Internet Engineering Task Force (IETF)
Request for Comments: 9689
Category: Informational
ISSN: 2070-1721

Z. Li
D. Dhody
Huawei Technologies
Q. Zhao
Etheric Networks
K. He
Tencent Holdings Ltd.
B. Khasanov
MTS Web Services (MWS)

December 2024

Use Cases for a PCE as a Central Controller (PCECC)

Abstract

The PCE is a core component of a Software-Defined Networking (SDN) system. It can be used to compute optimal paths for network traffic and update existing paths to reflect changes in the network or traffic demands. The PCE was developed to derive Traffic Engineering (TE) paths in MPLS networks, which are supplied to the headend of the paths using the Path Computation Element Communication Protocol (PCEP).

SDN has much broader applicability than signalled MPLS TE networks, and the PCE may be used to determine paths in a range of use cases including static Label-Switched Paths (LSPs), Segment Routing (SR), Service Function Chaining (SFC), and most forms of a routed or switched network. Therefore, it is 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 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 are required for the PCECC use cases, are covered in separate documents.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are candidates for any level of Internet Standard; see Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9689.

Table of Contents

1. Introduction

The PCE [RFC4655] was developed to offload the path computation function from routers in an MPLS Traffic Engineering (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 the PCE have grown to cover several other uses (such as GMPLS [RFC7025] or 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 the 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] outlines the architecture for the PCE as a central controller, extending the framework described in [RFC4655], and demonstrates how PCEP can serve as a general southbound control protocol between the PCE and Path Computation Client (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.

AS: Autonomous System

ASBR: Autonomous System Border Router

BGP-LS: Border Gateway Protocol - Link State [RFC9552]

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

LSP: Label-Switched Path

PCC: Path Computation Client (as per [RFC4655], any client application requesting a path computation to be performed by a PCE)

PCE: Path Computation Element (as per [RFC4655], an entity such as a 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 (an extension of a PCE to support SDN functions as per [RFC8283])

PST: Path Setup Type [RFC8408]

RR: Route Reflector [RFC4456]

SID: Segment Identifier [RFC8402]

SR: Segment Routing [RFC8402]

SRGB: Segment Routing Global Block [RFC8402]

SRLB: Segment Routing Local Block [RFC8402]

TE: Traffic Engineering [RFC9522]

3. Use Cases

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

* use of a PCECC to set up static TE LSPs (the PCEP extension for

this use case is in [RFC9050])

* use of a PCECC in SR [RFC8402]

* use of a PCECC to set up Multicast Point-to-Multipoint (P2MP) LSPs

* use of a PCECC to set up Service Function Chaining (SFC) [RFC7665]

* use of a PCECC in optical networks

Section 3.1 describes the general case of a 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 a PCECC.

It is interesting to note that some of the use cases listed can also be supported via BGP instead of PCEP. However, within the scope of this document, the focus is on the use of PCEP.

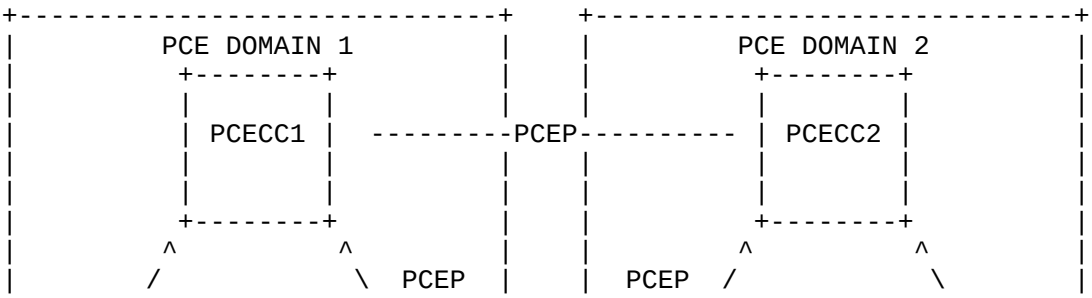
3.1. PCECC for Label Management

As per [RFC8283], in some cases, the PCECC 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 PCECC 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 [PCE-ID] for a possible PCEP extension to support the advertisement of the MPLS label space for the PCE to control).

[RFC8664] specifies extensions to PCEP that allow a stateful PCE to compute, update, or initiate SR-TE paths. [PCECC-SR] describes the mechanism for a PCECC to allocate and provision the node/prefix/adjacency label (Segment Routing Identifier (SID)) via PCEP. To make such an allocation, the PCE needs to be aware of the label space from the 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 the PCEP. The full SRGB/SRLB of a node could be learned via existing IGP or BGP-LS [RFC9552] mechanisms.

Further, there have been proposals for a global label range in MPLS as well as the use of PCECC architecture to learn the label space of each node to determine and provision the global label range.



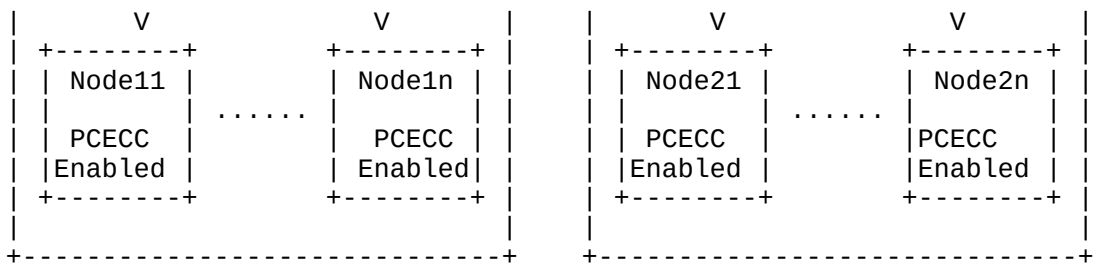


Figure 1: PCECC for MPLS Label Management

As shown in Figure 1:

* The PCC will advertise the PCECC capability to the PCECC [RFC9050].

* The PCECC could also learn the label range set aside by the PCC (via [PCE-ID]).

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

  - In the case that the shared global label range needs 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], the 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 SR

SR [RFC8402] leverages the source routing paradigm. Using SR, a source node steers a packet through a path without relying on hop-by-hop signalling 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 an intra-domain routing protocol (such as IS-IS or OSPF), an inter-domain protocol (such as BGP), or by any other means. PCEP could also be one of other protocols.

[RFC8664] specifies the PCEP extension specific to SR for SR over MPLS (SR-MPLS). The PCECC may further use the PCEP for distributing SR Segment Identifiers (SIDs) 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 an SDN environment.

When the PCECC is used for the distribution of the Node-SID and Adj-SID for SR-MPLS, the Node-SID is allocated from the SRGB of the node and the Adj-SID is allocated from the SRLB of the node as described in [PCECC-SR].

[RFC8355] identifies various protection and resiliency use cases for SR. Path protection lets the ingress node be in charge of the failure recovery (used for SR-TE [RFC8664]). 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 the case of the PCECC, the protection paths can be precomputed and set up by the PCE.

