TEAS Working Group
Internet-Draft
Intended status: Experimental
Expires: December 13, 2016

Quintin Zhao
Robin Li
Boris Khasanov
Huawei Technologies
King Ke
Tencent Holdings Ltd.
Luyuan Fang
Microsoft
Chao Zhou
Cisco Systems
Boris Zhang
Telus Communications
Artem Rachitskiy
Anton Gulida
Mobile TeleSystems JLLC

July 8, 2016

The Use Cases for Using PCE as the Central Controller(PCECC) of LSPs
draft-zhao-teas-pcecc-use-cases-01

Abstract

In certain networks deployment scenarios, service providers would like to keep all the existing MPLS functionalities in both MPLS and GMPLS network while removing the complexity of existing signaling protocols such as LDP and RSVP-TE. In this document, we propose to use the PCE as a central controller so that LSP can be calculated/signaled/initiated/downloaded/managed through a centralized PCE server to each network devices along the LSP path while leveraging the existing PCE technologies as much as possible.

This draft describes the use cases for using the PCE as the central controller where LSPs are calculated/setup/initiated/downloaded/ maintained through extending the current PCE architectures and extending the PCEP.

Requirements Language

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

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute

working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on December 13, 2016.

Table of Contents

1. Introduction

1.1. Background

In certain network deployment scenarios, service providers would like to have the ability to dynamically adapt to a wide range of customer's requests for the sake of flexible network service delivery, SDN has provides additional flexibility in how the network is operated comparing the traditional network.

The existing networking ecosystem has become awfully complex and highly demanding in terms of robustness, performance, scalability, flexibility, agility, etc. By migrating to the SDN enabled network from the existing network, service providers and network operators must have a solution which they can evolve easily from the existing network into the SDN enabled network while keeping the network services remain scalable, guarantee robustness and availability etc.

Taking the smooth transition between traditional network and the new SDN enabled network into account, especially from a cost impact assessment perspective, using the existing PCE components from the current network to function as the central controller of the SDN network is one choice, which not only achieves the goal of having a centralized controller to provide the functionalities needed for the central controller, but also leverages the existing PCE network components.

The Path Computation Element communication Protocol (PCEP) provides mechanisms for Path Computation Elements (PCEs) to perform route computations in response to Path Computation Clients (PCCs) requests. PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model draft [I-D. draft-ietf-pce-stateful-pce] describes a set of extensions to PCEP to enable active control of MPLS-TE and GMPLS tunnels.

[I-D.crabbe-pce-pce-initiated-lsp] describes the setup and teardown of PCE-initiated LSPs under the active stateful PCE model, without the need for local configuration on the PCC, thus allowing for a dynamic MPLS network that is centrally controlled and deployed.

[I-D.ali-pce-remote-initiated-gmpls-lsp] complements [I-D. draft-crabbe-pce-pce-initiated-lsp] by addressing the requirements for remote-initiated GMPLS LSPs.

SR technology leverages the source routing and tunneling paradigms. A source node can choose a path without relying on hop-by-hop signaling protocols such as LDP or RSVP-TE. Each path is specified as a set of "segments" advertised by link-state routing protocols

(IS-IS or OSPF). [I-D.filsfils-spring-segment-routing] provides an introduction to SR technology. The corresponding IS-IS and OSPF extensions are specified in [I-D.ietf-isis-segment-routing-extensions] and [I-D.psenak-ospf-segment-routing-extensions], respectively.

A Segment Routed path (SR path) can be derived from an IGP Shortest Path Tree (SPT). Segment Routed Traffic Engineering paths (SR-TE paths) may not follow IGP SPT. Such paths may be chosen by a suitable network planning tool and provisioned on the source node of the SR-TE path.

It is possible to use a stateful PCE for computing one or more SR-TE paths taking into account various constraints and objective functions. Once a path is chosen, the stateful PCE can instantiate an SR-TE path on a PCC using PCEP extensions specified in [I-D.crabbe-pce-pce-initiated-lsp] using the SR specific PCEP extensions described in [I-D.sivabalan-pce-segment-routing].

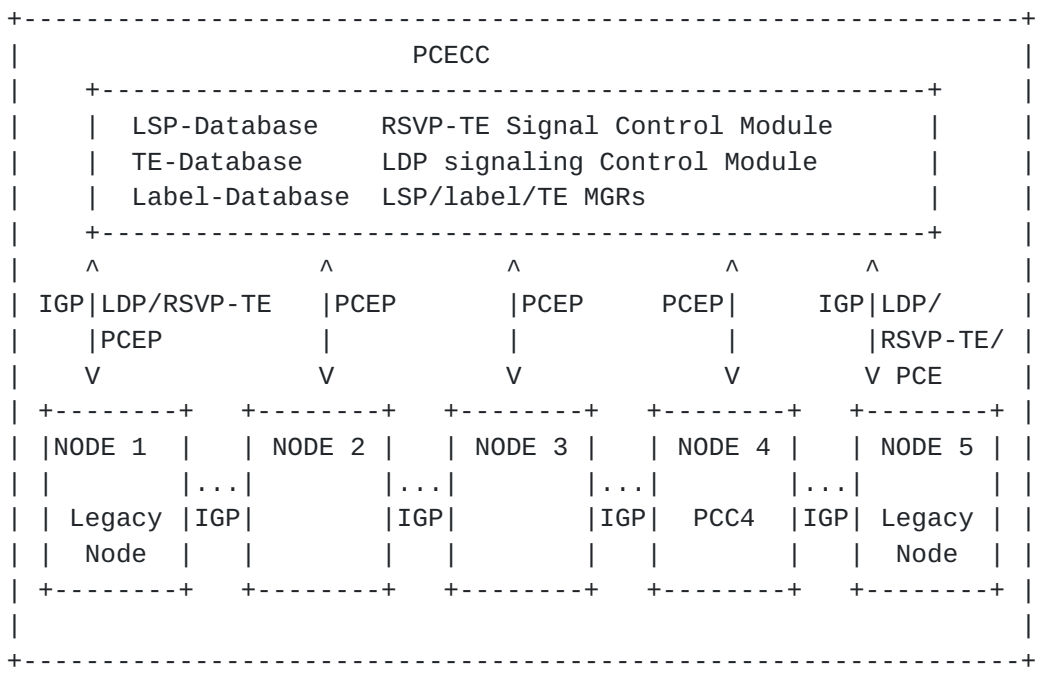
By using the solutions provided from above drafts, LSP in both MPLS and GMPLS network can be setup/delete/maintained/synchronized through a centrally controlled dynamic MPLS network. Since in these solutions, the LSP is need to be signaled through the head end LER to the tail end LER, there are either RSVP-TE signaling protocol need to be deployed in the MPLS/GMPLS network, or extend TGP protocol with node/adjacency segment identifiers signaling capability to be deployed.

The PCECC solution proposed in this document allow for a dynamic MPLS network that is eventually controlled and deployed without the deployment of RSVP-TE protocol or extended IGP protocol with node/adjacency segment identifiers signaling capability while providing all the key MPLS functionalities needed by the service providers. These key MPLS features include MPLS P2P LSP, P2MP/MP2MP LSP, MPLS protection mechanism etc. In the case that one LSP path consists legacy network nodes and the new network nodes which are centrally controlled, the PCECC solution provides a smooth transition step for users.

