Abstract

This document presents a forward search procedure for computing a path for a Point-to-MultiPoint (P2MP) Traffic Engineering (TE) Label Switched Path (LSP) crossing a number of domains through using multiple Path Computation Elements (PCEs). In addition, extensions to the Path Computation Element Communication Protocol (PCEP) for supporting the forward search procedure are described.

Status of This Memo

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

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

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 16, 2013.

Copyright Notice

Copyright (c) 2012 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
Internet-Draft  Forward Search P2MP Path  September 2012

described in the Simplified BSD License.

Table of Contents

1. Introduction .............................................. 3
2. Terminology ............................................... 3
3. Conventions Used in This Document ......................... 4
4. Requirements .............................................. 4
5. Forward Search P2MP Path Computation ....................... 5
   5.1. Overview of Procedure ............................... 5
   5.2. Description of Procedure ............................ 6
   5.3. Processing Request and Reply Messages ............... 8
6. Comparison with BRPC based Core Tree path computation
   procedure ................................................ 9
7. Extensions to PCEP ......................................... 10
   7.1. RP Object Extension ................................. 10
   7.2. Request Message Extension ........................... 11
   7.3. Reply Message Extension ............................. 11
8. Suggestion to improve performance .......................... 12
9. Manageability Considerations ............................... 12
   9.1. Control of Function and Policy ...................... 12
   9.2. Information and Data Models ......................... 13
   9.3. Liveness Detection and Monitoring ................... 13
   9.4. Verify Correct Operations ........................... 13
   9.5. Requirements On Other Protocols ..................... 13
10. Security Considerations .................................... 13
11. IANA Considerations ....................................... 13
12. Acknowledgement ........................................... 13
13. References ................................................ 13
    13.1. Normative References ............................... 13
    13.2. Informative References ............................. 14
1. Introduction

[RFC4105] "Requirements for Inter-Area MPLS TE" lists the requirements for computing a shortest path for a TE LSP crossing multiple IGP areas; and [RFC4216] "MPLS Inter-Autonomous System (AS) TE Requirements" describes the requirements for computing a shortest path for a TE LSP crossing multiple ASes. [RFC5671] "Applicability of PCE to P2MP MPLS and GMPLS TE" examines the applicability of PCE to path computation for P2MP TE LSPs in MPLS and GMPLS networks.

This document presents a forward search procedure to address these requirements for computing a path for a P2MP TE LSP crossing domains through using multiple Path Computation Elements (PCEs).

The procedure is called "Forward Search Shortest P2MP LSP Path Crossing Domains". The major characteristics of this procedure for computing a path for a P2MP TE LSP from a source node to a number of destination nodes crossing multiple domains include the following three ones.

1. It guarantees that the path computed from the source node to the destination nodes is shortest.

2. It does not depend on any domain path tree or domain sequences from the source node to the destination nodes.

3. Navigating a mesh of domains is simple and efficient.

2. Terminology

The following terminology is used in this document.

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

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

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

Entry BN of domain(n): a BN connecting domain(n-1) to domain(n) along the path found from the source node to the BN, where domain(n-1) is the previous hop domain of domain(n).
Exit BN of domain(n): a BN connecting domain(n) to domain(n+1) along
the path found from the source node to the BN, where domain(n+1)
is the next hop domain of domain(n).

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

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

LSP: Label Switched Path

LSR: Label Switching Router

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

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

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

TED: Traffic Engineering Database.

This document uses terminology defined in [RFC5440].

3. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

4. Requirements

This section summarizes the requirements specific for computing a
path for a Traffic Engineering (TE) LSP crossing multiple domains
(areas or ASes). More requirements for Inter-Area and Inter-AS MPLS
Traffic Engineering are described in [RFC4105] and [RFC4216].

A number of requirements specific for a solution to compute a path
for a TE LSP crossing multiple domains is listed as follows:

1. The solution SHOULD provide the capability to compute a shortest
   path dynamically, satisfying a set of specified constraints
   across multiple IGP areas.

2. The solution MUST provide the ability to reoptimize in a
   minimally disruptive manner (make before break) an inter-area TE
LSP, should a more optimal path appear in any traversed IGP area.

3. The solution SHOULD provide mechanism(s) to compute a shortest end-to-end path for a TE LSP crossing multiple ASes and satisfying a set of specified constraints dynamically.

4. Once an inter-AS TE LSP has been established, and should there be any resource or other changes inside anyone of the ASes, the solution MUST be able to re-optimize the LSP accordingly and non-disruptively, either upon expiration of a configurable timer or upon being triggered by a network event or a manual request at the TE tunnel Head-End.