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

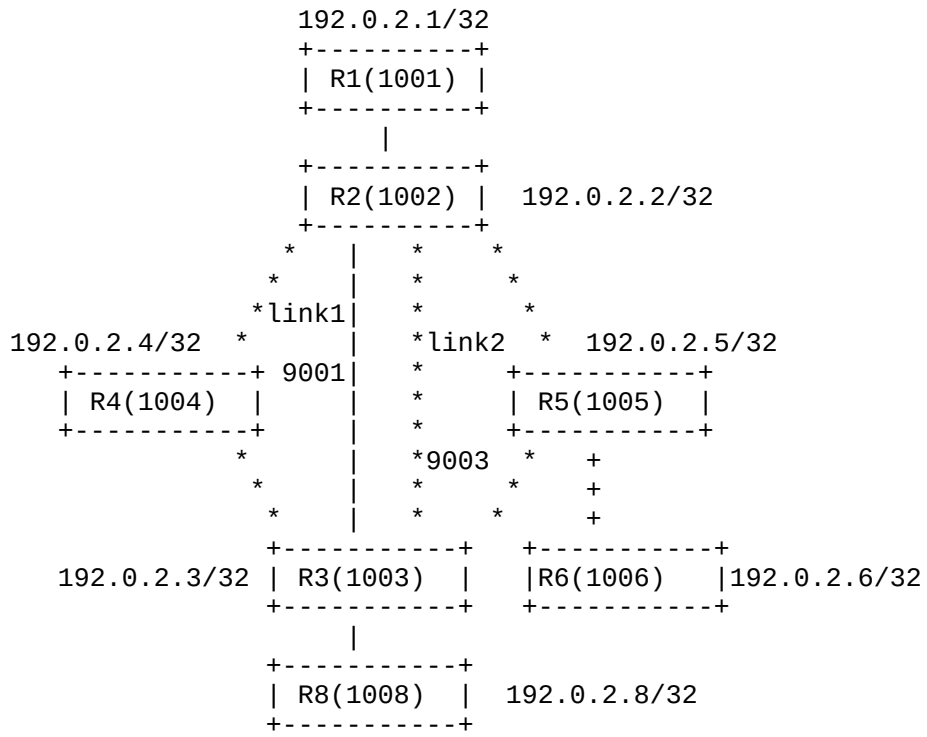


Figure 2: SR Topology

3.2.1. PCECC SID Allocation for SR-MPLS

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 next hop and download the label forwarding instructions accordingly. The forwarding behavior and the end result are the same as IGP shortest-path SR forwarding based on Node-SIDs. Thus, from anywhere in the domain, it enforces the ECMP-aware shortest-path forwarding of the packet towards the related node.

The PCECC can allocate an Adj-SID for each adjacency in the network. 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 [PCECC-SR].

3.2.2. PCECC for SR-MPLS Best Effort (BE) Paths

When using PCECC for SR-MPLS Best Effort (BE) Paths, the PCECC needs to allocate the Node-SID (without calculating the explicit path for the SR path). The ingress router of the forwarding path 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 / next hop calculation on the routers.

3.2.3. PCECC for SR-MPLS TE Paths

SR-TE paths may not follow an IGP shortest path tree (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 a strict traffic engineering path is needed, all the Adj-SIDs are stacked; otherwise, a combination of Node-SIDs or Adj-SIDs can be used for the SR-TE paths.

As shown in Figure 3, 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-SIDs of R2 and R8, respectively. 9001 is the Adj-SID for link1. The path should be: "R1-R2-link1-R3-R8".

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

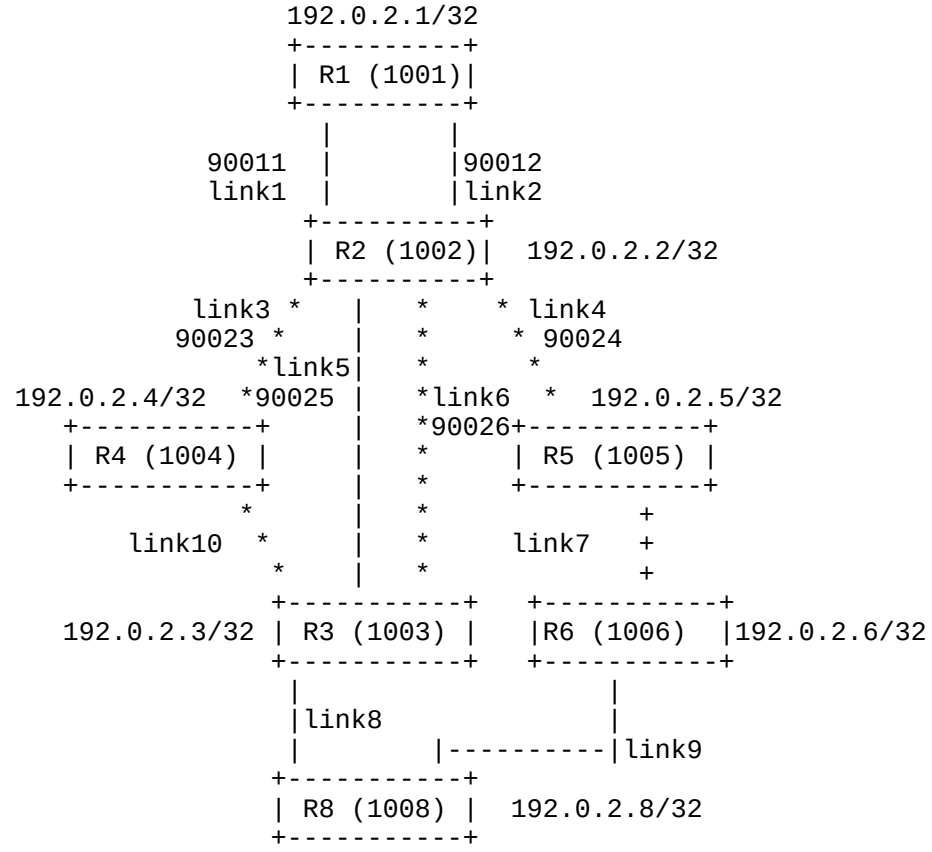


Figure 3: PCECC TE LSP Setup Example

Refer to 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.

* The PCECC will calculate the optimal path according to the given constraints (e.g., bandwidth (BW)).

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

* For the end-to-end protection, the 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 SR 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 headend 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 the 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 Service Level Agreement (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 for SR-MPLS, where the Node-SIDs are 100x and the Adj-SIDs are 900xx.

Let’s assume that R1 needs to have two disjoint SR Policies towards R8 based on different BWs. This means 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 the candidate path and segment list) will be signalled to a headend (R1) via PCEP [PCEP-POLICY] with the addition of an ASSOCIATION object. A Binding SID (BSID) [RFC8402] can be used for traffic steering of labelled traffic into an SR Policy; a BSID can be provisioned from the PCECC also via PCEP [RFC9604]. For non-labelled traffic steering into the SR Policy POL1 or POL2, a per-destination traffic steering will be used by means of the BGP Color Extended Community [RFC9012].

The procedure is as follows:

* The PCECC allocates Node-SIDs and Adj-SIDs using the mechanism described in Section 3.2.1 for all nodes and links.

* The PCECC calculates disjoint paths for POL1 and POL2 and create segment lists for them: {90011, 1002, 90023, 1004, 1003, 1008};{90012, 1002, 90024, 1005, 1006, 1008}.

* The PCECC forms both SR Policies POL1 and POL2.

* The PCECC sends both POL1 and POL2 to R1 via PCEP.

* The PCECC optionally allocates 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 color 200, will be steered to POL2 and follows the path: "R1-link2-R2-link4-R5-R6-R8". Due to the possibility of having many segment lists in the same candidate path of each POL1/POL2, the PCECC could provision more paths towards R8 and traffic will be balanced either as ECMP or as weighted-ECMP (W-ECMP). This is the advantage of SR Policy architecture.

Note that an SR Policy can be associated with multiple candidate paths. A candidate path is selected when it is valid and it is determined to be the best path of the SR Policy as described in [RFC9256].

3.2.4. PCECC for SRv6

As per [RFC8402], with SR, a node steers a packet through an ordered list of instructions, called segments. SR 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 SR over IPv6 (SRv6) Segment is a 128-bit value. "SRv6 SID" or simply "SID" are often used as a shorter reference for "SRv6 Segment". [RFC8986] defines the SRv6 SID as consisting of LOC:FUNCT:ARG.

