Workgroup: TEAS Working Group
Internet-Draft:
draft-ietf-teas-pcecc-use-cases-13
Published: 8 January 2023
Intended Status: Informational
Expires: 12 July 2023
Authors: Z. Li
Huawei Technologies
D. Dhody
Huawei Technologies
Q. Zhao
Etheric Networks
K. He
Tencent Holdings Ltd.
B. Khasanov
Yandex LLC

The Use Cases for Path Computation Element (PCE) as a Central Controller (PCECC).

Abstract

The Path Computation Element (PCE) is a core component of a Software-Defined Networking (SDN) systems. It can compute optimal paths for traffic across a network and update paths to reflect changes in the network or traffic demands. PCE was developed to derive paths for MPLS Label Switched Paths (LSPs), which are supplied to the head end of the LSP using the Path Computation Element Communication Protocol (PCEP).

SDN has a much broader applicability than signal MPLS traffic-engineered (TE) paths, and the PCE may be used to determine paths in a range of use cases including static LSPs, Segment Routing (SR), Service Function Chaining (SFC), and most forms of a routed or switched network. It is, therefore, reasonable to consider PCEP as a control protocol for use in these environments to allow the PCE to be fully enabled as a central controller.

A PCE as a Central Controller (PCECC) can simplify the processing of a distributed control plane by blending it with elements of SDN and without necessarily completely replacing it. This document describes general considerations for PCECC deployment and examines its applicability and benefits, as well as its challenges and limitations, through a number of use cases. PCEP extensions which required for the PCECC use cases are covered in separate documents.

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 12 July 2023.

Copyright Notice

Table of Contents

1. Introduction

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

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

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

[RFC9050] introduces the procedures and extensions for PCEP to support the PCECC architecture [RFC8283].

This document describes the various use cases for the PCECC architecture.

2. Terminology

The following terminology is used in this document.

IGP: Interior Gateway Protocol. In the document we assume either Open Shortest Path First (OSPF) [RFC2328][RFC5340] or Intermediate System to Intermediate System (IS-IS) [RFC1195] as IGP.

PCC: Path Computation Client. As per [RFC4655], any client application requesting a path computation to be performed by a Path Computation Element.

PCE: Path Computation Element. As per [RFC4655], 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.

PCECC: PCE as a central controller. Extension of PCE to support SDN functions as per [RFC8283].

TE: Traffic Engineering [I-D.ietf-teas-rfc3272bis].

3. Use Cases

[RFC8283] describes various use cases for PCECC such as:

*Use of PCECC to set up Static TE LSPs. The PCEP extension for this use case is in [RFC9050].

*Use of PCECC in Segment Routing [RFC8402].

*Use of PCECC to set up Multicast Point-to-Multipoint (P2MP) LSP.

*Use of PCECC to set up Service Function Chaining (SFC) [RFC7665].

*Use of PCECC in Optical Networks.

Section 3.1 describe the general case of PCECC being in charge of managing MPLS label space which is a prerequisite for further use cases. Further, various use cases (SR, Multicast etc) are described in the following sections to showcase scenarios that can benefit from the use of PCECC.

3.1. PCECC for Label Management

As per [RFC8283], in some cases, the PCE-based controller can take responsibility for managing some part of the MPLS label space for each of the routers that it controls, and it may take wider responsibility for partitioning the label space for each router and

allocating different parts for different uses, communicating the ranges to the router using PCEP.

[RFC9050] describes a mode where LSPs are provisioned as explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what label forwarding instructions to program and what resources to reserve. The controller uses PCEP to communicate with each router along the path of the end-to-end LSP. For this to work, the PCE-based controller will take responsibility for managing some part of the MPLS label space for each of the routers that it controls. An extension to PCEP could be done to allow a PCC to inform the PCE of such a label space to control. (See [I-D.li-pce-controlled-id-space] for a possible PCEP extension to support advertisement of the MPLS label space to the PCE to control.)

[RFC8664] specifies extensions to PCEP that allow a stateful PCE to compute, update or initiate SR-TE paths. [I-D.ietf-pce-pcep-extension-pce-controller-sr] describes the mechanism for PCECC to allocate and provision the node/prefix/ adjacency label (Segment Routing Identifier (SID)) via PCEP. To make such allocation PCE needs to be aware of the label space from Segment Routing Global Block (SRGB) or Segment Routing Local Block (SRLB) [RFC8402] of the node that it controls. A mechanism for a PCC to inform the PCE of such a label space to control is needed within PCEP. The full SRGB/SRLB of a node could be learned via existing IGP or BGP-LS mechanisms too.

Further, there have been proposals for a global label range in MPLS, the PCECC architecture could be used as means to learn the label space of nodes, and could also be used to determine and provision the global label range.

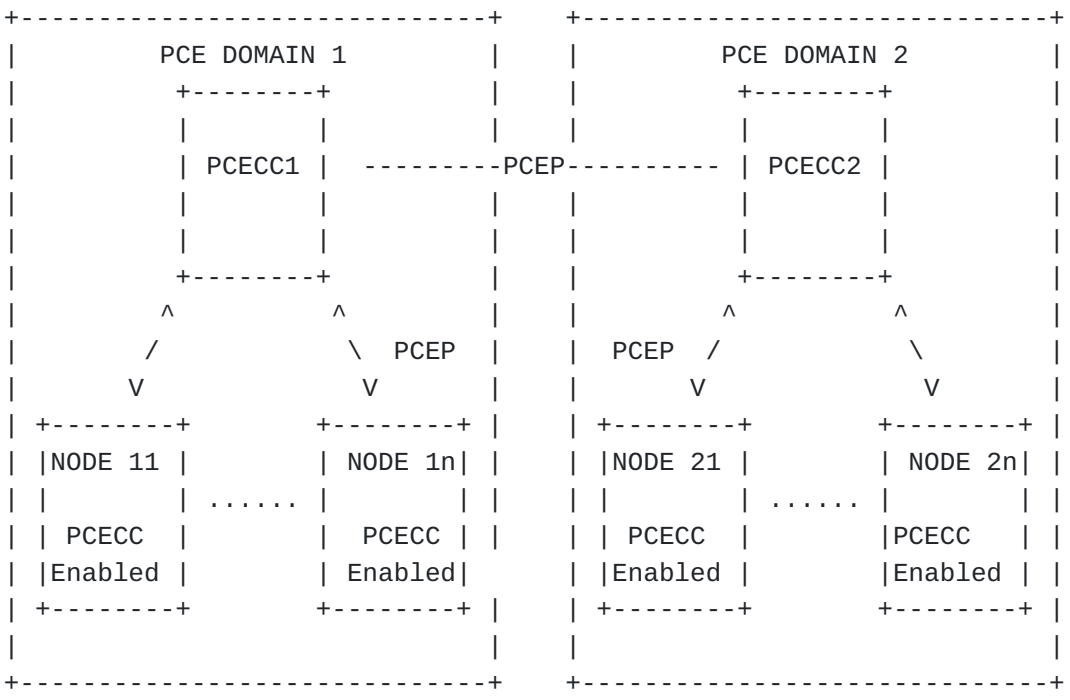

Figure 1: PCECC for Label Management

*As shown in Figure 1, PCC will advertise the PCECC capability to the PCE central controller (PCECC) [RFC9050].

*The PCECC could also learn the label range set aside by the PCC ([I-D.li-pce-controlled-id-space]).

*Optionally, the PCECC could determine the shared MPLS global label range for the network.

-In the case that the shared global label range need to be negotiated across multiple domains, the central controllers of these domains will also need to negotiate a common global label range across domains.

-The PCECC will need to set the shared global label range to all PCC nodes in the network.

As per [RFC9050], PCECC could also rely on the PCC to make label allocations initially and use PCEP to distribute it to where it is needed.

3.2. PCECC and Segment Routing

Segment Routing (SR) leverages the source routing paradigm. Using SR, a source node steers a packet through a path without relying on hop-by-hop signaling protocols such as LDP [RFC5036] or RSVP-TE [RFC3209]. Each path is specified as an ordered list of instructions

called "segments". Each segment is an instruction to route the packet to a specific place in the network, or to perform a specific service on the packet. A database of segments can be distributed through the network using a routing protocol (such as IS-IS or OSPF) or by any other means. PCEP (and PCECC) could also be one of them.

[RFC8664] specifies the SR specific PCEP extensions. PCECC may further use PCEP protocol for SR SIDs (Segment Identifiers) distribution to the SR nodes (PCC) with some benefits. If the PCECC allocates and maintains the SIDs in the network for the nodes and adjacencies; and further distributes them to the SR nodes directly via the PCEP session then it is more advantageous over the configurations on each SR node and flooding them via IGP, especially in a SDN environment.

When the PCECC is used for the distribution of the Node-SID and Adj-SID, the Node-SID is allocated from the SRGB of the node. For the allocation of Adj-SID, the allocation is from the SRLB of the node as described in [I-D.ietf-pce-pcep-extension-pce-controller-sr].

[RFC8355] identifies various protection and resiliency usecases for SR. Path protection lets the ingress node be in charge of the failure recovery (used for SR-TE). Also protection can be performed by the node adjacent to the failed component, commonly referred to as local protection techniques or fast-reroute (FRR) techniques. In case of PCECC, the protection paths can be pre-computed and setup by the PCE.

The Figure 2 illustrates the use case where the Node-SID and Adj-SID are allocated by the PCECC.

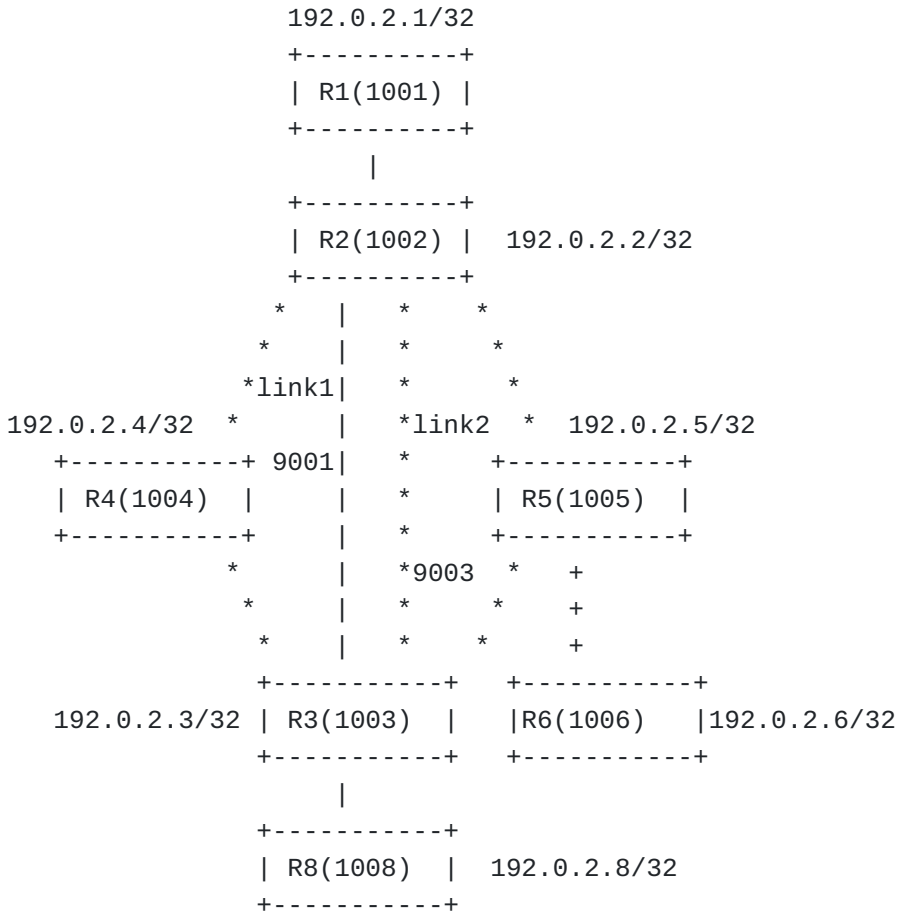

Figure 2: SR Topology

3.2.1. PCECC SID Allocation

Each node (PCC) is allocated a Node-SID by the PCECC. The PCECC needs to update the label mapping of each node to all the other nodes in the domain. After receiving the label mapping, each node (PCC) uses the local routing information to determine the nexthop and download the label forwarding instructions accordingly. The forwarding behavior and the end result is the same as IGP shortest-path SR forwarding based on Node-SID. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node.

