SPRING Working Group
Internet-Draft
Intended status: Informational
Expires: January 1, 2018

A. Farrel
J. Drake
Juniper Networks

June 30, 2017

Interconnection of Segment Routing Domains - Problem Statement and Solution Landscape
draft-farrel-spring-sr-domain-interconnect-00

Abstract

Segment Routing (SR) is now a popular forwarding paradigm for use in MPLS and IPv6 networks. It is typically deployed in discrete domains that may be data centers, access networks, or other networks that are under the control of a single operator and that can easily be upgraded to support this new technology.

Traffic originating in one SR domain often terminates in another SR domain, but must transit a backbone network that provides interconnection between those domains.

This document describes a mechanism for providing connectivity between SR domains to enable end-to-end or domain-to-domain traffic engineering.

The approach described: allows connectivity between SR domains, utilizes traffic engineering mechanisms (RSVP-TE or Segment Routing) across the backbone network, makes heavy use of pre-existing technologies requiring the specifications of very few additional mechanisms.

This document some background and a problem statement, explains the solution mechanism, and provides examples. It does not define any new protocol mechanisms.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Data Centers are a growing market sector. They are being set up by new specialist companies, by enterprises for their own use, by legacy ISPs, and by the new wave of network operators such as Microsoft and Amazon.

The networks inside Data Centers are currently well-planned, but the traffic loads can be unpredictable. There is a need to be able to direct traffic within a Data Center to follow a specific path.

Data Centers are attached to external ("backbone") networks to allow access by users and to facilitate communication among Data Centers. An individual Data Center may be attached to multiple backbone networks, and may have multiple points of attachment to each backbone network. Traffic to or from a Data Center may need to be directed to or from any of these points of attachment.

A variety of networking technologies exist and have been proposed to steer traffic within the Data Center and across the backbone networks. This document proposes an approach that builds on existing technologies to produce mechanisms that provide scalable and flexible interconnection of Data Centers, and that will be easy to operate.

Segment Routing (SR) is a new technology that places forwarding state into each packet as a stack of loose hops as distinct from other pre-existing techniques that require signaling protocols to install state in the network. SR is a popular option for building Data Centers, and is also seeing increasing traction in edge and access networks as well as in backbone networks.

This paper describes mechanisms to provide end-to-end SR connectivity between SR-capable domains across an MPLS backbone network that supports SR and/or MPLS-TE. This is the generalization of the requirement to provide inter-Data Center connectivity.

2. Problem Statement

Consider the network in Figure 1. Without loss of generality, this `figure can be used to represent the architecture and problem space for steering traffic within and between SR edge domains. The figure shows a single destination for all traffic that we will consider. In this figure we distinguish between the PEs that provide access to the backbone networks and the Gateways that provide access to the SR edge

domains: these may, in fact be the same equipment, and the PEs might be located at the domain edges.

In describing the problem space and the solution we use four terms for network nodes as follows:

SR edge domain : A collection of SR-capable nodes in an edge network attached to the backbone network through one or more gateways. Examples include, access networks, Data Center sites, and blessings of unicorns.

Host : A node within an edge domain. May be an end system or a transit node in the edge domain.

Gateway (GW) : Provides access to or from an edge domain. Examples are CEs, ASBRs, and Data Center gateways.

Provider Edge (PE) : Provides access to or from the backbone network.

Autonomous System Border Router (ASBR) : Provides access to one AS in the backbone network from another AS in the backbone network.