[RFC9603] extends [RFC8664] to support SR for the IPv6 data plane. Further, a PCECC could be extended to support SRv6 SID allocation and distribution. An example of how PCEP extensions could be extended for SRv6 for a PCECC is described in [PCECC-SRv6].

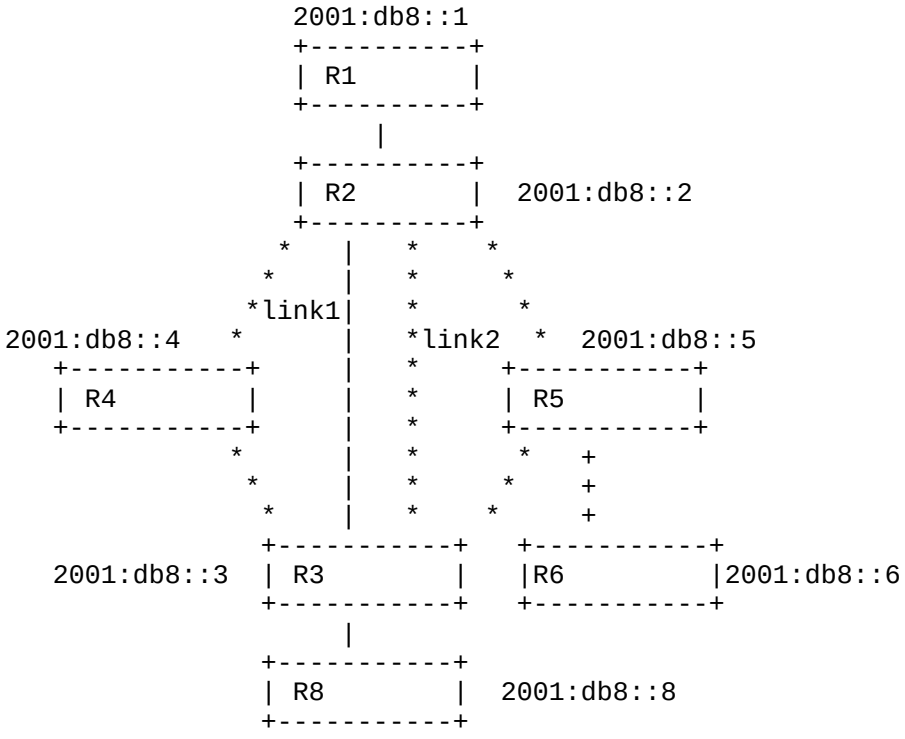


Figure 4: PCECC for SRv6

In this case, as shown in Figure 4, the PCECC could assign the SRv6 SID (in the form of an IPv6 address) to be used for node and

adjacency. Later, the SRv6 path in the form of a list of SRv6 SIDs could be used at the ingress. Some examples:

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

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

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

3.3. PCECC for Static TE LSPs

As described in Section 3.1.2 of [RFC8283], the PCECC architecture supports the provisioning of static TE LSPs. 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 PCECC 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.

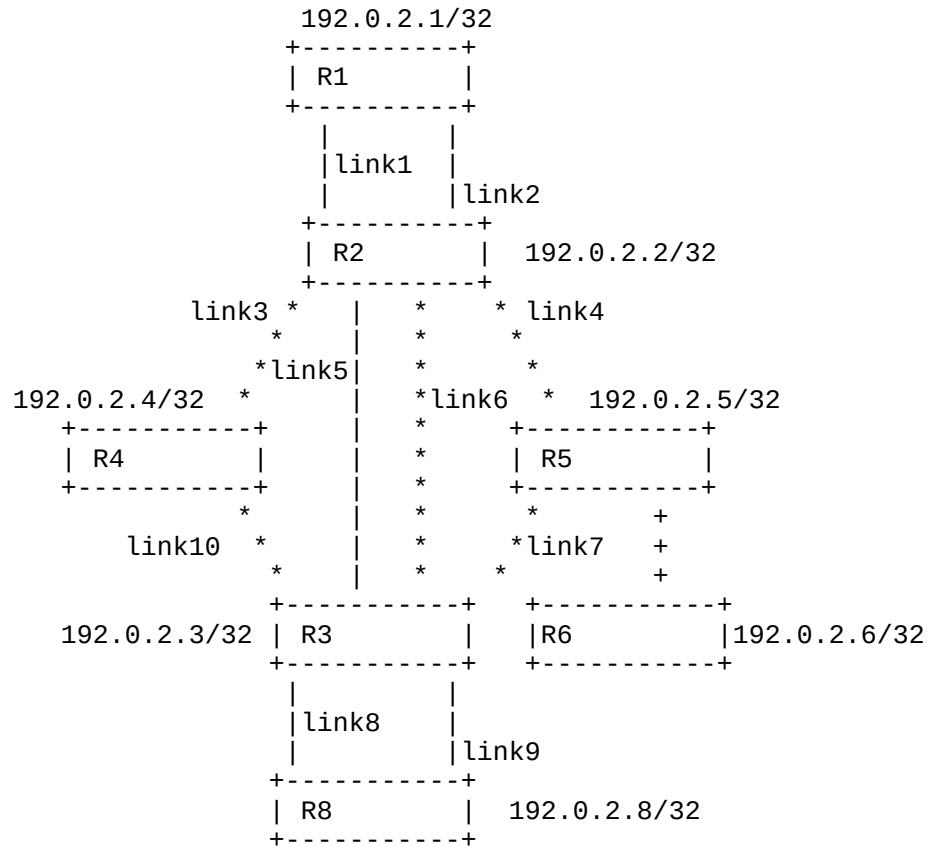


Figure 5: PCECC TE LSP Setup Example

Refer to 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.

* The PCECC will calculate the optimal path according to the given constraints (e.g., BW).

* The 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, the PCECC programs 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 set up by the PCECC. For example, use the primary path as above, then to protect the node R4 locally, the 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 the usage of TEs in the mobile backhaul (MBH). Consider the topology as shown in Figure 6 (where AGG 1...AGG N are Aggregation routers, and Core 1...Core N are Core routers).

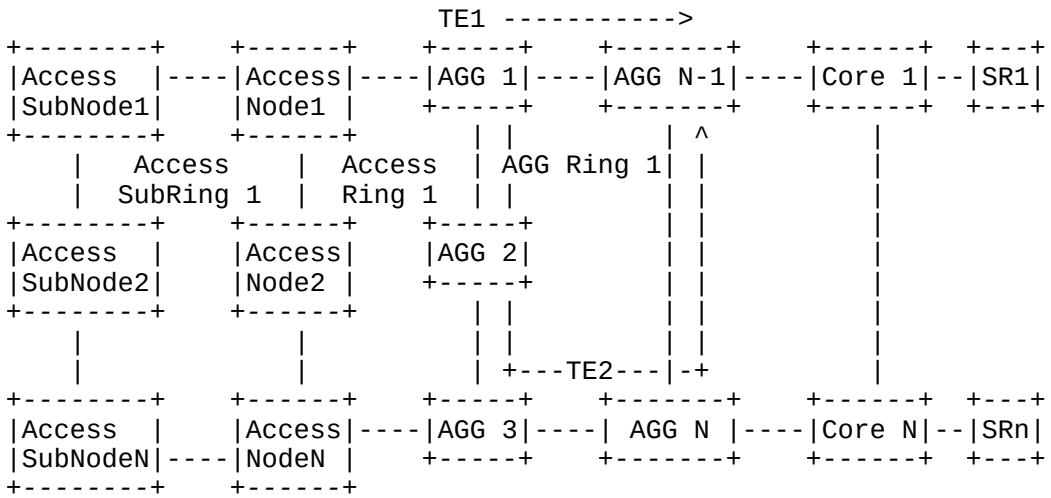


Figure 6: PCECC LB Use Case

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 and AGG N) that are connected to the Core domain (Core 1...Core N) via L2 interfaces.