1.2. Using the PCE as the Central Controller (PCECC) Approach

With PCECC, it not only removes the existing MPLS signaling totally from the control plane without losing any existing MPLS functionalities, but also PCECC achieves this goal through utilizing the existing PCEP without introducing a new protocol into the network.

The following diagram illustrates the PCECC architecture.



Through the draft, we call the combination of the functionality for global label range signaling and the functionality of LSP setup/download/cleanup using the combination of global labels and local labels as PCECC functionality.

Current MPLS label has local meaning. That is, MPLS label allocated locally and signaled through the LDP/RSVP-TE/BGP etc dynamic signaling protocol.

As the SDN(Service-Driven Network) technology develops, MPLS global label has been proposed again for new solutions. [I-D.li-mpls-global-label-usecases] proposes possible usecases of MPLS global label. MPLS global label can be used for identification of the location, the service and the network in different application scenarios. From these usecases we can see that no matter SDN or traditional application scenarios, the new solutions based on MPLS global label can gain advantage over the existing solutions to facilitate service provisions. The solution choices are described in [I-D.li-mpls-global-label-framework].

To ease the label allocation and signaling mechanism, also with the new applications such as concentrated LSP controller is introduced, PCE can be conveniently used as a central controller and MPLS global label range negotiator.

The later section of this draft describes the user cases for PCE server and PCE clients to have the global label range negotiation and local label range negotiation functionality.

To empower networking with centralized controllable modules, there are many choices for downloading the forwarding entries to the data plane, one way is the use of the OpenFlow protocol, which helps devices populate their forwarding tables according to a set of instructions to the data plane. There are other candidate protocols to convey specific configuration information towards devices also. Since the PCEP protocol is already deployed in some of the service network, to leverage the PCEP to populated the MPLS forwarding table is a possible good choice.

For the centralized network, the performance achieved through distributed system can not be easy matched if all of the forwarding path is computed, downloaded and maintained by the centralized controller. The performance can be improved by supporting part of the forwarding path in the PCECC network through the segment routing mechanism except that the adjacency IDs for all the network nodes and links are propagated through the centralized controller instead of using the IGP extension.

The node and link adjacency IDs can be negotiated through the PCECC with each PCECC clients and these IDs can be just taken from the global label range which has been negotiated already.

With the capability of supporting SR within the PCECC architecture, all the p2p forwarding path protection use cases described in the draft [I-D.ietf-spring-resiliency-use-cases] will be supported too within the PCECC network. These protection alternatives include end-to-end path protection, local protection without operator management and local protection with operator management.

With the capability of global label and local label existing at the same time in the PCECC network, PCECC will use compute, setup and maintain the P2MP and MP2MP LSP using the local label range for each network nodes.

With the capability of setting up/maintaining the P2MP/MP2MP LSP within the PCECC network, it is easy to provide the end-end managed path protection service and the local protection with the operation management in the PCECC network for the P2MP/MP2MP LSP, which includes both the RSVP-TE P2MP based LSP and also the mLDP based LSP.

2. Terminology

The following terminology is used in this document.

IGP: Interior Gateway Protocol. Either of the two routing protocols, Open Shortest Path First (OSPF) or Intermediate System to Intermediate System (IS-IS).

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

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

TE: Traffic Engineering.

3. PCEP Requirements

Following key requirements associated PCECC should be considered when designing the PCECC based solution:

1. Path Computation Element (PCE) clients supporting this draft MUST have the capability to advertise its PCECC capability to the PCECC.

2. Path Computation Element (PCE) supporting this draft MUST have the capability to negotiate a global label range for a group of clients.

3. Path Computation Client (PCC) MUST be able ask for global label range assigned in path request message .

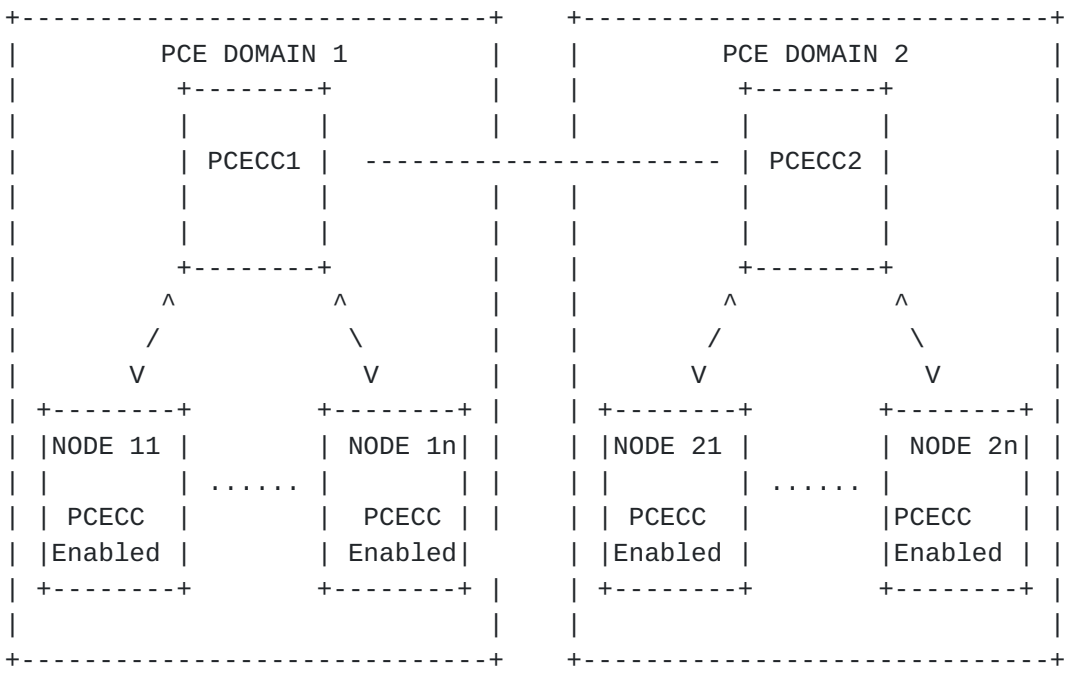
4. PCE are not required to support label reserve service. Therefore, it MUST be possible for a PCE to reject a Path Computation Request message with a reason code that indicates no support for label reserve service.

5. PCEP SHOULD provide a means to return global label range and LSP label assignments of the computed path in the reply message.

6. PCEP SHOULD provide a means to download the MPLS forwarding entry to the PCECC's clients.

4. Use Cases of PCECC for Label Resource Reservations

Example 1 to 2 are based on network configurations illustrated using the following figure:



Example 1: Shared Global Label Range Reservation

- PCECC Clients nodes report MPLS label capability to the central controller PCECC.
- The central controller PCECC collects MPLS label capability of all nodes. Then PCECC can calculate the shared MPLS global label range for all the PCECC client nodes.
- In the case that the shared global label range need to be negotiated across multiple domains, the central controllers of these domains need to be communicate to negotiate a common global label range.
- The central controller PCECC notifies the shared global label range to all PCECC client nodes.

Example 2: Global Label Allocation

- PCECC Client node1 send global label allocation request to the central controller PCECC1.

- The central controller PCECC1 allocates the global label for FEC1 from the shared global label range and sends the reply to the client node1.

- The central controller PCECC1 notifies the allocated label for FEC1 to all PCECC client nodes within domain 1.