These terms can be seen used in Figure 1 where the various sources and destinations are hosts.

```
    -------------------------------------------------------------------
   |                                                                   |
   |                              AS1                                  |
   |  ----    ----                                       ----    ----  |
    -|PE1a|--|PE1b|-------------------------------------|PE2a|--|PE2b|-
      ----    ----                                       ----    ----
      :        :   ------------           ------------      :      :
      :        :  | AS2        |         |        AS3 |     :      :
      :        :  |         ------     ------         |     :      :
      :        :  |        |ASBR2a|...|ASBR3a|        |     :      :
      :        :  |         ------     ------         |     :      :
      :        :  |            |         |            |     :      :
      :        :  |         ------     ------         |     :      :
      :        :  |        |ASBR2b|...|ASBR3b|        |     :      :
      :        :  |         ------     ------         |     :      :
      :        :  |            |         |            |     :      :
      :  ......:  |  ----      |         |      ----  |     :      :
      :  :         -|PE2a|-----           -----|PE3a|-      :      :
      :  :           ----                       ----        :      :
      :  :      ......:                           :.......  :      :
      :  :      :                                        :  :      :
      ----    ----                                       ----    ----
    -|GW1a|--|GW1b|-                                   -|GW2a|--|GW2b|-
   |  ----    ----  |                                 |  ----    ----  |
   |                |                                 |                |
   |                |                                 |                |
   |                |                                 | Source3        |
   |        Source2 |                                 |                |
   |                |                                 |        Source4 |
   | Source1        |                                 |                |
   |                |                                 |   Destination  |
   |                |                                 |                |
   | Dom1           |                                 |           Dom2 |
    ----------------                                   ----------------
```

Figure 1: Reference Architecture for SR Domain Interconnect

Traffic to the destination may be sourced from multiple sources within that domain (we show two such sources: Source3 and Source4). Furthermore, traffic intended for the destination may arrive from outside the domain through any of the points of attachment to the backbone networks (we show GW3a and GW3b). This traffic may need to be steered within the domain to achieve load-balancing across network resources, to avoid degraded or out-of-service resources (including planned service outages), and to achieve different qualities of service. Of course, traffic in a remote source domain may also need

to be steered within that domain. We class this problem as "Intra-Domain Traffic Steering".

Traffic across the backbone networks may need to be steered to conform to common Traffic Engineering paradigms. That is, the path across any network (shown in the figure as an AS) or across any collection of networks may need to be chosen. Furthermore, the points of inter-connection between networks may need to be selected and influence the path chosen for the data. We class this problem as "Inter-Domain Traffic Steering".

The composite end-to-end path comprises steering in the source domain, choice of source domain exit point, steering across the backbone networks, choice of network interconnections, choice of destination domain entry point, and steering in the destination domain. These issues may be inter-dependent (for example, the best traffic steering in the source domain may help select the best exit point from that domain, but the connectivity options across the backbone network may drive the selection of a different exit point). We class this combination of problems as "End-to-End Domain Interconnect Traffic Steering".

It should be noted that the solution to the End-to-End Domain Interconnect Traffic Steering problem depends on a number of factors:

- What technology is deployed in the domains.
- What technology is deployed in the backbone networks.
- How much information are the domains willing to share with each other.
- How much information are the backbone network operators and the domain operators are willing to share.

In some cases, the domains and backbone networks are all owned and operated by the same company (with the backbone network often being a private network). In other cases, the domains are operated by one company, with other companies operating the backbone.

3. Solution Technologies

Within the Data Center, Segment Routing (SR from the SPRING working group in the IETF [RFC7855] and [I-D.ietf-spring-segment-routing]) is becoming a dominant solution. SR introduces traffic steering capabilities into an MPLS network [I-D.ietf-spring-segment-routing-mpls] by utilizing existing data plane capabilities (label pop and packet forwarding - "pop and go")

in combination with additions to existing IGPs [I-D.ietf-ospf-segment-routing-extensions], [I-D.ietf-isis-segment-routing-extensions], BGP (as BGP-LU) [I-D.ietf-mpls-rfc3107bis], or a centralized controller to distribute "per-hop" labels. An MPLS label stack can be imposed on a packet to describe a sequence of links/nodes to be transited by the packet; as each hop is transited, the label that represents it is popped from the stack and the packet is forwarded. Thus, on a packet-by-packet basis, traffic can be steered within the Data Center network.