The 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 two tunnels (one way) from each AGG router to egress AGG routers. There are also many L2 access rings connected to AGG routers.

Service deployment is made by means of Layer 2 Virtual Private Networks (L2VPNs), Virtual Private LAN Services (VPLSs), 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 used as transport towards service routers in case of architecture based on seamless MPLS [MPLS-SEAMLESS].

Load Balancing (LB) between TE tunnels involves distributing network traffic across multiple TE tunnels to optimize the use of available network resources, enhance performance, and ensure reliability. Some common techniques include Equal-Cost Multipath (ECMP) and Unequal-Cost Multipath (UCMP) based on the BW of the TE tunnels.

There is a need to solve the following tasks:

* Perform automatic LB amongst TE tunnels according to current traffic loads.

* Manage TE BW by guaranteeing BW for specific services (such as High-Speed Internet (HSI), IPTV, etc.) and enabling time-based BW reservation (such as Bandwidth on Demand (BoD)).

* 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 LB tasks. LB tasks could be solved by means of the PCECC in the following ways:

* Applications, network services, or operators can ask the SDN controller (PCECC) for LSP-based LB between AGG X and AGG N/AGG N-1 (egress AGG routers that have connections to the core). Each of these will have associated constraints (such as BW, inclusion or exclusion of specific links or nodes, number of paths, Objective Function (OF), need for disjoint LSP paths, etc.).

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

* Depending on the given LSP PST, the 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 the case of SR.

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

* AGG X as an ingress router now has N LSPs towards AGG N and AGG

N-1, which are available for installation to the router's forwarding table and for LB traffic between them. Traffic distribution between those LSPs depends on the particular realization of the hash function on that router.

* Since the PCECC is aware of the Traffic Engineering Database (TED) (TE state) and the LSP Database (LSP-DB), it can manage and prevent possible over-subscriptions and limit the number of available load-balance states. Via a 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 signalling options for establishing Inter-AS TE LSPs: contiguous TE LSPs [RFC5151], stitched TE LSPs [RFC5150], and nested TE LSPs [RFC4206].

The 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 an Inter-AS and Intra-AS PCE Reference Model (as shown in Figure 7) that is provided below in shortened form for the sake of simplicity.

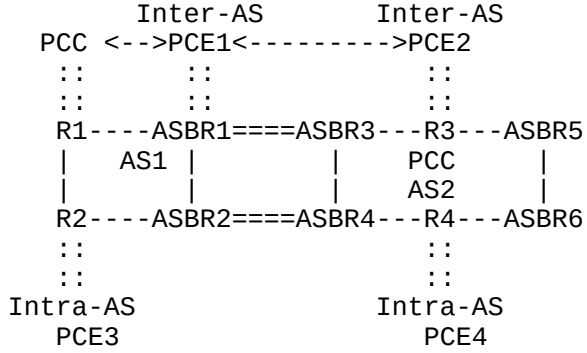
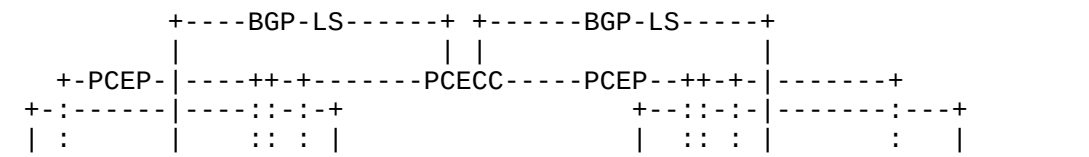


Figure 7: Shortened Form of the Inter-AS and Intra-AS PCE Reference Model

The PCECC belonging to the different domains can cooperate to set up Inter-AS TE LSPs. The stateful Hierarchical PCE (H-PCE) mechanism [RFC8751] could also be used to establish a per-domain PCECC LSP first. These could be stitched together to form an Inter-AS TE LSP as described in [PCE-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-AS and Intra-AS PCEs could be combined in one single PCE if such combined PCE performance is enough to handle 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. In Figure 8, the PCECC maintains a BGP-LS session with Route Reflectors (RRs) in each AS. This allows the RRs to redistribute routes to other BGP routers (clients) without requiring a full mesh. The RRs act as a BGP-LS Propagator, and the PCECC acts as a BGP-LS Consumer [RFC9552].



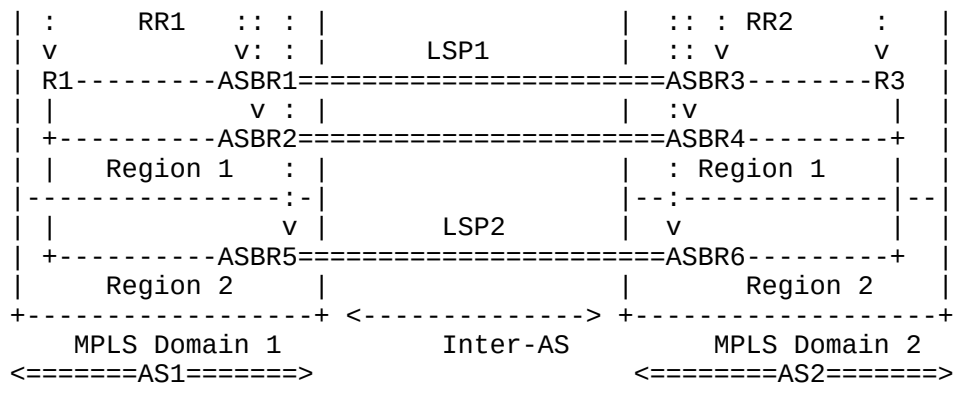


Figure 8: Particular Case of Inter-AS PCE

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

As per Figure 8, LSP1 from R1 to R3 goes via ASBR1 and ASBR3, and it is the primary Inter-AS LSP. LSP2 from R1 to R3 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 the PCE (SDN controller), the PCECC-based Inter-AS TE model should follow the PCECC use case for TE LSP including the requirements described in [RFC5376] with the following details:

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

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

* After the operator’s application or service orchestrator creates a request for tunnel creation of a specific service, the PCECC will receive that request via the Northbound Interface (NBI) (note that the NBI type is implementation-dependent; it could be NETCONF/ YANG, REST, etc.). Then, the PCECC will calculate the optimal path based on the OF and given constraints (i.e., PST, BW, etc.). These constraints include those from [RFC5376], such as priority, AS sequence, preferred ASBR, disjoint paths, and protection type. In this step, we will have two paths: "R1-ASBR1-ASBR3-R3, R1-ASBR5-ASBR6-R3".

* The PCECC will use central control download instructions to the PCC based on the PST. At this stage, it is assumed the PCECC is aware of the label space it controls, and in the case of SR, the SID allocation and distribution is already done.

* The PCECC will send a PCInitiate message [RFC8281] towards the ingress router R1 (PCC) in AS1 and receive the PCRpt message [RFC8231] back from it.

  - If the PST is SR-MPLS, the PCECC will include the SID stack as per [RFC8664]. Optionally, a BSID 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 a PCECC, the PCECC will assign labels along the calculated paths ("R1-ASBR1-ASBR3-R3", "R1-ASBR5-ASBR6-R3") and sets up the path by sending central controller instructions in a PCEP message to each node along the path of the LSPs as per [RFC9050]. After these steps, the PCECC will send a PCUpd message to the ingress R1 router with information about new LSPs and R1 will respond by a PCRpt with LSP(s) status.

