PCE-based Computation Procedure To Compute Shortest Constrained P2MP Inter-domain Traffic Engineering Label Switched Paths
draft-zhao-pce-pcep-inter-domain-p2mp-procedures-05

Abstract

The ability to compute paths for constrained point-to-multipoint (P2MP) Traffic Engineering Label Switched Paths (TE LSPs) across multiple domains has been identified as a key requirement for the deployment of P2MP services in MPLS and GMPLS networks. The Path Computation Element (PCE) has been recognized as an appropriate technology for the determination of inter-domain paths of P2MP TE LSPs.

This document describes the procedures and extensions to the PCE communication Protocol (PCEP) to handle requests and responses for the computation of inter-domain paths for P2MP TE LSPs.

Status of this Memo

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

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

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

This Internet-Draft will expire on January 12, 2011.

Copyright Notice
1. Introduction

Multicast services are increasingly in demand for high-capacity applications such as multicast Virtual Private Networks (VPNs), IP-television (IPTV) which may be on-demand or streamed, and content-rich media distribution (for example, software distribution, financial streaming, or data-sharing). The ability to compute constrained Traffic Engineering Label Switched Paths (TE LSPs) for point-to-multipoint (P2MP) LSPs in Multiprotocol Label Switching (MPLS) and Generalized MPLS (GMPLS) networks across multiple domains. A domain can be defined as a collection of network elements within a common sphere of address management or path computational responsibility such as an IGP area or an Autonomous Systems.

The applicability of the Path Computation Element (PCE) [RFC4655] for the computation of such paths is discussed in [RFC5671], and the requirements placed on the PCE communications Protocol (PCEP) for this are given in [PCE-P2MP-REQ].

This document describes how multiple PCE techniques can be combined to address the requirements. These mechanisms include the use of the per-domain path computation technique specified in [RFC5152], extensions to the backward recursive path computation (BRPC) technique specified in [RFC5441] for P2MP LSP path computation in an inter-domain environment, and a new procedure for core-tree based path computation defined in this document. These three mechanisms are suitable for different environments (topologies, administrative domains, policies, service requirements, etc.) and can also be effectively combined.

2. Terminology

Terminology used in this document is consistent with the related MPLS/GMPLS and PCE documents [RFC4461], [RFC4655], [RFC4875], [RFC5376], [RFC5440], [RFC5441]. [RFC5671], and [PCE-P2MP-REQ].

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

ASBR: Autonomous System Border Routers. Routers used to connect together ASes of the same or different Service Providers via one or more Inter-AS links.

Boundary Node (BN): a boundary node is either an ABR in the context of inter-area Traffic Engineering or an ASBR in the context of inter-AS Traffic Engineering.
Entry BN of domain(n): a BN connecting domain(n-1) to domain(n) along a determined sequence of domains.

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

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

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

LSR: Label Switching Router.

LSP: Label Switched Path.

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

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

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

TED: Traffic Engineering Database.

VSPT: Virtual Shortest Path Tree.

P2MP LSP Path Tree: A set of LSRs and TE links that comprise the path of a P2MP TE LSP from its ingress LSR to all of its egress LSRs.

Core Tree: The core tree is a P2MP tree where the root is the ingress LSR, the transit node and branch node are the BNs of the transit domains and the leaf nodes are the leaf BNs of the leaf domains.

Root Boundary Node: The egress LSR from the root domain on the path of the P2MP LSP.

Root Domain: The domain that includes the ingress (root) LSR.

Transit/branch Domain: A domain that has an upstream and one or more downstream neighbour domain.

Leaf Domain: A domain that doesn’t has a downstream neighbor domain.

Leaf Boundary Nodes: The entry boundary node in the leaf domain.

Leaf Nodes: The LSR which is the P2MP LSP’s final
Destination. The lead Nodes can be in Root Domain, Transit Domain and Leaf Domain.

OF: Objective Function: A set of one or more optimization criterion (criteria) used for the computation of a single path (e.g. path cost minimization), or the synchronized computation of a set of paths (e.g. aggregate bandwidth consumption minimization, etc.). See [RFC4655] and [PCE-OF].

Path Domain Sequence: The known sequence of domains for a path between root and leaf.

PCE Sequence: The known sequence of PCEs for calculating a path between root and leaf.

PCE Topology Tree: A list of PCE Sequences which has all the PCE Sequence for each path of the P2MP LSP path tree.

PCE(i): A PCE that performs path computations for domain(i).

VSPT: Virtual Shortest Path Tree [RFC5441].