For each adjacency in the network, a PCECC can allocate an Adj-SID. The PCECC sends a PCInitiate message to update the label mapping of each adjacency to the corresponding nodes in the domain. Each node (PCC) downloads the label forwarding instructions accordingly. The forwarding behavior and the end result are similar to IGP-based Adj-SID allocation and usage in SR.

These mechanisms are described in [I-D.ietf-pce-pcep-extension-pce-controller-sr].

3.2.2. PCECC for SR Best Effort (BE) Path

In this use case, the PCECC just needs to allocate the Node-SID (without calculating the explicit path for the SR path). The ingress router of the forwarding path just needs to encapsulate the destination Node-SID on top of the packet. All the intermediate nodes will forward the packet based on the destination Node-SID. It is similar to the LDP LSP.

R1 may send a packet to R8 simply by pushing an SR label with segment {1008} (Node-SID for R8). The path will be based on the routing/nexthop calculation on the routers.

3.2.3. PCECC for SR-TE Path

SR-TE paths may not follow an IGP SPT. Such paths may be chosen by a PCECC and provisioned on the ingress node of the SR-TE path. The SR header consists of a list of SIDs (or MPLS labels). The header has all necessary information so that, the packets can be guided from the ingress node to the egress node of the path; hence, there is no need for any signalling protocol. For the case where strict traffic engineering path is needed, all the Adj-SID are stacked, otherwise a combination of node-SID or adj-SID can be used for the SR-TE paths.

R1 may send a packet to R8 by pushing an SR header with segment list {1002, 9001, 1008}. Where, 1002 and 1008 are the Node-SID of R2 and R8 respectively. 9001 is the Adj-SID for the link1. The path should be: R1-R2-link1-R3-R8.

To achieve this, the PCECC first allocates and distribute SIDs as described in Section 3.2.1. [RFC8664] describe the mechanism for a PCE to compute, update, or initiate SR-TE paths.

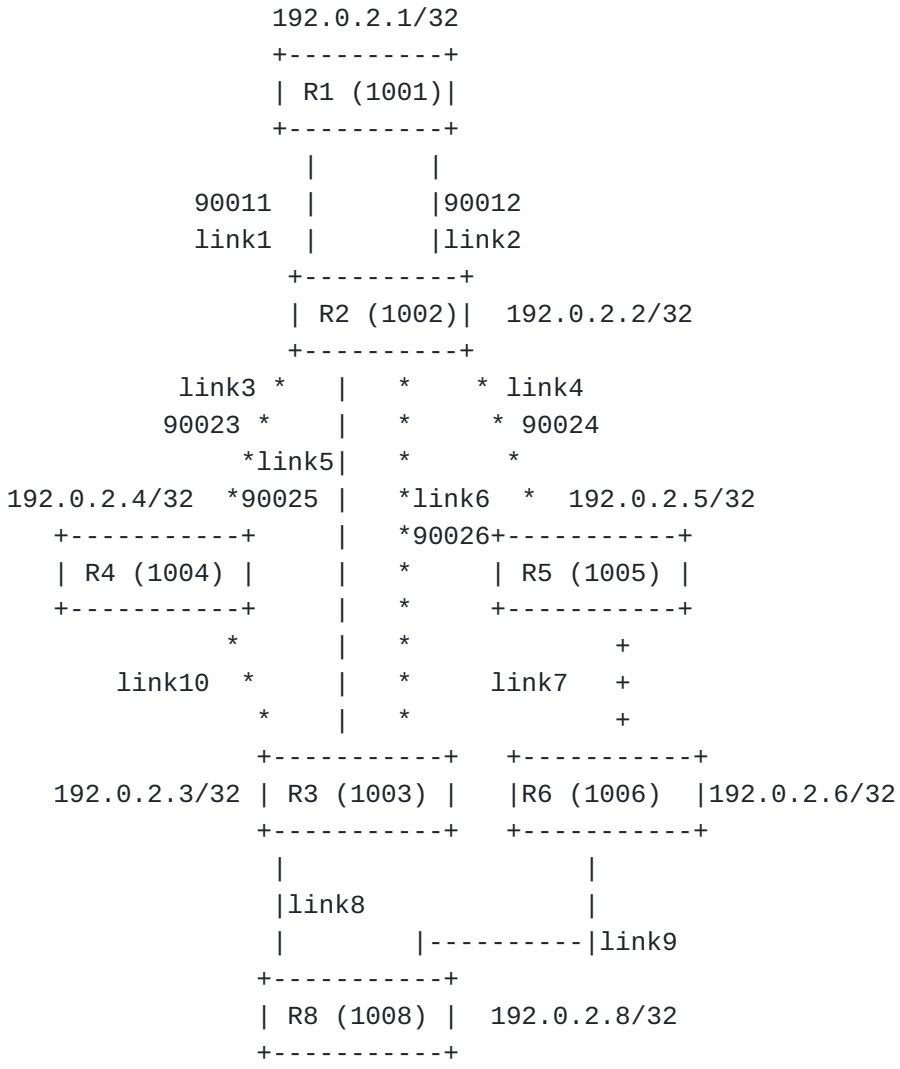

Figure 3: PCECC TE LSP Setup Example

Refer Figure 3 for an example of TE topology, where, 100x - are Node-SIDs and 900xx - are Adj-SIDs.

*The SID allocation and distribution are done by the PCECC with all Node-SIDs (100x) and all Adj-SIDs (900xx).

*Based on path computation request/delegation or PCE initiation, the PCECC receives a request with constraints and optimization criteria from a PCC.

*PCECC will calculate the optimal path according to given constrains (e.g. bandwidth).

*PCECC will provision SR-TE LSP (path R1-link1-R2-link6-R3-R8) at the ingress node: {90011,1002,90026,1003,1008}

*For the end-to-end protection, PCECC can provision the secondary path (R1-link2-R2-link4-R5-R8): {90012,1002,90024,1005,1008}.

3.2.3.1. PCECC for SR Policy

[RFC8402] defines Segment Routing architecture, which uses an SR Policy to steer packets from a node through an ordered list of segments. The SR Policy could be configured on the headed or instantiated by an SR controller. The SR architecture does not restrict how the controller programs the network. In this case, the focus is on PCEP as the protocol for SR Policy delivery from PCE to PCC.

An SR Policy architecture is described in [RFC9256]. An SR Policy is a framework that enables the instantiation of an ordered list of segments on a node for implementing a source routing policy for the steering of traffic for a specific purpose (e.g. for a specific SLA) from that node.

An SR Policy is identified through the tuple <headend, color, endpoint>.

Figure 3 is used as an example of PCECC application for SR Policy instantiation, where, 100x - are Node-SIDs and 900xx - are Adj-SIDs.

Let's assume that R1 needs to have two disjoint SR Policies towards R8 based on different bandwidth, the possible paths are:

POL1: {Headend R1, color 100, Endpoint R8; Candidate Path1: Segment List 1: {90011,1002,90023,1004,1003,1008}}

POL2: {Headend R1, color 200, Endpoint R8; Candidate Path1: Segment List 1: {90012,1002,90024,1005,1006,1008}}

Each SR Policy (including candidate path and segment list) will be signaled to a headend (R1) via PCEP [I-D.ietf-pce-segment-routing-policy-cp] with addition of an ASSOCIATION object. Binding SID (BSID) [RFC8402] can be used for traffic steering of labelled traffic into SR Policy, BSID can pe provisioned from PCECC also via PCEP [I-D.ietf-pce-binding-label-sid]. For non-labelled traffic steering into the SR Policy POL1 or POL2 a per-destination traffic steering will be used by means of BGP Color extended community [RFC9012]

The procedure:

PCECC allocates Node-SIDs and Adj-SIDs as described in Section 3.1 for all nodes and links.

PCECC will calculate disjoint paths for POL1 and POL2 and create Segment Lists for them:{90011,1002,90023,1004,1003,1008}; {90012,1002,90024,1005,1006,1008}.

PCECC will form both SR Policies POL1 and POL2.

PCECC will send both POL1 and POl2 to R1 via PCEP.

PCECC optionally can allocate BSIDs for the SR Policies.

The traffic from R1 to R8 which fits to color 100 will be steered to POL1 and follows the path: R1-link1-R2-link3-R4-R3-R8. The traffic from R1 to R8 which fits to color 200 will be steered to POL2 and follows the path: R1-link2-R2-link4-R5-R6-R8. Due to the possibility to have many Segment Lists in the same Candidate Path of each POL1/POL2, PCECC could provision more paths towards R8 and traffic will be balanced either as ECMP or as w/ECMP. This is the advantage of SR Policy architecture.

3.2.4. PCECC for SRv6

As per [RFC8402], with Segment Routing (SR), a node steers a packet through an ordered list of instructions, called segments. Segment Routing can be applied to the IPv6 architecture with the Segment Routing Header (SRH) [RFC8754]. A segment is encoded as an IPv6 address. An ordered list of segments is encoded as an ordered list of IPv6 addresses in the routing header. The active segment is indicated by the Destination Address of the packet. Upon completion of a segment, a pointer in the new routing header is incremented and indicates the next segment.

As per [RFC8754], an SRv6 Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". Further details are in An illustration is provided in [RFC8986] where SRv6 SID is represented as LOC:FUNCT.

[I-D.ietf-pce-segment-routing-ipv6] extends [RFC8664] to support SR for IPv6 data plane. Further a PCECC could be extended to support SRv6 SID allocation and distribution. PCECC PCEP extensions for SRv6 [I-D.dhody-pce-pcep-extension-pce-controller-srv6] will be used for that.