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

3.6. PCECC for Multicast LSPs

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

3.6.1. PCECC for the Setup of P2MP/MP2MP LSPs

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

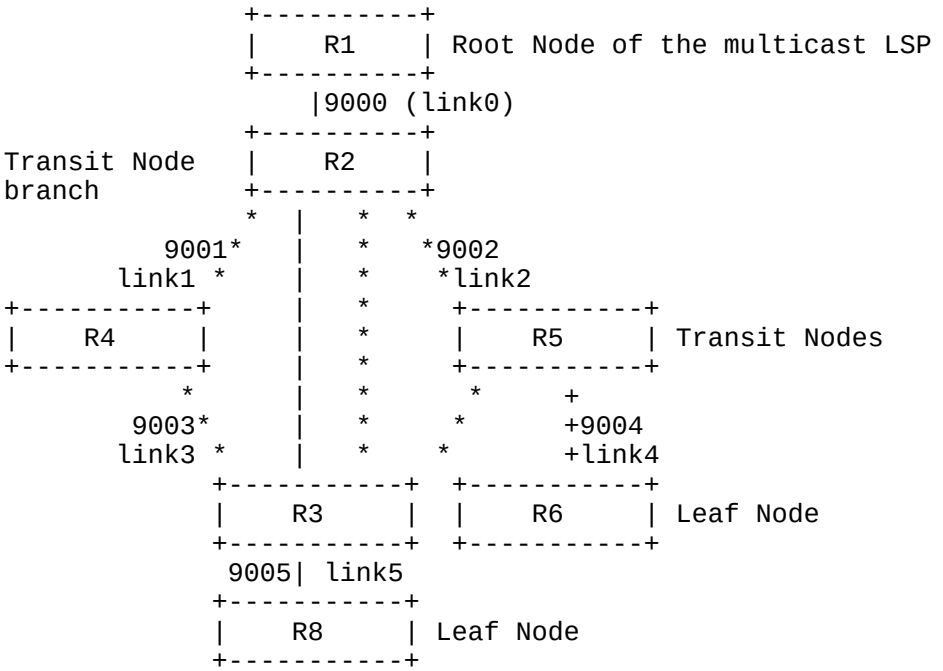


Figure 9: Using a PCECC for the Setup of P2MP/MP2MP LSPs

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

* The PCECC will calculate the optimal P2MP path according to given constraints (i.e., BW).

* The PCECC will provision each node along the path and assign incoming and outgoing labels from R1 to {R6, R8} with the path as "R1-link0-R2-link2-R5-link4-R6" and "R1-link0-R2-link1-R4-link3-R3-link5-R8":

  - R1: Outgoing label 9000 on link0

  - R2: Incoming label 9000 on link0

  - R2: Outgoing label 9001 on link1 (*)

  - R2: Outgoing label 9002 on link2 (*)

  - R5: Incoming label 9002 on link2

  - R5: Outgoing label 9004 on link4

  - R6: Incoming label 9004 on link4

  - R4: Incoming label 9001 on link1

  - R4: Outgoing label 9003 on link3

  - R3: Incoming label 9003 on link3

  - R3: Outgoing label 9005 on link5

  - R8: Incoming label 9005 on link5

* 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 a packet are sent towards R4 and R5 with 9001 and 9002 labels, respectively.

The packet forwarding involves the following:

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. At the same time, it will replicate the packet and swap the label 9000 to 9002 and forward it 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 the 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 it to 9005. 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

This section describes 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 the computation of the primary multicast trees, the 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.

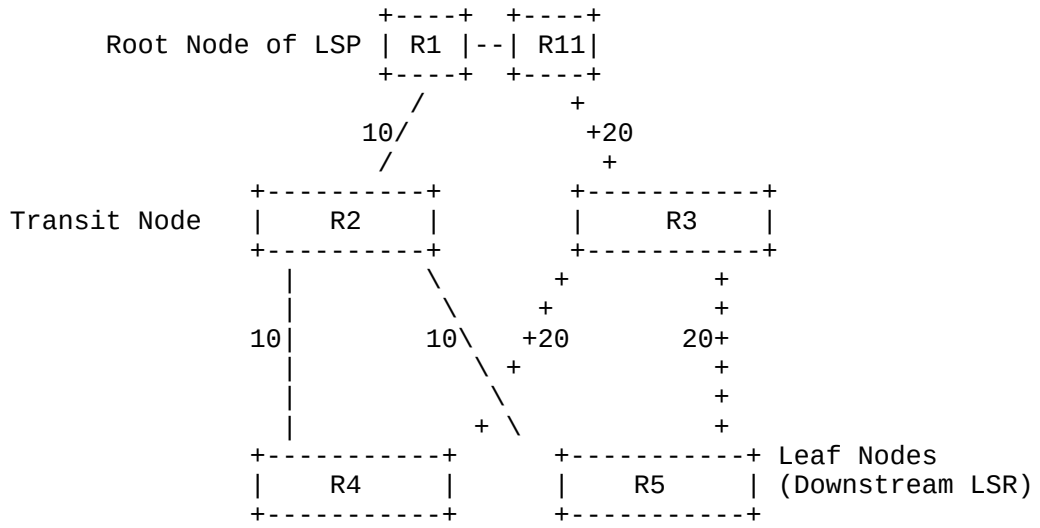


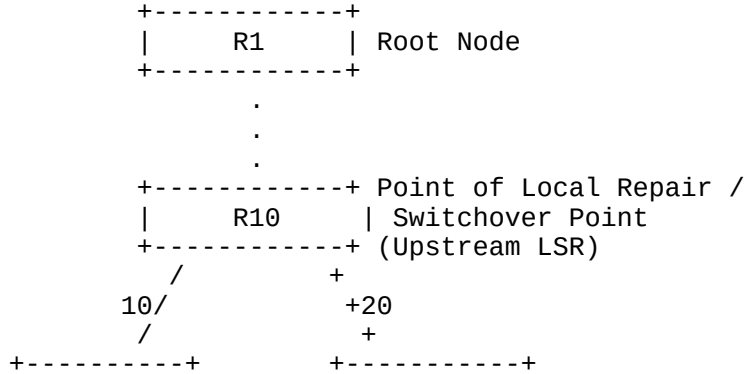
Figure 10: PCECC for the End-to-End Protection of P2MP/MP2MP LSPs

In Figure 10, when the PCECC sets up the primary multicast tree from the root node R1 to the leaves, which is R1->R2->{R4, R5}, it can set up the backup tree at the same time, which is R1->R11->R3->{R4, R5}. Both of them (the 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, path computation, label downloading, and finally forwarding can be done without the complex signalling used in the P2MP RSVP-TE or mLDP.

3.6.3. PCECC for the Local Protection of 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 the Point of Local Repair (PLR), 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 the PLR and protect the network from traffic congestion risks.