5. Forward Search P2MP Path Computation

This section gives an overview of the forward search path computation procedure to satisfy the requirements for computing a path for a P2MP TE LSP crossing multiple domains described above and describes the procedure in details.

5.1. Overview of Procedure

Simply speaking, the idea of the Forward Search P2MP inter-domain path computation method for computing a path for an MPLS TE P2MP LSP crossing multiple domains from a source node to a number of destination nodes includes:

Start from the source node and the source domain.

Consider the optimal path segment from the source node to every exit boundary and destination node of the source domain as a special link;

Consider the optimal path segment from an entry boundary node to every exit boundary node and destination node of a domain as a special link; and the optimal path segment is computed as needed.

The whole topology consisting of many domains can be considered as a special virtual topology, which contains those special links and the inter-domain links.

Compute a shortest path in this special topology from the source node to the multiple destination nodes using CSPF.

Forward Search P2MP inter-domain path computation method running at any PCE just grows the result path list/tree in the same way as normal CSPF on the special virtual topology. When the result path list/tree reaches all the destination nodes, the shortest path from the source node to the destination nodes is found and a PCRep message
with the path is sent to the PCE/PCC that sends the PCReq message eventually.

5.2. Description of Procedure

Suppose that we have the following variables:

A current PCE named as CurrentPCE which is currently computing the path.

A candidate node list named as CandidateNodeList, which contains the nodes to each of which the temporary optimal path from the source node is currently found and satisfies a set of given constraints. The information about each node C in CandidateNodeList consists of:

- the cost of the path from the source node to node C,
- the hopcount of the path from the source node to node C,
- the previous hop node P and the link between P and C,
- the domain list of C (ABR or ASBR) where C has visibility to multiple domains and clearly mark the domain by which C is added to CandidateNodeList,
- the PCE responsible for C (i.e., the PCE responsible for the domain containing C. Alternatively, we may use the above mentioned domain instead of the PCE.), and
- the flags for C. The flags include:
  * bit D indicating that C is a Destination node if it is set,
  * bit S indicating that C is the Source node if it is set,
  * bit T indicating that C is on result path Tree if it is set.

The nodes in CandidateNodeList are ordered by path cost.

Initially, CandidateNodeList contains a Source Node, with path cost 0, PCE responsible for the source domain, and flags with S bit set. It also contains every destination node, with path cost infinity and flags with D bit set.

A result path list or tree named as ResultPathTree, which contains the shortest paths from the source node to the boundary nodes and destination nodes. Initially, ResultPathTree is empty.
Alternatively, the result path list or tree can be combined into the candidate node list. We may set bit T to one in the NODE-FLAGS object (refer [FORWARD-SEARCH-P2P-PATH]) for the candidate node when grafting it into the existing result path list or tree. Thus all the candidate nodes with bit T set to one in the candidate list constitute the result path tree or list.

The Forward Search path computation method for computing a path for an MPLS TE P2MP LSP crossing a number of domains from a source node to a number of destination nodes can be described as follows:

Initially, a PCC sends a PCE responsible for the source domain a request with CandidateNodeList and ResultPathTree initialized.

When the PCE responsible for a domain (called current domain) receives a request for computing the path for the MPLS TE P2MP LSP, it checks whether the current PCE is the PCE responsible for the node C with the minimum cost in the CandidateNodeList (always expand node C only if it is an entry boundary node). If it is, then remove C from CandidateNodeList and graft it into ResultPathTree (i.e., set flag bit T of node C to one); otherwise, a PCReq message is sent to the PCE for node C.

Suppose that node C is in the current domain. The ResultPathTree is built from C in the following steps.

If node C is a destination node (i.e., the Destination Node (D) bit in the Flags is set), then check whether the path cost to node C is infinity. If it is, then we can not find any path for the P2MP LSP, and a reply message with failure reasons is sent; otherwise, if all the destinations are on the result path tree, then the shortest path is found and a PCRep message with the path is sent to the PCE/PCC which sends the request to the current PCE.

If node C is an entry boundary node or the source node, then the optimal path segments from node C to every destination node and every exit boundary node of the current domain that is not on the result path tree and satisfies the given constraints are computed through using CSPF and as special links.

For every node N connected to node C through a special link (i.e., the optimal path segment satisfying the given constraints), it is merged into CandidateNodeList. The cost to node N is the sum of the cost to node C and the cost of the special link (i.e., the path segment) between C and N. If node N is not in the candidate node list, then node N is added into the list with the cost to node N, node C as its previous hop node and a PCE for node N. The PCE for node N is the current PCE if node N is an ASBR; otherwise (node N is
an ABR, an exit boundary node of the current domain and an entry boundary node of the domain next to the current domain) the PCE for node N is the PCE for the next domain. If node N is in the candidate node list and the cost to node N through node C is less than the cost to node N in the list, then replace the cost to node N in the list with the cost to node N through node C and the previous hop to node N in the list with node C.

