.

For GMPLS purposes we note that the BANDWIDTH object used in the attribute-list is defined in [PCEP-GMPLS]. Furthermore, additional TLVs are defined for the LSPA object in [PCEP-GMPLS] and MAY be included to indicate technology-specific attributes. There are other technology-specific attributes that need to be conveyed in the <intended-attribute-list> of the <path> construct in the PCUpd message. Note that these path details in the PCUpd message are the same as the <attribute-list> of the PCRep message. See Section 4.2 for the details.

4.2. LSP Synchronization in GMPLS-controlled Networks

PCCs need to report the attributes of LSPs to the PCE to enable stateful operation of a GMPLS network. This process is known as LSP state synchronization. The LSP attributes include bandwidth, associated route, and protection information etc., are stored by the PCE in the LSP database (LSP-DB). Note that, as described in [RFC8231], the LSP state synchronization covers both the bulk

reporting of LSPs at initialization as well the reporting of new or modified LSP during normal operation. Incremental LSP-DB synchronization may be desired in a GMPLS-controlled network and it is specified in [RFC8232].

[RFC8231] describes mechanisms for LSP synchronization using the Path Computation State Report (PCRpt) message, but does not cover reporting of technology-specific attributes. As stated in [RFC8231], the <path> construct is further composed of a compulsory ERO object and a compulsory attribute-list and an optional RRO object. In order to report LSP states in GMPLS networks, this specification allows the use within a PCRpt message both of technology- and GMPLS-specific attribute objects and TLVs defined in [PCEP-GMPLS] as follows:

- IRO/XRO Extensions to support the inclusion/exclusion of labels and label sub-objects for GMPLS. (See Section 2.6 and 2.7 in [PCEP-GMPLS])

- END-POINTS (Generalized END-POINTS Object Type. See Section 2.5 in [PCEP-GMPLS])

- BANDWIDTH (Generalized BANDWIDTH Object Type. See Section 2.3 in [PCEP-GMPLS])

- LSPA (PROTECTION ATTRIBUTE TLV, See Section 2.8 in [PCEP-GMPLS].

The END-POINTS object SHOULD be carried within the attribute-list to specify the endpoints pertaining to the reported LSP. The XRO object MAY be carried to specify the network resources that the reported LSP avoids and a PCE SHOULD consider avoid these network resources during the process of re-optimizing after this LSP is delegated to the PCE. To be more specific, the <attribute-list> is updated as follows:

```
<attribute-list> ::= [<END-POINTS>]
                     [<LSPA>]
                     [<BANDWIDTH>]
                     [<metric-list>]
                     [<IRO>]
                     [<XRO>]

<metric-list>::= <METRIC>[<metric-list>]
```

If the LSP being reported protects another LSP, the PROTECTION-ATTRIBUTE TLV [PCEP-GMPLS] MUST be included in the LSPA object to describe its attributes and restrictions. Moreover, if the status of the protecting LSP changes from non-operational to operational, this SHOULD to be synchronized to the stateful PCE using a PCRpt message.

4.3. Modification of Existing PCEP Messages and Procedures

One of the advantages mentioned in [RFC8051] is that the stateful nature of a PCE simplifies the information conveyed in PCEP messages, notably between PCC and PCE, since it is possible to refer to PCE managed state for active LSPs. To be more specific, with a stateful PCE, it is possible to refer to an LSP with a unique identifier in the scope of the PCC-PCE session and thus use such identifier to refer to that LSP. Note this MAY also be applicable to packet networks.

4.3.1. Modification for LSP Re-optimization

The Request Parameters (RP) object on a Path Computation Request (PCReq) message carries the R bit. When set, this indicates that the PCC is requesting re-optimization of an existing LSP. Upon receiving such a PCReq, a stateful PCE SHOULD perform the re-optimization in the following cases:

- The existing bandwidth and route information of the LSP to be re-optimized is provided in the PCReq message using the BANDWIDTH object and the ERO.

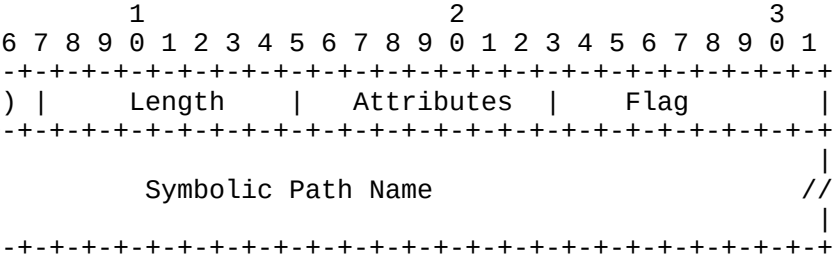
- The existing bandwidth and route information is not supplied in the PCReq message, but can be found in the PCE’s LSP-DB. In this case, the LSP MUST be identified using an LSP identifier carried in the PCReq message, and that fact requires that the LSP identifier was previously supplied either by the PCC in a PCRpt message or by the PCE in a PCRep message. [RFC8231] defines how this is achieved using a combination of the per-node LSP identifier (PLSP-ID) and the PCC’s address.

If no LSP state information is available to carry out re-optimization, the stateful PCE should report the error "LSP state information unavailable for the LSP re-optimization" (Error Type = TBD1, Error value= TBD2).

4.3.2. Modification for Route Exclusion

[RFC5521] defines a mechanism for a PCC to request or demand that specific nodes, links, or other network resources are excluded from paths computed by a PCE. A PCC may wish to request the computation of a path that avoids all link and nodes traversed by some other LSP.

To this end this document defines a new sub-object for use with route exclusion defined in [RFC5521]. The LSP exclusion sub-object is as follows:



X bit and Attribute fields are defined in [RFC5521].

X bit: indicates whether the exclusion is mandatory (X=1) and MUST be accommodated, or desired (X=0) and SHOULD be accommodated.

Type: Subobject Type for an LSP exclusion sub-object. Value of TBD3. To be assigned by IANA.

Length: The Length contains the total length of the subobject in bytes, including the Type and Length fields.

Attributes: indicates how the exclusion object is to be interpreted. Currently, Interface (Attributes = 0), Node (Attributes =1) and SRLG (Attributes =2) are defined in [RFC5521] and this document does not define new values.

Flags: This field may be used to further specify the exclusion constraint with regard to the LSP. Currently, no values are defined.

Symbolic Path Name: This is the identifier given to a LSP and is unique in the context of the PCC address as defined in [RFC8231].

Reserved: MUST be transmitted as zero and SHOULD be ignored on receipt.

This sub-object is OPTIONAL in the exclude route object (XRO) and can be present multiple times. When a stateful PCE receives a PCReq message carrying this sub-object, it SHOULD search for the identified LSP in its LSP-DB and then exclude it from the new path computation all resources used by the identified LSP. If the stateful PCE cannot recognize one or more of the received LSP identifiers, it should send an error message PCErr reporting "The LSP state information for route exclusion purpose cannot be found" (Error-type = TBD1, Error-value = TBD4). Optionally, it may provide with the unrecognized identifier information to the requesting PCC using the error reporting techniques described in [RFC5440].

4.3.3. Modification for SRP Object to indicate Bi-directional LSP

The format of the SRP object is defined in [RFC8231] and included here for easy reference with the addition of the new B flag. This SRP object is used in PCUpd and PCInit messages for GMPLS.

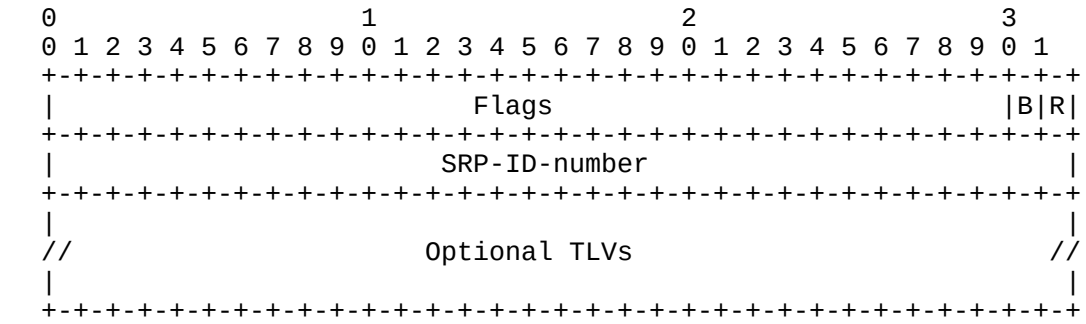


Figure 3: The SRP Object Format

A new flag is defined to indicate a bidirectional co-routed LSP setup operation initiated by the PCE:

B (Bidirectional LSP -- 1 bit): If set to 0, it indicates a request to create a uni-directional LSP. If set to 1, it indicates a request to create a bidirectional co-routed LSP.

4.4. Object Encoding

Note that, as is stated in Section 7 of [RFC8231], the P flag and the I flag of the PCEP objects used on PCUpd and PCRpt messages

SHOULD be set to 0 on transmission and SHOULD be ignored on receipt since these flags are exclusively related to path computation requests.

5. IANA Considerations

IANA is requested to allocate new Types for the TLV/Object defined in this document.

5.1. New PCEP Error Codes

IANA is requested to make the following allocation in the "PCEP-ERROR Object Error Types and Values" registry.

Error Type	Meaning	Reference
TBD1	LSP state information missing	[This.I-D]
Error-value TBD2:	LSP state information unavailable for the LSP re-optimization	[This.I-D]
Error-value TBD4:	LSP state information for route exclusion purpose cannot be found	[This.I-D]

5.2. New Subobject for the Exclude Route Object

IANA maintains the "PCEP Parameters" registry containing a subregistry called "PCEP Objects". This registry has a subregistry for the XRO (Exclude Route Object) listing the sub-objects that can be carried in the XRO. IANA is requested to assign a further sub-object that can be carried in the XRO as follows:

Value	Description	Reference
TBD3	LSP identifier sub-object	[This.I-D]

5.3. New "B" Flag in the SRP Object

IANA has created a new subregistry, named "SRP Object Flag Field", 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 is

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
TDB	Bi-directional co-routed LSP	[This.I-D]

6. Manageability Considerations

The description and functionality specifications presented related to stateful PCEs should also comply with the manageability specifications covered in Section 8 of [RFC4655]. Furthermore, a further list of manageability issues presented in [RFC8231] should also be considered.

Additional considerations are presented in the next sections.

6.1. Requirements on Other Protocols and Functional Components

When the detailed route information is included for LSP state synchronization (either at the initial stage or during LSP state report process), this requires the ingress node of an LSP carry the RRO object in order to enable the collection of such information.

7. Security Considerations

This draft provides additional extensions to PCEP so as to facilitate stateful PCE usage in GMPLS-controlled networks, on top of [RFC8231]. The PCEP extensions to support GMPLS-controlled networks should be considered under the same security as for MPLS networks, as noted in [RFC7025]. Therefore, the security considerations elaborated in [RFC5440] still apply to this draft. Furthermore, [RFC8231] provides a detailed analysis of the additional security issues incurred due to the new extensions and possible solutions needed to support for the new stateful PCE capabilities and they apply to this document as well.

8. Acknowledgement

We would like to thank Adrian Farrel and Cyril Margaria for the useful comments and discussions.

9. References

9.1. 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.

[RFC8231] Crabbe, E., Medved, J., Varga, R., Minei, I., "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, September 2017.

[PCEP-GMPLS] Margaria, C., Gonzalez de Dios, O., Zhang, F., "PCEP extensions for GMPLS", draft-ietf-pce-gmpls-pcep-extensions, work in progress.

9.2. Informative References

[RFC8051] Zhang, X., Minei, I., et al, "Applicability of Stateful Path Computation Element (PCE) ", RFC 8051, January 2017.

[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, September 2017.

10. Contributors' Address

Dhruv Dhody
Huawei Technology
India

EMail: dhruv.ietf@gmail.com

Yi Lin
Huawei Technologies
F3-5-B R&D Center, Huawei Base
Bantian, Longgang District
Shenzhen 518129 P.R.China

Phone: +86-755-28972914
Email: yi.lin@huawei.com

Authors' Addresses

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

Phone: +86-755-28972645
Email: zhang.xian@huawei.com

Young Lee (Editor)
Huawei
5340 Legacy Drive, Suite 170
Plano, TX 75023
US

Phone: +1 469 278 5838
EMail: leeyoung@huawei.com

Fatai Zhang
Huawei
F3-5-B R&D Center, Huawei Base
Bantian, Longgang District
P.R. China

Phone: +86-755-28972912
Email: zhangfatai@huawei.com

Ramon Casellas
CTTC
Av. Carl Friedrich Gauss n7
Castelldefels, Barcelona 08860
Spain

Phone:
Email: ramon.casellas@cttc.es

Oscar Gonzalez de Dios
Telefonica Investigacion y Desarrollo
Emilio Vargas 6
Madrid, 28045
Spain

Phone: +34 913374013

Email: ogondio@tid.es

Zafar Ali
Cisco Systems
Email: zali@cisco.com

PCE Working Group
Internet-Draft
Intended status: Informational
Expires: September 6, 2018

D. Dhody
Y. Lee
Huawei Technologies
D. Ceccarelli
Ericsson
J. Shin
SK Telecom
D. King
Lancaster University
O. Gonzalez de Dios
Telefonica I+D

March 5, 2018

Hierarchical Stateful Path Computation Element (PCE). draft-ietf-pce-stateful-hpce-04

Abstract

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

The Hierarchical Path Computation Element (H-PCE) architecture, provides an architecture to allow the optimum sequence of inter-connected domains to be selected, and network policy to be applied if applicable, via the use of a hierarchical relationship between PCEs.

Combining the capabilities of Stateful PCE and the Hierarchical PCE would be advantageous. This document describes general considerations and use cases for the deployment of Stateful PCE(s) using the Hierarchical PCE architecture.

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."

Copyright Notice

Copyright (c) 2018 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

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.

A stateful PCE is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

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

[RFC8231] also describes the active stateful PCE. The active PCE functionality allows a PCE to reroute an existing LSP or make changes to the attributes of an existing LSP, or delegate control of specific LSPs to a new PCE.

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 motivation for PCE development. [RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs). Within the Hierarchical PCE (H-PCE) architecture [RFC6805], the Parent PCE (P-PCE) is used to compute a multi-domain path based on the domain connectivity information. A Child PCE (C-PCE) may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its domain topology information.

This document presents general considerations for stateful PCE(s) in hierarchical PCE architecture. In particular, the behavior changes and additions to the existing stateful PCE mechanisms (including PCE-

initiated LSP setup and active PCE usage) in the context of networks using the H-PCE architecture.

The initial section of the document focuses on end to end (E2E) inter-domain TE LSP. Section 3.3.1 describe the operations for the Per Domain LSP that could be stitched.

1.1. Requirements Language

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

2. Terminology

The terminology is as per [RFC4655], [RFC5440], [RFC6805], [RFC8231], and [RFC8281].

3. Hierarchical Stateful PCE

As described in [RFC6805], in the hierarchical PCE architecture, a P-PCE maintains a domain topology map that contains the child domains (seen as vertices in the topology) and their interconnections (links in the topology). The P-PCE has no information about the content of the child domains. Each child domain has at least one PCE capable of computing paths across the domain. These PCEs are known as C-PCEs and have a direct relationship with the P-PCE. The P-PCE builds the domain topology map either via direct configuration (allowing network policy to also be applied) or from learned information received from each C-PCE.

[RFC8231] 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 H-PCE Architecture from a C-PCE towards a P-PCE (CE-PE) or from a P-PCE towards a C-PCE (PE-CE). All PCE types herein (i.e., PE or CE) are assumed to be 'stateful PCE'.

A number of interactions are expected in the Hierarchical Stateful PCE architecture, these include:

LSP State Report (CE-PE): a child stateful PCE sends an LSP state report to a Parent Stateful PCE whenever the state of a LSP

changes.

LSP State Synchronization (CE-PE): after the session between the Child and Parent stateful PCEs is initialized, the P-PCE must learn the state of C-PCE's TE LSPs.

LSP Control Delegation (CE-PE,PE-CE): a C-PCE grants to the P-PCE the right to update LSP attributes on one or more LSPs; the C-PCE may withdraw the delegation or the P-PCE may give up the delegation at any time.