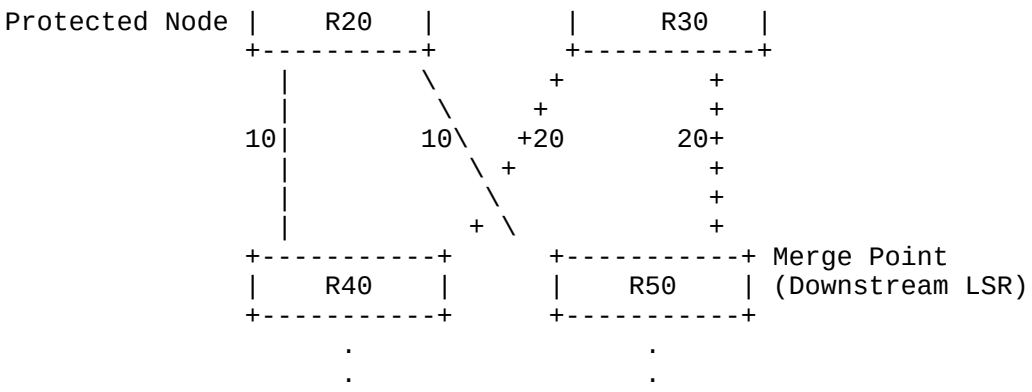


Figure 11: PCECC for the Local Protection of P2MP/MP2MP LSPs

In Figure 11, when the PCECC sets up the primary multicast path around the PLR node R10 to protect node R20, which is R10->R20->{R40, R50}, it can set up the backup path R10->R30->{R40, R50} at the same time. Both the primary forwarding path and the 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 the complex signalling 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 the following:

* MPLS traffic engineering (where it determines what traffic is forwarded into which LSPs),

* SR (where it is used to select which set of forwarding instructions (SIDs) to add to a packet), and

* 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 LB. 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.

The description of traffic flows by the combination of multiple Flow Specification components and their dissemination as traffic Flow Specifications is described for BGP in [RFC8955]. When a PCECC is used to initiate tunnels (such as TE LSPs or SR paths) using PCEP, it is important that the headend of the tunnels understands what traffic to place on each tunnel. [RFC9168] specifies a set of extensions to PCEP to support the dissemination of Flow Specification components where the instructions are passed from the PCECC to the routers using PCEP.

Along with traffic classification, there are a few more questions about the tunnels set up by the PCECC that need to be considered:

* how to use it,

* whether it is a virtual link,

* whether to advertise it in the IGP as a virtual link, and

* what bits of this information to signal to the tail end.

These are out of the 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 LB 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 PCECC, 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. The PCECC will be able to instruct each Service Function Forwarder (SFF) along the SFP.

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

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

* Provide SFC Proxy handling instruction.

The PCECC can play the role of setting the traffic classification rules (as per Section 3.7) at the classifier to impose the Network Service Header (NSH) [RFC8300]. It can also download the forwarding instructions to each SFF along the way so that they could process the NSH and forward accordingly. This includes instructions for the service classifier that handles the context header, metadata, etc. This metadata/context is shared amongst SFs and classifiers, between SFs, and between external systems (such as a PCECC) and SFs. As described in [RFC7665], the SFC encapsulation enables the sharing of metadata/context information along the SFP.

It is also possible to support SFC with SR in conjunction with or without an NSH such as described in [RFC9491] and [SR-SERVICE]. PCECC techniques 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 "Centralized 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 a native IP network, using multiple BGP sessions and a PCE as the centralized controller. It requires the PCECC to send the instructions to the PCCs to build multiple BGP sessions, distribute different prefixes on the established BGP sessions, and assign the different paths to the BGP next hops. The PCEP is used to transfer the key parameters between the PCE and the underlying network devices (PCC) using the PCECC technique. The central control instructions from the PCECC to PCC will identify which prefix should be advertised on which BGP session. There are PCEP extensions defined in [PCEP-NATIVE] for it.

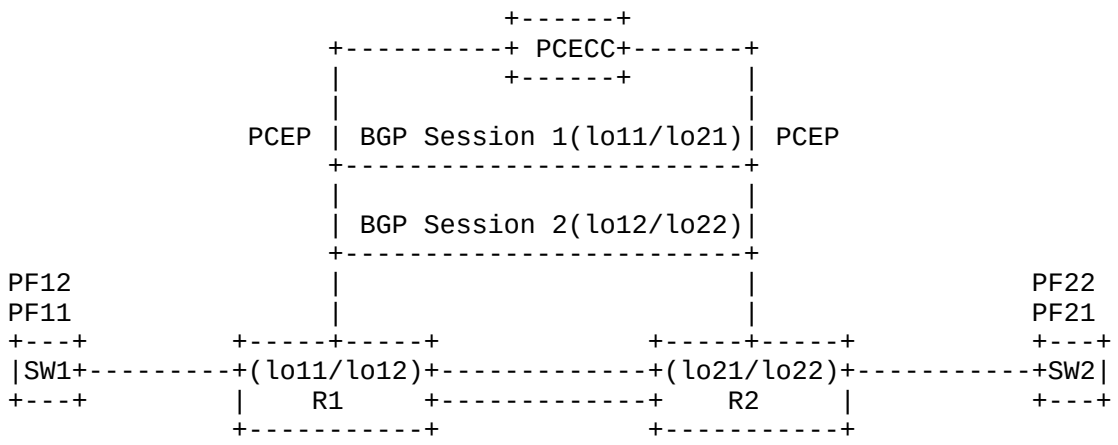


Figure 12: PCECC for Native IP

In the case as shown in Figure 12, the PCECC will instruct both R1 and R2 how to form BGP sessions with each other via PCEP 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 a 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 paths can be derived from a PCE and used at the ingress (a possible mechanism is described in [PCEP-BIER]).

The 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 controllers can use PCEP to instruct the BIER-capable routers on 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 sub-domain-id, BFR-id, etc.) for both node and adjacency.

A possible way to use the PCECC and PCEP extension is described in [PCECC-BIER].

4. IANA Considerations

This document has no IANA actions.

5. Security Considerations

[RFC8283] describes how the security considerations for a PCECC are a little different from those for any other PCE system. PCECC operations rely 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 on 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 PCECC on the network type and services being managed.

6. References

6.1. Normative References

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

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "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., and R. Varga, "Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE", RFC 8231, DOI 10.17487/RFC8231, September 2017, <https://www.rfc-editor.org/info/rfc8231>.

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

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

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

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402,

July 2018, <https://www.rfc-editor.org/info/rfc8402>.

6.2. Informative References

[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", DOI 10.1109/IPDPS.2011.315, May 2011, <https://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>.

[MPLS-SEAMLESS]
Leymann, N., Ed., Decraene, B., Filsfils, C., Konstantynowicz, M., Ed., and D. Steinberg, "Seamless MPLS Architecture", Work in Progress, Internet-Draft, draft-ietf-mpls-seamless-mpls-07, 28 June 2014, <https://datatracker.ietf.org/doc/html/draft-ietf-mpls-seamless-mpls-07>.

[PCE-ID] Li, C., Shi, H., Ed., 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-ietf-pce-controlled-id-space-01, 21 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-controlled-id-space-01>.

[PCE-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-05, 5 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-stateful-interdomain-05>.

[PCE-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://datatracker.ietf.org/doc/html/draft-cbrt-pce-stateful-local-protection-01>.

[PCECC-BIER]
Chen, R., Zhu, C., 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-06, 8 July 2024, <https://datatracker.ietf.org/doc/html/draft-chen-pce-pcep-extension-pce-controller-bier-06>.

[PCECC-SR] Li, Z., Peng, S., Negi, M. S., Zhao, Q., and C. Zhou, "PCE 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-09, 4 July 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-extension-pce-controller-sr-09>.

[PCECC-SRv6]
Li, Z., Peng, S., Geng, X., and M. S. Negi, "PCE 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-ietf-pce-pcep-extension-pce-controller-srv6-03, 18 August 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-extension-pce-controller-srv6-03>.

[PCEP-BIER]
Chen, R., Zhang, Z., Chen, H., Dhanaraj, S., Qin, F., and A. Wang, "PCEP Extensions for Tree Engineering for Bit Index Explicit Replication (BIER-TE)", Work in Progress, Internet-Draft, draft-ietf-pce-bier-te-01, 10 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-bier-te-01>.