Note that other Data Center data plane technologies also exist. While this document focuses on connecting domains that use MPLS Segment Routing, the techniques are equally applicable to non-MPLS domains (such as those using IP, VXLAN, and NVGRE). See Section 9 for details.

This document broadens the problem space to consider interconnection of any type of edge domain. These may be Data Center sites, but they may equally be access networks, VPN sites, or any other form of domain that includes packet sources and destinations. We particularly focus on "SR edge domains" being source or destination domains that utilize SR, but the domains could use other technologies as described in Section 9.

Backbone networks are commonly based on MPLS hardware. In these networks, a number of different options exist to establish TE paths. Among these options are static LSPs (perhaps set up by an SDN controller), LSP tunnels established using a signaling protocol (such as RSVP-TE), and inter-domain use of SR (as described above for intra-domain steering). Where traffic steering (without resource reservation) is needed, SR may be adequate. Where Traffic Engineering is needed (i.e., traffic steering with resource reservation) RSVP-TE or centralized SDN control are preferred. However, in a network that is fully managed and controlled through a centralized planning tool, resource reservation can be achieved and SR can be used for full Traffic Engineering. These solutions are already used in support of a number of edge-to-edge services such as L3VPN and L2VPN.

3.1. Characteristics of Solution Technologies

Each of the solution technologies mentioned in the previous section has certain characteristics, and the combined solution needs to recognize and address the characteristics in order to make a workable solution.

o When SR is used for traffic steering, the size of the MPLS label stack used in SR scales linearly with the length of the source

route. This can cause issues with MPLS implementations that only support label stacks of a limited size. For example, some MPLS implementations cannot push enough labels on the stack to represent an entire source route. Other implementations may be unable to do the proper "ECMP hashing" if the label stack is too long; they may be unable to read enough of the packet header to find an entropy label or to find the IP header of the payload. Increasing the packet header size also reduces the size of the payload that can be carried in an MPLS packet. There are techniques that can be used to reduce the size of the label stack. For example, a single label (known as a "binding SID") can be used to represent a sequence of nodes; this label can be replaced with a set of labels when the packet reaches the first node in the sequence. It is also possible to combine SR with conventional RSVP-TE by using a binding SID in the label stack to represent an LSP tunnel set up by RSVP-TE.

o Most of the work on using SR for traffic steering assumes that traffic only needs to be steered within a single administrative domain. If the backbone consists of multiple ASes that are part of a common administrative domain, the use of SR across the backbone may prove to be a challenge, and its use in the backbone may be limited to cases where private networks connect the domains, rather than cases where the domains are connected by third-party network operators or by the public Internet.

o RSVP-TE has been used to provide edge-to-edge tunnels through which flows to/from many endpoints can be routed, and this provides a reduction in state while still offering Traffic Engineering across the backbone network. However, this requires O(n2) connections and as the number of edge domains increases this becomes unsustainable.

o A centralized control system, while capable of producing more optimal results than a distributed control system, may present challenges in large and dynamic networks. It relies on all network state being held centrally, and it is difficult to make central control as robust and self-correcting as distributed control.

This paper introduces an approach that blends the best points of each of these solution technologies to achieve a trade-off where RSVP-TE tunnels in the backbone network are stitched together using SR, and end-to-end SR paths can be created under the control of a central controller with routing devolved to the constituent networks where possible.

4. Decomposing the Problem

It is important to decompose the problem to take account of different regions spanned by the end-to-end path. These regions may use different technologies and may be under different administrative control. The separation of administrative control is particularly important because the operator of one region may be unwilling to share information about their networks, and may be resistant to allowing a third party to exert control over their network resources.

Using the reference model in Figure 1, we can consider how to get a packet from Source1 to the Destination. The following decisions must be made:

- In which domain the Destination lies.

- Which exit point from Dom1 to use.

- Which entry point to Dom2 to use.

- How to reach the exit point of Dom1 from Source1.

- How to reach the entry point to Dom2 from the exit point of Dom1.

- How to reach the Destination from the entry point to Dom2.