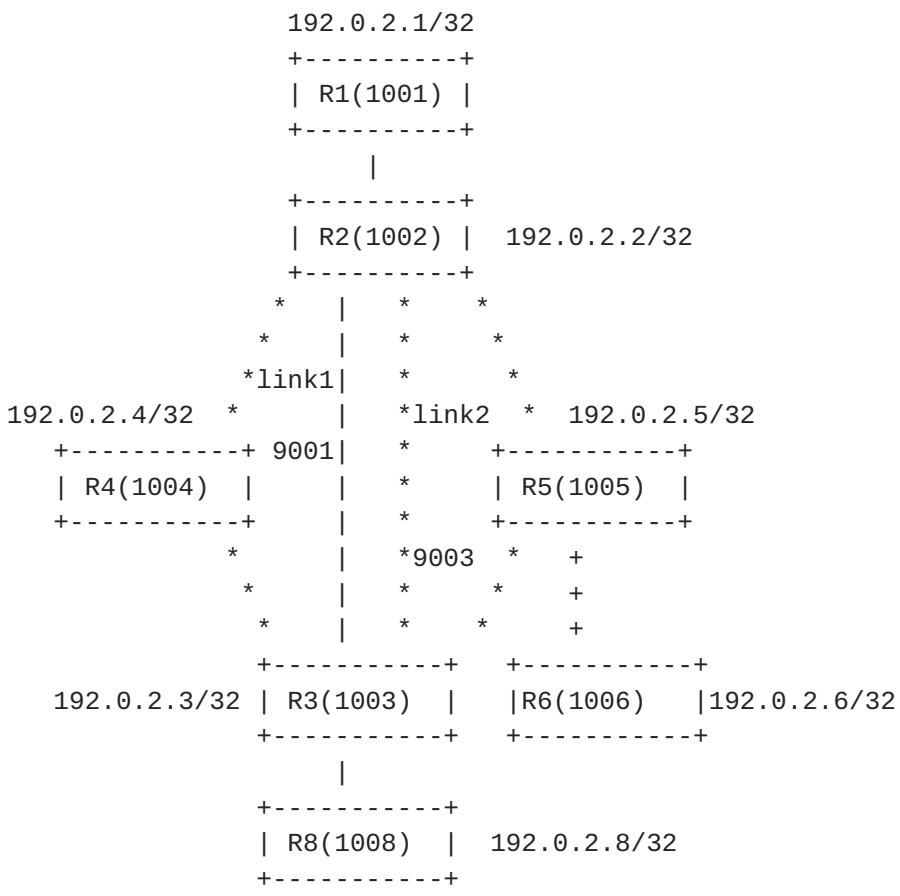
5. Using PCECC for SR without the IGP Extension

For the centralized network, the performance achieved through distributed system can not be easy matched if all of the forwarding path is computed, downloaded and maintained by the centralized controller. The performance can be improved by supporting part of the forwarding path in the PCECC network through the segment routing mechanism except that node segment ids and adjacency segment IDs for all the network are allocated dynamically and propagated through the centralized controller instead of using the IGP extension.

When the PCECC is used for the distribution of the node segment ID and adjacency segment ID, the node segment ID is allocated from the global label pool. For the allocation of adjacency segment ID, there are two choices, the first choice is that it is allocated from the local label pool, the second choice is that it is allocated from the global label pool. The advantage for the second choice is that the depth of the label stack for the forwarding path encoding will be reduced since adjacency segment ID can signal the forwarding path without adding the node segment ID in front of it. In this version of the draft, we use the fist choice for now. We may update the draft to reflect the use of the second choice.

Same as the SR solutions, when PCECC is used as the central controller, the support of FRR on any topology can be pre-computated and setup without any additional signaling (other than the regular IGP/BGP protocols) including the support of shared risk constraints, support of node and link protection and support of microloop avoidance.

The following example illustrate the use case where the node segment ID and adjacency segment ID are allocated from the global label allocated for SR path.



5.1. Use Cases of PCECC for SR Best Effort(BE) Path

In this mode of the solution, the PCECC just need to allocate the node segment ID and adjacency ID without calculating the explicit path for the SR path. The ingress of the forwarding path just need to encapsulate the destination node segment ID on top of the packet. All the intermediate nodes will forward the packet based on the final destination node segment id. It is similar to the LDP LSP forwarding except that label swapping is using the same global label both for the in segment and out segment in each hop.

The p2p SR BE path examples are explained as bellow:

Note that the node segment id for each node from the shared global labels ranges negotiated already.

Example 1:

R1 may send a packet to R8 simply by pushing an SR header with segment list {1008}. The path can be: R1-R2-R3-R8 or R1-R2-R5-R8 depending on the route calculation on node R2.

Example 2: local link/node protection:

For the packet which has destination of R3 and after that, R2 may preinstalled the backup forwarding entry to protect the R4 node, the pre-installed the backup path can go through either node5 or link1 or link2 between R2 and R3. The backup path calculation is locally decided by R2 and any existing IP FRR algorithms can be used here.

5.2. Use Cases of PCECC for SR Traffic Engineering (TE) Path

In the case of traffic engineering path is needed, the PCECC need to allocate the node segment ID and adjacency ID, and at the same time PCECC calculates the explicit path for the SR path and pass this explicit path represented with a sequence of node segment id and adjacency id. The ingress of the forwarding path need to encapsulate the stack of node segment id and adjacency id on top of the packet. For the case where strict traffic engineering path is needed, all the intermediate nodes and links will be specified through the stack of labels so that the packet is forwarded exactly as it is wanted.

Even though it is similar to TE LSP forwarding where forwarding path is engineered, but the Qos is only guaranteed through the enforce of the bandwidth admission control. As for the RSVP-TE LSP case, Qos is guaranteed through the link bandwidth reservation in each hop of the forwarding path.

The p2p SR traffic engineering path examples are explained as bellow:

Note that the node segment id for each node is allocated from the shared global labels ranges negotiated already and adjacency segment ids for each link are allocated from the local label pool for each node.

Example 1:

R1 may send a packet P1 to R8 simply by pushing an SR header with segment list {1008}. The path should be: R1-R2-R3-R8.

Example 2:

R1 may send a packet P2 to R8 by pushing an SR header with segment list {1002, 9001, 1008}. The path should be: R1-R2-(1)link-R3-R8.

Example 3:

R1 may send a packet P3 to R8 while avoiding the links between R2 and R3 by pushing an SR header with segment list {1004, 1008}. The path should be : R1-R2-R4-R3-R8

The p2p local protection examples for SR TE path are explained as below:

Example 4: local link protection:

- R1 may send a packet P4 to R8 by pushing an SR header with segment list {1002, 9001, 1008}. The path should be: R1-R2-(1)link-R3-R8.

- When node R2 receives the packet from R1 which has the header of R2- (1)link-R3-R8, and also find out there is a link failure of link1, then it will send out the packet with header of R3-R8 through link2.