LSP Update Request (PE-CE): a stateful P-PCE requests modification of attributes on a C-PCE's TE LSP.

PCE LSP Initiation Request (PE-CE): a stateful P-PCE requests C-PCE to initiate a TE LSP.

Note that this hierarchy is recursive and thus a Label Switching Router (LSR), as a PCC could delegate the control to a PCE, which may delegate to its parent, which may further delegate it to its parent (if it exist or needed). Similarly update operations could also be applied recursively.

[I-D.ietf-pce-hierarchy-extensions] defines the H-PCE capability TLV that should be used in the OPEN message to advertise the H-PCE capability. [RFC8231] defines the stateful PCE capability TLV. The presence of both TLVs represent the support for stateful H-PCE operations as described in this document.

[I-D.litkowski-pce-state-sync] describes the procedures to allow a stateful communication between PCEs for various use-cases. The procedures and extensions as described in Section 3 of [I-D.litkowski-pce-state-sync] are also applicable to Child and Parent PCE communication.

3.1. Passive Operations

Procedures as described in [RFC6805] are applied, where the ingress C-PCE sends a request to the P-PCE. The P-PCE selects a set of candidate domain paths based on the domain topology and the state of the inter-domain links. It then sends computation requests to the C-PCEs responsible for each of the domains on the candidate domain paths. Each C-PCE computes a set of candidate path segments across its domain and sends the results to the P-PCE. The P-PCE uses this information to select path segments and concatenate them to derive the optimal end-to-end inter-domain path. The end-to-end path is then sent to the C-PCE that received the initial path request, and this C-PCE passes the path on to the PCC that issued the original

request.

As per [RFC8231], PCC sends an LSP State Report carried on a PCRpt message to the C-PCE, indicating the LSP's status. The C-PCE MAY further propagate the State Report to the P-PCE. A local policy at C-PCE MAY dictate which LSPs to be reported to the P-PCE. The PCRpt message is sent from C-PCE to P-PCE.

State synchronization mechanism as described in [RFC8231] and [RFC8232] are applicable to PCEP session between C-PCE and P-PCE as well.

Taking the sample hierarchical domain topology example from [RFC6805] as the reference topology for the entirety of this document.

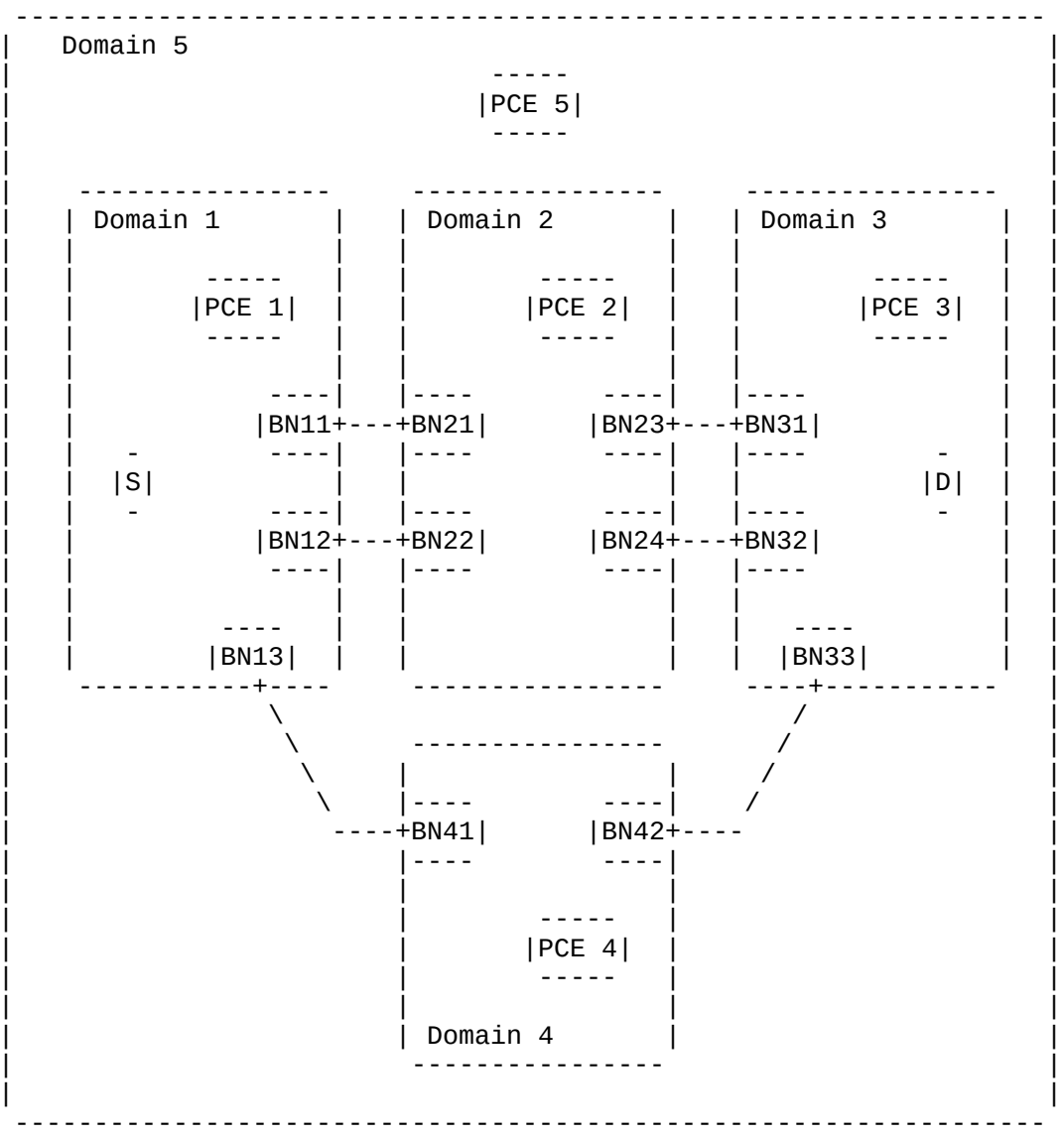


Figure 1: Sample Hierarchical Domain Topology

Steps 1 to 11 are exactly as described in section 4.6.2 (Hierarchical PCE End-to-End Path Computation Procedure) of [RFC6805], the following additional steps are added for stateful PCE:

(1) The Ingress LSR initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(2) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(3) The Ingress LSR notifies the LSP state to PCE1 when the state is "UP".

(4) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

3.2. Active Operations

[RFC8231] describes the case of active stateful PCE. The active PCE functionality uses two specific PCEP messages:

o Update Request (PCUpd)
o State Report (PCRpt)

The first is sent by the PCE to a Path Computation Client (PCC) for modifying LSP attributes. The PCC sends back a PCRpt to acknowledge the requested operation or report any change in LSP's state.

As per [RFC8051], Delegation is an operation to grant a PCE, temporary rights to modify a subset of LSP parameters on one or more PCC's LSPs. The C-PCE may further choose to delegate to P-PCE based on a local policy. The PCRpt message with "D" (delegate) flag is sent from C-PCE to P-PCE.

To update an LSP, a PCE send to the PCC, an LSP Update Request using a PCUpd message. For LSP delegated to the P-PCE via the child PCE, the P-PCE can use the same PCUpd message to request change to the C-PCE (the Ingress domain PCE), the PCE further propagates the update request to the PCC.

The P-PCE uses the same mechanism described in Section 3.1 to compute the end to end path using PCReq and PCRep messages.

The following additional steps are also initially performed, for active operations, again using the reference architecture described in Figure 1 (Sample Hierarchical Domain Topology).

(1) The Ingress LSR delegates the LSP to the PCE1 via PCRpt message with D flag set.

(2) The PCE1 further delegates the LSP to the P-PCE (PCE5).

Steps 4 to 10 of section 4.6.2 of [RFC6805] are executed to determine the end to end path.

(3) The P-PCE (PCE5) sends the update request to the C-PCE (PCE1) via PCUpd message.

(4) The PCE1 further updates the LSP to the Ingress LSR (PCC).

(5) The Ingress LSR initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(6) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(7) The Ingress LSR notifies the LSP state to PCE1 when the state is "UP".

(8) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

3.3. PCE Initiation Operation

[RFC8281] 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. To instantiate or delete an LSP, the PCE sends the Path Computation LSP Initiate Request (PCInitiate) message to the PCC. In case of inter-domain LSP in Hierarchical PCE architecture, the initiation operations can be carried out at the P-PCE. In which case after P-PCE finishes the E2E path computation, it can send the PCInitiate message to the C-PCE (the Ingress domain PCE), the PCE further propagates the initiate request to the PCC.

The following additional steps are also initially performed, for PCE initiated operations, again using the reference architecture described in Figure 1 (Sample Hierarchical Domain Topology):

(1) The P-PCE (PCE5) is requested to initiate a LSP.

Steps 4 to 10 of section 4.6.2 of [RFC6805] are executed to determine the end to end path.

(2) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE1) via PCInitiate message.

(3) The PCE1 further propagates the initiate message to the Ingress LSR (PCC).

(4) The Ingress LSR initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(5) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(6) The Ingress LSR notifies the LSP state to PCE1 when the state is "UP".

(7) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

3.3.1. Per Domain Stitched LSP

The Hierarchical PCE architecture as per [RFC6805] is primarily used for E2E LSP. With PCE-Initiated capability, another mode of operation is possible, where multiple intra-domain LSPs are initiated in each domain which are further stitched to form an E2E LSP. The P-PCE sends PCInitiate message to each C-PCE separately to initiate individual LSP segments along the domain path. These individual per domain LSP are stitched together by some mechanism, which is out of scope of this document (Refer [I-D.dugeon-pce-stateful-interdomain]).

The following additional steps are also initially performed, for the Per Domain stiched LSP operation, again using the reference architecture described in Figure 1 (Sample Hierarchical Domain Topology):

(1) The P-PCE (PCE5) is requested to initiate a LSP.

Steps 4 to 10 of section 4.6.2 of [RFC6805] are executed to determine the end to end path, which are broken into per-domain LSPs say -

- S-BN41
- BN41-BN33
- BN33-D

It should be noted that the P-PCE MAY use other mechanisms to determine the suitable per-domain LSPs (apart from [RFC6805]).

For LSP (BN33-D)

(2) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE3) via PCInitiate message for LSP (BN33-D).

(3) The PCE3 further propagates the initiate message to BN33.

(4) BN33 initiates the setup of the LSP as per the path and reports

to the PCE3 the LSP status ("GOING-UP").

(5) The PCE3 further reports the status of the LSP to the P-PCE (PCE5).

(6) The node BN33 notifies the LSP state to PCE3 when the state is "UP".

(7) The PCE3 further reports the status of the LSP to the P-PCE (PCE5).

For LSP (BN41-BN33)

(8) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE4) via PCInitiate message for LSP (BN41-BN33).

(9) The PCE4 further propagates the initiate message to BN41.

(10) BN41 initiates the setup of the LSP as per the path and reports to the PCE4 the LSP status ("GOING-UP").

(11) The PCE4 further reports the status of the LSP to the P-PCE (PCE5).

(l2) The node BN41 notifies the LSP state to PCE4 when the state is "UP".

(13) The PCE4 further reports the status of the LSP to the P-PCE (PCE5).

For LSP (S-BN41)

(14) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE1) via PCInitiate message for LSP (S-BN41).

(15) The PCE1 further propagates the initiate message to node S.

(16) S initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(17) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(18) The node S notifies the LSP state to PCE1 when the state is "UP".

(19) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

Additionally:

(20) Once P-PCE receives report of each per-domain LSP, it should use suitable stitching mechanism, which is out of scope of this document. In this step, P-PCE (PCE5) could also initiate an E2E LSP (S-D) by sending the PCInitiate message to Ingress C-PCE (PCE1). It is also possible to stitch the per-domain LSP at the same time as the per-domain LSPs are initiated as defined in [I-D.dugeon-pce-stateful-interdomain].

4. Other Considerations

4.1. Applicability to Inter-Layer

[RFC5623] describes a framework for applying the PCE-based architecture to inter-layer (G)MPLS traffic engineering. The H-PCE Stateful architecture with stateful P-PCE coordinating with the stateful C-PCEs of higher and lower layer is shown in the figure below.

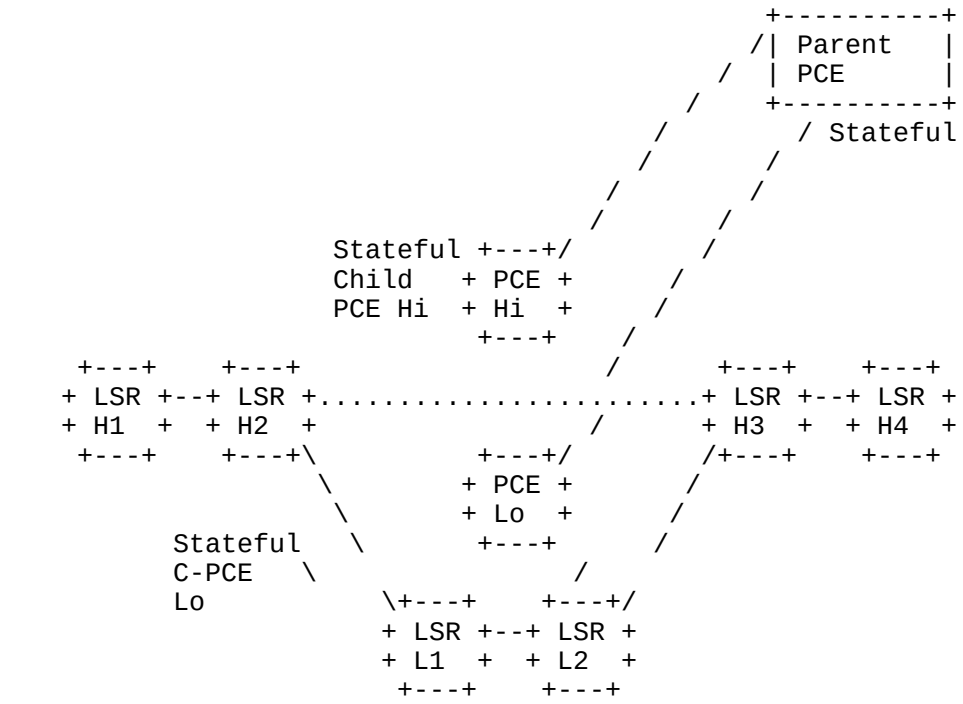


Figure 2: Sample Inter-Layer Topology

All procedures described in Section 3 are applicable to inter-layer

path setup as well.

4.2. Applicability to ACTN

[I-D.ietf-teas-actn-framework] describes framework for Abstraction and Control of TE Networks (ACTN), where each Provisioning Network Controller (PNC) is equivalent to C-PCE and P-PCE is the Multi-Domain Service Coordinator (MDSC). The Per domain stitched LSP as per the Hierarchical PCE architecture described in Section 3.3.1 and Section 4.1 is well suited for ACTN.

[I-D.ietf-pce-applicability-actn] examines the applicability of PCE to the ACTN framework. To support the function of multi domain coordination via hierarchy, the stateful hierarchy of PCEs plays a crucial role.

In ACTN framework, Customer Network Controller (CNC) can request the MDSC to check if there is a possibility to meet Virtual Network (VN) requirements (before requesting for VN provision). The H-PCE architecture as described in [RFC6805] can supports via the use of PCReq and PCRep messages between the P-PCE and C-PCEs.

5. Other Considerations

5.1. Scalability Considerations

It should be noted that if all the C-PCEs would report all the LSPs in their domain, it could lead to scalability issues for the P-PCE. Thus it is recommended to only report the LSPs which are involved in H-PCE, i.e. the LSPs which are either delegated to the P-PCE or initiated by the P-PCE. Scalability considerations for PCEP as per [RFC8231] continue to apply for the PCEP session between child and parent PCE.

5.2. Confidentiality

As described in section 4.2 of [RFC6805], information about the content of child domains is not shared for both scaling and confidentiality reasons. Along with the confidentiality during path computation, the child PCE could also conceal the path information, a C-PCE may replace a path segment with a path-key [RFC5520], effectively hiding the content of a segment of a path.