As already mentioned, these decisions may be inter-related. This enables us to break down the problem into three steps:

1. Get the packet from Source1 to the exit point of Dom1.

2. Get the packet from exit point of Dom1 to entry point of Dom2.

3. Get the packet from entry point of Dom2 to Destination.

The solution needs to achieve this in a way that allows:

- Adequate discovery of preferred elements in the end-to-end path (such as location of destination, destination domain entry point).

- Full control of the end-to-end path if all of the operators are willing.

- Re-use of existing techniques and technologies.

From a technology point of view we must support several functions and mixtures of those functions:

- If the domain uses MPLS Segment Routing, the labels within the domain may be populated by any means including BGP-LU [I-D.ietf-mpls-rfc3107bis], IGP, and central control. Source routes within the domain may be expressed as label stacks pushed by a controller or computed by a source router, or expressed as a single label and programmed into the domain routers by a controller.

- If the domain uses other (non-MPLS) forwarding, the domain processing is specific to that technology. See Section 9 for details.

- If the domains use Segment Routing, the source and destination domains may or may not be in the same Segment Routing domain, so that the prefix-SIDs may be the same or different in the two domains.

- The backbone network may be a single private network under the control of the owner of the domains and comprising one or more ASes, or may be a network operated by one or more third parties.

- The backbone network may utilize MPLS Traffic Engineering tunnels in conjunction with MPLS Segment Routing and the domain-to-domain source route may be provided by stitching TE LSPs.

- A single controller may be used to handle the source and destination domains as well as the backbone network, or there may be a different controller for the backbone network separate from that that controls the two domains, or there may be separate controllers for each network. The controllers may cooperate and share information to different degrees.

All of these different decompositions of the problem reflect different deployment choices and different commercial and operational practices, each with different functional trade-offs. For example, with separate controllers that do not share information and that only cooperate to a limited extent, it will be possible to achieve end-to-end connectivity with optimal routing at each step (domain or backbone AS), but the end-to-end path that is achieved might not be optimal.

5. Solution Space

5.1. Global Optimization of the Paths

Global optimization of the path from one domain to another requires either that the source controller has a complete view of the end-to-

end topology or some form of cooperation between controllers (such as in BRPC in RFC 5441 [RFC5441]).

BGP-LS [RFC7752] can be used to provide the "source" controller with a view of the topology of the backbone. This requires some of the BGP speakers in each AS to have BGP-LS sessions to the controller. Other means of obtaining this view are of course possible.

5.2. Figuring Out the GWs at a Destination Domain for a Given Prefix

Suppose GW1 and GW2 both advertise a route to prefix X, each setting itself as next hop. One might think that the GWs for X could be inferred from the routes' next hop fields, but typically both routes do not get distributed across the backbone, only the "best" route, as selected by BGP. But the best route according to the BGP selection process might not be the route via the GW that we want to use for traffic engineering purposes.

The obvious solution would be to use the ADD-PATH mechanism [RFC7911] to ensure that all routes to X get advertised. However, even if one does this, the identity of the GWs would get lost as soon as the routes got distributed through an ASBR that sets next hop self. And if there are multiple ASes in the backbone, not only will the next hop change several times, but the ADD-PATH mechanism experiences scaling issues. So this "obvious" solution only works within a single AS.

A better solution can be achieved using the Tunnel Encapsulation [I-D.ietf-idr-tunnel-encaps] attribute as follows:

We define a new tunnel type, "SR tunnel" and when the GWs to a given domain advertise a route to a prefix X within the domain, they each include a Tunnel Encapsulation attribute with multiple remote endpoint sub-TLVs each identifying a specific GW to the domain.

In other words, each route advertised by any GW identifies all of the GWs to the same domain (see Section 9 for a discussion of how GWs discover each other). Therefore, only one of the routes needs to be distributed to other ASes, and it doesn't matter how many times the next hop changes, the Tunnel Encapsulation attribute (and its remote endpoint sub-TLVs) remains unchanged.