Example 5: local node protection:

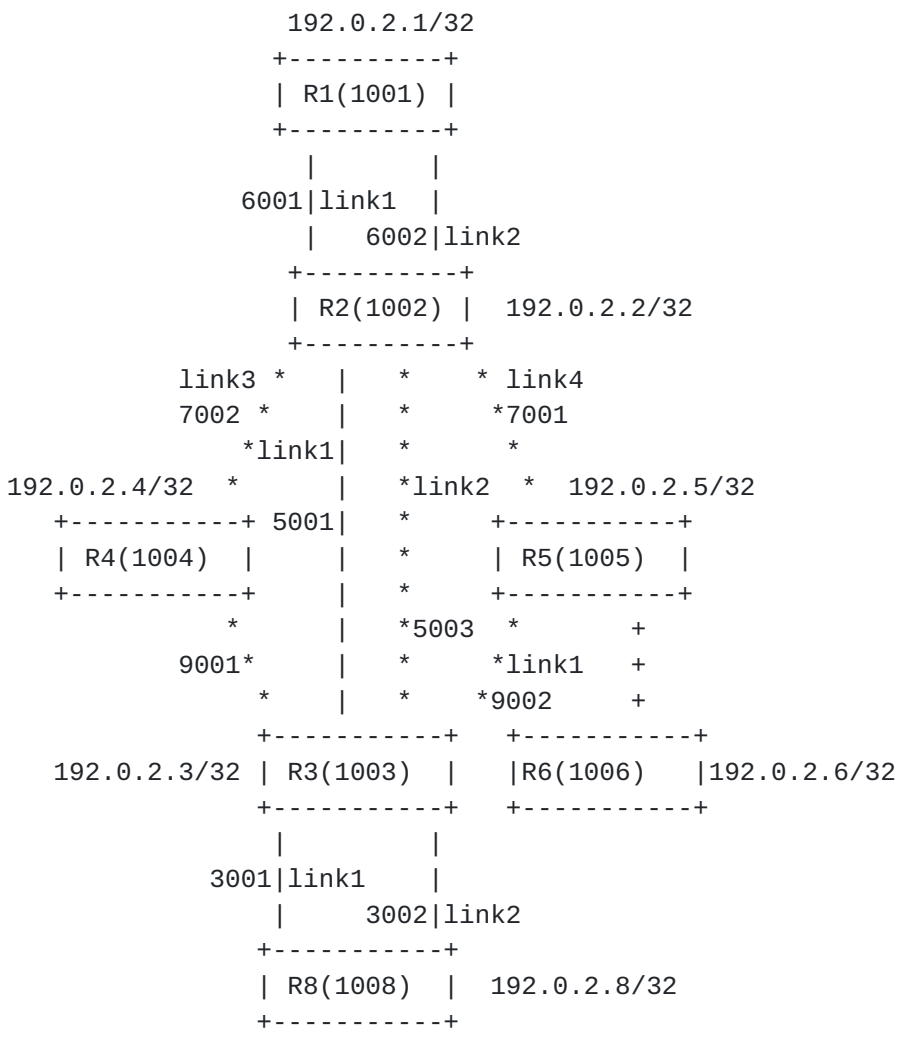
- R1 may send a packet P5 to R8 by pushing an SR header with segment list {1004, 1008}. The path should be : R1-R2-R4-R3-R8.

- When node R2 receives the packet from R1 which has the header of {1004, 1008}, and also find out there is a node failure for node4, then it will send out the packet with header of {1005, 1008} to node5 instead of node4.

6. Use Cases of PCECC for TE LSP

In the previous sections, we have discussed the cases where the SR path is setup through the PCECC. Although those cases give the simplicity and scalability, but there are existing functionalities for the traffic engineering path such as the bandwidth guarantee through the full forwarding path and the multicast forwarding path which SR based solution cannot solve. Also there are cases where the depth of the label stack may have been an issue for existing deployment and certain vendors.

So to address these issues, PCECC architecture should also support the TE LSP and multicast LSP functionalities. To achieve this, the existing PCEP can be used to communicate between the PCE server and PCE's client PCC for exchanging the path request and reply information regarding to the TE LSP info. In this case, the TE LSP info is not only the path info itself, but it includes the full forwarding info. Instead of letting the ingress of LSP to initiate the LSP setup through the RSVP-TE signaling protocol, with minor extensions, we can use the PCEP to download the complete TE LSP forwarding entries for each node in the network.



TE LSP Setup Example

- Node1 sends a path request message for the setup of TE LSP from R1 to R8.

- PCECC program each node along the path from R1 to R8 with the primary path: {R1, link1, 6001}, {R2, link3, 7002], {R4, link0, 9001}, {R3, link1, 3001}, {R8}.

- For the end to end protection, PCECC program each node along the path from R1 to R8 with the secondary path: {R1, link2, 6002}, {R2, link4, 7001], {R5, link1, 9002}, {R3, link2, 3002}, {R8}.

- It is also possible to have a secondary backup path for the local node protection setup by PCECC. For example, the primary path is still same as what we have setup so far, then to protect the node R4 locally, PCECC can program the secondary path like this: {R1,

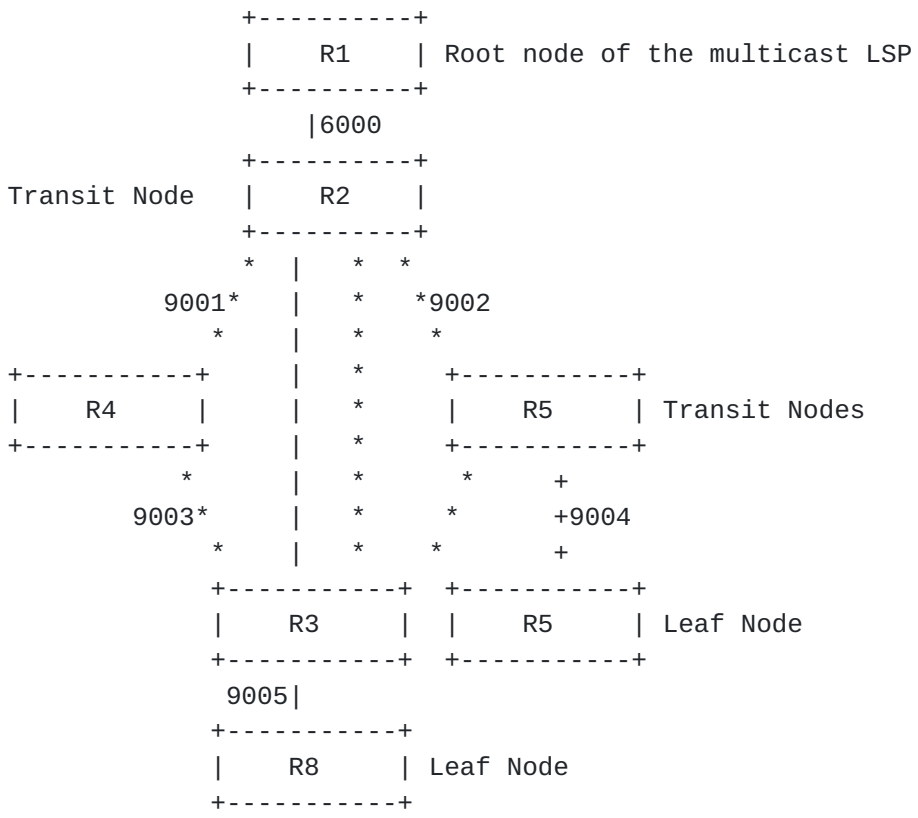
link1, 6001}, {R2, link1, 5001}, {R3, link1, 3001}, {R8}. By doing this, the node R4 is locally protected.

7. Use Cases of PCECC for Multicast LSPs

The current multicast LSPs are setup either using the RSVP-TE P2MP or mLDP protocols. The setup of these LSPs not only need a lot of manual configurations, but also it is also complex 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 signal lings, but also it can compute the disjoint primary path and secondary path efficiently.