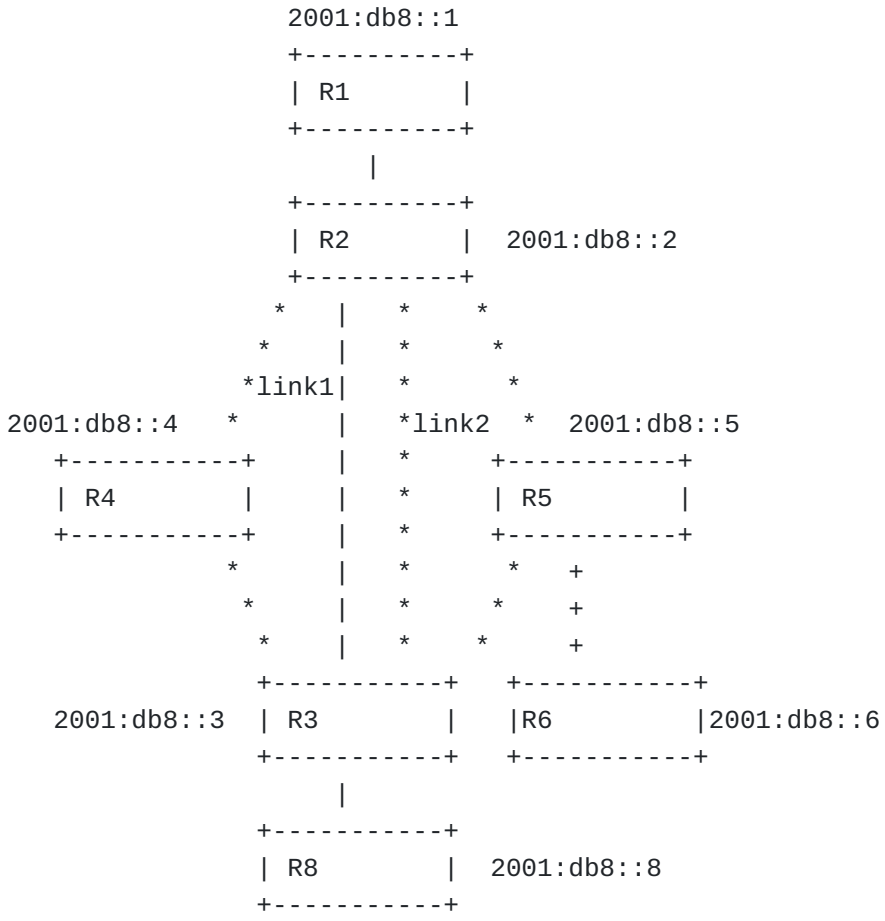

Figure 4: PCECC for SRv6

In the case, as shown in Figure 4, PCECC could assign the SRv6 SID (in form of an IPv6 address) to be used for node and adjacency. Later SRv6 path in form of a list of SRv6 SID could be used at the ingress. Some examples -

*SRv6 SID-List={2001:db8::8} - The best path towards R8

*SRv6 SID-List={2001:db8::5, 2001:db8::8} - The path towards R8 via R5

The rest of the procedures and mechanisms remain the same as SR-MPLS.

3.3. PCECC for Static TE LSP

As described in Section 3.1.2 of [RFC8283], PCECC architecture support the provisioning of static TE LSP. To achieve this, the existing PCEP can be used to communicate between the PCECC and nodes along the path to provision explicit label instructions at each hop on the end-to-end path. Each router along the path must be told what

label-forwarding instructions to program and what resources to reserve. The PCE-based controller keeps a view of the network and determines the paths of the end-to-end LSPs, and the controller uses PCEP to communicate with each router along the path of the end-to-end LSP.

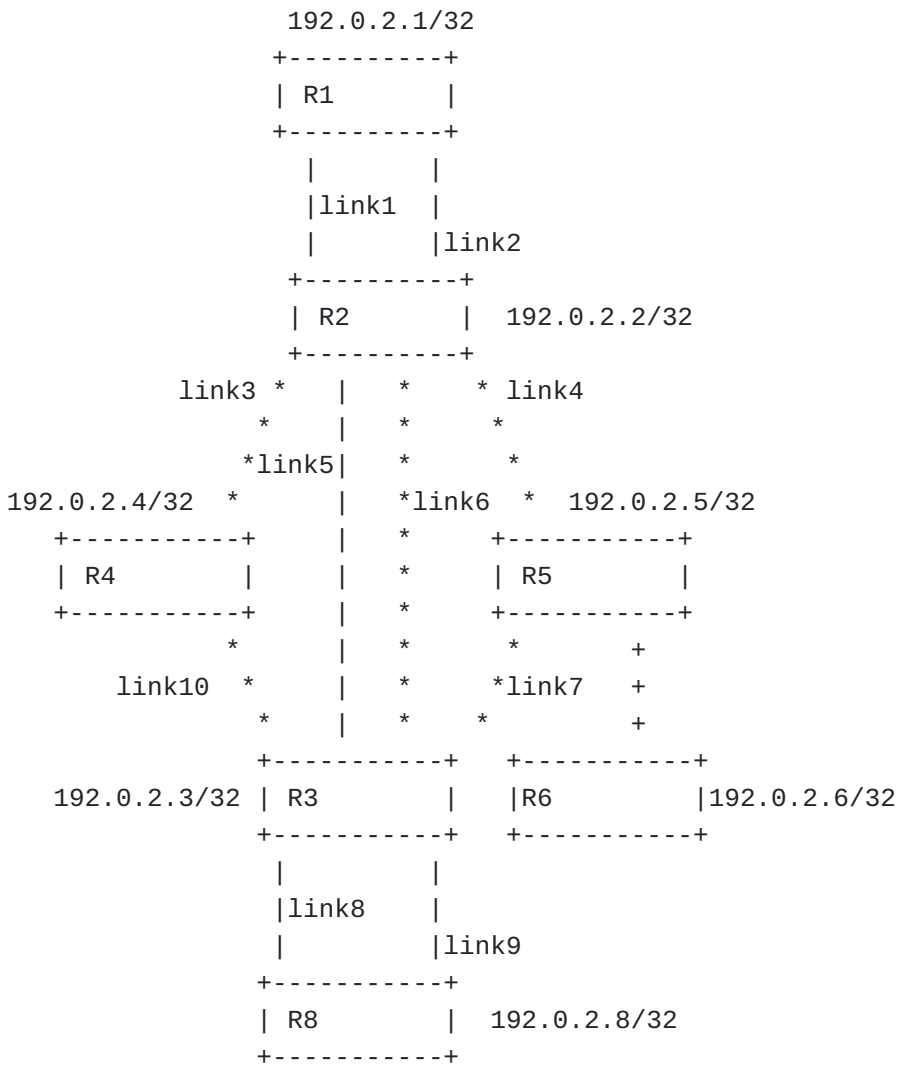

Figure 5: PCECC TE LSP Setup Example

Refer Figure 5 for an example TE topology.

*Based on path computation request/delegation or PCE initiation, the PCECC receives a request with constraints and optimization criteria.

*PCECC will calculate the optimal path according to given constrains (e.g. bandwidth).

*PCECC will provision each node along the path and assign incoming and outgoing labels from R1 to R8 with the path as "R1-link1-R2-link3-R4-link10-R3-link8-R8":

-R1: Outgoing label 1001 on link 1

-R2: Incoming label 1001 on link 1

-R2: Outgoing label 2003 on link 3

-R4: Incoming label 2003 on link 3

-R4: Outgoing label 4010 on link 10

-R3: Incoming label 4010 on link 10

-R3: Outgoing label 3008 on link 8

-R8: Incoming label 3008 on link 8

*This can also be represented as {R1, link1, 1001}, {1001, R2, link3, 2003], {2003, R4, link10, 4010}, {4010, R3, link8, 3008}, {3008, R8}.

*For the end to end protection, PCECC program each node along the path from R1 to R8 with the secondary path: {R1, link2, 1002}, {1002, R2, link4, 2004], {2004, R5, link7, 5007}, {5007, R3, link9, 3009}, {3009, R8}.

*It is also possible to have a bypass path for the local protection setup by the PCECC. For example, the primary path as above, then to protect the node R4 locally, PCECC can program the bypass path like this: {R2, link5, 2005}, {2005, R3}. By doing this, the node R4 is locally protected at R2.

3.4. PCECC for Load Balancing (LB)

Very often many service providers use TE tunnels for solving issues with non-deterministic paths in their networks. One example of such applications is usage of TEs in the mobile backhaul (MBH). Consider the topology as shown in Figure 6 (AGG1...AGGN are Aggregation Routers, Core 1...Core N are Core routers) -

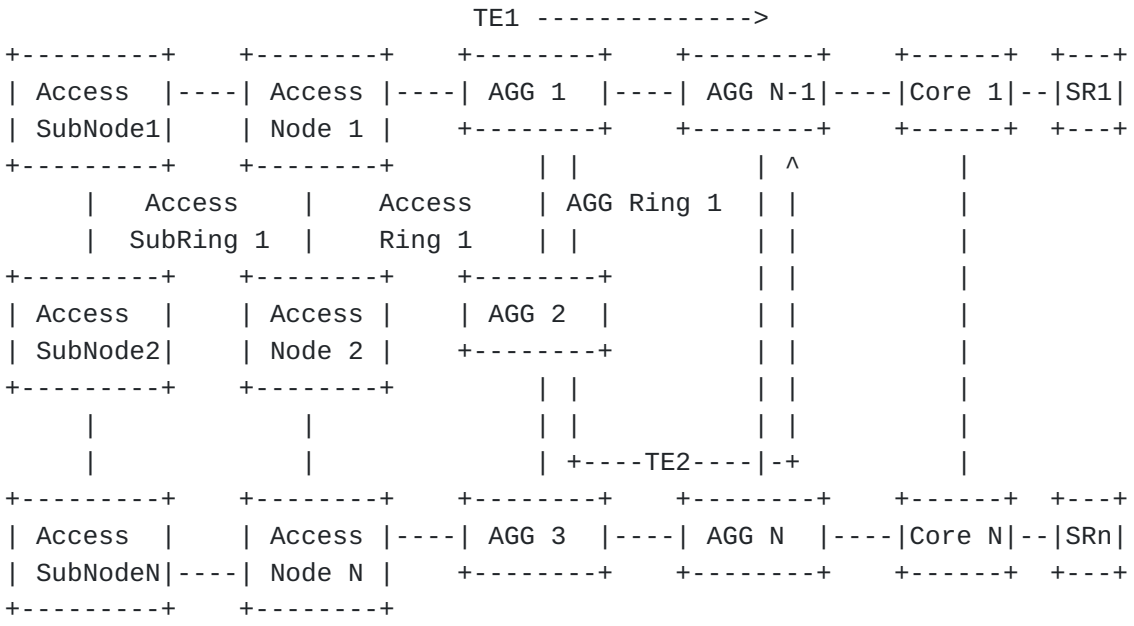

Figure 6: PCECC Load Balancing (LB) Use Case

This MBH architecture uses L2 access rings and sub-rings. L3 starts at the aggregation layer. For the sake of simplicity, the figure shows only one access sub-ring. The access ring and aggregation ring are connected by Nx10GE interfaces. The aggregation domain runs its own IGP. There are two Egress routers (AGG N-1,AGG N) that are connected to the Core domain (Core 1...Core N) via L2 interfaces. Core also has connections to service routers, RSVP-TE or SR-TE is used for MPLS transport inside the ring. There could be at least 2 tunnels (one way) from each AGG router to egress AGG routers. There are also many L2 access rings connected to AGG routers.

Service deployment, made by means of Layer 2 Virtual Private Networks (L2VPNs) (Virtual Private LAN Service (VPLS)), Layer 3 Virtual Private Networks (L3VPNs) or Ethernet VPNs (EVPNs). Those services use MPLS TE (or SR-TE) as transport towards egress AGG routers. TE tunnels could be also used as transport towards service routers in case of seamless MPLS ([I-D.ietf-mpls-seamless-mpls]) based architecture.

There is a need to solve the following tasks:

*Perform automatic load-balance amongst TE tunnels according to current traffic load.

*TE bandwidth (BW) management: Provide guaranteed BW for specific services: High Speed Data Service (HSI)), IPTV, etc., provide time-based BW reservation (BW on demand (BoD)) for other services.

*Simplify the development of TE tunnels by automation without any manual intervention.

*Provide flexibility for Service Router placement (anywhere in the network by the creation of transport LSPs to them).