Further, when a packet destined for prefix X is sent on a TE path to GW1 we want the packet to arrive at GW1 carrying, at the top of its label stack, GW1's label for prefix X. To achieve this we will place the SID/SRGB in a sub-TLV of the Tunnel Encapsulation attribute. We will define the prefix-SID sub-TLV to be essentially identical in syntax to the prefix-SID attribute (see

[I-D.ietf-idr-bgp-prefix-sid]), but the semantics are somewhat different.

It is also possible to define an "MPLS Label Stack" sub-TLV for the Tunnel Encapsulation attribute, and put this in the "SR tunnel" TLV. This allows the destination GW to specify a label stack that it wants packets destined for prefix X to have. This label stack represents a source route through the destination domain.

5.3. Figuring Out the Backbone Egress ASBRs

We need to figure out the backbone egress ASBRs that are attached to a given GW at the destination domain this out in order to properly engineer the path across the backbone.

The "cleanest" way to figure this out is to have the backbone egress ASBRs distribute the information to the source controller using the EPE extensions of BGP-LS [I-D.ietf-idr-bgpls-segment-routing-epe]. The EPE extensions to BGP-LS allow a BGP speaker to say, "Here is a list of my EBGP neighbors, and here is a (locally significant) adjacency-SID for each one."

It may also be possible to consider utilizing cooperating PCEs or a Hierarchical PCE approach in RFC 6805 [RFC6805]. But it should be observed that this question is dependent on the question in Section 5.2. That is, it is not possible to even start the selection of egress ASBRs until it is known which GWs at the destination domain provide access to a given prefix. Once that question has been answered, any number of PCE approaches can be used to select the right egress ASBR and, more generally, the ASBR path across the backbone.

5.4. Making use of RSVP-TE LSPs Across the Backbone

There are a number of ways to carry traffic across the backbone from one domain to another. RSVP-TE is a popular tunneling mechanism in similar scenarios (e.g., L3VPN) because it allows for reservation of resources as well as traffic steering.

A controller can cause an RSVP-TE LSP to be set up by using PCEP to talk to the LSP headend, using PCEP extensions [I-D.ietf-pce-pce-initiated-lsp]. That draft specifies an "LSP-initiate" message that the controller uses to specify the RSVP-TE LSP endpoints, the ERO, a "symbolic pathname", and optionally other attributes (specified in the PCEP specification, RFC 5440 [RFC5440]) such as bandwidth.

When the headend receives an LSP-initiate message, it sets up the RSVP-TE LSP, assigns it a "PLSP-id", and reports the PLSP-id back to the controller in a PCRpt message [I-D.ietf-pce-stateful-pce]. The PCRpt message also contains the symbolic name that the controller assigned to the LSP, as well as containing some information identifying the LSP-initiate message from the controller, and details of exactly how the LSP was set up (RRO, bandwidth, etc.).

The headend can add to the PCRpt message a TE-PATH-BINDING TLV [I-D.sivabalan-pce-binding-label-sid]. This allows the headend to assign a "binding SID" to the LSP, and to report to the controller that a particular binding SID corresponds to a particular LSP. The binding SID is locally scoped to the headend.

The controller can make this label be part of the label stack that it tells the source (or the GW at the source domain) to put on the data packets being sent to prefix X. When the headend receives a packet with this label at the top of the stack it will send the packet onward on the LSP.

5.5. Data Plane

Consolidating all of the above, consider what happens when we want to move a data packet from Source to Destination in Figure 1via the following source route:

```
Source1---GW1b---PE2a---ASBR2a---ASBR3a---PE3a---GW2a---Destination
```

Further, assume that there is an RSVP-TE LSP from PE2a to ASBR2a that we want to use, as well as an RSVP-TE LSP from ASBR3a to PE3a that we want to use.