[PCEP-NATIVE]
Wang, A., Khasanov, B., Fang, S., Tan, R., and C. Zhu, "Path Computation Element Communication Protocol (PCEP) Extensions for Native IP Networks", Work in Progress, Internet-Draft, draft-ietf-pce-pcep-extension-native-ip-40, 10 September 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-pcep-extension-native-ip-40>.

[PCEP-POLICY]
Koldychev, M., Sivabalan, S., Barth, C., Peng, S., and H. Bidgoli, "Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing (SR) Policy Candidate Paths", Work in Progress, Internet-Draft, draft-ietf-pce-segment-routing-policy-cp-18, 14 October 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-pce-segment-routing-policy-cp-18>.

[RFC1195] Callon, R., "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., "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., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, DOI 10.17487/RFC3209, December 2001, <https://www.rfc-editor.org/info/rfc3209>.

[RFC3985] Bryant, S., Ed. and P. Pate, Ed., "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. and Y. Rekhter, "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. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP

(IBGP)", RFC 4456, DOI 10.17487/RFC4456, April 2006, <https://www.rfc-editor.org/info/rfc4456>.

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

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

[RFC5150] Ayyangar, A., Kompella, K., Vasseur, JP., and A. Farrel, "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., and JP. Vasseur, "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., and A. Lindem, "OSPF for IPv6", RFC 5340, DOI 10.17487/RFC5340, July 2008, <https://www.rfc-editor.org/info/rfc5340>.

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

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

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

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

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

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

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

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

[RFC8355] Filsfils, C., Ed., Previdi, S., Ed., Decraene, B., and R. Shakir, "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>.

[RFC8408] Sivabalan, S., Tantsura, J., Minei, I., Varga, R., and J. Hardwick, "Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages", RFC 8408, DOI 10.17487/RFC8408, July 2018, <https://www.rfc-editor.org/info/rfc8408>.

[RFC8664] Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "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., and P. Mi, "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., and D. King, "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., and D. Voyer, "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., and H. Chen, "PCE-Based Traffic Engineering (TE) in Native IP Networks", RFC 8821, DOI 10.17487/RFC8821, April 2021, <https://www.rfc-editor.org/info/rfc8821>.

[RFC8955] Loibl, C., Hares, S., Raszuk, R., McPherson, D., and M. Bacher, "Dissemination of Flow Specification Rules", RFC 8955, DOI 10.17487/RFC8955, December 2020, <https://www.rfc-editor.org/info/rfc8955>.

[RFC8986] Filsfils, C., Ed., Camarillo, P., Ed., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "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., and J. Scudder, "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., and C. Zhou, "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.,

and D. Afanasiev, "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., and Z. Li, "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., and P. Mattes, "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., and G. Cauchie, "Tree Engineering for Bit Index Explicit Replication (BIER-TE)", RFC 9262, DOI 10.17487/RFC9262, October 2022, <https://www.rfc-editor.org/info/rfc9262>.

[RFC9491] Guichard, J., Ed. and J. Tantsura, Ed., "Integration of the Network Service Header (NSH) and Segment Routing for Service Function Chaining (SFC)", RFC 9491, DOI 10.17487/RFC9491, November 2023, <https://www.rfc-editor.org/info/rfc9491>.

[RFC9522] Farrel, A., Ed., "Overview and Principles of Internet Traffic Engineering", RFC 9522, DOI 10.17487/RFC9522, January 2024, <https://www.rfc-editor.org/info/rfc9522>.

[RFC9552] Talaulikar, K., Ed., "Distribution of Link-State and Traffic Engineering Information Using BGP", RFC 9552, DOI 10.17487/RFC9552, December 2023, <https://www.rfc-editor.org/info/rfc9552>.

[RFC9603] Li, C., Ed., Kaladharan, P., Sivabalan, S., Koldychev, M., and Y. Zhu, "Path Computation Element Communication Protocol (PCEP) Extensions for IPv6 Segment Routing", RFC 9603, DOI 10.17487/RFC9603, July 2024, <https://www.rfc-editor.org/info/rfc9603>.

[RFC9604] Sivabalan, S., Filsfils, C., Tantsura, J., Previdi, S., and C. Li, Ed., "Carrying Binding Label/SID in PCE-Based Networks", RFC 9604, DOI 10.17487/RFC9604, August 2024, <https://www.rfc-editor.org/info/rfc9604>.

[SR-SERVICE] Clad, F., Ed., Xu, X., Ed., 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-10, 23 August 2024, <https://datatracker.ietf.org/doc/html/draft-ietf-spring-sr-service-programming-10>.

Appendix A. Other Use Cases of the PCECC

This section lists some more use cases of the PCECC that were proposed by operators and discussed within the working group but are not in active development at the time of publication. They are listed here for future consideration.

A.1. PCECC for Network Migration

One of the main advantages of the PCECC solution is its backward compatibility. The PCE server can function as a proxy node of the MPLS network for all the new nodes that no longer support the

signalling protocols.

As 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 signalling protocols such as LDP and RSVP-TE, and the new nodes will set up their portion of the forwarding path through the PCECC directly. With the PCECC function as the proxy of these new nodes, MPLS signalling can populate through the network for both old and new nodes.

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

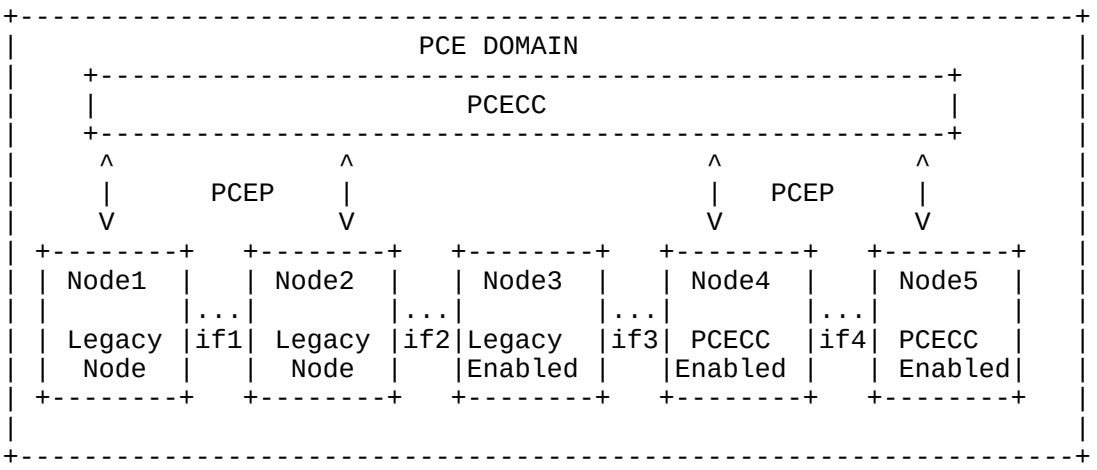


Figure 13: PCECC-Initiated LSP Setup in the Network Migration

In this example, there are five nodes for the TE LSP from the headend (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 the LSP with the destination as Node5.

* The PCECC sends a reply message to Node1 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. With the help of the PCECC, Node3 could proxy the signalling.

* 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 L3VPN [RFC4364] or EVPNs [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 PCECC 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 Provider Edge (PE) router instead

of using the BGP protocols.