3. Problem Statement

The Path Computation Element (PCE) defined in [RFC4655] is an entity that is capable of computing a network path or route based on a network graph, and applying computational constraints. A Path Computation Client (PCC) may make requests to a PCE for paths to be computed.

[RFC4875] describes how to set up P2MP TE LSPs for use in MPLS and GMPLS networks. The PCE is identified as a suitable application for the computation of paths for P2MP TE LSPs [RFC5671].

[RFC5441] specifies a procedure relying on the use of multiple PCEs to compute (P2P) inter-domain constrained shortest paths across a predetermined sequence of domains, using a backward recursive path computation technique. The technique can be combined with the use of path keys [RFC5520] to preserve confidentiality across domains, which is sometimes required when domains are managed by different Service Providers.

The PCE communication Protocol (PCEP) [RFC5440] is extended for point-to-multipoint (P2MP) path computation requests and in [PCE-P2MP-EXT]. However, that specification does not provide all the necessary mechanisms to request the computation of inter-domain P2MP TE LSPs.
As discussed in [RFC4461], a P2MP tree is a graphical representation of all TE links that are committed for a particular P2MP LSP. In other words, a P2MP tree is a representation of the corresponding P2MP tunnel on the TE network topology. A sub-tree is a part of the P2MP tree describing how the root or an intermediate P2MP LSP minimizes packet duplication when P2P TE sub-LSPs traverse common links. As described in [RFC5671] the computation of a P2MP tree requires three major pieces of information. The first is the path from the ingress LSR of a P2MP LSP to each of the egress LSRs, the second is the traffic engineering related parameters, and the third is the branch capability information.

Generally, an inter-domain P2MP tree (i.e., a P2MP tree with source and at least one destination residing in different domains) is particularly difficult to compute even for a distributed PCE architecture. For instance, while the BRPC recursive path computation may be well-suited for P2P paths, P2MP path computation involves multiple branching path segments from the source to the multiple destinations. As such, inter-domain P2MP path computation may result in a plurality of per-domain path options that may be difficult to coordinate efficiently and effectively between domains. That is, when one or more domains have multiple ingress and/or egress border nodes, there is currently no known technique for one domain to determine which border routers another domain will utilize for the inter-domain P2MP tree, and no way to limit the computation of the P2MP tree to those utilized border nodes.

A trivial solution to the computation of inter-domain P2MP tree would be to compute shortest inter-domain P2P paths from source to each destination and then combine them to generate an inter-domain, shortest-path-to-destination P2MP tree. This solution, however, cannot be used to trade cost to destination for overall tree cost (i.e., it cannot produce a Steiner tree) and in the context of inter-domain P2MP LSPs it cannot be used to reduce the number of domain border nodes that are transited.

Computing P2P LSPs individually is not an acceptable solution for computing a P2MP tree. Even per domain path computation [RFC5152] can be used to compute P2P multi-domain paths, but it does not guarantee to find the optimal path which crosses multiple domains. Furthermore, constructing a P2MP tree from individual source to leaf P2P LSPs does not guarantee to produce a least-cost tree. This approach may also be considered to have scaling issues during LSP setup. That is, the LSP to each leaf is signaled separately, and each border node must perform path computation for each leaf.

Apart from path computation difficulties faced due to the inter-domain topology visibility issues, the computation of the minimum
P2MP Steiner tree, i.e. one which guarantees the least cost resulting tree, is an NP-complete problem. Moreover, adding and/or removing a single destination to/from the tree may result in an entirely different tree. In this case, the frequent Steiner I tree computation process may prove computationally intensive, and the resulting frequent tunnel reconfiguration may even cause network destabilization. There are several heuristic algorithms presented in the literature that approximate the result within polynomial time that are applicable within the context of a single-domain.

This document presents a solution, and procedures and extensions to PCEP to support P2MP inter-domain path computation.

4. Assumptions

It is assumed that due to deployment and commercial limitations (e.g., inter-AS peering agreements) the sequence of domains for a path (the path domain tree) will be known in advance.

The examples and scenarios used in this document are also based on the following assumptions:

- The PCE that serves each domain in the path domain tree is known, and the set of PCEs and their relationships is propagated to each PCE during the first exchange of path computation requests;
- Each PCE knows about any leaf LSRs in the domain it serves;
- The boundary nodes to use on the LSP are pre-determined and form path of the path domain tree. In this version of the document we do not consider multi-homed domains.

Additional assumptions are documented in [RFC5441] and will not be repeated here.