Let's suppose that the Source pushes a label stack following instructions from the controller (for example, using BGP-LU [I-D.ietf-mpls-rfc3107bis]). We won't worry for now about source routing through the domains themselves: that is, in practice there may be additional labels in the stack to cover the source route from the Source to GW1b and from GW2a to the Destination, but we will focus only on the labels necessary to leave the source domain, traverse the backbone, and enter the egress domain. So we only care what the stack looks like when the packet gets to GW1b.

When the packet gets to GW1b, the stack should have six labels:

Top Label:

Peer-SID or adjacency-SID identifying link or links to PE2a. These SIDs are distributed from GW1b to the controller via the EPE

extensions of BGP-LS. (This label will get popped by GW1b, which will then send the packet to PE2a.)

Second Label:

Binding SID advertised by PE2a to the controller for the RSVP-TE LSP to ASBR2a. This binding SID is advertised via the PCEP extensions discussed above. (This label will get swapped by PE2a for the label that the LSP's next hop has assigned to the LSP.)

Third Label:

Peer-SID or adjacency-SID identifying link or links to ASBR3a, as advertised to the controller by ASBR2a using the BGP-LS EPE extensions. (This label gets popped by ASBR2a, which then sends the packet to ASBR3a.)

Fourth Label:

Binding SID advertised by ASBR3a for the RSVP-TE LSP to PE3a. This binding SID is advertised via the PCEP extensions discussed above. ASBR3a treats this label just like PE2a treated the second label above.

Fifth label:

Peer-SID or adjacency-SID identifying link or links to GW2a, as advertised to the controller by ASBR3a using the BGP-LS EPE extensions. ASBR3a pops this label and sends the packet to GW2a.

Sixth Label:

Prefix-SID or other label identifying the Destination advertised in a Tunnel Encapsulation attribute by GW2a. (This can be omitted if GW2a is happy to accept IP packets, or prefers a VXLAN tunnel for example. That would be indicated through the Tunnel Encapsulation attribute of course.)

Note that the size of the label stack is proportional to the number of RSVP-TE LSPs that get stitched together by SR.

See Section 7 for some detailed examples that show the concrete use of labels in a sample topology.

In the above example, all labels except the sixth are locally significant labels: peer-SIDs, binding SIDs, or adjacency-SIDs. Only the sixth label, a prefix-SID, has a domain-wide unique value. To impose that label, the source needs to know the SRGB of GW2a. If all

nodes have the same SRGB, this is not a problem. Otherwise, there are a number of different ways GW3a can advertise its SRGB. This can be done via the segment routing extensions of BGP-LS, or it can be done using the prefix-SID attribute or BGP-LU [I-D.ietf-mpls-rfc3107bis], or it can be done using the BGP Tunnel Encapsulation attribute. The exact technique to be used will depend on the details of the deployment scenario.

The reason the above example is primarily based on locally significant labels is that it creates a "strict source route", and it presupposes the EPE extensions of BGP-LS. In some scenarios, the EPE extension to BGP-LS might not be available (or BGP-LS might not be available at all). In other scenarios, it may be desirable to steer a packet through a "loose source route". In such scenarios, the label stack imposed by the source will be based upon a sequence of domain-wide unique "node-SIDs", each representing one of the hops of source route. Each label has to be computed by adding the corresponding node-SID to the SRGB of the node that will act upon the label. One way to learn the node-SIDs and SRGBs is to use the segment routing extensions of BGP-LS. Another way is to use BGP-LU as follows. Each node that may be part of a source route would originate a BGP-LU route with one of its own loopback addresses as the prefix. The BGP prefix-SID attribute would be attached to this route. The prefix-SID attribute would contain a SID, which is the domain-wide unique SID corresponding to the node's loopback address. The attribute would also contain the node's SRGB.

While this technique is useful when BGP-LS is not available, it presupposes that the source controller has some other means of discovering the topology. In this document, we focus primarily on the scenario where BGP-LS, rather than BGP-LU, is used.