6. Security Considerations

The security considerations listed in [RFC8231],[RFC6805] and [RFC5440] apply to this document as well. As per [RFC6805], it is

expected that the parent PCE will require all child PCEs to use full security when communicating with the parent.

Any multi-domain operation necessarily involves the exchange of information across domain boundaries. This is bound to represent a significant security and confidentiality risk especially when the child domains are controlled by different commercial concerns. PCEP allows individual PCEs to maintain confidentiality of their domain path information using path-keys [RFC5520], and the hierarchical PCE architecture is specifically designed to enable as much isolation of domain topology and capabilities information as is possible. The LSP state in the PCRpt message SHOULD continue to use this.

The security consideration for PCE-Initiated LSP as per [RFC8281] is also applicable from P-PCE to C-PCE.

Thus securing the PCEP session (between the P-PCE and the C-PCE) using mechanism like TCP Authentication Option (TCP-AO) [RFC5925] or Transport Layer Security (TLS) [RFC8253] is RECOMMENDED.

7. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440], [RFC6805], [RFC8231], and [RFC8281] apply to Stateful H-PCE defined in this document. In addition, requirements and considerations listed in this section apply.

7.1. Control of Function and Policy

Support of the hierarchical procedure will be controlled by the management organization responsible for each child PCE. The parent PCE must only accept path computation requests from authorized child PCEs. If a parent PCE receives report from an unauthorized child PCE, the report should be dropped. All mechanism as described in [RFC8231] and [RFC8281] continue to apply.

7.2. Information and Data Models

An implementation SHOULD allow the operator to view the stateful and H-PCE capabilities advertised by each peer. The PCEP YANG module [I-D.ietf-pce-pcep-yang] can be extended to include details stateful H-PCE.

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] and [RFC8231].

7.5. Requirements On Other Protocols

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

7.6. Impact On Network Operations

Mechanisms defined in [RFC5440] and [RFC8231] also apply to PCEP extensions defined in this document.

The stateful H-PCE technique brings the applicability of stateful PCE as described in [RFC8051], for the LSP traversing multiple domains.

8. IANA Considerations

There are no IANA considerations.

9. Acknowledgments

Thanks to Manuela Scarella, Haomian Zheng, Sergio Marmo, Stefano Parodi, Giacomo Agostini, Jeff Tantsura and Rajan Rao for suggestions.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://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, <http://www.rfc-editor.org/info/rfc5440>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the

Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <http://www.rfc-editor.org/info/rfc6805>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

10.2. Informative References

[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>.

[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, <http://www.rfc-editor.org/info/rfc5520>.

[RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 5623, DOI 10.17487/RFC5623, September 2009, <http://www.rfc-editor.org/info/rfc5623>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <http://www.rfc-editor.org/info/rfc5925>.

[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>.

[RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <https://www.rfc-editor.org/info/rfc8232>.

[RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <https://www.rfc-editor.org/info/rfc8253>.

[I-D.ietf-teas-actn-framework] Ceccarelli D. and Y. Lee, "Framework for Abstraction and Control of Transport Networks", draft-ietf-teas-actn-framework-11 (work in progress), October 2017.

[I-D.ietf-pce-applicability-actn] Dhody, D., Lee, Y., and D. Ceccarelli, "Applicability of Path Computation Element (PCE) for Abstraction and Control of TE Networks (ACTN)", draft-ietf-pce-applicability-actn-03 (work in progress), March 2018.

[I-D.litkowski-pce-state-sync] Litkowski, S., Sivabalan, S., and D. Dhody, "Inter Stateful Path Computation Element communication procedures", draft-litkowski-pce-state-sync-02 (work in progress), August 2017.

[I-D.ietf-pce-hierarchy-extensions] Zhang, F., Zhao, Q., Dios, O., Casellas, R., and D. King, "Extensions to Path Computation Element Communication Protocol (PCEP) for Hierarchical Path Computation Elements (PCE)", draft-ietf-pce-hierarchy-extensions-03 (work in progress), July 2016.

[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-06 (work in progress), January 2018.

[I-D.dugeon-pce-stateful-interdomain] Dugeon, O. and J. Meuric, "PCEP Extension for Stateful Inter-Domain Tunnels", draft-dugeon-pce-stateful-interdomain-00 (work in progress), October 2017.

Appendix A. Contributor Addresses

Avantika
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: s.avantika.avantika@gmail.com

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

EMail: zhang.xian@huawei.com

Udayasree Palle
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: udayasreereddy@gmail.com

Authors' Addresses

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

EMail: dhruv.ietf@gmail.com

Young Lee
Huawei Technologies
5340 Legacy Drive, Building 3
Plano, TX 75023
USA

EMail: leeyoung@huawei.com

Daniele Ceccarelli
Ericsson

Torshamnsgatan,48
Stockholm
Sweden

EMail: daniele.ceccarelli@ericsson.com

Jongyoon Shin
SK Telecom
6 Hwangsaeul-ro, 258 beon-gil, Bundang-gu, Seongnam-si,
Gyeonggi-do 463-784
Republic of Korea

EMail: jongyoon.shin@sk.com

Daniel King
Lancaster University
UK

EMail: d.king@lancaster.ac.uk

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid, 28045
Spain

Phone: +34913128832
Email: ogondio@tid.es

PCE Working Group
Internet-Draft
Intended status: Informational
Expires: December 20, 2018

D. Dhody
Y. Lee
Huawei Technologies
D. Ceccarelli
Ericsson
J. Shin
SK Telecom
D. King
Lancaster University
O. Gonzalez de Dios
Telefonica I+D
June 18, 2018

Hierarchical Stateful Path Computation Element (PCE). draft-ietf-pce-stateful-hpce-05

Abstract

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

The Hierarchical Path Computation Element (H-PCE) architecture, provides an architecture to allow the optimum sequence of inter-connected domains to be selected, and network policy to be applied if applicable, via the use of a hierarchical relationship between PCEs.

Combining the capabilities of Stateful PCE and the Hierarchical PCE would be advantageous. This document describes general considerations and use cases for the deployment of Stateful PCE(s) using the Hierarchical PCE architecture.

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."

Copyright Notice

Copyright (c) 2018 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

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.

A stateful PCE is capable of considering, for the purposes of path computation, not only the network state in terms of links and nodes (referred to as the Traffic Engineering Database or TED) but also the status of active services (previously computed paths, and currently reserved resources, stored in the Label Switched Paths Database (LSP-DB).

[RFC8051] describes general considerations for a stateful PCE deployment and examines its applicability and benefits, as well as its challenges and limitations through a number of use cases.

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

[RFC8231] also describes the active stateful PCE. The active PCE functionality allows a PCE to reroute an existing LSP or make changes to the attributes of an existing LSP, or delegate control of specific LSPs to a new PCE.

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 motivation for PCE development. [RFC6805] describes a Hierarchical PCE (H-PCE) architecture which can be used for computing end-to-end paths for inter-domain MPLS Traffic Engineering (TE) and GMPLS Label Switched Paths (LSPs). Within the Hierarchical PCE (H-PCE) architecture [RFC6805], the Parent PCE (P-PCE) is used to compute a multi-domain path based on the domain connectivity information. A Child PCE (C-PCE) may be responsible for a single domain or multiple domains, it is used to compute the intra-domain path based on its domain topology information.

This document presents general considerations for stateful PCE(s) in hierarchical PCE architecture. In particular, the behavior changes and additions to the existing stateful PCE mechanisms (including PCE-

initiated LSP setup and active PCE usage) in the context of networks using the H-PCE architecture.

The initial section of the document focuses on end to end (E2E) inter-domain TE LSP. Section 3.3.1 describe the operations for the Per Domain LSP that could be stitched.

1.1. Requirements Language

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

2. Terminology

The terminology is as per [RFC4655], [RFC5440], [RFC6805], [RFC8231], and [RFC8281].

3. Hierarchical Stateful PCE

As described in [RFC6805], in the hierarchical PCE architecture, a P-PCE maintains a domain topology map that contains the child domains (seen as vertices in the topology) and their interconnections (links in the topology). The P-PCE has no information about the content of the child domains. Each child domain has at least one PCE capable of computing paths across the domain. These PCEs are known as C-PCEs and have a direct relationship with the P-PCE. The P-PCE builds the domain topology map either via direct configuration (allowing network policy to also be applied) or from learned information received from each C-PCE.

[RFC8231] 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 H-PCE Architecture from a C-PCE towards a P-PCE (CE-PE) or from a P-PCE towards a C-PCE (PE-CE). All PCE types herein (i.e., PE or CE) are assumed to be 'stateful PCE'.

A number of interactions are expected in the Hierarchical Stateful PCE architecture, these include:

LSP State Report (CE-PE): a child stateful PCE sends an LSP state report to a Parent Stateful PCE whenever the state of a LSP

changes.

LSP State Synchronization (CE-PE): after the session between the Child and Parent stateful PCEs is initialized, the P-PCE must learn the state of C-PCE's TE LSPs.

LSP Control Delegation (CE-PE,PE-CE): a C-PCE grants to the P-PCE the right to update LSP attributes on one or more LSPs; the C-PCE may withdraw the delegation or the P-PCE may give up the delegation at any time.

LSP Update Request (PE-CE): a stateful P-PCE requests modification of attributes on a C-PCE's TE LSP.

PCE LSP Initiation Request (PE-CE): a stateful P-PCE requests C-PCE to initiate a TE LSP.

Note that this hierarchy is recursive and thus a Label Switching Router (LSR), as a PCC could delegate the control to a PCE, which may delegate to its parent, which may further delegate it to its parent (if it exist or needed). Similarly update operations could also be applied recursively.

[I-D.ietf-pce-hierarchy-extensions] defines the H-PCE capability TLV that should be used in the OPEN message to advertise the H-PCE capability. [RFC8231] defines the stateful PCE capability TLV. The presence of both TLVs represent the support for stateful H-PCE operations as described in this document.

[I-D.litkowski-pce-state-sync] describes the procedures to allow a stateful communication between PCEs for various use-cases. The procedures and extensions as described in Section 3 of [I-D.litkowski-pce-state-sync] are also applicable to Child and Parent PCE communication. The SPEAKER-IDENTITY-TLV (defined in [RFC8232]) is included in the LSP object to identify the Ingress (PCC). The PLSP-ID used in the forwarded PCRpt by the C-PCE to P-PCE is same as the original one used by the PCC.

3.1. Passive Operations

Procedures as described in [RFC6805] are applied, where the ingress C-PCE sends a request to the P-PCE. The P-PCE selects a set of candidate domain paths based on the domain topology and the state of the inter-domain links. It then sends computation requests to the C-PCEs responsible for each of the domains on the candidate domain paths. Each C-PCE computes a set of candidate path segments across its domain and sends the results to the P-PCE. The P-PCE uses this

information to select path segments and concatenate them to derive the optimal end-to-end inter-domain path. The end-to-end path is then sent to the C-PCE that received the initial path request, and this C-PCE passes the path on to the PCC that issued the original request.

As per [RFC8231], PCC sends an LSP State Report carried on a PCRpt message to the C-PCE, indicating the LSP's status. The C-PCE MAY further propagate the State Report to the P-PCE. A local policy at C-PCE MAY dictate which LSPs to be reported to the P-PCE. The PCRpt message is sent from C-PCE to P-PCE.

State synchronization mechanism as described in [RFC8231] and [RFC8232] are applicable to PCEP session between C-PCE and P-PCE as well.

Taking the sample hierarchical domain topology example from [RFC6805] as the reference topology for the entirety of this document.

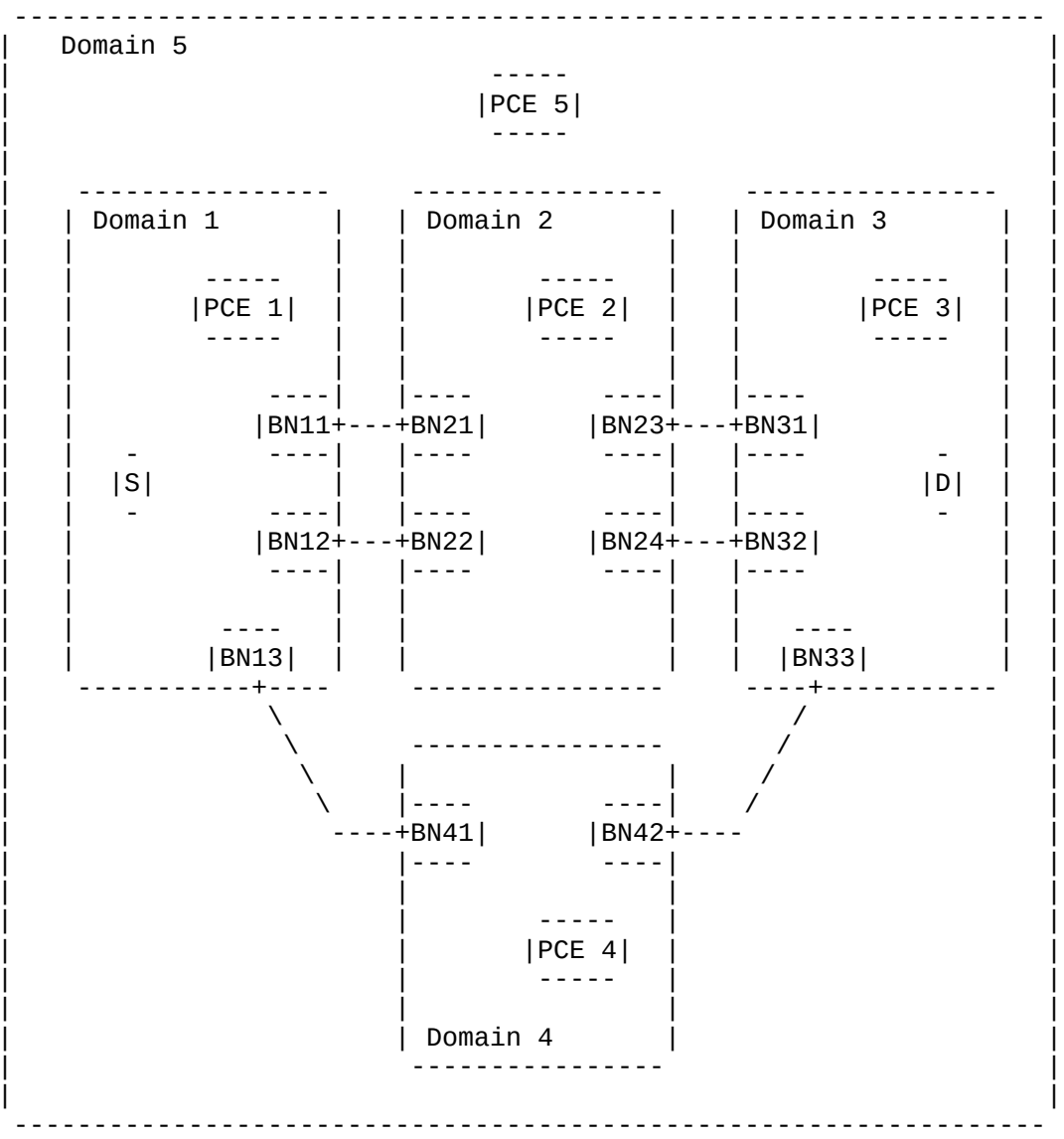


Figure 1: Sample Hierarchical Domain Topology

Steps 1 to 11 are exactly as described in section 4.6.2 (Hierarchical PCE End-to-End Path Computation Procedure) of [RFC6805], the following additional steps are added for stateful PCE:

(1) The Ingress LSR initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(2) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(3) The Ingress LSR notifies the LSP state to PCE1 when the state is "UP".

(4) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

The Ingress LSR could trigger path re-optimization by sending the path computation request as described in [RFC6805], at this time it can include the LSP object in the PCReq message as described in [RFC8231].

3.2. Active Operations

[RFC8231] describes the case of active stateful PCE. The active PCE functionality uses two specific PCEP messages:

o Update Request (PCUpd)
o State Report (PCRpt)

The first is sent by the PCE to a Path Computation Client (PCC) for modifying LSP attributes. The PCC sends back a PCRpt to acknowledge the requested operation or report any change in LSP’s state.

As per [RFC8051], Delegation is an operation to grant a PCE, temporary rights to modify a subset of LSP parameters on one or more PCC’s LSPs. The C-PCE may further choose to delegate to P-PCE based on a local policy. The PCRpt message with "D" (delegate) flag is sent from C-PCE to P-PCE.