5. Requirements

This section summarizes the requirements specific to computing inter-domain P2MP paths. In these requirements we note that the actual computation times by any PCE implementation are outside the scope of this document, but we observe that reducing the complexity of the required computations has a beneficial effect on the computation time regardless of implementation. Additionally, reducing the number of message exchanges and the amount of information exchanged will reduce the overall computation time for the entire P2MP tree. We refer to the "Complexity of the computation" as the impact on these aspects of
path computation time as various parameters of the topology and the P2MP LSP are changed.

It's also important that the solution preserves confidentiality across domains, which is required when domains are managed by different Service Providers.

Other than the requirements specified in [RFC5376], a number of requirements specific to P2MP are detailed below:

1. The computed P2MP LSP should be optimal when only considering the paths among the BNs.

2. Grafting and pruning of multicast destinations in a domain should have no impact on other domains and on the paths among BNs.

3. The complexity of the computation for each sub-tree within each domain should be dependent only on the topology of the domain and it should be independent of the domain sequence.

4. The number of PCEP request and reply messages should be independent of the number of multicast destinations in each domain.

5. Specifying the domain entry and exit nodes.

6. Specifying which nodes should be used as branch nodes.

7. Reoptimization of existing sub-trees.

8. Computation of P2MP paths that need to be diverse from existing P2MP paths.

6. Object Functions

During the computation of a single or a set of P2MP TE LSPs a request to meet specific optimization criteria, called an Objective Function (OF), may be requested.

The computation of one or more P2MP TE-LSPs maybe subject to an OF in order to select the "best" candidate paths. A variety of objective functions have been identified as being important during the computation of inter-domain P2MP LSPs. These are:

1. The sub-tree within each domain should be optimized, which can be either the Minimum cost tree [PCE-P2MP-REQ] or Shortest path tree [PCE-P2MP-REQ].
2. The P2MP LSP paths should be optimal while only considering the entry and exit nodes of each domain as the transit, branch and leaf nodes of the P2MP LSP path. (That is, the Core Tree should be optimized.)

3. It should be possible to limit the number of entry points to a domain.

4. It should be possible to force the branches for all leaves within a domain to be in that domain.

7. P2MP Path Computation Procedures

The following sections describe the core tree based procedures to satisfy the requirements specified in the previous section.

A core tree based solution provides an optimal inter-domain P2MP TE LSP and meets the requirements and OFs outlined in previous sections.

A core tree is a path tree with nodes from each domain corresponding to the PCE topology which satisfies the following conditions:

- The root of the core tree is the ingress LSR in the root domain;
- The leaf of the core tree is the entry node in the leaf domain;
- The transit and branch nodes of the core tree are from the entry and exit nodes from the transit and branch domains.

Procedure Phase 1: Build the P2MP LSP Core Tree.

In this phase, the core tree, where the root is the ingress LSR, the transit node and branch node are the BNs of the transit domains and the leaf nodes are the leaf BNs of the leaf domains, is computed.

Procedure Phase 2: Grafting destinations to the P2MP LSP Core Tree.

Once the core tree is built, the grafting of all the leaf nodes from each domain to the core tree can be achieved by a number of algorithms. One algorithm for doing this phase is that the root PCE will send the request with C bit set for the path computation to the destination(s) directly to the PCE where the destination(s) belong(s) along with the core tree computed from the phase 1.
7.1. Core Tree Computation Procedures

Computing the complete P2MP LSP path tree is done in two phases:

The algorithms to compute the optimal large core tree are outside scope of this document. In the case that the number of domains and the number of BNs are not big, the following extended BRPC based procedure can be used to compute the core tree.

BRPC Based Core Tree Path Computation Procedure:

1. Using the BRPC procedures to compute the VSPT(i) for each leaf BN(i), i=1 to n, where n is the total number of entry nodes for all the leaf domains. In each VSPT(i), there are a number of P(i) paths.

2. When the root PCE has computed all the VSPT(i), i=1 to n, take one path from each VSPT and form a set of paths, we call it a PathSet(j), j=1 to M, where M=P(1)xP(2)...xP(n);

3. For each PathSet(j), there are n S2L (Source to Leaf BN) paths and form these n paths into a Core Tree(j);

4. There will be M number of Core Trees computed from step3. Apply the OF to each of these M Core Trees and find the optimal Core Tree.

7.2. Sub Tree Computation Procedures

The algorithms to compute the optimal large sub tree are outside scope of this document. In the case that the number of destinations and the number of BNs within a domain are not big, the incremental procedure based on p2p path computation usign the OSPF can be used.