5.6. Centralized and Distributed Controllers

A controller or set of controllers are needed to collate topology and TE information from the constituent networks, to apply policies and service requirements to compute paths across those networks, to select an end-to-end path, and to program key nodes in the network to take the right forwarding actions (pushing label stacks, stitching LSPs, forwarding traffic).

o It is commonly understood that a fully optimal end-to-end path can only be computed with full knowledge of the end-to-end topology and available Traffic Engineering resources. Thus, one option is for all information about the domain networks and backbone network to be collected by a central controller that makes all path computations and is responsible for issuing the necessary programming commands. Such a model works best when there is no

commercial or administrative impediment (for example, where the domains and the backbone network are owned and operated by the same organization). There may, however, be some scaling concerns if the component networks are large.

In this mode of operation, each network may use BGP-LS to export Traffic Engineering and topology information to the central controller, and the controller may use PCEP to program the network behavior.

- A similar centralized control mechanism can be used with a scalability improvement that risks a reduction in optimality. In this case, the domain networks can export to the controller just the feasibility of connectivity between data source/sink and gateway, perhaps enhancing this with some information about the Traffic Engineering metrics of the path.

  This approach allows the central controller to understand the end-to-end path that it is selecting, but not to control it fully. The source route from data source to domain egress gateway is left to the source host or a controller in the source domain, while the source route from domain ingress gateway to destination is left as a decision for the domain ingress gateway or to a controller in the destination domain.

  This mode of operation still leaves overall control with a centralized server and that may not be considered suitable when there is separate commercial or administrative control of the networks.

- When there is separate commercial or administrative control of the networks the domain operator will not want the backbone operator to have control of the source routes within the domain and may be reluctant to disclose any information about the topology or resource availability within the domains. Conversely, the backbone operator may be very unwilling to allow the domain operator (a customer) any control over or knowledge about the backbone network.

  This "problem" has already been solved for Traffic Engineering in MPLS networks that span multiple administrative domains and leads to multiple potential solutions:

  * Per-domain path computation in RFC 5152 [RFC5152] can be seen as "best effort optimization". In this mode the controller for each domain is responsible for finding the best path to the next domain, but has no way of knowing which is the best exit

point from the local domain. The resulting path may end up significantly sub-optimal or even blocked.

* Backward recursive path computation (BRPC) in RFC 5441 [RFC5441] is a mechanism that allows controllers to cooperate across a small set of domains (such as ASes) to build a tree of possible paths and so allow the controller for the ingress domain to select the optimal path. The details of the paths within each domain that might reveal confidential information can be hidden using Path Keys in RFC 5520 [RFC5520] BRPC produces optimal paths but scales poorly with an increase in domains and with an increase in connectivity between domains. It can also lead to slow computation times.

* Hierarchical PCE (H-PCE) in RFC 6805 [RFC6805] is a two-level cooperation process between PCEs. The child PCEs remain responsible for computing paths across their domains, and they coordinate with a parent PCE that stitches these paths together to form the end-to-end path. This approach has many similarities with BRPC but can scale better through the maintenance of "domain topology" that shows how the domains are interconnected, and through the ability to pipe-line computation requests to all of the child domains. It has the drawback that some party has to own and operate the parent PCE.

* An alternative approach is documented by the TEAS working group [RFC7926]. In this model each network advertises to controllers for adjacent networks (using BGP-LS) selected information about potential connectivity across the network. It does not have to show full topology and can make its own decisions about which paths it considers optimal for use by its different neighbors and customers. This approach is suitable for the End-to-End Domain Interconnect Traffic Steering problem where the backbone is under different control from the domains because it allows the overlay nature of the use of the backbone network to be treated as a peer network relationship by the controllers of the domains - the domains can be operated using a single controller or a separate controller for each domain.

It is also possible to operate domain interconnection when some or all domains do not have a controller. Segment Routing is capable of routing a packet toward the next hop based on the top label on the stack, and that label does not need to indicate an immediately adjacent node or link. In these cases, the packet may be forwarded untouched, or the forwarding router may impose a locally-determined additional set of labels that define the path to the next hop.