In this section, the focus is on load balancing (LB) task. LB task could be solved by means of PCECC in the following way:

*Application or network service or operator can ask SDN controller (PCECC) for LSP based load balancing between AGG X and AGG N/AGG N-1 (egress AGG routers which have connections to core). Each of these will have associated constrains (i.e. bandwidth, inclusion or exclusion specific links or nodes, number of paths, objective function (OF), need for disjoint LSP paths etc.);

*PCECC could calculate multiple (say N) LSPs according to given constrains, calculation is based on results of Objective Function (OF) [RFC5541], constraints, endpoints, same or different bandwidth (BW) , different links (in case of disjoint paths) and other constrains.

*Depending on given LSP Path setup type (PST), PCECC will download instructions to the PCC. At this stage it is assumed the PCECC is aware of the label space it controls and SID allocation and distribution is already done in case of SR.

*PCECC will send PCInitiate message [RFC8281] towards ingress AGG X router(PCC) for each of N LSPs and receives PCRpt PCEP message [RFC8231] back from PCCs. If PST is PCECC-SR, the PCECC will include a SID stack as per [RFC8664]. If PST is PCECC (basic), then the PCECC will assign labels along the calculated path and set up the path by sending central controller instructions in PCEP message to each node along the path of the LSP as per [RFC9050] and then send PCUpd message to the ingress AGG X router with information about new LSP. AGG X(PCC) will respond with PCRpt with LSP status.

*AGG X as ingress router now have N LSPs towards AGG N and AGG N-1 which are available for installing to router's forwarding table and load-balance traffic between them. Traffic distribution between those LSPs depends on particular realization of hash-function on that router.

*Since PCECC is aware of TEDB (TE state) and LSP-DB, it can manage and prevent possible over-subscriptions and limit number of available load-balance states. Via PCECC mechanism the control can take quick actions into the network by directly provisioning the central control instructions.

3.5. PCECC and Inter-AS TE

There are various signaling options for establishing Inter-AS TE LSP: contiguous TE LSP [RFC5151], stitched TE LSP [RFC5150], and nested TE LSP [RFC4206].

Requirements for PCE-based Inter-AS setup [RFC5376] describe the approach and PCEP functionality that is needed for establishing Inter-AS TE LSPs.

[RFC5376] also gives Inter- and Intra-AS PCE Reference Model (as shown in Figure 7) that is provided below in shortened form for the sake of simplicity.

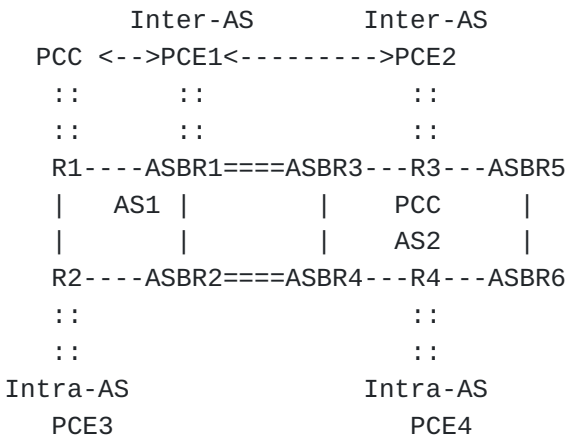

Figure 7: Shortened form of Inter- and Intra-AS PCE Reference Model

The PCECC belonging to the different domains can co-operate to set up inter-AS TE LSP. The stateful H-PCE [RFC8751] mechanism could also be used to establish a per-domain PCECC LSP first. These could be stitched together to form inter-AS TE LSP as described in [I-D.ietf-pce-stateful-interdomain].

For the sake of simplicity, here the focus is on a simplified Inter-AS case when both AS1 and AS2 belong to the same service provider administration. In that case, Inter and Intra-AS PCEs could be combined in one single PCE if such combined PCE performance is enough for handling the load. The PCE will require interfaces (PCEP and BGP-LS) to both domains. PCECC redundancy mechanisms are described in [RFC8283]. Thus routers (PCCs) in AS1 and AS2 can send PCEP messages towards the same PCECC.

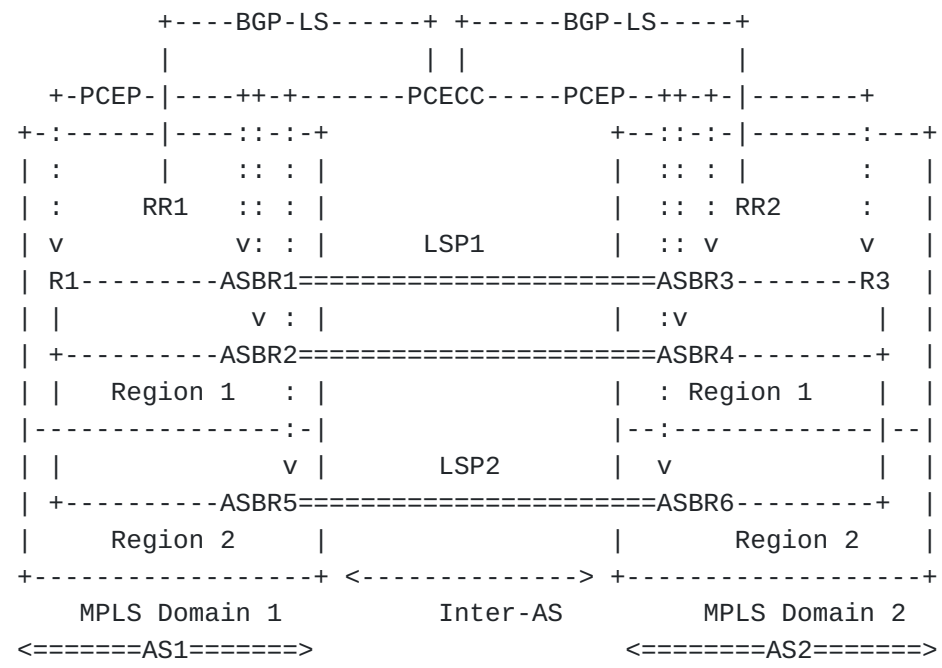

Figure 8: Particular case of Inter-AS PCE

In a case of PCECC Inter-AS TE scenario (as shown in Figure 8) where service provider controls both domains (AS1 and AS2), each of them have own IGP and MPLS transport. There is a need to setup Inter-AS LSPs for transporting different services on top of them (Voice, L3VPN etc.). Inter-AS links with different capacity exist in several regions. The task is not only to provision those Inter-AS LSPs with given constrains but also calculate the path and pre-setup the backup Inter-AS LSPs that will be used if primary LSP fails.

As per the Figure 8, LSP1 from R1 to R3 goes via ASBR1 and ASBR3, and it is the primary Inter-AS LSP. R1-R3 LSP2 that goes via ASBR5 and ASBR6 is the backup one. In addition there could also be a bypass LSP setup to protect against ASBR or inter-AS link failures.

After the addition of PCECC functionality to PCE (SDN controller), PCECC-based Inter-AS TE model should follow the PCECC usecase for TE LSP including requirements of [RFC5376] with the following details:

*Since PCECC needs to know the topology of both domains AS1 and AS2, PCECC can utilize the BGP-LS peering with routers (or RRs) in both domains.

*PCECC needs to establish PCEP connectivity with all routers in both domains (see also section 4 in [RFC5376]).

*After operator's application or service orchestrator will create request for tunnel creation of specific service, PCECC will

receive that request via NBI (NBI type is implementation dependent, could be NETCONF/Yang, REST etc.). Then PCECC will calculate the optimal path based on Objective Function (OF) and given constraints (i.e. path setup type, bandwidth etc.), including those from [RFC5376]: priority, AS sequence, preferred ASBR, disjoint paths, protection type. On this step we will have two paths: R1-ASBR1-ASBR3-R3, R1-ASBR5-ASBR6-R3

*Depending on given LSP PST (PCECC or PCECC-SR), PCECC will use central control download instructions to the PCC. At this stage it is assumed the PCECC is aware of the label space it controls and in case of SR the SID allocation and distribution is already done.

*PCECC will send PCInitiate message [RFC8281] towards the ingress router R1 (PCC) in AS1 and receives PCRpt PCEP message [RFC8231] back from. If the PST is PCECC-SR, the PCECC will include the SID stack as per [RFC8664]. Optionally, a binding SID or BGP Peering-SID [RFC9087] can also be included on the AS boundary. The backup SID stack can be installed at ingress R1 but more importantly each node along the SR path could also do the local protection just based on the top segment. If the PST is PCECC (basic), when the PCECC will assign labels along the calculated paths (R1-ASBR1-ASBR3-R3, R1-ASBR5-ASBR6-R3); and set up the path by sending central controller instructions in PCEP message to each node along the path of the LSPs as per [RFC9050]. Then PCECC will send PCUpd message to the ingress R1 router with an information about new LSPs and R1 will respond by PCRpt with LSP(s) status.

*After that step R1 now have primary and backup TEs (LSP1 and LSP2) towards R3. It is up to router implementation how to make switchover to backup LSP2 if LSP1 fails.

3.6. PCECC for Multicast LSPs

The multicast LSPs can be setup via the RSVP-TE P2MP or Multipoint LDP (mLDP) protocols. The setup of these LSPs may require manual configurations and complex signaling when the protection is considered. By using the PCECC solution, the multicast LSP can be computed and setup through centralized controller which has the full picture of the topology and bandwidth usage for each link. It not only reduces the complex configurations comparing the distributed RSVP-TE P2MP or mLDP signaling, but also it can compute the disjoint primary path and secondary P2MP path efficiently.

3.6.1. PCECC for P2MP/MP2MP LSPs' Setup

It is assumed the PCECC is aware of the label space it controls for all nodes and makes allocations accordingly.

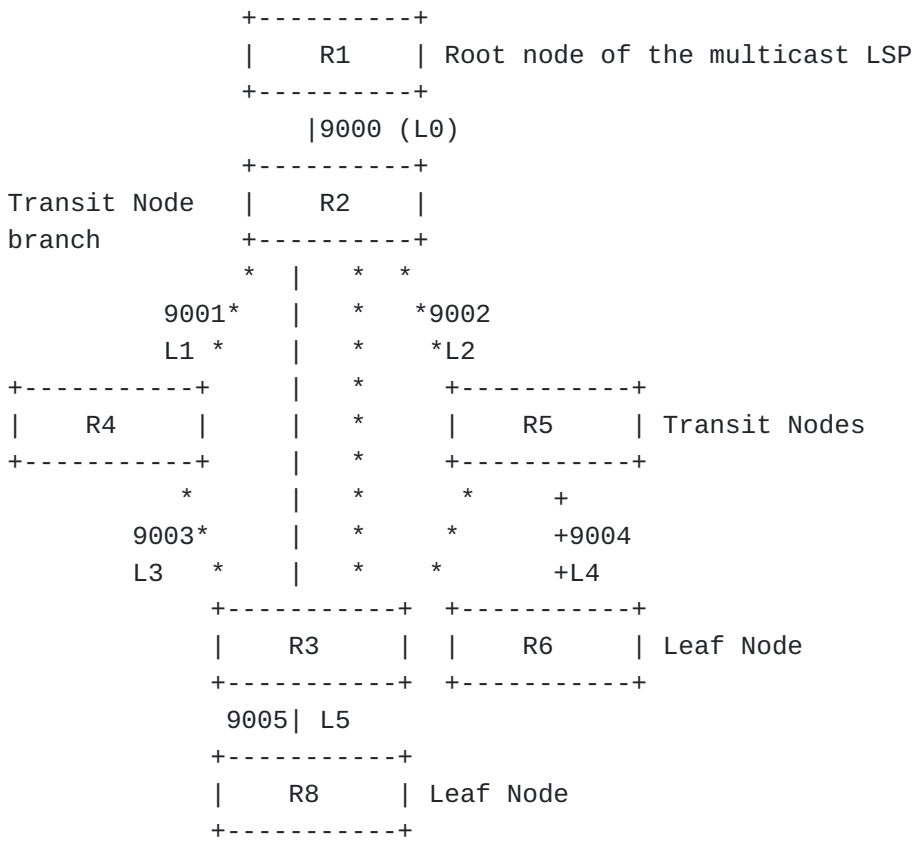

Figure 9: Using PCECC for P2MP/MP2MP LSPs' Setup

The P2MP examples (based on Figure 9) are explained here, where R1 is the root and the router R8 and R6 are the leaves.