7.1. Using PCECC for P2MP/MP2MP LSPs' Setup

With the capability of global label and local label existing at the same time in the PCECC network, PCECC will use compute, setup and maintain the P2MP and MP2MP lsp using the local label range for each network nodes.



The P2MP examples are explained here:

Step1: R1 may send a packet P1 to R2 simply by pushing an label of 6000 to the packet.

Step2: After R2 receives the packet with label 6000, it will forwarding to R4 by pushing header of 9001 and R5 by pusing header of 9002.

Step3: After R4 receives the packet with label 9001, it will forwarding to R3 by pushing header of 9003. After R5 receives the packet with label 9002, it will forwarding to R5 by pushing header of 9004.

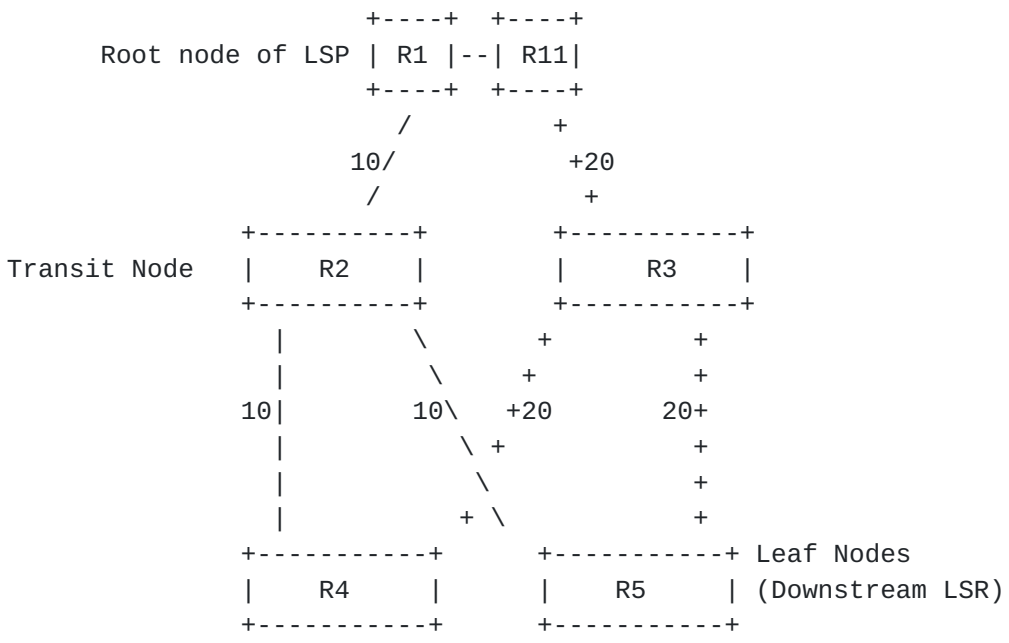
Step3: After R3 receives the packet with label 9003, it will forwarding to R8 by pushing header of 9005

7.2. Use Cases of PCECC for the Resiliency of P2MP/MP2MP LSPs

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

In this section we describe the end-end managed path protection service and the local protection with the operation management in the PCECC network for the P2MP/MP2MP LSP, which includes both the RSVP-TE P2MP based LSP and also the mLDP based LSP.

An end-to-end protection (for nodes and links) principle can be applied for computing backup P2MP or MP2MP LSPs. During computation of the primarily multicast trees, PCECC server may also be taken into consideration to compute a secondary tree. A PCE may compute the primary and backup P2MP or MP2Mp LSP together or sequentially.

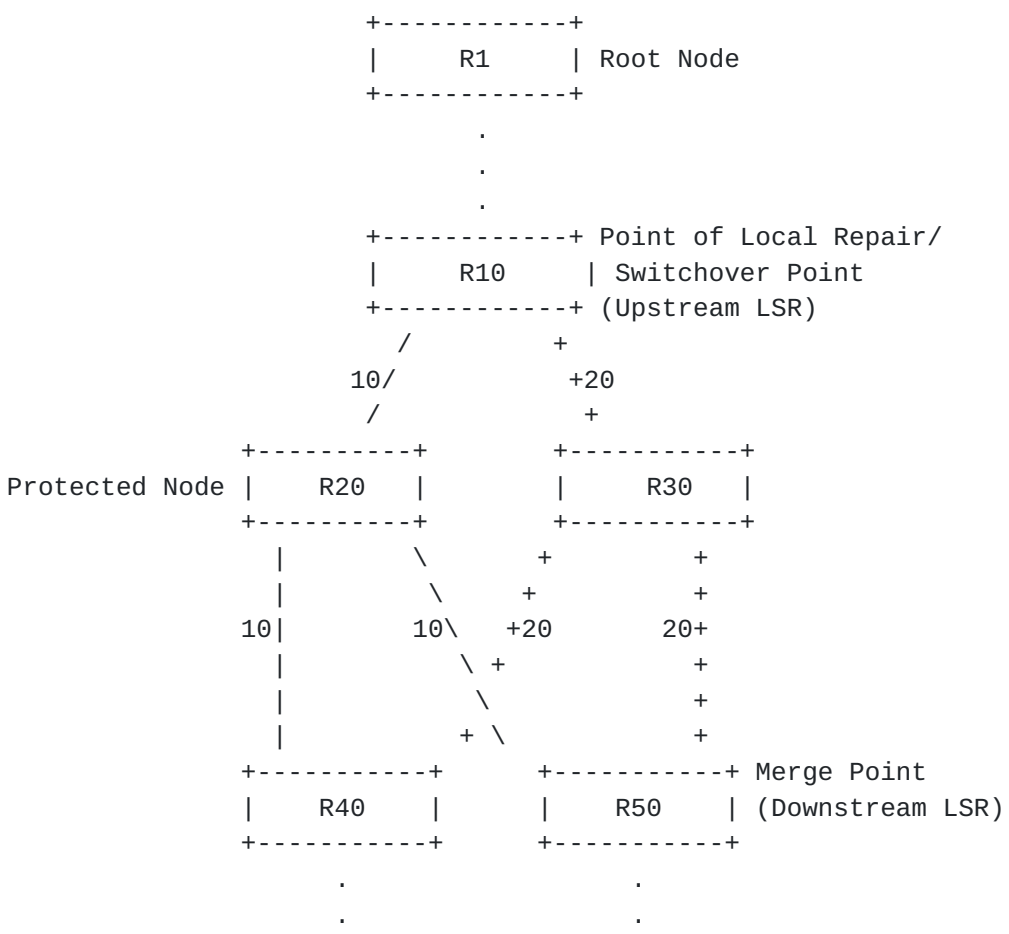


In the example above, when the PCECC setup the primary multicast tree from the root node R1 to the leafs, which is R1->R2->{R4, R5}, at same time, it can setup the backup tree, which is R11->R3->{R4, R5}. Both the these two primary forwarding tree and secondary forwarding tree will be downloaded to each routers along the primary path and the secondary path. The traffic will be forwarded through the R1->R2->{R4, R5} path normally, and when there is a node in the primary tree, then the root node R1 will switch the flow to the backup tree, which is R11->R3->{R4, R5}. By using the PCECC, the path computation and forwarding path downloading can all be done without the complex signaling used in the P2MP RSVP-TE or mLDP.