The example described in this section is based on network configurations illustrated in Figure 14:

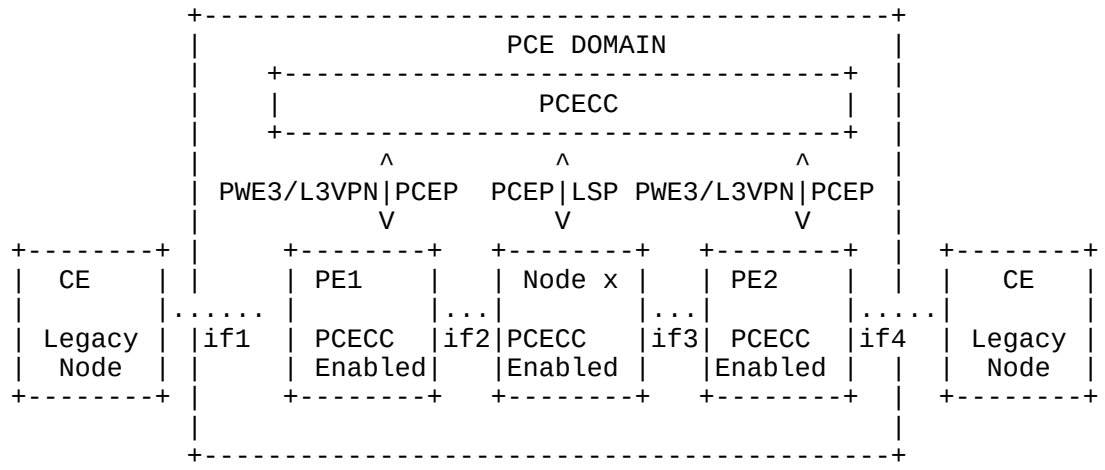


Figure 14: PCECC for L3VPN and PWE3

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

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

[PCE-PROTECTION] claims that there is a 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)

The MapReduce model of distributed computations in computing clusters is widely deployed. In Hadoop (https://hadoop.apache.org/) 1.0 architecture, MapReduce operations occur on big data in the Hadoop Distributed File System (HDFS), where NameNode knows about resources of the cluster and where actual data (chunks) for a particular task are located (which DataNode). Each chunk of data (64 MB or more) should have three saved copies in different DataNodes based on their proximity.

The proximity level currently has a semi-manual allocation and is based on Rack IDs (the assumption is that closer data is better because of access speed / smaller latency).

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

There is a need to go beyond one data center (DC) for Hadoop cluster creation and move towards distributed clusters. In that case, one needs to handle performance and latency issues. Latency depends on the speed of light in the fiber links and on the latency introduced by intermediate devices in between. The latter is closely correlated with network device architecture and performance. The current performance of routers based on Network Processing Unit (NPU) should be enough for creating distributed Hadoop clusters with predicted latency. The performance of software-based routers (mainly Virtual Network Functions (VNFs)) with additional hardware features such as the Data Plane Development Kit (DPDK) is promising but requires additional research and testing.

The main question is how to create a simple but effective architecture for a distributed Hadoop cluster.

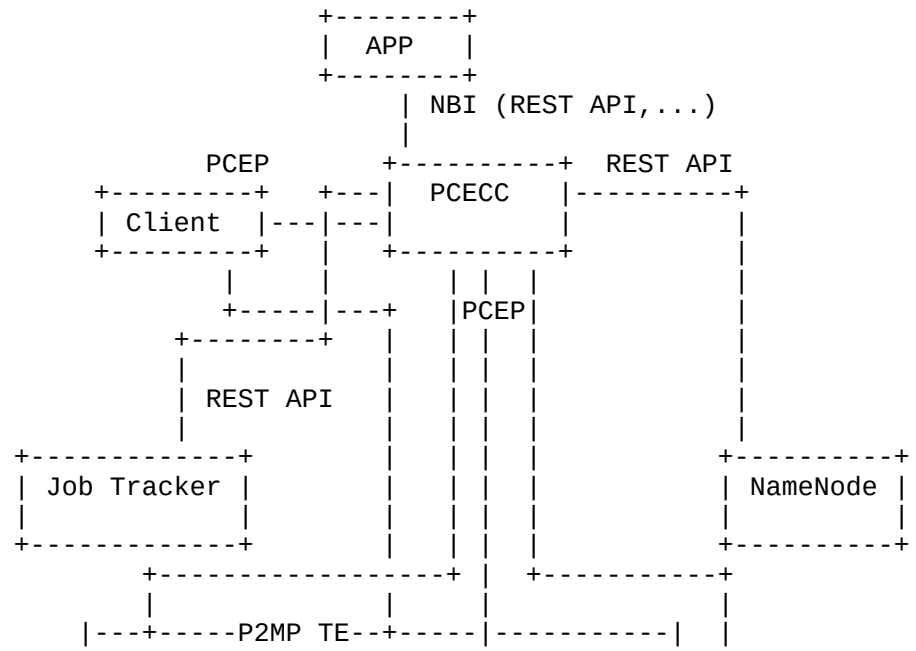
There is research [MAP-REDUCE] that shows how usage of the multicast tree could improve the speed of resource or cluster members' discovery inside the cluster as well as increased redundancy in communications between cluster nodes.

The conventional IP-based multicast may not be appropriate because it requires an additional control plane (IGMP, PIM) and a lot of signalling, which is not suitable for high-performance computations that are very sensitive to latency.

P2MP TE tunnels are more suitable as a potential solution for the creation of multicast-based communications between NameNode as the root and DataNodes as leaves inside the cluster. These P2MP tunnels could be dynamically created and turned down (with no manual intervention). Here, the PCECC comes into play with the main objective of creating an optimal topology for each particular request for MapReduce computation and creating P2MP tunnels with needed parameters such as BW and delay.

This solution will require the use of MPLS label-based forwarding inside the cluster. The 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, and MPLS also exists on Linux and Open vSwitch (OVS).

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



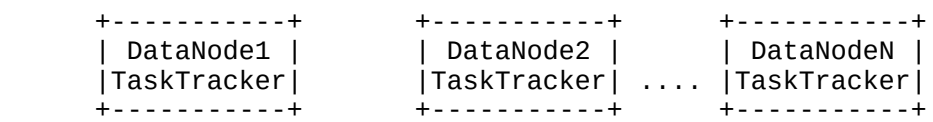


Figure 15: Using Reliable P2MP TE-Based Multicast Delivery for Distributed Computations (MapReduce-Hadoop)

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

* Phase 1: Distributed cluster resource discovery occurs during this phase. JobTracker and NameNode should identify and find available DataNodes according to computing requests from the application (APP). NameNode should query the PCECC about available DataNodes, and NameNode may provide additional constraints to the PCECC such as topological proximity and redundancy level.

  The PCECC should analyze the topology of the distributed cluster and perform a constraint-based path calculation from the client towards the most suitable NameNodes. The PCECC should reply to NameNode with the list of the most suitable DataNodes and their resource capabilities. The topology discovery mechanism for the PCECC will be added later to that framework.

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

* Phase 3: NameNode should send this information to the client, and the PCECC should inform the client about the optimal P2MP path towards DataNodes via a PCEP message.

* Phase 4: The client sends data blocks to those DataNodes for writing via the created P2MP tunnel.

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

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 and Carlos Pignataro for the RTGDIR review. Thanks to Derrell Piper for the SECDIR review. Thanks to Sue Hares for GENART review.

Thanks to Vishnu Pavan Beeram for being the document shepherd and Jim Guichard for being the responsible AD.

Thanks to Roman Danyliw for the IESG review comments.

Contributors

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
AWS
Germany
Email: arachyts@gmail.com

Anton Hulida
AWS
Australia
Email: hulidant@amazon.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
India
Email: dhruv.ietf@gmail.com

Quintin Zhao
Etheric Networks
1009 S Claremont St.
San Mateo, CA 94402
United States of America
Email: quintinzhao@gmail.com

King He
Tencent Holdings Ltd.
Shenzhen
China
Email: kinghe@tencent.com

Boris Khasanov
MTS Web Services (MWS)
Andropova Ave. 18, building 9
Moscow
Russian Federation
Email: bhassanov@yahoo.com