*Based on the P2MP path computation request / delegation or PCE initiation, the PCECC receives the request with constraints and optimization criteria.

*PCECC will calculate the optimal P2MP path according to given constrains (i.e.bandwidth).

*PCECC will provision each node along the path and assign incoming and outgoing labels from R1 to {R6, R8} with the path as "R1-L0-R2-L2-R5-L4-R6" and "R1-L0-R2-L1-R4-L3-R3-L5-R8":

-R1: Outgoing label 9000 on link L0

-R2: Incoming label 9000 on link L0

-R2: Outgoing label 9001 on link L1 (*)

-R2: Outgoing label 9002 on link L2 (*)

-R5: Incoming label 9002 on link L2

-R5: Outgoing label 9004 on link L4

-R6: Incoming label 9004 on link L4

-R4: Incoming label 9001 on link L1

-R4: Outgoing label 9003 on link L3

-R3: Incoming label 9003 on link L3

-R3: Outgoing label 9005 on link L5

-R8: Incoming label 9005 on link L5

*This can also be represented as : {R1, 6000}, {6000, R2, {9001,9002}}, {9001, R4, 9003}, {9002, R5, 9004} {9003, R3, 9005}, {9004, R6}, {9005, R8}. The main difference (*) is in the branch node instruction at R2 where two copies of packet are sent towards R4 and R5 with 9001 and 9002 labels respectively.

The packet forwarding involves -

Step 1: R1 sends a packet to R2 simply by pushing the label of 9000 to the packet.

Step 2: When R2 receives the packet with label 9000, it will forward it to R4 by swapping label 9000 to 9001 and at the same time, it will replicate the packet and swap label 9000 to 9002 and forward to R5

Step 3: When R4 receives the packet with label 9001, it will forward it to R3 by swapping 9001 to 9003. When R5 receives the packet with label 9002, it will forward it to R6 by swapping 9002 to 9004.

Step 4: When R3 receives the packet with label 9003, it will forward it to R8 by swapping to 9005 and when R5 receives the packet with label 9002, it will be swapped to 9004 and sent to R6.

Step 5: When R8 receives the packet with label 9005, it will pop the label; when R6 receives the packet with label 9004, it will pop the label.

3.6.2. PCECC for the End-to-End Protection of P2MP/MP2MP LSPs

In this section, the end-to-end managed path protection service as well as the local protection with the operation management in the PCECC network for the P2MP/MP2MP LSP.

An end-to-end protection principle can be applied for computing backup P2MP or MP2MP LSPs. During computation of the primary multicast trees, PCECC could also take the computation of a secondary tree into consideration. A PCECC could compute the primary and backup P2MP (or MP2MP) LSPs together or sequentially.

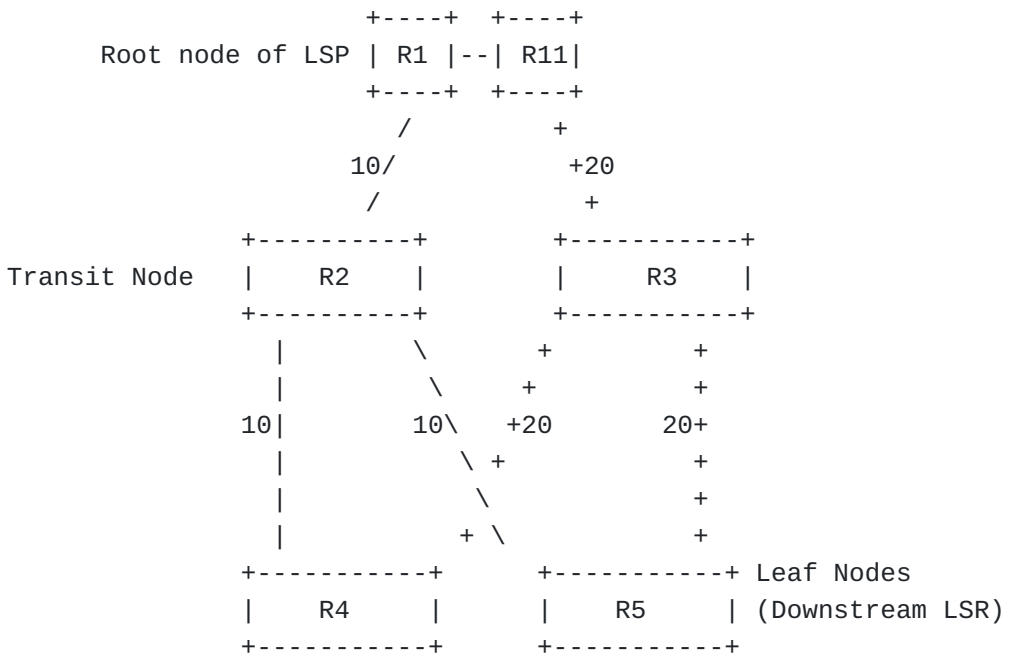

Figure 10: PCECC for the End-to-End Protection of the P2MP/MP2MP LSPs

In the Figure 10, when the PCECC setups the primary multicast tree from the root node R1 to the leaves, which is R1->R2->{R4, R5}, at the same time, it can setup the backup tree, which is R1->R11->R3->{R4, R5}. Both of them (primary forwarding tree and secondary forwarding tree) will be downloaded to each router along the primary path and the secondary path. The traffic will be forwarded through the R1->R2->{R4, R5} path normally, but when a node in the primary tree fails (say R2) the root node R1 will switch the flow to the backup tree, which is R1->R11->R3->{R4, R5}. By using the PCECC a path computation, label downloading and finally forwarding can be done without complex signaling used in the P2MP RSVP-TE or mLDP.

3.6.3. PCECC for the Local Protection of the P2MP/MP2MP LSPs

In this section we describe the local protection service in the PCECC network for the P2MP/MP2MP LSP.

While the PCECC sets up the primary multicast tree, it can also build the backup LSP between Point of Local Repair (PLR), the protected node and Merge Points (MPs) (the downstream nodes of the protected node). In the cases where the amount of downstream nodes

is huge, this mechanism can avoid unnecessary packet duplication on PLR and protect the network from traffic congestion risk.

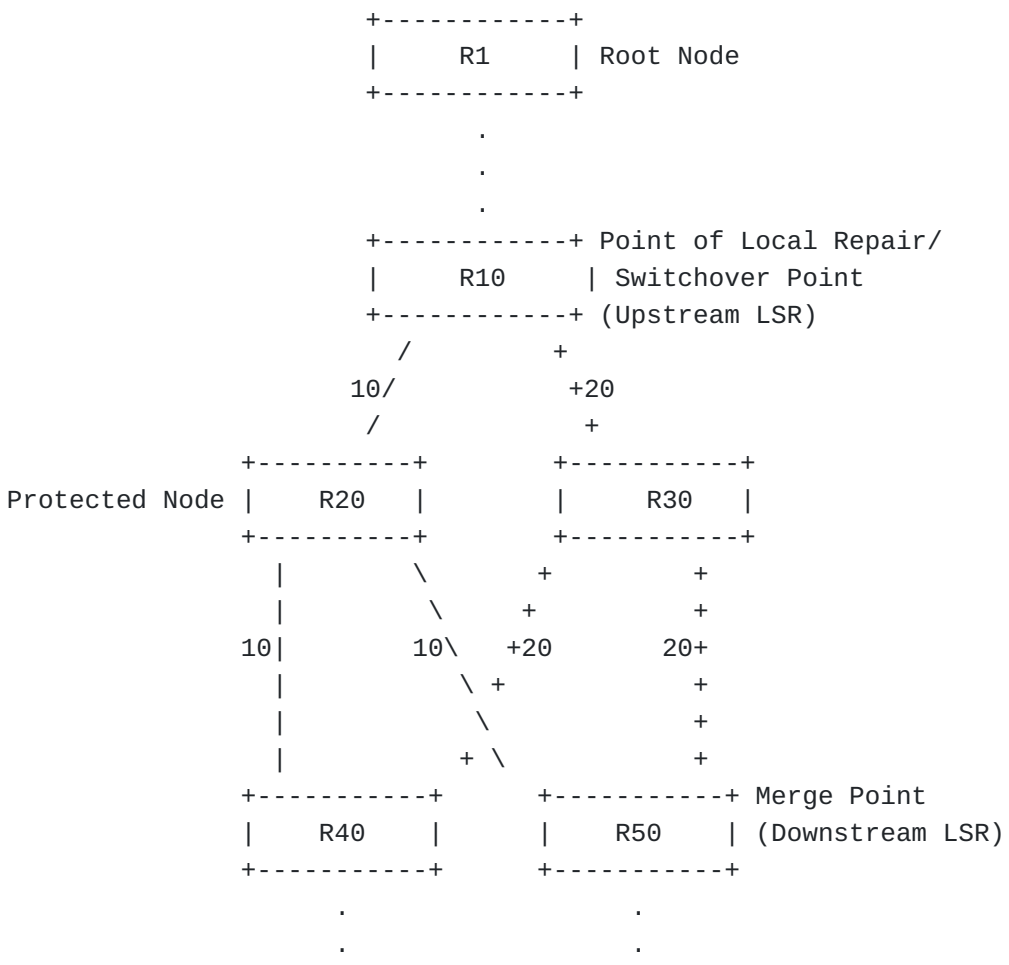

Figure 11: PCECC for the Local Protection of the P2MP/MP2MP LSPs

In Figure 11, when the PCECC setups the primary multicast path around the PLR node R10 to protect node R20, which is R10->R20->{R40, R50}, at the same time, it can set up the backup path R10->R30->{R40, R50}. Both the primary forwarding path and secondary bypass forwarding path will be downloaded to each router along the primary path and the secondary bypass path. The traffic will be forwarded through the R10->R20->{R40, R50} path normally and when there is a node failure for node R20, the PLR node R10 will switch the flow to the backup path, which is R10->R30->{R40, R50}. By using the PCECC, path computation, label downloading and finally forwarding can be done without complex signaling used in the P2MP RSVP-TE or mLDP.

3.7. PCECC for Traffic Classification

As described in [RFC8283], traffic classification is an important part of traffic engineering. It is the process of looking into a packet to determine how it should be treated while it is forwarded through the network. It applies in many scenarios including MPLS traffic engineering (where it determines what traffic is forwarded into which LSPs); segment routing (where it is used to select which set of forwarding instructions (SIDs) to add to a packet); SFC (where it indicates how a packet should be forwarded across which service function path). In conjunction with traffic engineering, traffic classification is an important enabler for load balancing. Traffic classification is closely linked to the computational elements of planning for the network functions because it determines how traffic is balanced and distributed through the network. Therefore, selecting what traffic classification mechanism should be performed by a router is an important part of the work done by a PCECC.

Instructions can be passed from the controller to the routers using PCEP. These instructions tell the routers how to map traffic to paths or connections. Refer [RFC9168].

Along with traffic classification, there are few more questions that needs to be considered after path setup:

*how to use it

*Whether it is a virtual link

*Whether to advertise it in the IGP as a virtual link

*What bits of this information to signal to the tail end

These are out of scope of this document.