7.2.2. 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 back LSP among PLR, the protected node, and MPs (the downstream nodes of the protected node). In the cases where the amount of downstream nodes are huge, this mechanism can avoid unnecessary packet duplication on PLR, so that protect the network from traffic congestion risk.



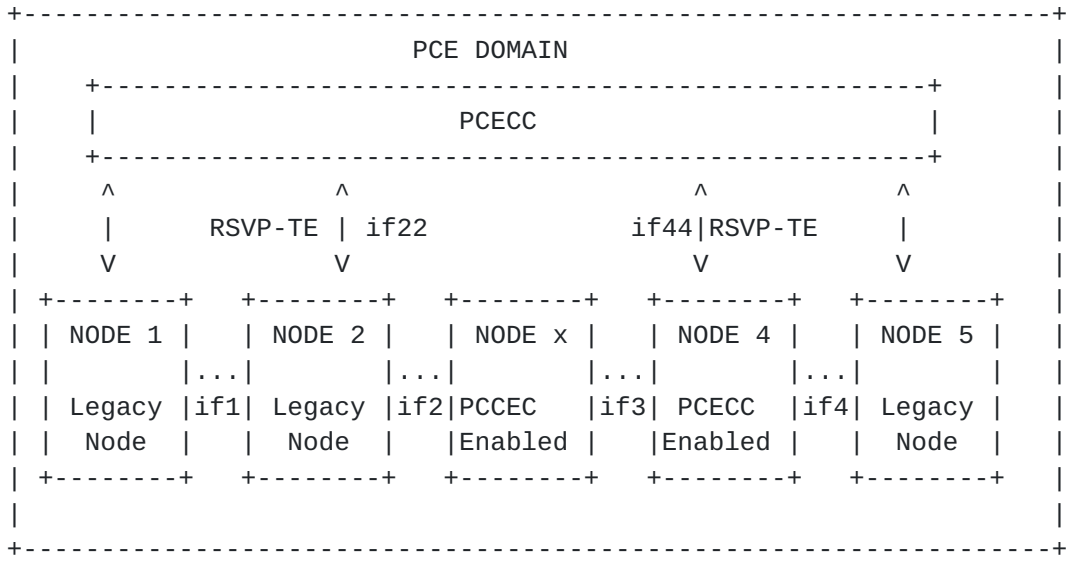
In the example above, when the PCECC setup the primary multicast path around the PLR node R10 to protect node R20, which is R10->R20->{R40, R50}, at same time, it can setup the backup path R10->R30->{R40, R50}. Both the these two primary forwarding path and secondary forwarding path will be downloaded to each routers along the primary path and the secondary path. The traffic will be forwarded through the R10->R20->{R40, R50} path normally, and when there is a node failure for node R20, then the PLR node R10 will switch the flow to the backup path, which is R10->R30->{R40, R50}. By using the PCECC, the path computation and forwarding path downloading can all be done without the complex signaling used in the P2MP RSVP-TE or mLDP.

8. Use Cases of PCECC for LSP in the Network Migration

One of the main advantages for PCECC solution is that it has backward compatibility naturally since the PCE server itself can function as a proxy node of MPLS network for all the new nodes which don't support the existing MPLS signaling protocol anymore.

As it is illustrated in the following example, the current network will migrate to a total PCECC controlled network gradually by replacing the legacy nodes. During the migration, the legacy nodes still need to signal using the existing MPLS protocol such as LDP and RSVP-TE, and the new nodes setup 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 network as normal.

Example described in this section is based on network configurations illustrated using the following figure:



Example: PCECC Initiated LSP Setup In the Network Migration

In this example, there are five nodes for the TE LSP from head end (node1) to the tail end (node5). Where the NodeX is central controlled and other nodes are legacy nodes.

- Node1 sends a path request message for the setup of LSP destinating to Node5.

- PCECC sends a reply message for LSP setup with path (node1, if1), (node2, if22), (node-PCECC, if44), (node4, if4), Nnode5.

- Node1, Node2, Node-PCECC, Node 5 will setup the LSP to Node5 normally using the local label as normal.

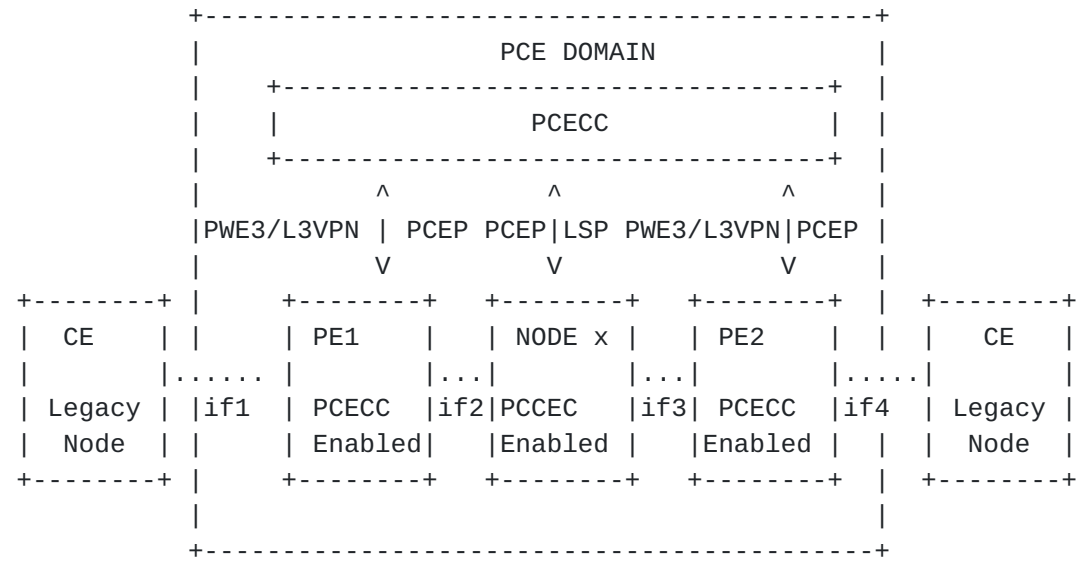
- Then the PCECC will program the outsegment of Node2, the insegment of Node4, and the insegment/outsegment for NodeX.

9. Use Cases of PCECC for L3VPN and PWE3

The existing services using MPLS LSP tunnels based on MPLS signalling mechanism such L3VPN, PWE3 and IPv6 can be simplified by using the PCECC to negoitate the label assignments for the L3VPN, PWE3 and Ipv6.

In the case of L3VPN, VPN labels can be negotiated and distributed through the PCECC PCEP among the PE router instead of using the BGP protocols.