To update an LSP, a PCE send to the PCC, an LSP Update Request using a PCUpd message. For LSP delegated to the P-PCE via the child PCE, the P-PCE can use the same PCUpd message to request change to the C-PCE (the Ingress domain PCE), the PCE further propagates the update request to the PCC.

The P-PCE uses the same mechanism described in Section 3.1 to compute the end to end path using PCReq and PCRep messages.

The following additional steps are also initially performed, for active operations, again using the reference architecture described in Figure 1 (Sample Hierarchical Domain Topology).

(1) The Ingress LSR delegates the LSP to the PCE1 via PCRpt message with D flag set.

(2) The PCE1 further delegates the LSP to the P-PCE (PCE5).

Steps 4 to 10 of section 4.6.2 of [RFC6805] are executed to determine the end to end path.

(3) The P-PCE (PCE5) sends the update request to the C-PCE (PCE1) via PCUpd message.

(4) The PCE1 further updates the LSP to the Ingress LSR (PCC).

(5) The Ingress LSR initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(6) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(7) The Ingress LSR notifies the LSP state to PCE1 when the state is "UP".

(8) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

3.3. PCE Initiation Operation

[RFC8281] 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. To instantiate or delete an LSP, the PCE sends the Path Computation LSP Initiate Request (PCInitiate) message to the PCC. In case of inter-domain LSP in Hierarchical PCE architecture, the initiation operations can be carried out at the P-PCE. In which case after P-PCE finishes the E2E path computation, it can send the PCInitiate message to the C-PCE (the Ingress domain PCE), the PCE further propagates the initiate request to the PCC.

The following additional steps are also initially performed, for PCE initiated operations, again using the reference architecture described in Figure 1 (Sample Hierarchical Domain Topology):

(1) The P-PCE (PCE5) is requested to initiate a LSP.

Steps 4 to 10 of section 4.6.2 of [RFC6805] are executed to determine the end to end path.

(2) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE1) via PCInitiate message.

(3) The PCE1 further propagates the initiate message to the Ingress LSR (PCC).

(4) The Ingress LSR initiates the setup of the LSP as per the path and reports to the PCE1 the LSP status ("GOING-UP").

(5) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(6) The Ingress LSR notifies the LSP state to PCE1 when the state is "UP".

(7) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

The Ingress LSR (PCC) generates the PLSP-ID for the LSP and inform the C-PCE, which is propagated to the P-PCE as described in [I-D.litkowski-pce-state-sync]

3.3.1. Per Domain Stitched LSP

The Hierarchical PCE architecture as per [RFC6805] is primarily used for E2E LSP. With PCE-Initiated capability, another mode of operation is possible, where multiple intra-domain LSPs are initiated in each domain which are further stitched to form an E2E LSP. The P-PCE sends PCInitiate message to each C-PCE separately to initiate individual LSP segments along the domain path. These individual per domain LSP are stitched together by some mechanism, which is out of scope of this document (Refer [I-D.dugeon-pce-stateful-interdomain]).

The following additional steps are also initially performed, for the Per Domain stitched LSP operation, again using the reference architecture described in Figure 1 (Sample Hierarchical Domain Topology):

(1) The P-PCE (PCE5) is requested to initiate a LSP.

Steps 4 to 10 of section 4.6.2 of [RFC6805] are executed to determine the end to end path, which are broken into per-domain LSPs say -

- S-BN41
- BN41-BN33
- BN33-D

It should be noted that the P-PCE MAY use other mechanisms to

determine the suitable per-domain LSPs (apart from [RFC6805]).

For LSP (BN33-D)

(2) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE3) via PCInitiate message for LSP (BN33-D).

(3) The PCE3 further propagates the initiate message to BN33.

(4) BN33 initiates the setup of the LSP as per the path and reports to the PCE3 the LSP status ("GOING-UP").

(5) The PCE3 further reports the status of the LSP to the P-PCE (PCE5).

(6) The node BN33 notifies the LSP state to PCE3 when the state is "UP".

(7) The PCE3 further reports the status of the LSP to the P-PCE (PCE5).

For LSP (BN41-BN33)

(8) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE4) via PCInitiate message for LSP (BN41-BN33).

(9) The PCE4 further propagates the initiate message to BN41.

(10) BN41 initiates the setup of the LSP as per the path and reports to the PCE4 the LSP status ("GOING-UP").

(11) The PCE4 further reports the status of the LSP to the P-PCE (PCE5).

(l2) The node BN41 notifies the LSP state to PCE4 when the state is "UP".

(13) The PCE4 further reports the status of the LSP to the P-PCE (PCE5).

For LSP (S-BN41)

(14) The P-PCE (PCE5) sends the initiate request to the child PCE (PCE1) via PCInitiate message for LSP (S-BN41).

(15) The PCE1 further propagates the initiate message to node S.

(16) S initiates the setup of the LSP as per the path and reports to

the PCE1 the LSP status ("GOING-UP").

(17) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

(18) The node S notifies the LSP state to PCE1 when the state is "UP".

(19) The PCE1 further reports the status of the LSP to the P-PCE (PCE5).

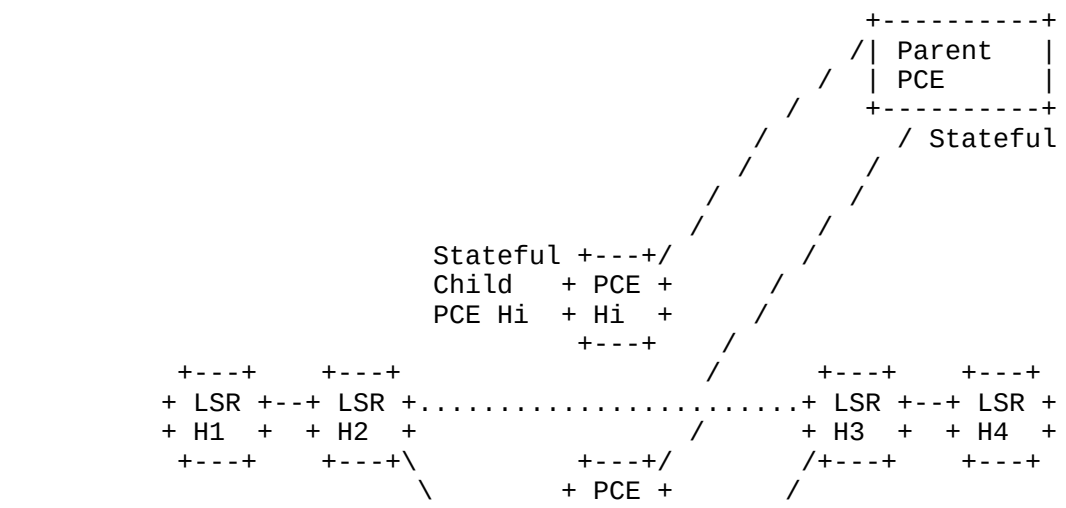
Additionally:

(20) Once P-PCE receives report of each per-domain LSP, it should use suitable stitching mechanism, which is out of scope of this document. In this step, P-PCE (PCE5) could also initiate an E2E LSP (S-D) by sending the PCInitiate message to Ingress C-PCE (PCE1). It is also possible to stitch the per-domain LSP at the same time as the per-domain LSPs are initiated as defined in [I-D.dugeon-pce-stateful-interdomain].

4. Other Considerations

4.1. Applicability to Inter-Layer

[RFC5623] describes a framework for applying the PCE-based architecture to inter-layer (G)MPLS traffic engineering. The H-PCE Stateful architecture with stateful P-PCE coordinating with the stateful C-PCEs of higher and lower layer is shown in the figure below.



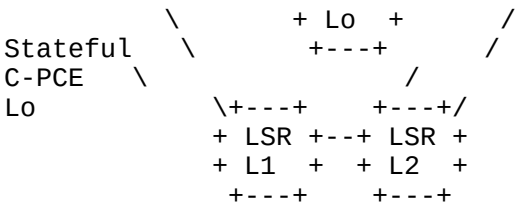


Figure 2: Sample Inter-Layer Topology

All procedures described in Section 3 are applicable to inter-layer path setup as well.

4.2. Applicability to ACTN

[I-D.ietf-teas-actn-framework] describes framework for Abstraction and Control of TE Networks (ACTN), where each Provisioning Network Controller (PNC) is equivalent to C-PCE and P-PCE is the Multi-Domain Service Coordinator (MDSC). The Per domain stitched LSP as per the Hierarchical PCE architecture described in Section 3.3.1 and Section 4.1 is well suited for ACTN.

[I-D.ietf-pce-applicability-actn] examines the applicability of PCE to the ACTN framework. To support the function of multi domain coordination via hierarchy, the stateful hierarchy of PCEs plays a crucial role.

In ACTN framework, Customer Network Controller (CNC) can request the MDSC to check if there is a possibility to meet Virtual Network (VN) requirements (before requesting for VN provision). The H-PCE architecture as described in [RFC6805] can supports via the use of PCReq and PCRep messages between the P-PCE and C-PCEs.

5. Other Considerations

5.1. Scalability Considerations

It should be noted that if all the C-PCEs would report all the LSPs in their domain, it could lead to scalability issues for the P-PCE. Thus it is recommended to only report the LSPs which are involved in H-PCE, i.e. the LSPs which are either delegated to the P-PCE or initiated by the P-PCE. Scalability considerations for PCEP as per [RFC8231] continue to apply for the PCEP session between child and parent PCE.

5.2. Confidentiality

As described in section 4.2 of [RFC6805], information about the

content of child domains is not shared for both scaling and confidentiality reasons. Along with the confidentiality during path computation, the child PCE could also conceal the path information, a C-PCE may replace a path segment with a path-key [RFC5520], effectively hiding the content of a segment of a path.

6. Security Considerations

The security considerations listed in [RFC8231],[RFC6805] and [RFC5440] apply to this document as well. As per [RFC6805], it is expected that the parent PCE will require all child PCEs to use full security when communicating with the parent.

Any multi-domain operation necessarily involves the exchange of information across domain boundaries. This is bound to represent a significant security and confidentiality risk especially when the child domains are controlled by different commercial concerns. PCEP allows individual PCEs to maintain confidentiality of their domain path information using path-keys [RFC5520], and the hierarchical PCE architecture is specifically designed to enable as much isolation of domain topology and capabilities information as is possible. The LSP state in the PCRpt message SHOULD continue to use this.

The security consideration for PCE-Initiated LSP as per [RFC8281] is also applicable from P-PCE to C-PCE.

Thus securing the PCEP session (between the P-PCE and the C-PCE) using mechanism like TCP Authentication Option (TCP-AO) [RFC5925] or Transport Layer Security (TLS) [RFC8253] is RECOMMENDED.

7. Manageability Considerations

All manageability requirements and considerations listed in [RFC5440], [RFC6805], [RFC8231], and [RFC8281] apply to Stateful H-PCE defined in this document. In addition, requirements and considerations listed in this section apply.

7.1. Control of Function and Policy

Support of the hierarchical procedure will be controlled by the management organization responsible for each child PCE. The parent PCE must only accept path computation requests from authorized child PCEs. If a parent PCE receives report from an unauthorized child PCE, the report should be dropped. All mechanism as described in [RFC8231] and [RFC8281] continue to apply.

7.2. Information and Data Models

An implementation SHOULD allow the operator to view the stateful and H-PCE capabilities advertised by each peer. The PCEP YANG module [I-D.ietf-pce-pcep-yang] can be extended to include details stateful H-PCE.

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] and [RFC8231].

7.5. Requirements On Other Protocols

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

7.6. Impact On Network Operations

Mechanisms defined in [RFC5440] and [RFC8231] also apply to PCEP extensions defined in this document.

The stateful H-PCE technique brings the applicability of stateful PCE as described in [RFC8051], for the LSP traversing multiple domains.

8. IANA Considerations

There are no IANA considerations.

9. Acknowledgments

Thanks to Manuela Scarella, Haomian Zheng, Sergio Marmo, Stefano Parodi, Giacomo Agostini, Jeff Tantsura and Rajan Rao for suggestions.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate

Requirement Levels", BCP 14, RFC 2119,
DOI 10.17487/RFC2119, March 1997,
<http://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, <http://www.rfc-editor.org/info/rfc5440>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <http://www.rfc-editor.org/info/rfc6805>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

10.2. Informative References

[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>.

[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, <http://www.rfc-editor.org/info/rfc5520>.

[RFC5623] Oki, E., Takeda, T., Le Roux, JL., and A. Farrel, "Framework for PCE-Based Inter-Layer MPLS and GMPLS Traffic Engineering", RFC 5623, DOI 10.17487/RFC5623,

September 2009, <http://www.rfc-editor.org/info/rfc5623>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <http://www.rfc-editor.org/info/rfc5925>.

[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>.

[RFC8232] Crabbe, E., Minei, I., Medved, J., Varga, R., Zhang, X., and D. Dhody, "Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE", RFC 8232, DOI 10.17487/RFC8232, September 2017, <https://www.rfc-editor.org/info/rfc8232>.

[RFC8253] Lopez, D., Gonzalez de Dios, O., Wu, Q., and D. Dhody, "PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)", RFC 8253, DOI 10.17487/RFC8253, October 2017, <https://www.rfc-editor.org/info/rfc8253>.

[I-D.ietf-teas-actn-framework]
Ceccarelli D. and Y. Lee, "Framework for Abstraction and Control of Transport Networks", draft-ietf-teas-actn-framework-15 (work in progress), May 2018.

[I-D.ietf-pce-applicability-actn]
Dhody, D., Lee, Y., and D. Ceccarelli, "Applicability of Path Computation Element (PCE) for Abstraction and Control of TE Networks (ACTN)", draft-ietf-pce-applicability-actn-06 (work in progress), June 2018.

[I-D.litkowski-pce-state-sync]
Litkowski, S., Sivabalan, S., and D. Dhody, "Inter Stateful Path Computation Element communication procedures", draft-litkowski-pce-state-sync-03 (work in progress), April 2018.

[I-D.ietf-pce-hierarchy-extensions]
Zhang, F., Zhao, Q., Dios, O., Casellas, R., and D. King, "Extensions to Path Computation Element Communication Protocol (PCEP) for Hierarchical Path Computation Elements (PCE)", draft-ietf-pce-hierarchy-extensions-05 (work in progress), June 2018.

[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-08 (work in progress), June 2018.

[I-D.dugeon-pce-stateful-interdomain]
Dugeon, O. and J. Meuric, "PCEP Extension for Stateful Inter-Domain Tunnels", draft-dugeon-pce-stateful-interdomain-00 (work in progress), October 2017.

Appendix A. Contributor Addresses

Avantika
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: s.avantika.avantika@gmail.com

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

EMail: zhang.xian@huawei.com

Udayasree Palle
Huawei Technologies
Divyashree Techno Park, Whitefield
Bangalore, Karnataka 560066
India

EMail: udayasreereddy@gmail.com

Authors' Addresses

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

EMail: dhruv.ietf@gmail.com

Young Lee
Huawei Technologies
5340 Legacy Drive, Building 3
Plano, TX 75023
USA

EMail: leeyoung@huawei.com

Daniele Ceccarelli
Ericsson

Torshamnsgatan,48
Stockholm
Sweden

EMail: daniele.ceccarelli@ericsson.com

Jongyoon Shin
SK Telecom
6 Hwangsaeul-ro, 258 beon-gil, Bundang-gu, Seongnam-si,
Gyeonggi-do 463-784
Republic of Korea

EMail: jongyoon.shin@sk.com

Daniel King
Lancaster University
UK

EMail: d.king@lancaster.ac.uk

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid, 28045
Spain

Phone: +34913128832
Email: ogondio@tid.es

PCE Working Group
Internet Draft
Intended status: Standard Track
Expires: August 2018

Francesco Lazzeri
Daniele Ceccarelli
Ericsson
Young Lee
Dhruv Dhody
Huawei

February 26, 2018

Extensions to the Path Computation Element Protocol (PCEP) for residual path bandwidth support

draft-lazzeri-pce-residual-bw-01

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 26, 2018.