3.8. PCECC for SFC

Service Function Chaining (SFC) is described in [RFC7665]. It is the process of directing traffic in a network such that it passes through specific hardware devices or virtual machines (known as service function nodes) that can perform particular desired functions on the traffic. The set of functions to be performed and the order in which they are to be performed is known as a service function chain. The chain is enhanced with the locations at which the service functions are to be performed to derive a Service Function Path (SFP). Each packet is marked as belonging to a specific SFP, and that marking lets each successive service function node know which functions to perform and to which service function node to send the packet next. To operate an SFC network, the service

function nodes must be configured to understand the packet markings, and the edge nodes must be told how to mark packets entering the network. Additionally, it may be necessary to establish tunnels between service function nodes to carry the traffic. Planning an SFC network requires load balancing between service function nodes and traffic engineering across the network that connects them. As per [RFC8283], these are operations that can be performed by a PCE-based controller, and that controller can use PCEP to program the network and install the service function chains and any required tunnels.

A possible mechanism could add support for SFC-based central control instructions. PCECC will be able to instruct the each SFF along the SFP.

*Service Path Identifier (SPI): Uniquely identifies a SFP.

*Service Index (SI): Provides location within the SFP.

*SFC Proxy handling

PCECC can play the role for setting the traffic classification rules at the classifier to impose the Network Service Header (NSH) as well as downloading the forwarding instructions to each SFFs along the way so that they could process the NSH and forward accordingly. Including instructions for the service classifier that handle the context header, meta data etc.

It is also possible to support SFC with SR in conjunction with or without NSH such as [I-D.ietf-spring-nsh-sr] and [I-D.ietf-spring-sr-service-programming]. PCECC technique can also be used for service function related segments and SR service policies.

3.9. PCECC for Native IP

[RFC8735] describes the scenarios and simulation results for the "Centrally Control Dynamic Routing (CCDR)" solution, which integrates the advantage of using distributed protocols (IGP/BGP) and the power of a centralized control technology (PCE/SDN), providing traffic engineering for native IP networks. [RFC8821] defines the framework for CCDR traffic engineering within Native IP network, using multiple BGP sessions and a PCE as the centralized controller. PCEP protocol will be used to transfer the key parameters between PCE and the underlying network devices (PCC) using PCECC technique. The central control instructions from PCECC to PCC will identify which prefix should be advertised on which BGP session. There are PCEP extensions defined in [I-D.ietf-pce-pcep-extension-native-ip] for it.

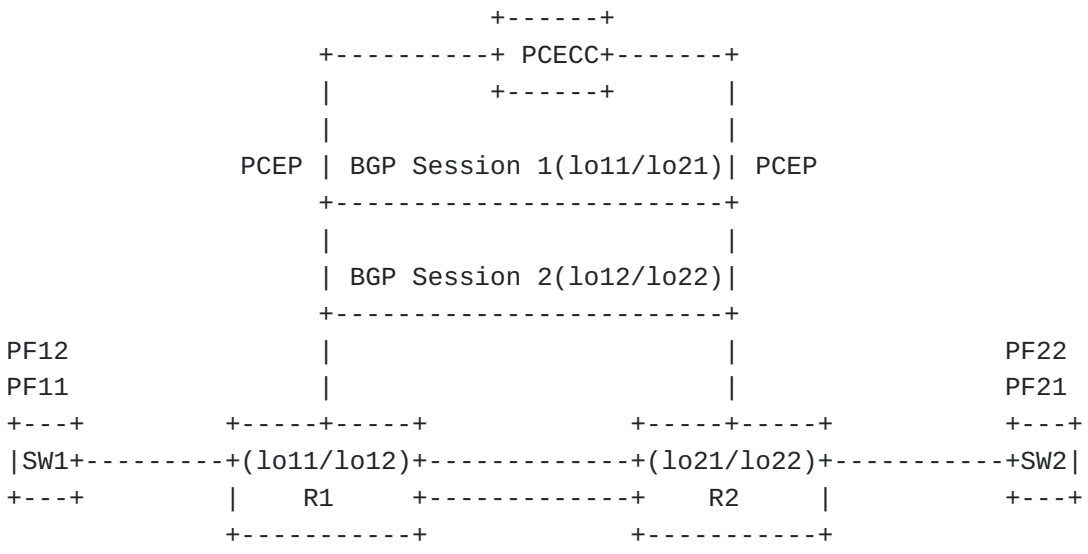

Figure 12: PCECC for Native IP

In the case, as shown in Figure 12, PCECC will instruct both R1 and R2 via PCEP how to form BGP sessions with each other and which IP prefixes need to be advertised via which BGP session.

3.10. PCECC for BIER

Bit Index Explicit Replication (BIER) [RFC8279] defines an architecture where all intended multicast receivers are encoded as a bitmask in the multicast packet header within different encapsulations. A router that receives such packet will forward that packet based on the bit position in the packet header towards the receiver(s) following a precomputed tree for each of the bits in the packet. Each receiver is represented by a unique bit in the bitmask.

BIER-TE [RFC9262] shares architecture and packet formats with BIER. BIER-TE forwards and replicates packets based on a BitString in the packet header, but every BitPosition of the BitString of a BIER-TE packet indicates one or more adjacencies. BIER-TE Path can be derived from a PCE and used at the ingress as described in [I-D.chen-pce-bier].

PCECC mechanism could be used for the allocation of bits for the BIER router for BIER as well as for the adjacencies for BIER-TE. PCECC-based controller can use PCEP to instruct the BIER capable routers the meaning of the bits as well as other fields needed for BIER encapsulation. The PCECC could be used to program the BIER router with various parameters used in the BIER encapsulation such as BIER subdomain-ID, BFR-ID, BIER Encapsulation etc. for both node and adjacency.

Detailed procedures of PCECC usage and extensions are described in [I-D.chen-pce-pcep-extension-pce-controller-bier].

4. IANA Considerations

This document does not require any action from IANA.

5. Security Considerations

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

As per [RFC9050], the use of Transport Layer Security (TLS) in PCEP is recommended, as it provides support for peer authentication, message encryption, and integrity. It further provides mechanisms for associating peer identities with different levels of access and/or authoritativeness via an attribute in X.509 certificates or a local policy with a specific accept-list of X.509 certificates. This can be used to check the authority for the PCECC operations.

It is expected that each new document that is produced for a specific use case will also include considerations of the security impacts of the use of a PCE-based central controller on the network type and services being managed.

6. Acknowledgments

Thanks to Adrian Farrel, Aijun Wang, Robert Tao, Changjiang Yan, Tieying Huang, Sergio Belotti, Dieter Beller, Andrey Elperin and Evgeniy Brodskiy for their useful comments and suggestions.

Thanks to Mach Chen for RTGDIR review.

7. References

7.1. Normative References

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

(PCEP)", RFC 5440, DOI 10.17487/RFC5440, March 2009, <https://www.rfc-editor.org/info/rfc5440>.

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

[RFC8283] Farrel, A., Ed., Zhao, Q., Ed., Li, Z., Zhou, C., and RFC Publisher, "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>.

7.2. Informative References

[I-D.cbrt-pce-stateful-local-protection] Barth, C. and R. Torvi, "PCEP Extensions for RSVP-TE Local-Protection with PCE-Stateful", Work in Progress, Internet-Draft, draft-cbrt-pce-stateful-local-protection-01, 29 June 2018, <https://www.ietf.org/archive/id/draft-cbrt-pce-stateful-local-protection-01.txt>.

[I-D.chen-pce-bier] Chen, R., Zhang, Z., Chen, H., Dhanaraj, S., Qin, F., and A. Wang, "PCEP Extensions for BIER-TE", Work in Progress, Internet-Draft, draft-chen-pce-bier-09, 12 July 2021, <https://www.ietf.org/archive/id/draft-chen-pce-bier-09.txt>.

[I-D.chen-pce-pcep-extension-pce-controller-bier] Chen, R., Xu, B., Chen, H., and A. Wang, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of BIER", Work in Progress, Internet-Draft, draft-chen-pce-pcep-extension-pce-controller-bier-03, 7 March 2022, <https://www.ietf.org/archive/id/draft-chen-pce-pcep-extension-pce-controller-bier-03.txt>.

[I-D.dhody-pce-pcep-extension-pce-controller-srv6] Li, Z., Peng, S., Geng, X., and M. S. Negi, "Path Computation Element Communication Protocol (PCEP) Extensions for Using the PCE as a Central Controller (PCECC) for Segment Routing over IPv6 (SRv6) Segment Identifier (SID) Allocation and Distribution.", Work in Progress, Internet-Draft, draft-dhody-pce-pcep-extension-pce-controller-srv6-09, 10 July

2022, <https://www.ietf.org/archive/id/draft-dhody-pce-pcep-extension-pce-controller-srv6-09.txt>.

[I-D.ietf-mpls-seamless-mpls]
Leymann, N., Decraene, B., Filsfils, C., Konstantynowicz, M., and D. Steinberg, "Seamless MPLS Architecture", Work in Progress, Internet-Draft, draft-ietf-mpls-seamless-mpls-07, 28 June 2014, <https://www.ietf.org/archive/id/draft-ietf-mpls-seamless-mpls-07.txt>.

[I-D.ietf-pce-binding-label-sid]
Sivabalan, S., Filsfils, C., Tantsura, J., Previdi, S., and C. Li, "Carrying Binding Label/Segment Identifier (SID) in PCE-based Networks.", Work in Progress, Internet-Draft, draft-ietf-pce-binding-label-sid-15, 20 March 2022, <https://www.ietf.org/archive/id/draft-ietf-pce-binding-label-sid-15.txt>.

[I-D.ietf-pce-pcep-extension-native-ip] Wang, A., Khasanov, B., Fang, S., Tan, R., and C. Zhu, "PCEP Extension for Native IP Network", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-extension-native-ip-19, 21 September 2022, <https://www.ietf.org/archive/id/draft-ietf-pce-pcep-extension-native-ip-19.txt>.

[I-D.ietf-pce-pcep-extension-pce-controller-sr]
Li, Z., Peng, S., Negi, M. S., Zhao, Q., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) Extensions for Using PCE as a Central Controller (PCECC) for Segment Routing (SR) MPLS Segment Identifier (SID) Allocation and Distribution.", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-extension-pce-controller-sr-05, 10 July 2022, <https://www.ietf.org/archive/id/draft-ietf-pce-pcep-extension-pce-controller-sr-05.txt>.

[I-D.ietf-pce-segment-routing-ipv6]
Li, C., Negi, M. S., Sivabalan, S., Koldychev, M., Kaladharan, P., and Y. Zhu, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing leveraging the IPv6 dataplane", Work in Progress, Internet-Draft, draft-ietf-pce-segment-routing-ipv6-15, 23 October 2022, <https://www.ietf.org/archive/id/draft-ietf-pce-segment-routing-ipv6-15.txt>.

[I-D.ietf-pce-segment-routing-policy-cp]
Koldychev, M., Sivabalan, S., Barth, C., Peng, S., and H. Bidgoli, "PCEP extension to support Segment Routing Policy Candidate Paths", Work in Progress, Internet-

Draft, draft-ietf-pce-segment-routing-policy-cp-08, 24 October 2022, <https://www.ietf.org/archive/id/draft-ietf-pce-segment-routing-policy-cp-08.txt>.

[I-D.ietf-pce-stateful-interdomain] Dugeon, O., Meuric, J., Lee, Y., and D. Ceccarelli, "PCEP Extension for Stateful Inter-Domain Tunnels", Work in Progress, Internet-Draft, draft-ietf-pce-stateful-interdomain-03, 4 March 2022, <https://www.ietf.org/archive/id/draft-ietf-pce-stateful-interdomain-03.txt>.