7.3. PCEP Protocol Extensions

7.3.1. The Extension of RP Object

The extended format of the RP object body to include the C bit is as follows:

The C bit is added in the flag bits field of the RP object to signal the receiver of the message that the request/reply is for inter-domain P2MP Core Tree or not.
The following flag is added in this draft:

C bit (P2MP Core Tree bit - 1 bit):

0: This indicates that this is normal PCReq/PCRrep for P2MP.

1: This indicates that this is PCReq or PCRep message for inter-domain Core Tree P2MP. When the C bit is set, then the request message should have the Core Tree passed along with the destinations which and then graphed to the tree.

7.3.2. The PCE Sequence Object

The PCE Sequence Object is added to the existing PCE protocol. A list of this objects will represent the PCE topology tree. A list of Sequence Objects can be exchanged between PCEs during the PCE capability exchange or on the first path computation request message between PCEs. In this case, the request message format needs to be changed to include the list of PCE Sequence Objects for the PCE inter-domain P2MP calculation request.

Each PCE Sequence can be obtained from the domain sequence for a specific path. All the PCE sequences for all the paths of P2MP inter-domain form the PCE Topology Tree of the P2MP LSP.

The format of the new PCE Sequence Object for IPv4 (Object-Type 3) is as follows:
The format of the new PCE Sequence Object for IPv6 (Object-Type 3) is as follows:

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Object-Class | OT |Res|P|I|   Object Length (bytes)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address for root PCE                                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address for the downstream PCE                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address for the downstream PCE                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| !!                                                                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| IPv6 address for the PCE corresponding to the leafDomain             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: The New PCE Sequence Object Body Format for IPv6
8. Manageability Considerations

[PCE-P2MP-REQ] describes various manageability requirements in support of P2MP path computation when applying PCEP. This section describes how manageability requirements mentioned in [PCE-P2MP-REQ] are supported in the context of PCEP extensions specified in this document.

Note that [RFC5440] describes various manageability considerations in PCEP, and most of manageability requirements mentioned in [PCE-P2MP-REQ] are already covered there.

9. Control of Function and Policy

In addition to configuration parameters listed in [RFC5440], the following parameters MAY be required.

- P2MP path computations enabled or disabled.
- Advertisement of P2MP path computation capability enabled or disabled (discovery protocol, capability exchange).

10. Information and Data Models

As described in [PCE-P2MP-REQ], MIB objects MUST be supported for PCEP extensions specified in this document.

11. Liveness Detection and Monitoring

There are no additional considerations beyond those expressed in [RFC5440], since [PCE-P2MP-REQ] does not address any additional requirements.

12. Verifying Correct Operation

There are no additional considerations beyond those expressed in [RFC5440], since [PCE-P2MP-REQ] does not address any additional requirements.

13. Requirements on Other Protocols and Functional Components

As described in [PCE-P2MP-REQ], the PCE MUST obtain information about the P2MP signaling and branching capabilities of each LSR in the
network.

Protocol extensions specified in this document does not provide such capability. Other mechanisms MUST be present.


It is expected that use of PCEP extensions specified in this document will not have significant impact on network operations.

15. Security Considerations

As described in [PCE-P2MP-REQ], P2MP path computation requests are more CPU-intensive and also use more link bandwidth. Therefore, it may be more vulnerable to denial of service attacks. Therefore, it is more important that implementations conform to security requirements of [RFC5440], and the implementer utilize those security features.

16. IANA Considerations

A new flag of the RP object (specified in [RFC5440]) is defined in this document.

TBD.

17. Acknowledgements

The authors would like to thank Adrian Farrel and Dan Tappan for their valuable comments on this draft.

18. References

18.1. Normative References


18.2. Informative References

Authors’ Addresses

Quintin Zhao
Huawei Technology
125 Nagog Technology Park
Acton, MA 01719
US
Email: qzhao@huawei.com

Zafar Ali
Cisco Systems
Email: zali@cisco.com

Tarek Saad
Cisco Systems
US
Email: tsaad@cisco.com

Daniel King
Old Dog Consulting
UK
Email: daniel@olddog.co.uk

Contributors’ Addresses

David Amzallag
British Telecommunications plc
UK
Email: david.Amzallag@bt.com

Fabien Verhaeghe
Thales Communication France
160 Bd Valmy 92700 Colombes
France
Email: fabien.verhaeghe@gmail.com

Kenji Kumaki
KDDI R&D Laboratories, Inc.
Japan
Email: ke-kumaki@kddi.com