Copyright Notice

Copyright (c) 2018 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

The PCEP protocol has objective functions to optimize path attributes like the residual bandwidth. While this is enough for some applications, it’s not possible to return the computed values of such attributes to the PCC, or put bounds on them.

This document describes extensions to the PCE Communication Protocol (PCEP) providing new path-related bandwidth metrics allowing a PCE to compute paths taking into account and returning to the PCC information about the remaining bandwidth along the computed paths.

Table of Contents

Introduction

The objective of this document is to define an extension to the PCEP [RFC5440] providing information about the bandwidth still available for future reservations on a given path, that is the minimum unreserved bandwidth and the minimum residual bandwidth among all the links of that path.
This is not a new concept to PCEP. In [RFC5541] two objective functions are defined, called minimum load path (MLP) and maximum residual bandwidth path (MBP). Both of them allow to find paths with optimal value of bandwidth-related metrics, defined on a per-link basis, considering the links traversed by that path.
For example, the residual bandwidth of a path is defined as the minimum value of the residual bandwidth on each link in the path. Specifying that OF inside the SVEC object of a PCReq message, the PCE tries and finds the path with the maximum value of the path residual bandwidth.
Unfortunately, being an objective function, MBP can only be used to find a path that optimizes the residual bandwidth, but its value cannot be returned for a path computed with some other objectives (and also when MBP itself is used), or used as a bound.
The same applies to the unreserved bandwidth. The difference between residual and unreserved bandwidth is well described in [RFC7471]:

> "The calculation of Residual Bandwidth is different than that of Unreserved Bandwidth [RFC3630]. Residual Bandwidth subtracts tunnel reservations from Maximum Bandwidth (i.e., the link capacity) [RFC3630] and provides an aggregated remainder across priorities. Unreserved Bandwidth, on the other hand, is subtracted from the Maximum Reservable Bandwidth (the bandwidth that can theoretically be reserved) and provides per priority remainders. Residual Bandwidth and Unreserved Bandwidth [RFC3630] can be used concurrently, and each has a separate use case (e.g., the former can be used for applications like Weighted ECMP while the latter can be used for call admission control)".

Having this information would allow a PCC to reuse a path resulting from a path computation to route additional LSPs without requesting new path computations(with the same end-points and constraints), until the maximum path unreserved bandwidth is taken (or a path deployment fails).

1. Requirements for managing the residual bandwidth as a metric

Path computation with optimization of the load or of the residual bandwidth has been defined as important objective functions in [RFC5541].

Managing the unreserved bandwidth (related to the load) and the residual bandwidth of a path as additional metrics, adds the capability to return their value, or putting a bound on their value. This is an added value in distributed PCE applications, like e.g. in ACTN architecture [ACTN-FW] and [PCE-APP]. The following associated key requirements are identified for PCEP:

1. A PCE supporting this draft MUST have the capability to compute end-to-end (E2E) paths with either unreserved bandwidth or with residual bandwidth constraints. It MUST also support the combination of these new constraints with existing constraints, like IGP metric, TE metric, hop limit, and network performance constraints as defined in [RFC5440] and [PCEP-SERV-AWARE].

2. A PCC MUST be able to specify either unreserved bandwidth or residual bandwidth constraints in a Path Computation Request (PCReq) message to be applied during the path computation.

3. A PCC MUST be able to request that a PCE optimizes a path using either unreserved bandwidth or residual bandwidth as objective metric.

4. A PCE that supports this specification is not required to provide unreserved bandwidth or residual bandwidth path computation to any PCC at any time.

   Therefore, it MUST be possible for a PCE to reject a PCReq message with reason codes that indicate unreserved bandwidth or residual bandwidth is not supported. Furthermore, a PCE that does not support this specification will either ignore or reject such requests using pre-existing mechanisms, therefore the requests MUST be identifiable to legacy PCEs and rejections by legacy PCEs MUST be acceptable within this specification.

5. A PCE that supports this specification MUST be able to return unreserved or residual bandwidth information of the computed path in a Path Computation Reply (PCRep) message.

2. New metrics definition

2.1. Link and Path Unreserved bandwidth

The unreserved bandwidth of a link is the bandwidth available for future allocation on the link at a given priority, that is the difference between the Maximum Reservable Bandwidth of the link and total bandwidth used on that link by LSPs with priority equal or lower (higher value) than the specified priority. In order to define the path unreserved bandwidth, the following concepts and notation need to be introduced:

- A network comprises of a set of N links {Li, (i=1...N)}.
- A path of a point to point (P2P) LSP is a list of K links {Lpi,(i=1...K)}.
- The maximum reservable bandwidth of the link Li, named Ri.
- The bandwidth allocated to LSPs at priority p on the link Li is the sum of the bandwidth of all the LSPs passing through the link Li with priority >= p, named Bi(p).
- The unreserved bandwidth at priority p of the link Li is Ui(p) = Ri - Bi(p)

The path unreserved bandwidth at a given priority k is defined as the minimum value of the unreserved bandwidth at priority k among all the links along the P2P path. Specifically, extending on the above mentioned terminology:

- Path unreserved bandwidth metric at priority is defined as:
  PU(p) = min {Ui(p), (i=1...K)}

2.2. Link and Path Residual bandwidth

The residual bandwidth of a link is the bandwidth physically left free for future allocation on the link. In order to define the path residual bandwidth, the following concepts and notation need to be introduced:

- A network comprises of a set of N links {Li, (i=1...N)}.

- A path of a point to point (P2P) LSP is a list of K links {Lpi,(i=1...K)}
- The maximum bandwidth of the link Li, named Bi.
- The sum of the bandwidth of all the LSPs passing through the link Li, that is the bandwidth allocated on the link, named Ai.
- The residual bandwidth of the link Li is r(i) = Bi - Ai.

The path residual bandwidth is defined as the minimum value of the residual bandwidth among all the links along the P2P path. Specifically, extending on the above mentioned terminology:

- Path residual bandwidth metric for the P2P path is defined as: PB = min {r(Lpi), (i=1...K)}

3. PCEP protocol extensions

This section defines PCEP extensions to fulfill the requirements outlined in Section 2. The proposed solution is used to support path unreserved bandwidth and path residual bandwidth as additional metrics of the PCEP protocol.
The METRIC object is defined in section 7.8 of [RFC5440], comprising metric-value, metric-type (T field) and a flags field comprising a number of bit-flags.

This document defines two new types for the METRIC object:

T = TBD1: Path Unreserved Bandwidth

When the T field is set to TBD1, the value of the metric-value field is set to the Path Unreserved Bandwidth for the traffic type and priority requested in the PCReq message.
The same format used by [RFC5440] for the BANDWIDTH object body is used here to represent the value of a path unreserved bandwidth bound or returned value, as shown in the following:

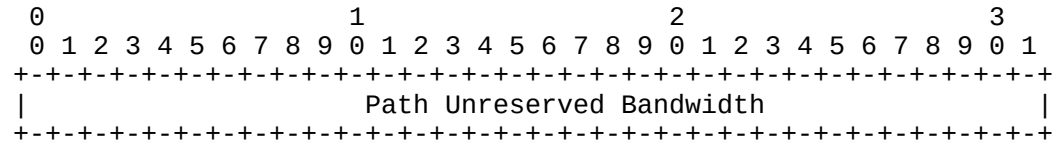


Figure 1: PATH UNRESERVED BANDWIDTH value format

Path Unreserved Bandwidth (32 bits): The path unreserved bandwidth is encoded in 32 bits in IEEE floating point format (see [IEEE.754.1985]), expressed in bytes per second.

The PATH UNRESERVED BANDWIDTH value has a fixed length of 4 bytes.

T = TBD2: Path Residual Bandwidth

When the T field is set to TBD2, the value of the metric-value field is set to the Path Residual Bandwidth for the traffic type requested in the PCReq message.

When the T field is set to TBD2, the value of the metric-value field is set to the Path Residual Bandwidth for the traffic type requested in the PCReq message.

The same format used by [RFC5440] for the BANDWIDTH object body is used here to represent the value of a path residual bandwidth bound or returned value, as shown in the following:

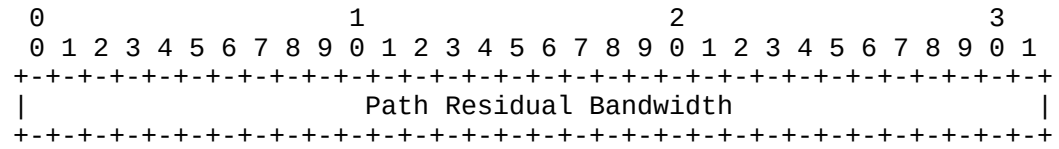


Figure 1: PATH RESIDUAL BANDWIDTH value format

Path Residual Bandwidth (32 bits): The path residual bandwidth is encoded in 32 bits in IEEE floating point format (see [IEEE.754.1985]), expressed in bytes per second.

The PATH RESIDUAL BANDWIDTH value has a fixed length of 4 bytes.

Editor NOTE: these definitions provide support only of PSC signal type. For other signal types (e.g. ODU, WDM) these fields can be filled with the number of unreserved or residual fixed containers (e.g. 3 ODU0) related to the type of traffic specified in the PCReq. This has to be discussed.

A PCC MAY use the path unreserved or residual bandwidth in a PCReq message to request a path meeting the end to end unreserved or residual bandwidth requirement. In this case, the B bit MUST be set to suggest a bound (a minimum) for the path residual bandwidth metric that must be guaranteed for the PCC to consider the computed path as acceptable. The path unreserved or residual bandwidth metrics must be greater than or equal to the value specified in the metric-value field.
The P bit MAY be set to specify the constraint as mandatory, or MAY be left cleared to specify the bound as optional.

A PCC can also use this metric to ask PCE to optimize (that is maximize) the path residual bandwidth during path computation.
In this case, the B bit MUST be cleared.

A PCE MAY use the path residual bandwidth metric in a PCRep message along with a NO-PATH object in the case where the PCE cannot compute a path meeting this constraint.
A PCE can also use this metric to send the computed path residual bandwidth metric to the PCC.

4. Non-Understanding/Non-Support Residual Bandwidth

If a PCE receives a PCReq message containing a METRIC object with type PATH UNRESERVED BANDWIDTH or PATH RESIDUAL BANDWIDTH and the PCE does not understand or support those metric types, and the P bit is clear in the METRIC object header then the PCE SHOULD simply ignore the METRIC object as per the processing specified in [RFC5440].

If the PCE does not understand the new METRIC types, and the P bit is set in the METRIC object header, then the PCE MUST send a PCErr message containing a PCEP-ERROR Object with Error-Type = 4

(Not supported object) and Error-value = 4 (Unsupported parameter) [RFC5440][RFC5541].

If the PCE understands but does not support the new METRIC type, and the P bit is set in the METRIC object header, then the PCE MUST send a PCErr message containing a PCEP-ERROR Object with Error-Type = 4 (Not supported object) with Error-value = TBD3 (Unsupported path unreserved bandwidth constraint) or TBD4 (Unsupported path residual bandwidth constraint).
The path computation request MUST then be cancelled.

If the PCE understands the new METRIC type, but the local policy has been configured on the PCE to not allow network performance constraint, and the P bit is set in the METRIC object header, then the PCE MUST send a PCErr message containing a PCEP-ERROR Object with Error-Type = 5 (Policy violation) with Error-value = TBD5 (Not Allowed path unreserved bandwidth constraint) or TBD6 (Not Allowed path residual bandwidth constraint). The path computation request MUST then be cancelled.

4.1. Mode of Operation

As explained in [RFC5440], the METRIC object is optional and can be used for several purposes. In a PCReq message, a PCC MAY insert one or more METRIC objects:

- To indicate the metric (path unreserved or path residual bandwidth) that MUST be optimized by the path computation algorithm.

- To indicate a bound on the METRIC (path unreserved or path residual bandwidth) that MUST NOT be exceeded for the path to be considered as acceptable by the PCC.

In a PCRep message, the PCE MAY insert the METRIC object with an Explicit Route Object (ERO) so as to provide the METRIC (residual bandwidth) for the computed path.
The PCE MAY also insert the METRIC object with a NO-PATH object to indicate that the metric constraint could not be satisfied.

The path computation algorithmic aspects used by the PCE to optimize a path with respect to a specific metric are outside the scope of this document.

All the rules of processing the METRIC object as explained in [RFC5440] are applicable to the new metric types as well.

5. Procedures

The new metrics defined in this document don’t add or change the procedures already defined for PCEP protocol in [RFC5440] and [RFC5541].

In particular, the existing objective function MBP is still usable as appropriate, being equivalent to the usage of the Path Residual Bandwidth metric with the B bit cleared.

The new metric can be used to define new procedures especially in the scope of SDN and ACTN, which are out of the scope of this document.

5.1. Use cases

The first use case is the application of the residual bandwidth to simplify the computation of an end-to-end path across a multi-domain network.
The ability of a hierarchy of PCEs to compute accurate end-to-end paths across multiple domains is recognized as an important requirement in many applications.
In particular, this is a key requirement for networks with a centralized path computation function (e.g. hierarchical PCE or SDN. In such scenarios, a hierarchy of PCEs is often implemented, where, as illustrated in [RFC6805], a parent H-PCE coordinates the operations of a set of child (domain) PCEs in order to compute end-to-end paths across the network.
An H-PCE (either stateful or stateless) can make the best of residual bandwidth metrics, using paths from erstwhile path computations to deploy multiple LSPs (having the same end-points and constraints) without additional requests, until either the remaining In a hierarchical architecture of PCEs, domain PCEs just know the topology of their domains, while the parent PCE has in general detailed information about the managed domains and the relevant inter-domain links, but not necessarily enough information about the internals of each domain, so that it’s capable to compute accurately an end-to-end path.

The residual bandwidth information would also be beneficial for implementing abstractions of the domain topology, building the

abstract connectivity incrementally, based only on really used constraints, as soon as path computation results are returned.
One of the key features of SDN is the support of network abstraction, that is, as described in [RFC7926], the capability of applying policy to a set of information about a network, in order to produce selective information that represents the potential ability to connect across the domain.
The process of abstraction produces a connectivity graph, which can be used by the parent PCE to compute an accurate path based on the abstracted topology. The main issue is that the connectivity graph can be huge, depending on the size of the domain topology and the number of end-points defined on the edge of the domain.
One way to provide similar information is to store the result of path computations requested to the child PCEs (performed by e.g. TE-tunnels "compute only") and try reusing them if possible to save further path computation iterations between parent and child PCEs.
In any case a selection of path computation constraints has to be defined against the abstract topology in order to reduce the number of the abstract links or TE-tunnels exported by the connectivity graph, as it’s impractical to compute or pre-compute all the constraints combinations. It’s also very important to reduce the number of updates of such connectivity information to the parent PCE in order not to flood it with a continuous stream of updates.

6. IANA considerations

6.1. METRIC types

IANA maintains the "Path Computation Element Protocol (PCEP) Numbers" at <http://www.iana.org/assignments/pcep>. Within this registry IANA maintains one sub-registry for "METRIC object T field".
Two new metric types are defined in this document for the METRIC object (specified in [RFC5440]).

IANA is requested to make the following allocations:

Value	Description	Reference
TBD1	Path unreserved bandwidth metric	[This I.D.]

TBD2	Path residual bandwidth metric	[This I.D.]

6.2. New Error-Values

IANA maintains a registry of Error-Types and Error-values for use in PCEP messages. This is maintained as the "PCEP-ERROR Object Error Types and Values" sub-registry of the "Path Computation Element Protocol (PCEP) Numbers" registry.

IANA is requested to make the following allocations:

Four new Error-values are defined for the Error-Type "Not supported object" (type 4) and "Policy violation" (type 5).

Error-Type	Meaning and error values	Reference
4	Not supported object	
	Error-value=TBD3 Unsupported Path unreserved bandwidth constraint	[This I.D.]
	Error-value=TBD4 Unsupported Path residual bandwidth constraint	
5	Policy violation	
	Error-value=TBD5 Not allowed Path unreserved bandwidth constraint	[This I.D.]
	Error-value=TBD6 Not allowed Path residual bandwidth constraint	

7. Security Considerations

This document defines new METRIC types, which do not add any new security concerns beyond those discussed in [RFC5440] and [RFC5541] in itself.

In some scenarios, path unreserved bandwidth and path residual bandwidth information could be considered sensitive and could be used to influence path computation and setup with adverse effect.