[I-D.ietf-spring-nsh-sr] Guichard, J. and J. Tantsura, "Integration of Network Service Header (NSH) and Segment Routing for Service Function Chaining (SFC)", Work in Progress, Internet-Draft, draft-ietf-spring-nsh-sr-11, 20 April 2022, <https://www.ietf.org/archive/id/draft-ietf-spring-nsh-sr-11.txt>.

[I-D.ietf-spring-sr-service-programming]
Clad, F., Xu, X., Filsfils, C., Bernier, D., Li, C., Decraene, B., Ma, S., Yadlapalli, C., Henderickx, W., and S. Salsano, "Service Programming with Segment Routing", Work in Progress, Internet-Draft, draft-ietf-spring-sr-service-programming-06, 9 June 2022, <https://www.ietf.org/archive/id/draft-ietf-spring-sr-service-programming-06.txt>.

[I-D.ietf-teas-rfc3272bis]
Farrel, A., "Overview and Principles of Internet Traffic Engineering", Work in Progress, Internet-Draft, draft-ietf-teas-rfc3272bis-22, 27 October 2022, <https://www.ietf.org/archive/id/draft-ietf-teas-rfc3272bis-22.txt>.

[I-D.li-pce-controlled-id-space] Li, C., Shi, H., Wang, A., Cheng, W., and C. Zhou, "Path Computation Element Communication Protocol (PCEP) extension to advertise the PCE Controlled Identifier Space", Work in Progress, Internet-Draft, draft-li-pce-controlled-id-space-14, 10 November 2022, <https://www.ietf.org/archive/id/draft-li-pce-controlled-id-space-14.txt>.

[MAP-REDUCE] Lee, K., Choi, T., Ganguly, A., Wolinsky, D., Boykin, P., and R. Figueiredo, "Parallel Processing Framework on a P2P System Using Map and Reduce Primitives", , May 2011, <http://leeky.me/publications/mapreduce_p2p.pdf>.

[MPLS-DC] Afanasiev, D. and D. Ginsburg, "MPLS in DC and inter-DC networks: the unified forwarding mechanism for network

programmability at scale", , March 2014, <https://www.slideshare.net/DmitryAfanasiev1/yandex-nag201320131031>.

[RFC1195] Callon, R. and RFC Publisher, "Use of OSI IS-IS for routing in TCP/IP and dual environments", RFC 1195, DOI 10.17487/RFC1195, December 1990, <https://www.rfc-editor.org/info/rfc1195>.

[RFC2328] Moy, J. and RFC Publisher, "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., Swallow, G., and RFC Publisher, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[RFC3985] Bryant, S., Ed., Pate, P., Ed., and RFC Publisher, "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", RFC 3985, DOI 10.17487/RFC3985, March 2005, <https://www.rfc-editor.org/info/rfc3985>.

[RFC4206] Kompella, K., Rekhter, Y., and RFC Publisher, "Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE)", RFC 4206, DOI 10.17487/RFC4206, October 2005, <https://www.rfc-editor.org/info/rfc4206>.

[RFC4364] Rosen, E., Rekhter, Y., and RFC Publisher, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

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

[RFC5036] Andersson, L., Ed., Minei, I., Ed., Thomas, B., Ed., and RFC Publisher, "LDP Specification", RFC 5036, DOI 10.17487/RFC5036, October 2007, <https://www.rfc-editor.org/info/rfc5036>.

[RFC5150] Ayyangar, A., Kompella, K., Vasseur, JP., Farrel, A., and RFC Publisher, "Label Switched Path Stitching with Generalized Multiprotocol Label Switching Traffic Engineering (GMPLS TE)", RFC 5150, DOI 10.17487/RFC5150, February 2008, <https://www.rfc-editor.org/info/rfc5150>.

[RFC5151] Farrel, A., Ed., Ayyangar, A., Vasseur, JP., and RFC Publisher, "Inter-Domain MPLS and GMPLS Traffic Engineering -- Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 5151, DOI 10.17487/RFC5151, February 2008, <https://www.rfc-editor.org/info/rfc5151>.

[RFC5340] Coltun, R., Ferguson, D., Moy, J., Lindem, A., and RFC Publisher, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

[RFC5376] Bitar, N., Zhang, R., Kumaki, K., and RFC Publisher, "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>.

[RFC5541] Le Roux, JL., Vasseur, JP., Lee, Y., and RFC Publisher, "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>.

[RFC7025] Otani, T., Ogaki, K., Caviglia, D., Zhang, F., Margaria, C., and RFC Publisher, "Requirements for GMPLS Applications of PCE", RFC 7025, DOI 10.17487/RFC7025, September 2013, <https://www.rfc-editor.org/info/rfc7025>.

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

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., Henderickx, W., and RFC Publisher, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

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

[RFC7665] Halpern, J., Ed., Pignataro, C., Ed., and RFC Publisher, "Service Function Chaining (SFC) Architecture", RFC 7665,

DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[RFC8231] Crabbe, E., Minei, I., Medved, J., Varga, R., and RFC Publisher, "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>.

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

[RFC8281] Crabbe, E., Minei, I., Sivabalan, S., Varga, R., and RFC Publisher, "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>.

[RFC8355] Filsfils, C., Ed., Previdi, S., Ed., Decraene, B., Shakir, R., and RFC Publisher, "Resiliency Use Cases in Source Packet Routing in Networking (SPRING) Networks", RFC 8355, DOI 10.17487/RFC8355, March 2018, <https://www.rfc-editor.org/info/rfc8355>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., Shakir, R., and RFC Publisher, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., Hardwick, J., and RFC Publisher, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing", RFC 8664, DOI 10.17487/RFC8664, December 2019, <https://www.rfc-editor.org/info/rfc8664>.

[RFC8735] Wang, A., Huang, X., Kou, C., Li, Z., Mi, P., and RFC Publisher, "Scenarios and Simulation Results of PCE in a Native IP Network", RFC 8735, DOI 10.17487/RFC8735, February 2020, <https://www.rfc-editor.org/info/rfc8735>.

[RFC8751] Dhody, D., Lee, Y., Ceccarelli, D., Shin, J., King, D., and RFC Publisher, "Hierarchical Stateful Path Computation Element (PCE)", RFC 8751, DOI 10.17487/RFC8751, March 2020, <https://www.rfc-editor.org/info/rfc8751>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., Voyer, D., and RFC Publisher, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

[RFC8821] Wang, A., Khasanov, B., Zhao, Q., Chen, H., and RFC Publisher, "PCE-Based Traffic Engineering (TE) in Native IP Networks", RFC 8821, DOI 10.17487/RFC8821, April 2021, <https://www.rfc-editor.org/info/rfc8821>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., Li, Z., and RFC Publisher, "Segment Routing over IPv6 (SRv6) Network Programming", RFC 8986, DOI 10.17487/RFC8986, February 2021, <https://www.rfc-editor.org/info/rfc8986>.

[RFC9012] Patel, K., Van de Velde, G., Sangli, S., Scudder, J., and RFC Publisher, "The BGP Tunnel Encapsulation Attribute", RFC 9012, DOI 10.17487/RFC9012, April 2021, <https://www.rfc-editor.org/info/rfc9012>.

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

[RFC9087] Filsfils, C., Ed., Previdi, S., Dawra, G., Ed., Aries, E., Afanasiev, D., and RFC Publisher, "Segment Routing Centralized BGP Egress Peer Engineering", RFC 9087, DOI 10.17487/RFC9087, August 2021, <https://www.rfc-editor.org/info/rfc9087>.

[RFC9168] Dhody, D., Farrel, A., Li, Z., and RFC Publisher, "Path Computation Element Communication Protocol (PCEP) Extension for Flow Specification", RFC 9168, DOI 10.17487/RFC9168, January 2022, <https://www.rfc-editor.org/info/rfc9168>.

[RFC9256] Filsfils, C., Talaulikar, K., Ed., Voyer, D., Bogdanov, A., Mattes, P., and RFC Publisher, "Segment Routing Policy Architecture", RFC 9256, DOI 10.17487/RFC9256, July 2022, <https://www.rfc-editor.org/info/rfc9256>.

[RFC9262] Eckert, T., Ed., Menth, M., Cauchie, G., and RFC Publisher, "Tree Engineering for Bit Index Explicit

Replication (BIER-TE)", RFC 9262, DOI 10.17487/RFC9262,
October 2022, <https://www.rfc-editor.org/info/rfc9262>.

Appendix A. Other Use Cases of PCECC

This section list some more advanced use cases of PCECC that were discussed and could be worked on in future.

A.1. PCECC for Network Migration

One of the main advantages of PCECC solution is that it has backward compatibility since the PCE server itself can function as a proxy node of the MPLS network for all the new nodes which may no longer support the signaling protocols.

As is illustrated in the following example, the current network could migrate to a total PCECC-controlled network gradually by replacing the legacy nodes. During the migration, the legacy nodes still need to use the existing MPLS protocols signaling such as LDP and RSVP-TE, and the new nodes will set up their portion of the forwarding path through PCECC directly. With the PCECC function as the proxy of these new nodes, MPLS signaling can populate through the network for both: old and new nodes.

The example described in this section is based on network configurations illustrated using the Figure 13:

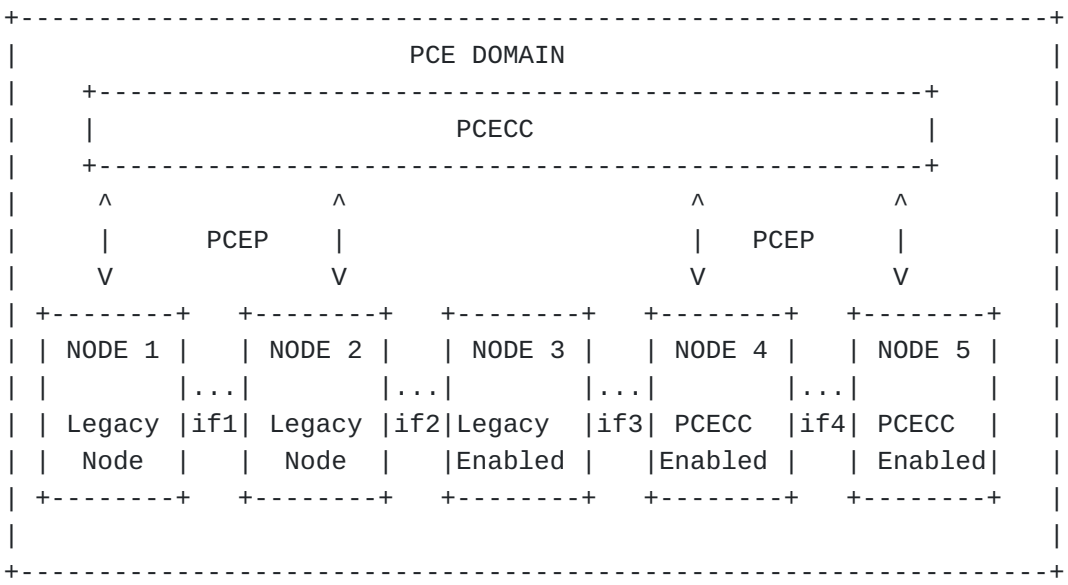

Figure 13: PCECC Initiated LSP Setup In the Network Migration

In this example, there are five nodes for the TE LSP from the head end (Node1) to the tail end (Node5). Where Node4 and Node5 are centrally controlled and other nodes are legacy nodes.