If node C is an exit boundary node and there are inter-domain links connecting to it (i.e., node C is an ASBR) and satisfying the constraints, then for every node N connecting to C, satisfying the constraints and not on the result path tree, it is merged into the candidate node list. The cost to node N is the sum of the cost to node C and the cost of the link between C and N. If node N is not in the candidate node list, then node N is added into the list with the cost to node N, node C as its previous hop node and the PCE for node N. If node N is in the candidate node list and the cost to node N through node C is less than the cost to node N in the list, then replace the cost to node N in the list with the cost to node N through node C and the previous hop to node N in the list with node C.

If the CurrentPCE is the same as the PCE for the node D with the minimum cost in CandidateNodeList, then the node D is removed from CandidateNodeList and grafted to ResultPathTree (i.e., set flag bit T of node D to one), and the above steps are repeated; otherwise, a request message is to be sent to the PCE for node D.

Note that if node C has visibility to multiple domains, do not remove it from the CandidateNodeList until it is expanded in all domains. Also mark in the domain list of node C, for which domains it is already expanded.

### 5.3. Processing Request and Reply Messages

In this section, we describe the processing of the request and reply messages with Forward search bit set for forward search inter-domain path computation. Each of the request and reply messages mentioned below has its Forward search bit set even though we do not indicate this explicitly.

In the case that a reply message is a final reply, which contains the optimal path from the source to the destination, the reply message is sent toward the PCC along the path that the request message goes from the PCC to the current PCE in reverse direction.

In the case that a request message is to be sent to the PCE for node D with the minimum cost in the candidate node list and there is a PCE
session between the current domain and the next domain containing node D, the current PCE sends the PCE for node D through the session a request message with the source node, the destination node, CandidateNodeList and ResultPathTree.

In the case that a request message is to be sent to the PCE for node D and there is not any PCE session between the current PCE and the PCE for node D, a reply message is sent toward a branch point on the result path tree from the current domain along the path that the request message goes from the PCC to the current PCE in reverse direction. From the branch point, there is a downward path to the domain containing the previous hop node of node D on the result path tree and to the domain containing node D. At this branch point, the request message is sent to the PCE for node D along the downward path.

Suppose that node D has the minimum cost in CandidateNodeList when a PCE receives a request message or a reply message containing CandidateNodeList.

When a PCE (current PCE) for a domain (current domain) receives a reply message PCRep, it checks whether the reply is a final reply with the optimal path from the source to the destination(s). If the reply is the final reply, the current PCE sends the reply to the PCE that sends the request to the current PCE; otherwise, it checks whether there is a path from the current domain to the domain containing the previous hop node of node D on ResultPathTree and to the domain containing node D. If there is a path, the PCE sends a request PCReq to the PCE responsible for the next domain along the path; otherwise, it sends a reply PCRep to the PCE that sends the request to the current PCE.

When a PCE receives a request PCReq, it checks whether the current domain contains node D. If it does, then node D is removed from CandidateNodeList and grafted to ResultPathTree (i.e., set flag bit T of node D to one), and the above steps in the previous sub section are repeated; otherwise, the PCE sends a request PCReq to the PCE responsible for the next domain along the path from the current domain to the domain containing the previous hop node of node D on ResultPathTree and to the domain containing node D.

6. Comparision with BRPC based Core Tree path computation procedure

[RFC5441] describes the Backward Recursive Path Computation (BRPC) algorithm or procedure for computing an MPLS TE P2P LSP path from a source node to a destination node crossing multiple domains. [PCE-P2MP-PROCEDURES] describes extended BRPC based procedures to compute Core Tree.
Comparing to Core Tree, there are a number of differences between Core Tree and the Forward-Search P2MP TE LSP Inter-Domain Path Computation. Some of the differences are briefed below.

At first, Core Tree is for computing an optimal path from source node to each entry node of destination domain(s) by crossing multiple domains. Once the core tree is built, then graft the destination nodes to core tree. The Forward-Search P2MP TE LSP Inter-Domain Path Computation is for computing a shortest path from a source node to a number of destination nodes crossing multiple domains.

Secondly, for Core Tree to compute an optimal path from a source node to each entry node of destination domain(s) by crossing multiple domains, we MUST provide a sequence of domains from the source node to each destination node in advance. The Forward-Search P2MP TE LSP Inter-Domain Path Computation does not need any sequence of domains for computing a shortest inter-domain P2MP path.