Snooping of PCEP messages with such data, or using PCEP messages for network reconnaissance, may give an attacker sensitive information about the capabilities of the network. Thus, such deployment should employ suitable PCEP security mechanisms like TCP Authentication Option (TCP-AO) [RFC5925] or [PCEPS].
The Transport Layer Security (TLS) based procedure in [PCEPS] is considered as a security enhancement and thus much better suited for the sensitive residual bandwidth information.

8. References

8.1. Normative References

[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>.

[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, <http://www.rfc-editor.org/info/rfc5541>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <http://www.rfc-editor.org/info/rfc5925>.

[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>.

[PCEPS] Lopez, D.Lopez, D., Dios, O., Wu, W., and D. Dhody, "Secure Transport for PCEP", draft-ietf-pce-pceps-11 (work in progress), January 2017.

[IEEE.754.1985] IEEE, "Standard for Binary Floating-Point Arithmetic", IEEE 754, August 1985

8.2. Informative References

[RFC6805] King, D., Ed., and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, November 2012, <http://www.rfc-editor.org/info/rfc6805>.

[RFC7471] Giacalone S., Ward D., Drake J., Atlas A. and S. Previdi, "OSPF Traffic Engineering (TE) Metric Extensions", RFC 7471, DOI 10.17487/RFC7471, March 2015, <http://www.rfc-editor.org/info/rfc7471>

[RFC7926] Farrel, A. et al., "Problem Statement and Architecture for Information Exchange Between Interconnected Traffic Engineered Networks", RFC 7926, July 2016.

[ACTN-FW] Ceccarelli, D. and Y. Lee, "Framework for Abstraction and Control of Traffic Engineered Networks", draft-ietf-teas-actn-framework-03 (work in progress), February 2017.

[PCE-APP] Dhody, D. Lee, Y. Ceccarelli, D. "Applicability of Path Computation Element (PCE) for Abstraction and Control of TE Networks (ACTN)" draft-dhody-pce-applicability-actn-02

9. Contributors

Authors' Addresses

Francesco Lazzeri
Ericsson
Via Melen 77
Genova - Italy
Email: francesco.lazzeri@ericsson.com

Daniele Ceccarelli
Ericsson AB
Gronlandsgatan 21
Kista - Sweden
Email: daniele.ceccarelli@ericsson.com

Young Lee
Huawei Technologies
5340 Legacy Drive
Plano, TX 75023, USA
Phone: (469)277-5838
Email: leeyoung@huawei.com

Dhruv Dhody
Huawei Technologies,
Divyashree Technopark, Whitefield
Bangalore, India
Email: dhruv.ietf@gmail.com

Intellectual Property Statement

The IETF Trust takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in any IETF Document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights.

Copies of Intellectual Property disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or

users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement any standard or specification contained in an IETF Document. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

All IETF Documents and the information contained therein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION THEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

MPLS Working Group
Internet-Draft
Intended status: Informational
Expires: March 7, 2019

L. Andersson
Bronze Dragon Consulting
S. Bryant
A. Malis
Huawei Technologies
N. Leymann
Deutsche Telekom
G. Swallow
Independent

September 3, 2018

Deprecating MD5 for LDP
draft-nslag-mpls-deprecate-md5-03

Abstract

When the MPLS Label Distribution Protocol (LDP) was specified circa 1999, there were very strong requirements that LDP should use a cryptographic hash function to sign LDP protocol messages. MD5 was widely used at that time, and was the obvious choices.

However, even when this decision was being taken there were concerns as to whether MD5 was a strong enough signing option. This discussion was briefly reflected in section 5.1 of RFC 5036 [RFC5036] (and also in RFC 3036 [RFC3036]).

Over time it has been shown that MD5 can be compromised. Thus, there is a concern shared in the security community and the working groups responsible for the development of the LDP protocol that LDP is no longer adequately secured.

This document deprecates MD5 as the signing method for LDP messages. The document also selects a future method to secure LDP messages - the choice is TCP-AO. In addition, we specify that the TBD cryptographic mechanism is to be the default TCP-AO security method.

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 7, 2019.

Copyright Notice

Copyright (c) 2018 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

RFC 3036 was published in January 2001 as a Proposed Standard, and it was replaced by RFC 5035, which is a Draft Standard, in October 2007. Two decades after LDP was originally specified there is a concern shared by the security community and the IETF working groups that develop the LDP protocol that LDP is no longer adequately secured.

LDP currently uses MD5 to cryptographically sign its messages for security security purposes. However, MD5 is a hash function that is no longer considered adequate to meet current security requirements.

1.1. Requirement Language

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

2. Background

2.1. LDP in RFC 5036

In Section 5.1 "Spoofing" of RFC 5036 [RFC5036], in list item 2 "Session communication carried by TCP" the following statements are made:

> LDP specifies use of the TCP MD5 Signature Option to provide for the authenticity and integrity of session messages.
>
> RFC 2385 [RFC2385] asserts that MD5 authentication is now considered by some to be too weak for this application. It also points out that a similar TCP option with a stronger hashing algorithm (it cites SHA-1 as an example) could be deployed. To our knowledge, no such TCP option has been defined and deployed. However, we note that LDP can use whatever TCP message digest techniques are available, and when one stronger than MD5 is specified and implemented, upgrading LDP to use it would be relatively straightforward.

2.2. MD5 in BGP

There has been a similar discussion among working groups developing the BGP protocol. BGP has already replaced MD5 with TCP-AO. This was specified in RFC 7454 [RFC7454].

To secure LDP the same approach will be followed, TCP-AO will be used for LDP also.

As far as we are able to ascertain, there is currently no recommended, mandatory to implement, cryptographic function specified. We are concerned that without such a mandatory function, implementations will simply fall back to MD5 and nothing will really be changed. The MPLS working group will need the expertise of the

security community to specify a viable security function that is suitable for wide scale deployment on existing network platforms.

2.3. Prior Art

RFC 6952 [RFC6952] dicusses a set of routing protocols that all are using TCP for transport of protocol messages, according to guidelines set forth in Section 4.2 of "Keying and Authentication for Routing Protocols Design Guidelines", RFC 6518 [RFC6518].

RFC 6952 takes a much broader approach than this document, it discusses several protcols and also securing the LDP session initialization. This document has a narrower scope, securing LDP session messages only. LDP in initialization mode is addressed in RFC 7349 [RFC7349].

RFC 6952 and this document, basically suggest the same thing, move to TCP-AO and deploy a strong cryotoigraphic algorithm.

All the protcols discuseed in RFC 6952 should adopt the approach to securing protocol messages over TCP.

3. Securing LDP

Implementations conforming to this RFC MUST implement TCP-AO to secure the TCP sessions carrying LDP in addition to the currently required TCP MD5 Signature Option.

A TBD cryptographic mechanism must be implemented and provided to TCP-AO to secure LDP messages.

The TBD mechanism is the preferred option, and MD5 SHOULD only to be used when TBD is unavailable.

Note: The authors are not experts on this part of the stack, but it seems that TCP security negotiation is still work in progress. If we are wrong, then we need to include a requirement that such negotiation is also required. In the absence of a negotiation protocol, however, we need to leave this as a configuration process until such time as the negotiation protocol work is complete. On completion of a suitable negotiation protocol we need to issue a further update requiring its use.

Cryptographic mechanisms do not have an indefinite lifetime, the IETF hence anticipates updating default cryptographic mechanisms over time.

The TBD default security function will need to be chosen such that it can reasonably be implemented on a typical router route processor, and which will provide adequate security without significantly degrading the convergence time of a Label Switching Router (LSR).

Without a function that does not significantly impact router convergence we simply close one vulnerability and open another.

Note: As experts on the LDP protocol, but not on security mechanisms, we need to ask the security area for a review of our proposed approach, and help correcting any misunderstanding of the security issues or our misunderstanding of the existing security mechanisms. We also need a recommendation on a suitable security function (TBD in the above text).

4. Security Considerations

This document is entirely about LDP operational security. It describes best practices that one should adopt to secure LDP messages and the TCP based LDP sessions between LSRs.

This document does not aim to describe existing LDP implementations, their potential vulnerabilities, or ways they handle errors. It does not detail how protection could be enforced against attack techniques using crafted packets.

5. IANA Considerations

There are no requests for IANA actions in this document.

Note to the RFC Editor - this section can be removed before publication.

6. Acknowledgements

-

-

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, DOI 10.17487/RFC2385, August 1998, <https://www.rfc-editor.org/info/rfc2385>.

[RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed., "LDP Specification", RFC 5036, DOI 10.17487/RFC5036, October 2007, <https://www.rfc-editor.org/info/rfc5036>.

[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>.

7.2. Informative References

[RFC3036] Andersson, L., Doolan, P., Feldman, N., Fredette, A., and B. Thomas, "LDP Specification", RFC 3036, DOI 10.17487/RFC3036, January 2001, <https://www.rfc-editor.org/info/rfc3036>.

[RFC6518] Lebovitz, G. and M. Bhatia, "Keying and Authentication for Routing Protocols (KARP) Design Guidelines", RFC 6518, DOI 10.17487/RFC6518, February 2012, <https://www.rfc-editor.org/info/rfc6518>.

[RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <https://www.rfc-editor.org/info/rfc6952>.

[RFC7349] Zheng, L., Chen, M., and M. Bhatia, "LDP Hello Cryptographic Authentication", RFC 7349, DOI 10.17487/RFC7349, August 2014, <https://www.rfc-editor.org/info/rfc7349>.

[RFC7454] Durand, J., Pepelnjak, I., and G. Doering, "BGP Operations and Security", BCP 194, RFC 7454, DOI 10.17487/RFC7454, February 2015, <https://www.rfc-editor.org/info/rfc7454>.

Authors' Addresses

Loa Andersson
Bronze Dragon Consulting

Email: loa@pi.nu

Stewart Bryant
Huawei Technologies

Email: stewart.bryant@gmail.com

Andrew G. Malis
Huawei Technologies

Email: agmalis@gmail.com

Nicolai Leymann
Deutsche Telekom

Email: N.Leymann@telekom.de

George Swallow
Independent

Email: swallow.ietf@gmail.com

TEAS Working Group
Internet Draft

A.Wang
China Telecom
Boris Khasanov
Huawei Technologies
Sudhir Cheruathur
Juniper Networks
Chun Zhu
ZTE Company

Intended status: Standard Track
Expires: August 13, 2018

February 14, 2018

PCEP Extension for Native IP Network
draft-wang-pce-pcep-extension-native-ip-01.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 https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 13, 2018.

Copyright Notice

Copyright (c) 2018 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.

Abstract

This document defines the PCEP extension for CCDR application in Native IP network. The scenario and architecture of CCDR in native IP is described in [draft-ietf-teas-native-ip-scenarios] and [draft-ietf-teas-pce-native-ip]. This draft describes the key information that is transferred between PCE and PCC to accomplish the end2end traffic assurance in Native IP network under central control mode.

Table of Contents

1. Introduction

Traditionally, MPLS-TE traffic assurance requires the corresponding network devices support MPLS or the complex RSVP/LDP/Segment Routing etc. technologies to assure the end-to-end traffic performance. But in native IP network, there will be no such signaling protocol to synchronize the action among different network devices. It is necessary to use the central control mode that described in [draft-ietf-teas-pce-control-function] to correlate the forwarding behavior among different network devices. Draft [draft-ietf-teas-pce-native-ip] describes the architecture and solution philosophy for the end2end traffic assurance in Native IP network via Dual/Multi BGP solution. This draft describes the corresponding PCEP extension to transfer the key information about peer address list, peer prefix association and the explicit peer route on on-path router.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

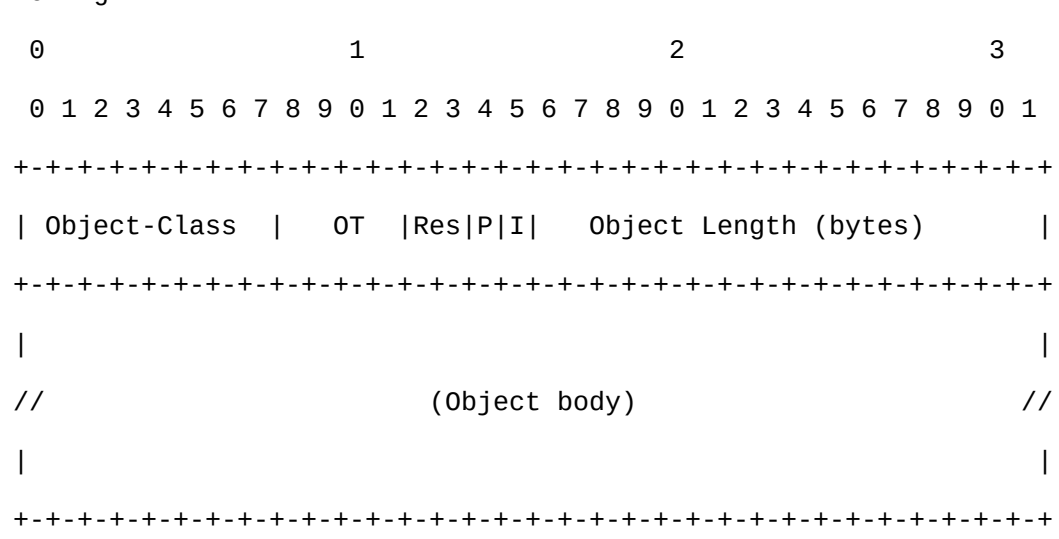
3. New Objects Extension

Three new objects are defined in this draft; they are Peer Address List Object (PAL Object), Peer Prefix Association Object (PPA Object) and Explicit Peer Route object (EPR Object).

Peer Address List object is used to tell the network device which peer it should be peered with dynamically, Peer Prefix Association is used to tell which prefixes should be advertised via the corresponding peer and Explicit Peer Route object is used to point out which route should be to taken to arrive to the peer.

4. Object Formats.

Each extension object takes the similar format, that is to say, it began with the common object header defined in [RFC5440] as the following:



Different object-class, object type and the corresponding object body is defined separated in the following section.

4.1. Peer Address List object.

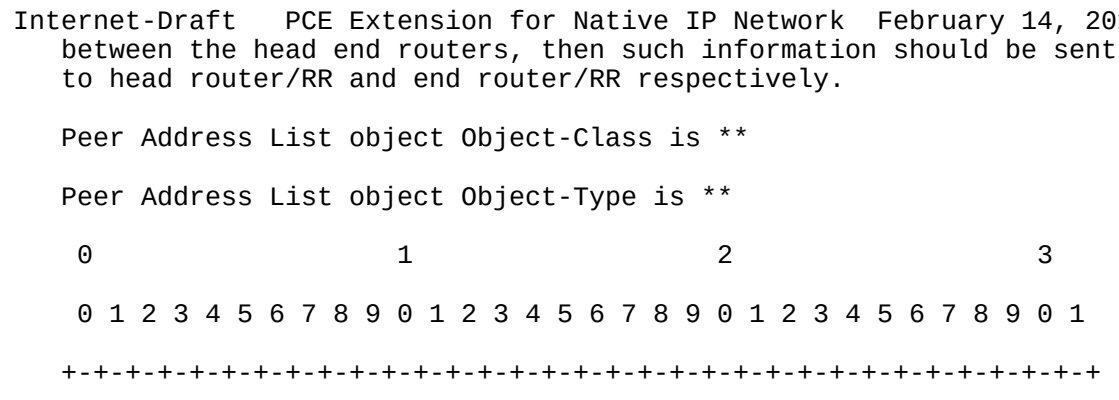
The Peer Address List object is used in a PCE Initiate message [draft-ietf-pce-pce-initiated-lsp] to specify the ip address of peer that the received network device should establish the BGP relationship with.

This Object should only be sent to the head and end router of the end2end path in case there is no RR involved. If the RR is used

between the head end routers, then such information should be sent to head router/RR and end router/RR respectively.

Peer Address List object Object-Class is **

Peer Address List object Object-Type is **



Peer Num (8 bits): Peer Address Number on the advertised router.

Peer-Id(8 bits): To distinguish the different peer pair, will be referenced in Peer Prefix Association, if the PCE use multi-BGP solution for different QoS assurance requirement.