Example described in this section is based on network configurations illustrated using the following figure:



Example: Using PCECC for L3VPN and PWE3

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

10. Using PCECC for Traffic Classification Information

When a TE-LSP is set up, the head end needs to know:

o how to use it

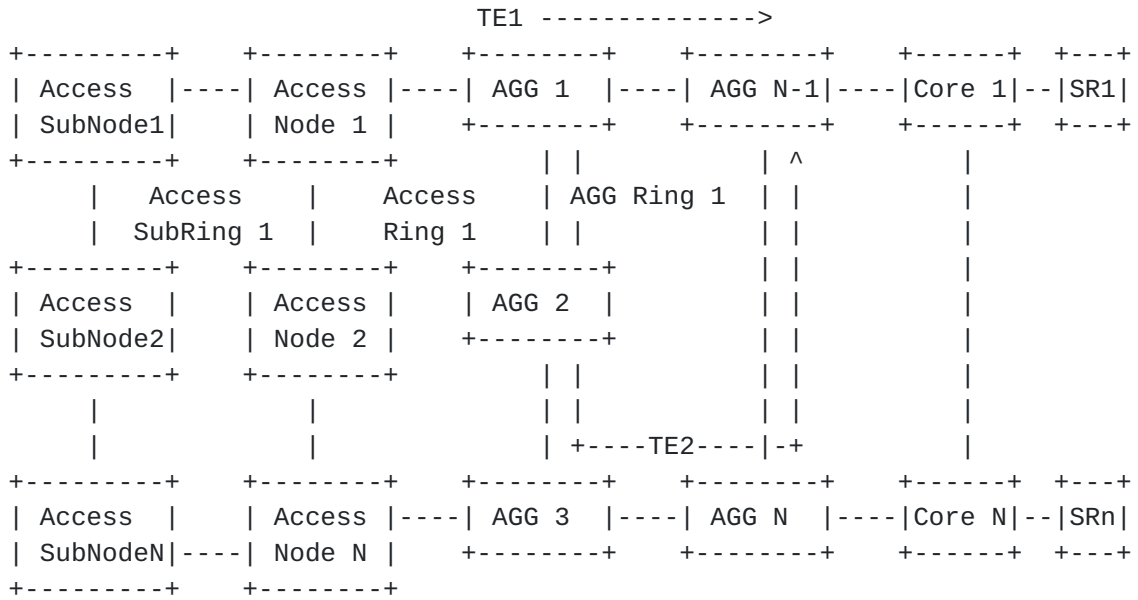
- What traffic to send on the LSP
- Whether it is a virtual link
- Whether to advertise it in the IGP
- What bits of this information to signal to the tail end

PCEP allows an Active PCE to set up or modify LSPs. But we have no way to tell the head end how to use the LSP This is because of history. It used to be the LER that made the request of the PCE, so it knew why it wanted the LSP.

With the PCECC architecture by extending the PCEP protocols, it is easy to carry this information such as how to use the LSP, how to advertise the LSP and other extra signaling information.

11. PCECC Load Balancing (LB) Use Case

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).
Let's consider the following typicall topology.



This MBH architecture uses L2 access rings and subrings. L3 starts at aggregation. For the sake of simplicity here we have only one access subring,access ring and aggregation ring (AGG1...AGGN), connected by Nx10GE interfaces. Aggregation domain runs its own IGP. There are two Egress routers (AGG N-1,AGG N) that are connected to the Core domain via L2 interfaces. Core also have connections to service routers (SRs),

RSVP TEs are 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 either L2VPNs (VPLS) or L3VPNs. Those services use MPLS TE as transport towards egress AGG routers. TE tunnels could be also used as transport towards SRs in case of seamless MPLS-like architecture in the future.

There is a need to solve the following tasks:

- o Perform automatic LB amongst TE tunnels according to current traffic load
- o TE bandwidth (BW) management: Provide guaranteed BW for specific service: HSI,IPTV, etc., provide time-based BW reservation (BoD)
- o Simplify development of TE tunnels (go away from manual provisioning)
- o Provide flexibility for Service Router (SR) placement (anywhere in the network by creation of transport LSPs to them)

Since other tasks are considered in other PCECC use cases above, hereafter we will focus here only on load balancing (LB) task. LB task could be solved by means of PCECC in the following way:

- o After application or network service or operator will ask SDN controller (PCECC) for LSP based LB between AGG X and AGG N/AGG N-1 (egress AGG routers which have connections to core) via North Bound Interface (NBI such as REST API), PCECC SHOULD ask for constrains for that particular calculation (i.e. LSP type: traditional CR-LSP or SR-TE LSP, bandwidth, inclusion or exclusion specific links or nodes, number of paths, shortest path or minimum cost tree, need for disjoint LSP paths etc.).
- o PCECC MUST calculate N P2P LSPs according to given constrains, calculation is based on results of Objective Function (OF), that includes same source and destination routers IDs, same or different bandwidth (BW) , different links (in case of disjoint paths) and other constrains from Step 1.
- o Depending on given LSP type (CR-LSP or SR-TE), PCECC SHOULD create different labels (aka different label spaces, it MAY also require label space negotiation procedure between PCECC and PCCs) for calculated LSPs from egress nodes AGG N-1 and AGG N towards ingress AGG X node.
- o PCECC SHOULD send PCInitiate PCEP message [I-D.crabbe-pce-pce-initiated-lsp] towards ingress AGG X router(PCC) for each of N LSPs and receives PCRpt PCEP message [I-D.ietf-pce-stateful-pce] back from him.
- o If LSP type is CR-LSP, PCECC MUST send PCLabelUpd [I-D.zhao-pce-pcep-extension-for-pce-controller] PCEP message to each node along the path with label information for each of N LSPs. If LSP type is SR-TE, PCECC also MUST send PCLabelUpd PCEP message

to each node along the path with label information (Node-ID and Adjacency-ID segment (label) list) specific to that node. Then PCECC SHOULD send PCUpd PCEP message to the ingress AGG X router with information about new LSP and AGG X(PCC) SHOULD send PCEP PCRpt back with LSP status:Up.

o Now each router along the LSP has corresponding label forwarding state for each of N LSPs.

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

o Since PCECC MUST know as LSDB as TEDB (TE state) he can manage and prevent possible oversubscriptions and limit number of available LB states.

12. 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 performs by means of Master-Slave architecture 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?

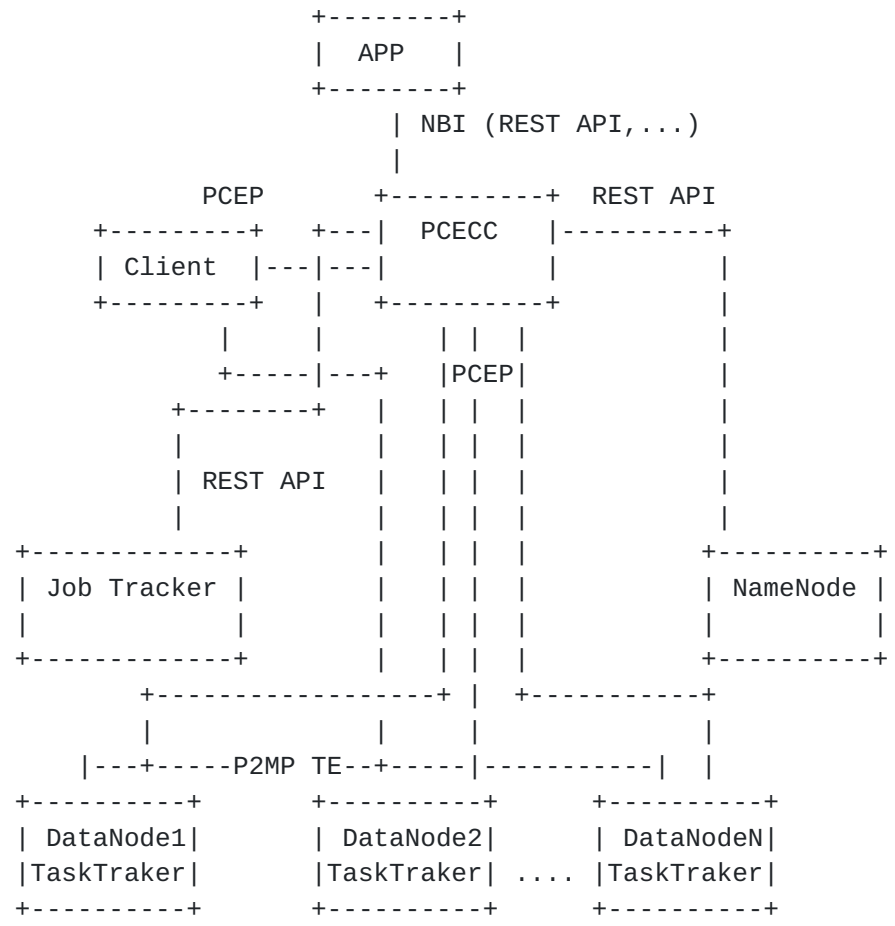
There are number of researches [Multicast Tree Map-Reduce...] which show how usage of multicast tree could improve speed of resource or cluster members discovery inside the cluster as well as increase redundancy in communications between cluster nodes.