*Node1 sends a path request message for the setup of LSP with the destination as Node5.

*PCECC sends to Node1 a reply message for LSP setup with the path: (Node1, if1),(Node2, if2), (Node3, if3), (Node4, if4), Node5.

*Node1, Node2, and Node3 will set up the LSP to Node5 using the local labels as usual. Node 3 with help of PCECC could proxy the signaling.

*Then the PCECC will program the out-segment of Node3, the in-segment/ out-segment of Node4, and the in-segment for Node5.

A.2. PCECC for L3VPN and PWE3

As described in [RFC8283], various network services may be offered over a network. These include protection services (including Virtual Private Network (VPN) services (such as Layer 3 VPNs [RFC4364] or Ethernet VPNs [RFC7432]); or Pseudowires [RFC3985]. Delivering services over a network in an optimal way requires coordination in the way where network resources are allocated to support the services. A PCE-based central controller can consider the whole network and all components of a service at once when planning how to deliver the service. It can then use PCEP to manage the network resources and to install the necessary associations between those resources.

In the case of L3VPN, VPN labels could also be assigned and distributed through PCEP among the PE router instead of using the BGP protocols.

Example described in this section is based on network configurations illustrated using the Figure 14:

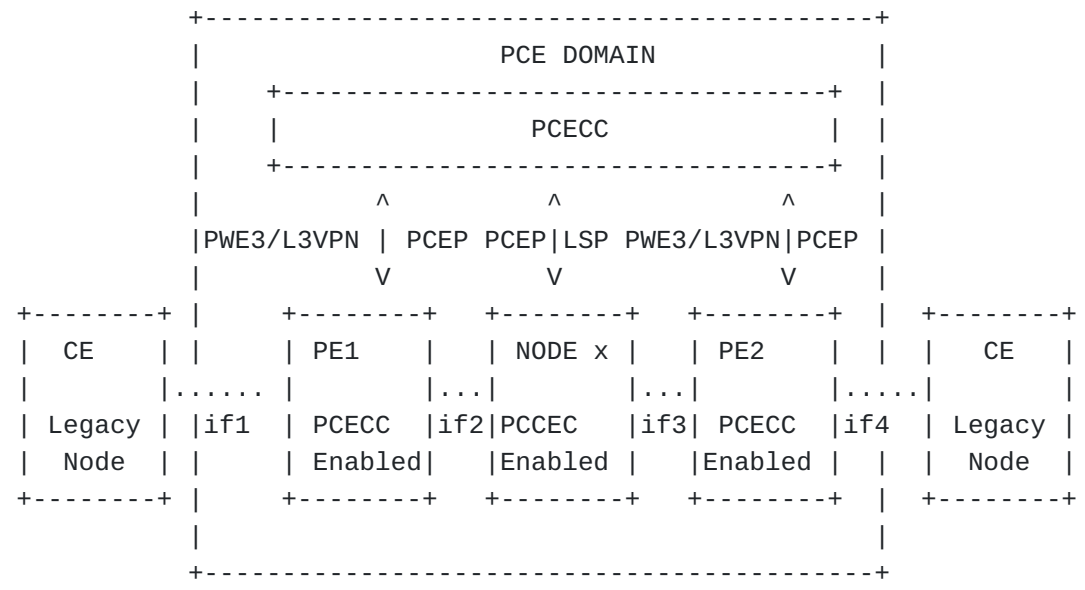

Figure 14: PCECC for L3VPN and PWE3

In the case PWE3, instead of using the LDP signaling protocols, the label and port pairs assigned to each pseudowire can be assigned through PCECC among the PE routers and the corresponding forwarding entries will be distributed into each PE routers through the extended PCEP protocols and PCECC mechanism.

A.3. PCECC for Local Protection (RSVP-TE)

[I-D.cbrt-pce-stateful-local-protection] describes the need for the PCE to maintain and associate the local protection paths for the RSVP-TE LSP. Local protection requires the setup of a bypass at the PLR. This bypass can be PCC-initiated and delegated, or PCE-initiated. In either case, the PLR needs to maintain a PCEP session with the PCE. The Bypass LSPs need to be mapped to the primary LSP. This could be done locally at the PLR based on a local policy but there is a need for a PCE to do the mapping as well to exert greater control.

This mapping can be done via PCECC procedures where the PCE could instruct the PLR to the mapping and identify the primary LSP for which bypass should be used.

A.4. Using reliable P2MP TE based multicast delivery for distributed computations (MapReduce-Hadoop)

MapReduce model of distributed computations in computing clusters is widely deployed. In Hadoop 1.0 architecture MapReduce operations on big data in the Hadoop Distributed File System (HDFS), where NameNode has the knowledge about resources of the cluster and where actual data (chunks) for particular task are located (which

DataNode). Each chunk of data (64MB or more) should have 3 saved copies in different DataNodes based on their proximity.

Proximity level currently has semi-manual allocation and based on Rack IDs (Assumption is that closer data are better because of access speed/smaller latency).

JobTracker node is responsible for computation tasks, scheduling across DataNodes and also have Rack-awareness. Currently transport protocols between NameNode/JobTracker and DataNodes are based on IP unicast. It has simplicity as pros but has numerous drawbacks related with its flat approach.

It is clear that we should go beyond of one DC for Hadoop cluster creation and move towards distributed clusters. In that case we need to handle performance and latency issues. Latency depends on speed of light in fiber links and also latency introduced by intermediate devices in between. The last one is closely correlated with network device architecture and performance. Current performance of NPU based routers should be enough for creating distribute Hadoop clusters with predicted latency. Performance of SW based routers (mainly as VNF) together with additional HW features such as DPDK are promising but require additional research and testing.

Main question is how can we create simple but effective architecture for distributed Hadoop cluster?

There is research [MAP-REDUCE] which show how usage of multicast tree could improve speed of resource or cluster members discovery inside the cluster as well as increase redundancy in communications between cluster nodes.

Is traditional IP based multicast enough for that? We doubt it because it requires additional control plane (IGMP, PIM) and a lot of signaling, that is not suitable for high performance computations, that are very sensitive to latency.

P2MP TE tunnels looks much more suitable as potential solution for creation of multicast based communications between NameNode as root and DataNodes as leaves inside the cluster. Obviously these P2MP tunnels should be dynamically created and turned down (no manual intervention). Here, the PCECC comes to play with main objective to create optimal topology of each particular request for MapReduce computation and also create P2MP tunnels with needed parameters such as bandwidth and delay.

This solution will require to use MPLS label based forwarding inside the cluster. Usage of label based forwarding inside DC was proposed by Yandex [MPLS-DC]. Technically it is already possible because MPLS

on switches is already supported by some vendors, MPLS also exists on Linux and OVS.

A possible framework for this task is shown in Figure 15:

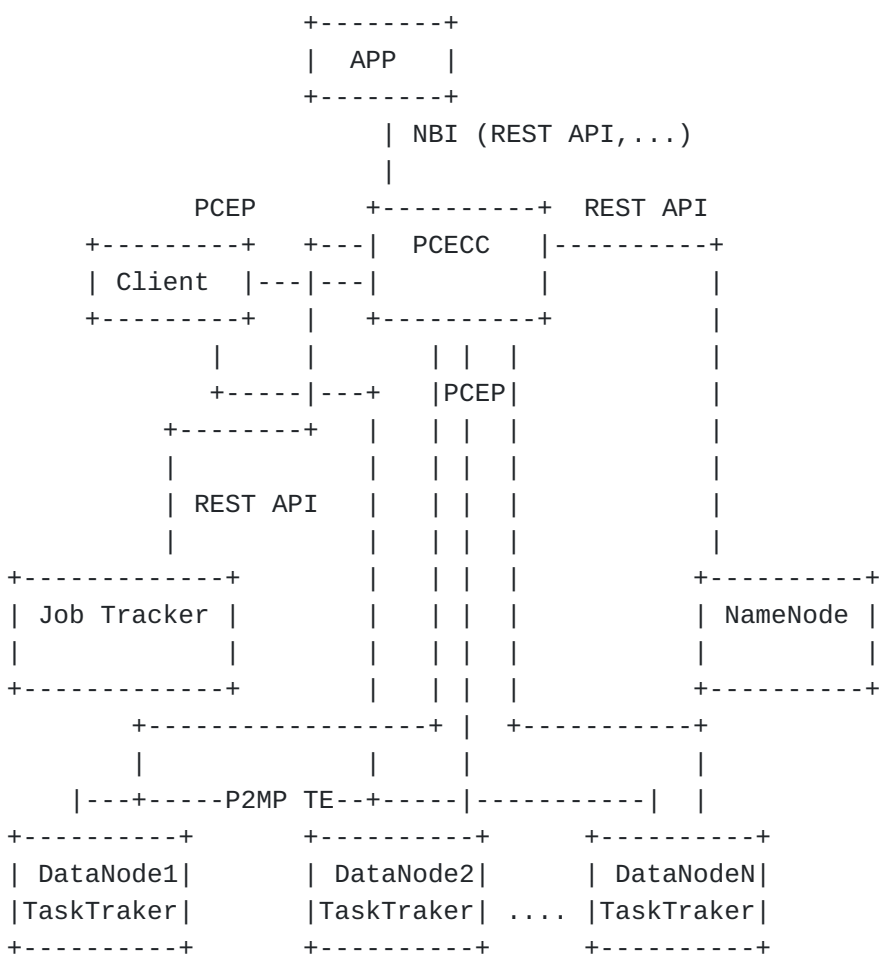

Figure 15: Using reliable P2MP TE based multicast delivery for distributed computations (MapReduce-Hadoop)

Communication between JobTracker, NameNode and PCECC can be done via REST API directly or via cluster manager such as Mesos.

Phase 1: Distributed cluster resources discovery During this phase JobTracker and NameNode should identify and find available DataNodes according to computing request from application (APP). NameNode should query PCECC about available DataNodes, NameNode may provide additional constrains to PCECC such as topological proximity, redundancy level.

PCECC should analyze the topology of distributed cluster and perform constrain based path calculation from client towards most suitable NameNodes. PCECC should reply to NameNode the list of most suitable

DataNodes and their resource capabilities. Topology discovery mechanism for PCECC will be added later to that framework.

Phase 2: PCECC should create P2MP LSP from client towards those DataNodes by means of PCEP messages following previously calculated path.

Phase 3. NameNode should send this information to client, PCECC informs client about optimal P2MP path towards DataNodes via PCEP message.

Phase 4. Client sends data blocks to those DataNodes for writing via created P2MP tunnel.

When this task will be finished, P2MP tunnel could be turned down.

Appendix B. Contributor Addresses

Following authors contributed text for this document and should be considered as co-authors:

Luyuan Fang
United States of America

Email: luyuanf@gmail.com

Chao Zhou
HPE

Email: chaozhou_us@yahoo.com

Boris Zhang
Amazon

Email: zhangyud@amazon.com

Artsiom Rachytski
Belarus

Email: arachyts@gmail.com

Anton Gulida
EPAM Systems, Inc.
Belarus

Email: Anton_Hulida@epam.com

Authors' Addresses

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

Email: lizhenbin@huawei.com

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

Email: dhruv.ietf@gmail.com

Quintin Zhao
Etheric Networks
1009 S CLAREMONT ST
SAN MATEO, CA 94402
United States of America

Email: qzhao@ethericnetworks.com

King He
Tencent Holdings Ltd.
Shenzhen
China

Email: kinghe@tencent.com

Boris Khasanov
Yandex LLC
Ulitsa Lva Tolstogo 16
Moscow

Email: bhassanov@yahoo.com