AT(8 bits): Address Type. To indicate the address type of Peer.
Equal to 4, if the following IP address of peer is belong to IPv4;
Equal to 6 if the following IP address of peer is belong to IPv6.

Resv(8 bits): Reserved for future use.

Local IP Address(4/16 Bytes): IPv4 address of the local router, used to peer with other end router. When AT equal to 4, length is 32bit; when AT equal to 16, length is 128bit;

Peer IP Address(4/16 Bytes): IPv4 address of the peer router, used to peer with the local router. When AT equal to 4, length is 32bit; IPv6 address of the peer when AT equal to 16, length is 128bit;

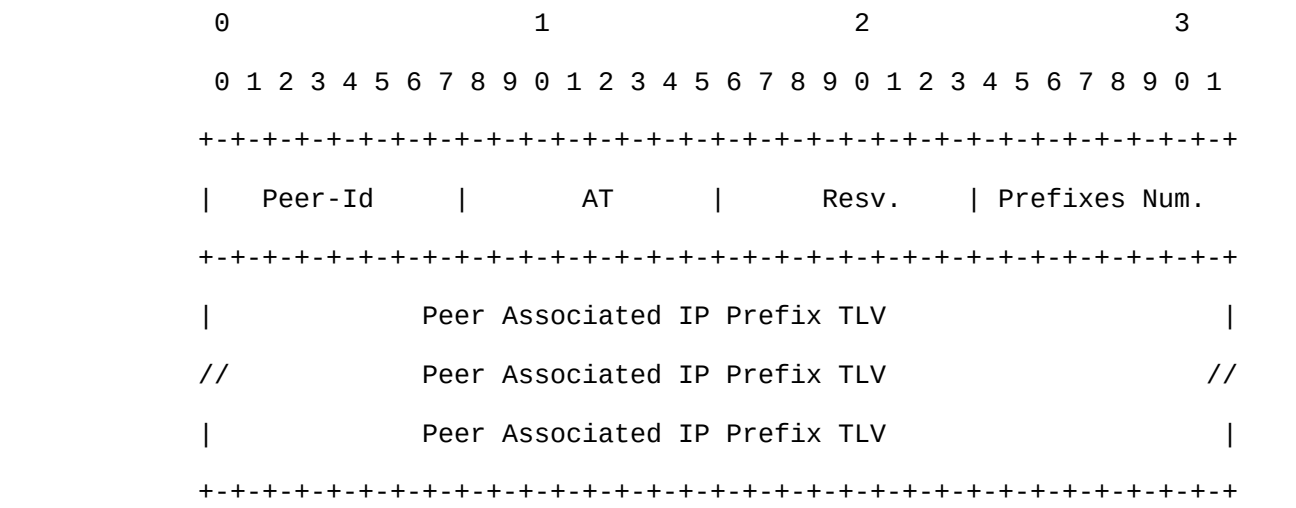
4.2. Peer Prefix Association

THE Peer Prefix Association object is carried within in a PCE Initiate message [draft-ietf-pce-pce-initiated-lsp] to specify the IP prefixes that should be advertised by the corresponding Peer.

This Object should only be sent to the head and end router of the end2end path in case there is no RR involved. If the RR is used between the head end routers, then such information should be sent to head router/RR and end router/RR respectively.

Peer Prefix Association object Object-Class is **

Peer Prefix Association object Object-Type is **



Peer-Id(8 bits): To indicate which peer should be used to advertise the following IP Prefix TLV. This value is assigned in the Peer Address List object and is referred in this object.

AT(8 bits): Address Type. To indicate the address type of Peer. Equal to 4, if the following IP address of peer is belong to IPv4; Equal to 6 if the following IP address of peer is belong to IPv6.

Resv(8 bits): Reserved for future use.

Prefixes Num(8 bits): Number of prefixes that advertised by the corresponding Peer. It should be equal to num of the following IP prefix TLV.

Peer Associated IP Prefix TLV: Variable Length, use the TLV format to indicate the advertised IP Prefix.

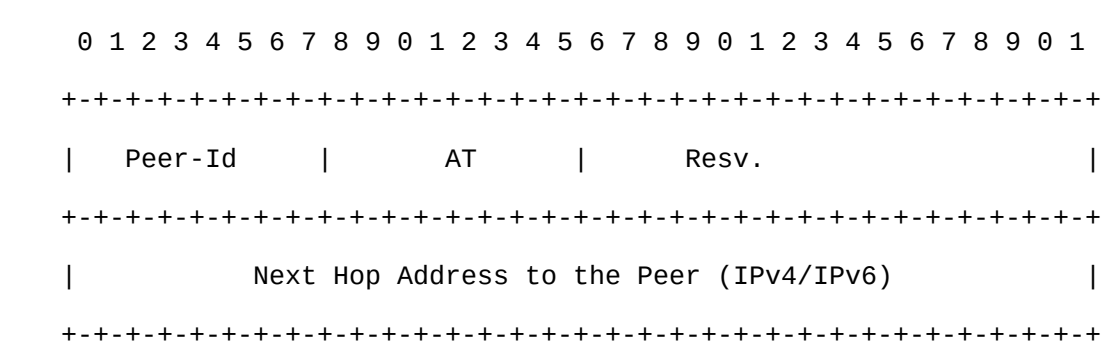
4.3. EXPLICIT PEER ROUTE Object

THE EXPLICIT PEER ROUTE Object is carried in a PCE Initiate message [draft-ietf-pce-pce-initiated-lsp] to specify the explicit peer route to the corresponding peer address on each device that is on the end2end assurance path.

This Object should be sent to all the devices that locates on the end2end assurance path that calculated by PCE.

EXPLICIT PEER ROUTE Object Object-Class is **

EXPLICIT PEER ROUTE Object Object-Type is **



Peer-Id(8 bits): To indicate the peer that the following next hop address point to. This value is assigned in the Peer Address List object and is referred in this object.

AT(8 bits): Address Type. To indicate the address type of explicit peer route. Equal to 4, if the following next hop address to the peer is belong to IPv4; Equal to 6 if the following next hop address to the peer is belong to IPv6.

Resv(16 bits): Reserved for future use.

Next Hop Address to the Peer TLV: Variable Length, use the TLV format to indicate the next hop address to the corresponding peer that indicated by the Peer-Id.

5. Management Consideration.

6. Security Considerations

TBD

7. IANA Considerations

TBD

8. Conclusions

TBD

9. References

9.1. Normative References

[RFC4655] Farrel, A., Vasseur, J.-P., and J. Ash, "A Path Computation Element (PCE)-Based Architecture", RFC 4655, August 2006,<http://www.rfc-editor.org/info/rfc4655>.

[RFC5440]Vasseur, JP., Ed., and JL. Le Roux, Ed., "Path Computation Element (PCE) Communication Protocol (PCEP)", RFC 5440, March 2009, <http://www.rfc-editor.org/info/rfc5440>.

9.2. Informative References

[I-D.draft-ietf-pce-pce-initiated-lsp-07]
E.Crabbe, I.Minei, S.Sivabalan, R.Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", https://tools.ietf.org/html/draft-ietf-pce-pce-initiated-lsp-07 (work in progress), July, 2016

[I-D. draft-ietf-teas-native-ip-scenarios]
Wang, X.Huang et al. "CCDR Scenario, Simulation and Suggestion" https://datatracker.ietf.org/doc/draft-ietf-teas-native-ip-scenarios/ (work in progress), February, 2018

[I-D. draft-ietf-teas-pce-native-ip]
Aijun Wang, Quintin Zhao, Boris Khasanov, Huaimo Chen,Raghavendra Mallya, Shaofu Peng "PCE in Native IP Network", https://datatracker.ietf.org/doc/draft-ietf-teas-pce-native-ip/

(work in progress), February, 2018

[I-D.draft-ietf-teas-pce-control-function]
Farrel, Q.Zhao "An Architecture for use of PCE and PCEP in a Network with Central Control"
https://tools.ietf.org/html/draft-ietf-teas-pce-central-control-01

(work in progress),December, 2016

10. Acknowledgments

TBD

Authors' Addresses

Aijun Wang
China Telecom
Beiqijia Town, Changping District
Beijing,China

Email: wangaj.bri@chinatelecom.cn

Boris Khasanov
Huawei Technologies
Moskovskiy Prospekt 97A
St.Petersburg 196084
Russia

EMail: khasanov.boris@huawei.com

Sudhir Cheruathur
Juniper Networks
1133 Innovation Way
Sunnyvale, California 94089 USA

Email: scheruathur@juniper.net

Chun Zhu
ZTE Corporation
50 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China
Email:zhu.chun1@zte.com.cn

PCE WG
Internet-Draft
Intended status: Standards Track
Expires: September 6, 2018

Quan Xiong
Fangwei Hu
Shuangping Zhan
ZTE Corporation

March 5, 2018

PCEP extensions for SR-TP
draft-xiong-pce-pcep-extension-sr-tp-00.txt

Abstract

This document proposes a set of extensions to PCEP for Segment Routing in MPLS Transport Profile (SR-TP) networks and defines a mechanism to create the bi-directional SR tunnel in SR-TP networks with PCE.

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 September 6, 2018.

Copyright Notice

Copyright (c) 2018 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

The Path Computation Element Communication Protocol (PCEP) defined in [RFC5440] provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

[I-D.ietf-pce-segment-routing] proposes extensions to PCEP that allow a stateful PCE to compute Traffic Engineering (TE) paths in segment routing (SR) networks. But it is applicable to Multi-protocol Label Switching (MPLS) networks. [I-D.hu-spring-sr-tp-use-case] describes the use case of SR tunnel to be deployed in MPLS Transport Profile (SR-TP) network. It is required to extend the PCEP protocol to meet the new requirement for SR-TP.

This document proposes a set of extensions to PCEP for Segment Routing in MPLS Transport Profile (SR-TP) networks and defines a mechanism to create the bi-directional SR tunnel in SR-TP networks with PCE.

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].

1.2. Terminology

The terminology is defined as [RFC5440], [I-D.ietf-pce-segment-routing] and [I-D.hu-spring-sr-tp-use-case].

2. The SR-TP Architecture with PCE

As described in [I-D.hu-spring-sr-tp-use-case], in SR-TP networks, the centralized controller may calculate the end to end SR paths, and creates the ordered segment list. The centralized controller may be replaced to PCE as the Figure 1 shown. The PCE can calculate the SR paths and initiate a SR path on a PCC.

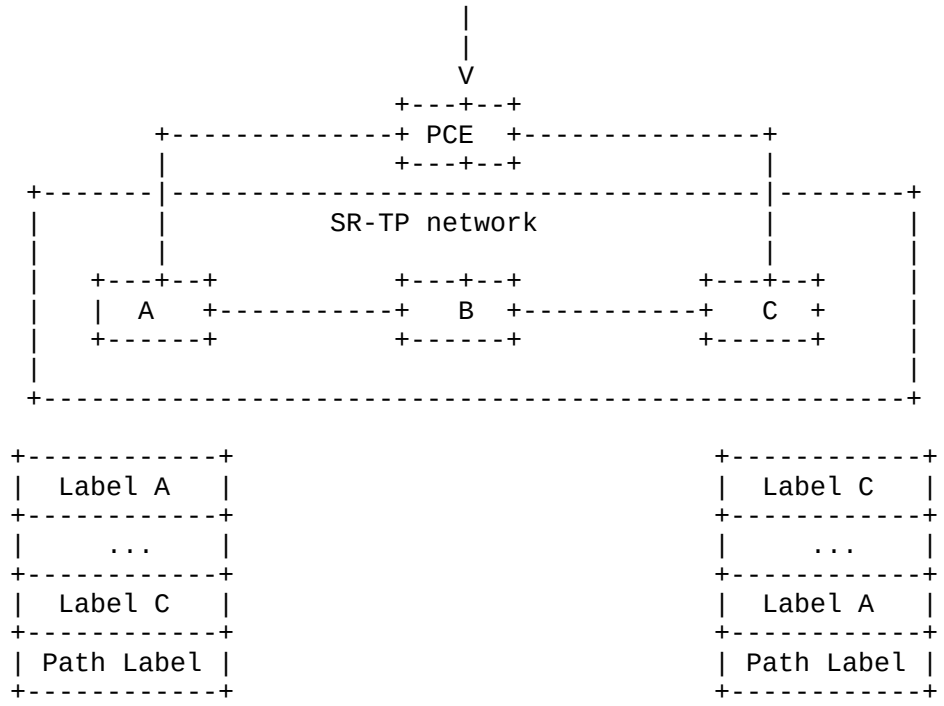


Figure 1 The SR-TP Architecture with PCE

It is required to support bi-direction tunnel to meet the requirement of SP-TP networks. A label named Path segment at both ends of the paths was defined to identify the direction of the SR paths as described in [I-D.cheng-spring-mpls-path-segment]. It mainly aims to bind two unidirectional SR paths to a single bi-directional tunnel.

3. PCEP extensions for SR-TP

3.1. Bi-directional LSP extension

3.1.1. The B flag in SRP Object

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

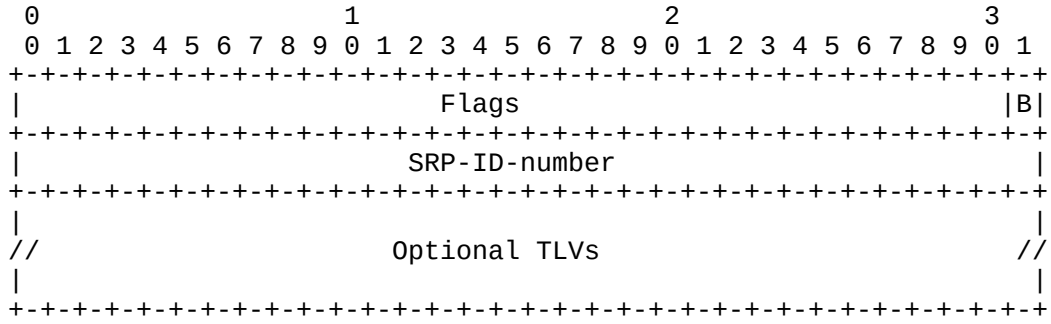


Figure 2 The SRP Object Format

A new flag is defined to indicate a bi-directional LSP operation initiated by the PCE:

B(Bi-directional -- 1 bit):when set, the PCE specifies that the request relates to a bi-directional TE LSP that has the same traffic engineering requirements including fate sharing, protection and restoration, LSRs, TE links, and resource requirements (e.g., latency and jitter) in each direction. When cleared, the TE LSP is unidirectional.

3.2. SR-TP ERO extension

As described in [I-D.hu-spring-sr-tp-use-case], it is required to support bi-directional tunnel to meet the requirement of SP-TP networks. But it is the uni-directional tunnel for SR and engineering traffic network as discussed in [I-D.ietf-pce-segment-routing]. The SR path is carried in the Segment Routing Explicit Route Object (SR-ERO), which consists of a sequence of SR subobjects. This document proposes the extension of the SR-ERO Subobject to carry the bi-directional tunnel information as the Figure 3 shown.

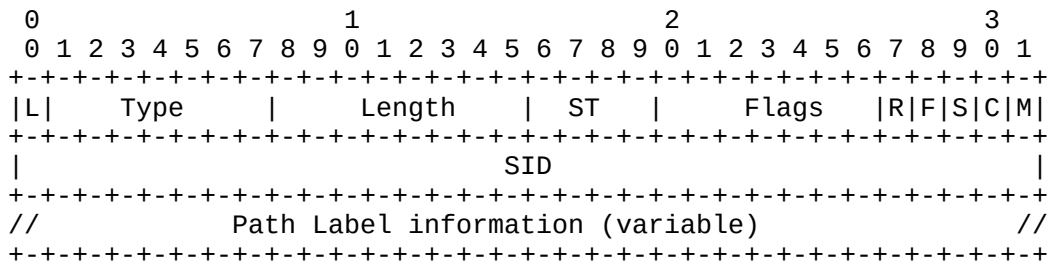


Figure 3 Extension of SR-ERO Subobject format

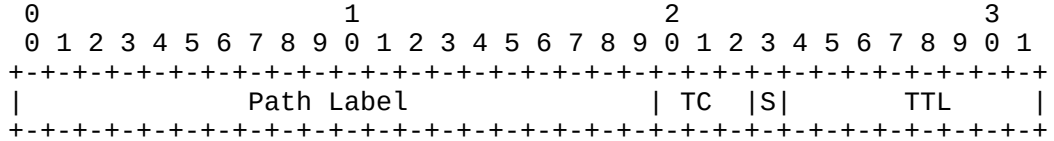


Figure 4 Path Label information

ST (SID Type -- 4 bit):TBD, indicates the type of information associated with the Path Label contained in the object body. when the ST indicates the Path Label type, the NAI is filled with the Path Label information as the Figure 4 shown.