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 Master and Slave nodes inside the cluster.
Obviously these P2MP tunnels should be dynamically created and turned down (no manual intervention). Here is there PCECC comes to play. His main task is to create optimal topology of each partucular request for MapReduce computation and also create P2MP tunnels with needed parameters such as badnwidth and delay.

This solution would require to use MPLS label based forwarding inside the cluster.
Usage of label based forwarding inside DC was proposed by Yandex [MPLS in DC...]
Technically it is already possible because mpls on switches is already supported by some vendors, mpls aslo exists on Linux and OVS.

The following framework can make this task:



Communication between Master nodes (JobTracker and NameNode) and PCECC via REST API MAY be either done directly or via cluster manager such as Mesos.

Phase 1: Distributed cluster resources discovery
During this phase Master Nodes SHOULD identify and find available Slave nodes 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 [RFC7334] 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 PCLabelUpd [I-D.zhao-pce-pcep-extension-for-pce-controller] 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 PCUpd message.

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

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

13. The Considerations for PCECC Procedure and PCEP extensions

The PCECC's procedures and PCEP extensions is defined in [I-D.zhao-pce-pcep-extension-for-pce-controller].

14. IANA Considerations

This document does not require any action from IANA.

15. Security Considerations

TBD.

16. Acknowledgments

We would like to thank Robert Tao, Changjiang Yan, Tieying Huang and Adrian Farrel for their useful comments and suggestions.

17. References

17.1. Normative References

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

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

17.2. Informative References

[RFC5441] Vasseur, JP., Ed., Zhang, R., Bitar, N., and JL. Le Roux, "A Backward-Recursive PCE-Based Computation (BRPC) Procedure to Compute Shortest Constrained Inter-Domain Traffic Engineering Label Switched Paths", RFC 5441, DOI 10.17487/RFC5441, April 2009, <http://www.rfc-editor.org/info/rfc5441>.

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

[I-D.filsfils-spring-segment-routing] Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R., Ytti, S., Henderickx, W., Tantsura, J., and E. Crabbe, "Segment Routing Architecture", draft-filsfils-spring-segment-routing-04 (work in progress), July 2014.

[I-D.ietf-pce-stateful-pce] Crabbe, E., Minei, I., Medved, J., and R. Varga, "PCEP Extensions for Stateful PCE", draft-ietf-pce-stateful-pce-14 (work in progress), May 2016.

[I-D.crabbe-pce-pce-initiated-lsp] Crabbe, E., Minei, I., Sivabalan, S., and R. Varga, "PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model", draft-crabbe-pce-pce-initiated-lsp-05 (work in progress), October 2015.

[I-D.ali-pce-remote-initiated-gmpls-lsp] Ali, Z., Sivabalan, S., Filsfils, C., Varga, R., Lopez, V., Dios, O., and X. Zhang, "Path Computation Element Communication Protocol (PCEP) Extensions for remote-initiated GMPLS LSP Setup", draft-ali-pce-remote-initiated-gmpls-lsp-03 (work in progress), February 2014.

[I-D.ietf-isis-segment-routing-extensions] Previdi, S., Filsfils, C., Bashandy, A., Gredler, H., Litkowski, S., Decraene, B., and J. Tantsura, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-06 (work in progress), December 2015.

[I-D.psenak-ospf-segment-routing-extensions]
Psenak, P., Previdi, S., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", draft-psenak-ospf-segment-routing-extensions-05 (work in progress), June 2014.

[I-D.sivabalan-pce-segment-routing]
Sivabalan, S., Medved, J., Filsfils, C., Crabbe, E., Raszuk, R., Lopez, V., and J. Tantsura, "PCEP Extensions for Segment Routing", draft-sivabalan-pce-segment-routing-03 (work in progress), July 2014.

[I-D.li-mpls-global-label-usecases]
Li, Z., Zhao, Q., Yang, T., Raszuk, R., and L. Fang, "Usecases of MPLS Global Label", draft-li-mpls-global-label-usecases-03 (work in progress), October 2015.

[I-D.li-mpls-global-label-framework]
Li, Z., Zhao, Q., Chen, X., Yang, T., and R. Raszuk, "A Framework of MPLS Global Label", draft-li-mpls-global-label-framework-02 (work in progress), July 2014.

[I-D.zhao-pce-pcep-extension-for-pce-controller]
Zhao, Q., Li, Z., Dhody, D., and C. Zhou, "PCEP Procedures and Protocol Extensions for Using PCE as a Central Controller (PCECC) of LSPs", draft-zhao-pce-pcep-extension-for-pce-controller-03 (work in progress), March 2016.

[I-D.ietf-spring-resiliency-use-cases]
Francois, P., Filsfils, C., Decraene, B., and R. Shakir, "Use-cases for Resiliency in SPRING", draft-ietf-spring-resiliency-use-cases-02 (work in progress), December 2015.

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

[Multicast Tree Map-Reduce...]
Lee, Kyungyong., Dr. Boykin, P. Oscar., Dr.Figueiredo, Renato J., "Multicast Tree Map-Reduce: Self-organizing Resource Discovery and Monitoring using Structured P2P Systems"

Authors' Addresses

Quintin Zhao
Huawei Technologies

125 Nagog Technology Park
Acton, MA 01719
US

EMail: quintin.zhao@huawei.com

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

EMail: lizhenbin@huawei.com

Boris Khasanov
Huawei Technologies
Moskovskiy Prospekt 97A
St.Petersburg 196084
Russia

EMail: khasanov.boris@huawei.com

King Ke
Tencent Holdings Ltd.
Shenzhen
China

EMail: kinghe@tencent.com

Luyuan Fang
Microsoft

EMail: lufang@microsoft.com

Chao Zhou
Cisco Systems

EMail: chao.zhou@cisco.com

Boris Zhang
Telus Communications

EMail: Boris.zhang@telus.com

Artem Rachitskiy
Mobile TeleSystems JLLC
Nezavisimosti ave., 95
Minsk 220043
Belarus

EMail: arachitskiy@mts.by

Anton Gulida
Mobile TeleSystems JLLC
Nezavisimosti ave., 95
Minsk 220043
Belarus

EMail: agulida@mts.by