Moreover, for a given sequence of domains domain(1), domain(2), ..., domain(n), BRPC computes an optimal path from domain(n-1), to domain(n-2), until domain(1) for destination node(n) and the same is repeated for all the destination nodes. So each sub-path insures that it is an optimal path along the given sequence of domains. It may not be an optimal path if the optimal path from the source to the destination node (n) is not in the given sequence of domains. Thus the combination of these sub-paths into a P2MP tree may not be optimal. The Forward-Search P2MP TE LSP Inter-Domain Path Computation calculates a shortest path in a special topology from the source node to the destination nodes using CSPF and it guarantees the path is optimal.

7. Extensions to PCEP

This section describes the extensions to PCEP for Forward Search P2MP Path Computation. The extensions include the definition of a new flag in the RP object, a result path list and a candidate node list in the PCReq and PCRep message which are explained in details in [FORWARD-SEARCH-P2P-PATH].

7.1. RP Object Extension

[FORWARD-SEARCH-P2P-PATH] added following flags into the RP Object:

The F bit is added in the flag bits field of the RP object to tell the receiver of the message that the request/reply is for Forward Search Path Computation.

The T bit is added in the flag bits field of the RP object to tell
the receiver of the message that the reply is for transferring a request message to the domain containing the node with minimum cost in the candidate list.

Setting Transfer request T-bit in a RP Object to one indicates that a reply message containing the RP Object is for transferring a request message to the domain containing the node with minimum cost in the candidate list.

This F bit with the N bit defined in [RFC6006] can indicate whether the request/reply is for Forward Search Path Computation of an MPLS TE P2P LSP or an MPLS TE P2MP LSP.

7.2. Request Message Extension

Below is the message format for a request message with the extension of result-path-list and candidate-node-list:

```plaintext
<PCReq Message>::= <Common Header>
    <request>
    <request>::= <RP>
        <end-point-rro-pair-list>
            [<OF>]
            [<LSPA>]
            [<BANDWIDTH>]
            [<metric-list>]
            [<IRO>]
            [<LOAD-BALANCING>]
            [<result-path-list>]
            [<candidate-node-list >]

where:

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

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

The result-path-list and candidate-node-list are as defined in [FORWARD-SEARCH-P2P-PATH].

7.3. Reply Message Extension

Below is the message format for a reply message with the extension of result-path-list and candidate-node-list:
<PCRep Message> ::= <Common Header>
<response>
<response> ::= <RP>
[<end-point-path-pair-list>]
[<NO-PATH>]
[<attribute-list>]
[<result-path-list>]
[<candidate-node-list >]

where:

<end-point-path-pair-list>::=[<END-POINTS>]
<path>
[<end-point-path-pair-list>]

<path> ::= (<ERO>|<SERO>) [<path>]

<attribute-list>::=[<OF>]
[<LSFA>]
[<BANDWIDTH>]
[<metric-list>]
[<IRO>]

The result-path-list and candidate-node-list are as defined in [FORWARD-SEARCH-P2P-PATH].

If the path from the source to the destination is not found yet and there are still chances to find a path (i.e., the candidate list is not empty), the reply message contains candidate-node-list consisting of the information of the candidate-node-list and result-path-list, which is encoded. In this case, the Transfer request T-bit in the RP Object is set to one.

If the path from the source to the destination is found, the reply message contains path-list comprising the information of the path.

8. Suggestion to improve performance

To get much better performance all the candidate nodes of current domain can be expanded before moving on to a new domain. Note in this, after expanding the least cost candidate node, PCE can look for and expand any other candidates (irrespective of cost) in this domain.

9. Manageability Considerations

9.1. Control of Function and Policy

TBD
9.2. Information and Data Models

TBD

9.3. Liveness Detection and Monitoring

TBD

9.4. Verify Correct Operations

TBD

9.5. Requirements On Other Protocols

TBD


TBD

10. Security Considerations

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

11. IANA Considerations

This are no new IANA allocation.

12. Acknowledgement

The authors would like to thank Julien Meuric, Daniel King, Cyril Margaria, Ramon Casellas, Olivier Dugeon, Oscar Gonzalez de Dios, Udayasree Palle, Reeja Paul and Sandeep Kumar Boina for their valuable comments on this draft.

13. References

13.1. Normative References


13.2. Informative References


Authors’ Addresses

Huaimo Chen
Huawei Technologies
Boston, MA,
USA

EMail: Huaimochen@huawei.com

Dhruv Dhody
Huawei Technologies
Leela Palace
Bangalore, Karnataka 560008
INDIA

EMail: dhruv.dhody@huawei.com