R (Reverse Flag -- 1 bit): indicates the SR path direction, when it is clear, it indicates the forward direction and when it is set, it indicates the reverse direction.

The definition of other fields is the same with [I-D.ietf-pce-segment-routing].

3.3. Processing Rules

As discussed in [I-D.cheng-spring-mpls-path-segment], the bi-directional SR tunnel is created from two binding unidirectional SR paths. As defined in [RFC8281], the stateful PCE calculates the SR paths and initiates the bi-directional LSP with Initiate Request message (PCInitiate).

The B bit in SRP Object MUST be set and the two unidirectional SR paths may be computed from the forward and reverse direction and sent to the source and destination PCC respectively in SR-ERO object. The path labels which binding the paths may be generated in PCE and sent to the related PCC carried in the bottom of the SR-ERO. When the PCCs at both ends receiving the PCInitiate message with the labels in

SR-ERO subobjects, they may forward the packets from bi-directional tunnel in SR-TP networks.

4. Security Considerations

TBD.

5. IANA Considerations

TBD.

6. Acknowledgements

TBD.

7. References

7.1. Informative References

[I-D.hu-spring-sr-tp-use-case]
hu, f., Xiong, Q., Mirsky, G., and W. Cheng, "Segment Routing Transport Profile Use Case", draft-hu-spring-sr-tp-use-case-01 (work in progress), March 2018.

7.2. Normative References

[I-D.cheng-spring-mpls-path-segment]
Cheng, W., Wang, L., Li, H., Chen, M., Zigler, R., and S. Zhan, "Path Segment in MPLS Based Sement Routing Network", draft-cheng-spring-mpls-path-segment-01 (work in progress), March 2018.

[I-D.ietf-pce-segment-routing]
Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-11 (work in progress), November 2017.

[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>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

Authors' Addresses

Quan Xiong
ZTE Corporation
No.6 Huashi Park Rd
Wuhan, Hubei 430223
China

Phone: +86 27 83531060
Email: xiong.quan@zte.com.cn

Fangwei Hu
ZTE Corporation
No.889 Bibo Rd
Shanghai 201203
China

Phone: +86 21 68896273
Email: hu.fangwei@zte.com.cn

Shuangping Zhan
ZTE Corporation
Liuxian Rd
Shenzhen 518057
China

Phone: +86 755 26773770
Email: zhan.shuangping@zte.com.cn

PCE WG
Internet-Draft
Intended status: Standards Track
Expires: February 28, 2019

Quan Xiong
Fangwei Hu
Shuangping Zhan
ZTE Corporation

August 27, 2018

PCEP extensions for SR-TP
draft-xiong-pce-pcep-extension-sr-tp-01

Abstract

This document proposes a set of extensions to PCEP for Segment Routing in MPLS Transport Profile (SR-TP) networks and defines a mechanism to create the bi-directional SR tunnel in SR-TP networks with PCE.

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 February 28, 2019.

Copyright Notice

Copyright (c) 2018 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

The Path Computation Element Communication Protocol (PCEP) defined in [RFC5440] provides mechanisms for Path Computation Elements (PCEs) to perform path computations in response to Path Computation Clients (PCCs) requests.

[I-D.ietf-pce-segment-routing] proposes extensions to PCEP that allow a stateful PCE to compute Traffic Engineering (TE) paths in segment routing (SR) networks. But it is applicable to Multi-protocol Label Switching (MPLS) networks. [I-D.hu-spring-sr-tp-use-case] describes the use case of SR tunnel to be deployed in MPLS Transport Profile (SR-TP) network. It is required to extend the PCEP protocol to meet the new requirements for SR-TP. One of the requirements is the bidirectional SR tunnel described in [I-D.cheng-spring-mpls-path-segment].

This document proposes a set of extensions to PCEP for Segment Routing in MPLS Transport Profile (SR-TP) networks and defines a mechanism to create the bidirectional SR tunnel in SR-TP networks with PCE.

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].

1.2. Terminology

The terminology is defined as [RFC5440], [I-D.ietf-pce-segment-routing] , [I-D.cheng-spring-mpls-path-segment] and [I-D.hu-spring-sr-tp-use-case].

2. The SR-TP Architecture with PCE

As described in [I-D.hu-spring-sr-tp-use-case], in SR-TP networks, the centralized controller may calculate the end to end SR paths, and creates the ordered segment list. The centralized controller may be replaced to PCE as the Figure 1 shown. The PCE can calculate the SR paths and a SR path can be initiated by PCE or PCC.

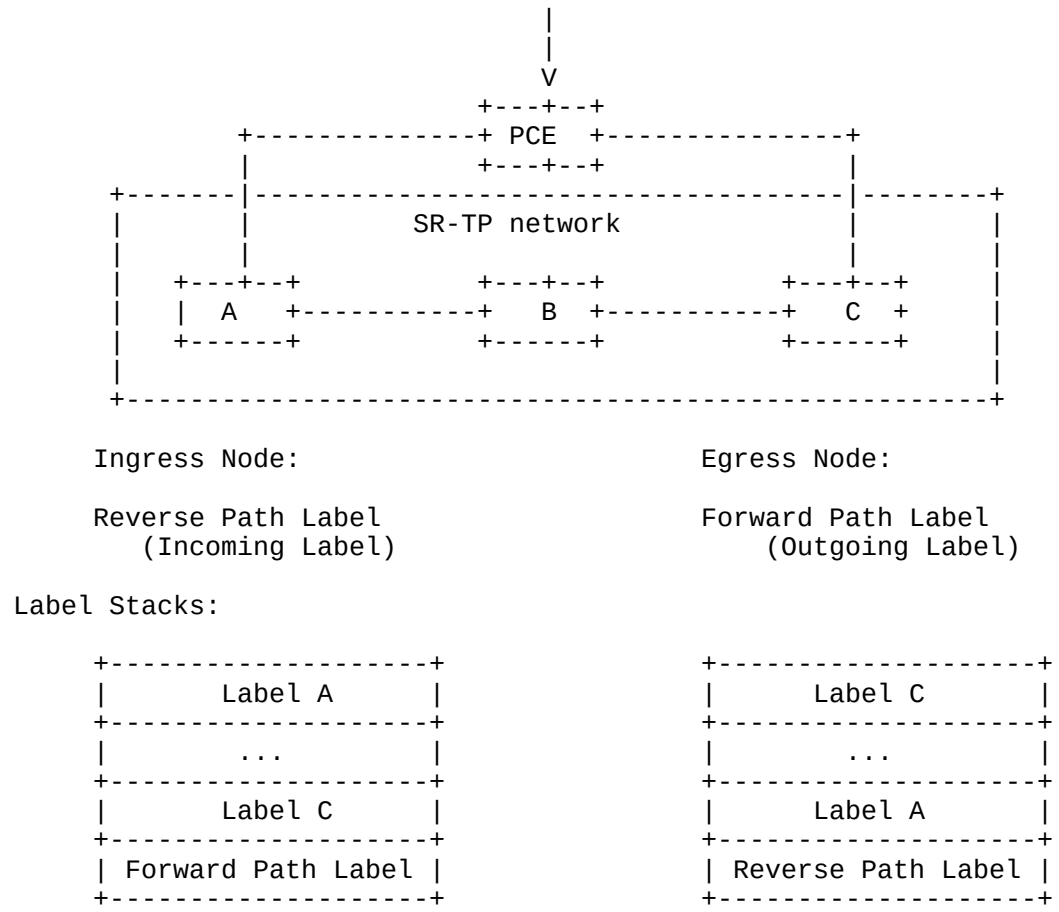


Figure 1 The SR-TP Architecture with PCE

It is required to support bidirectional tunnel to meet the requirement of SP-TP networks. A label named path segment at both ends of the paths was defined to identify the direction of the SR paths as defined in [I-D.cheng-spring-mpls-path-segment]. It mainly aims to bind two unidirectional SR paths to a single bidirectional tunnel.

As the Figure 1 shown, the forward and backward directions of the bidirectional SR tunnel are identified by the forward and reverse path label respectively. For the ingress node, the forward path label shall be added to the bottom of the label stack and the reverse path label shall be configured to the data plane as incoming label for the SR LSP. And for the egress node, the reverse path label need to be the last one label of the label stack and the forward path label shall be used as outgoing label.

2.1. SR Path SID

[RFC8402] defined the IGP, BGP, and Binding segments for the SR-MPLS and SRv6 data planes which can be referred to by Segment Identifier (SID). And [I-D.cheng-spring-mpls-path-segment] defined a new type of segment named path segment. So the path segment can also be identified by SID called SR path SID. The path segment may be associated with a unidirectional path.

3. Overview of Bi-directional SR tunnel

As [RFC5654] defined, MPLS-TP MUST support unidirectional, co-routed bidirectional, and associated bidirectional point-to-point transport paths. Based on the defination of co-routed bidirectional path, the forward and backward directions follow the same route (links and nodes) across the network and must be setup, monitored and protected as a single entity.

However, as [RFC8402] defined, segment routing leverages the source routing paradigm and the sourse node steers a packet through an ordered segment list along a unidirectional path. So for bidirectional SR tunnel, the forward and backward directional paths may be setup by the source node and destination node seperately. So the co-routed birectional SR paths can not be provisioned by PCE.

3.1. Associated Bidirectional SR tunnel

As described in [I-D.ietf-pce-association-bidir], two reverse unidirectional LSPs can be associated as an associated bidirectional tunnel which can be initialed by single-sided and double-sided methods. Based on the discussion above, the associated bidirectional

SR tunnel can only be provisioned on both ingress and egress node (PCCs).

The Double-sided initiation can be initiated by PCCs or PCE.The forward and reverse LSPs of the SR tunnel are initiated with the Association Type set to "Double-sided Bidirectional LSP Association" and the "Bidirectional LSP Association Group TLV" carried in initial messages.

The forward and reverse directional paths can be co-routed or non-corouted. The SR bidirectional tunnel may follow the same path in the forward and reverse directions and initialed as a co-routed associated bidirectional LSP.

4. PCEP extensions for SR-TP

4.1. SR-TP ERO extension

As described in [I-D.hu-spring-sr-tp-use-case], it is required to support bi-directional tunnel to meet the requirement of SP-TP networks. But it is the uni-directional tunnel for SR and engineering traffic network as discussed in [I-D.ietf-pce-segment-routing]. The SR path is carried in the Segment Routing Explicit Route Object (SR-ERO), which consists of a sequence of SR subobjects. This document proposes the extension of the SR-ERO Subobject to carry the bi-directional tunnel information as the Figure 3 shown. The subobjects with path SIDs need to be added to the list of the SR-ERO subobjects.

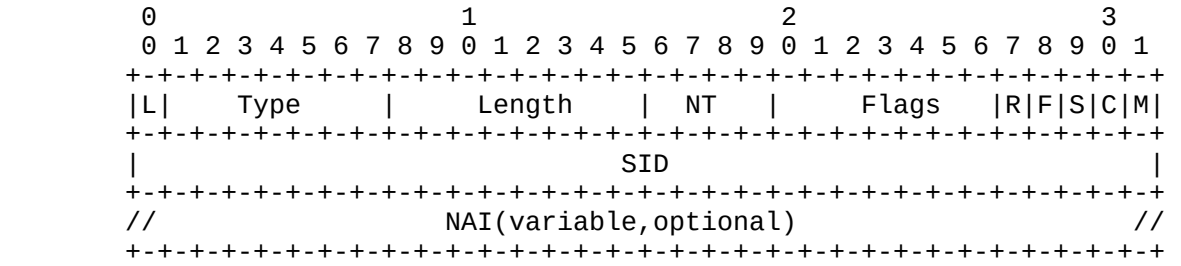


Figure 3 Extension of SR-ERO Subobject format

NAI Type (NT) : A new type of NT = 6 is added in this document and it indicates the type and format of the NAI associated with the path SID contained in the object body. When NT is set to 6, the format of NAI field is shown as figure 4.

R (Reverse Flag -- 1 bit): indicates the SR path direction, when it is clear, it indicates the forward direction and when it is set, it indicates the reverse direction.

The definition of other fields is the same with [I-D.ietf-pce-segment-routing].

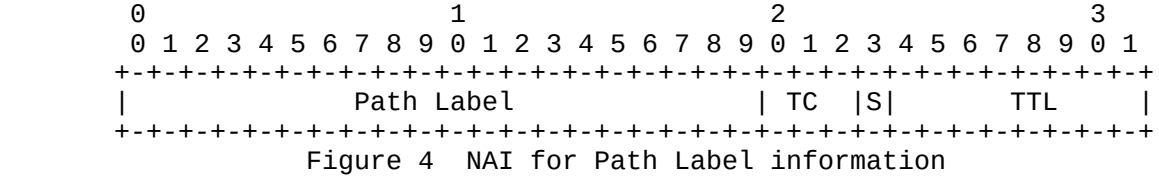


Figure 4 NAI for Path Label information

The format of Path Label information is specified as [I-D.cheng-spring-mpls-path-segment].

4.2. Processing Rules

As discussed in [I-D.cheng-spring-mpls-path-segment], the bi-directional SR tunnel is created from two binding unidirectional SR paths. As defined in [RFC8281], the stateful PCE calculates the SR paths and initiates the bi-directional LSP with Initiate Request message (PCInitiate).

The B bit in SRP Object MUST be set and the two unidirectional SR paths may be computed from the forward and reverse direction and sent to the source and destination PCC respectively in SR-ERO object. The path labels which binding the paths may be generated in PCE and sent to the related PCC carried in the bottom of the SR-ERO. When the PCCs at both ends receiving the PCInitiate message with the labels in SR-ERO subobjects, they may forward the packets from bi-directional tunnel in SR-TP networks.

5. Security Considerations

TBD.

6. IANA Considerations

TBD.

7. Acknowledgements

TBD.

8. References

8.1. Informative References

[I-D.hu-spring-sr-tp-use-case]
hu, f., Xiong, Q., Mirsky, G., and W. Cheng, "Segment Routing Transport Profile Use Case", draft-hu-spring-sr-tp-use-case-01 (work in progress), March 2018.

8.2. Normative References

[I-D.cheng-spring-mpls-path-segment]
Cheng, W., Wang, L., Li, H., Chen, M., Zigler, R., Zhan, S., and R. Gandhi, "Path Segment in MPLS Based Segment Routing Network", draft-cheng-spring-mpls-path-segment-02 (work in progress), July 2018.

[I-D.ietf-pce-association-bidir]
Barth, C., Gandhi, R., and B. Wen, "PCEP Extensions for Associated Bidirectional Label Switched Paths (LSPs)", draft-ietf-pce-association-bidir-01 (work in progress), May 2018.

[I-D.ietf-pce-segment-routing]
Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-12 (work in progress), June 2018.

[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>.

[RFC5654] Niven-Jenkins, B., Ed., Brungard, D., Ed., Betts, M., Ed., Sprecher, N., and S. Ueno, "Requirements of an MPLS Transport Profile", RFC 5654, DOI 10.17487/RFC5654, September 2009, <https://www.rfc-editor.org/info/rfc5654>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model", RFC 8281, DOI 10.17487/RFC8281, December 2017, <https://www.rfc-editor.org/info/rfc8281>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

Authors' Addresses

Quan Xiong
ZTE Corporation
No.6 Huashi Park Rd
Wuhan, Hubei 430223
China

Phone: +86 27 83531060
Email: xiong.quan@zte.com.cn

Fangwei Hu
ZTE Corporation
No.889 Bibo Rd
Shanghai 201203
China

Phone: +86 21 68896273
Email: hu.fangwei@zte.com.cn

Shuangping Zhan
ZTE Corporation
Liuxian Rd
Shenzhen 518057
China

Phone: +86 755 26773770
Email: zhan.shuangping@zte.com.cn