PCE can be used to instruct the source host or a transit node on what label stacks to add to packets. That is, a node that needs to impose labels (either to start routing the packet from the source host, or to advance the packet from a transit router toward the destination) can determine the label stack to use based on local function or can have that stack supplied by a PCE. The PCE Protocol (PCEP) has been extended to allow the PCE to supply a label stack for reaching a specific destination either in response to a request or in an unsolicited manner [I-D.ietf-pce-segment-routing].

6. BGP-LS Considerations

This section gives an overview of the use of BGP-LS to export an abstraction (or summary) of the connectivity across the backbone network by means of two figures that show different views of a sample network.

Figure 2 shows a more complex reference architecture.

Figure 3 represents the minimum set of nodes and links that need to be advertised in BGP-LS with SR in order to perform Domain Interconnect with traffic engineering across the backbone network: the PEs, ASBRs, and gateways (GWs), and the links between them. In particular, EPE [I-D.ietf-idr-bgpls-segment-routing-epe] and TE information with associated segment IDs is advertised in BGP-LS with SR.

Links that are advertised may be physical links, links realized by LSP tunnels, or abstract links. It is assumed that intra-AS links are either real links, RSVP-TE LSPs with allocated bandwidth, or SR TE policies as described in [I-D.previdi-idr-segment-routing-te-policy]. Additional nodes internal to an AS and their links to PEs, ASBRs, and/or GWs may also be advertised (for example to avoid full mesh problems).

```
    -------------------------------------------------------------------
   |                                                                   |
   |                              AS1                                  |
   |  ----    ----                                       ----    ----  |
    -|PE1a|--|PE1b|-------------------------------------|PE2a|--|PE2b|-
      ----    ----                                       ----    ----
      :        :   ------------           ------------     :     : :
      :        :  | AS2        |         |        AS3 |    :     : :
      :        :  |         ------.....------         |    :     : :
      :        :  |        |ASBR2a|   |ASBR3a|        |    :     : :
      :        :  |         ------  ..:------         |    :     : :
      :        :  |            |    :    |            |    :     : :
      :        :  |         ------..:  ------         |    :     : :
      :        :  |        |ASBR2b|...|ASBR3b|        |    :     : :
      :        :  |         ------     ------         |    :     : :
      :        :  |            |         |            |    :     : :
      :        :  |            |       ------         |    :     : :
      :        :  |            |    ..|ASBR3c|        |    :     : :
      :        :  |            |    :  ------         |    : ....: :
      :  ......:  |  ----      |    :    |      ----  |    : :     :
      :  :         -|PE2a|-----     :     -----|PE3b|-     : :     :
      :  :           ----           :           ----       : :     :
      :  :     .......:             :             :....... : :     :
      :  :     :                   ------                : : :     :
      :  :     :              ----|ASBR4b|----           : : :     :
      :  :     :             |     ------     |          : : :     :
      :  :     :           ----               |          : : :     :
      :  :     : .........|PE4b|          AS4 |          : : :     :
      :  :     : :         ----               |          : : :     :
      :  :     : :           |      ----      |          : : :     :
      :  :     : :            -----|PE4a|-----           : : :     :
      :  :     : :                  ----                 : : :     :
      :  :     : :                ..:  :..               : : :     :
      :  :     : :                :      :               : : :     :
      ----    ----              ----    ----             ----:   ----
    -|GW1a|--|GW1b|-          -|GW2a|--|GW2b|-         -|GW3a|--|GW3b|-
   |  ----    ----  |        |  ----    ----  |       |  ----    ----  |
   |                |        |                |       |                |
   |                |        |                |       |                |
   | Host1a  Host1b |        | Host2a  Host2b |       | Host3a  Host3b |
   |                |        |                |       |                |
   |                |        |                |       |                |
   | Dom1           |        | Dom2           |       |           Dom3 |
    ----------------          ----------------         ----------------
```

Figure 2: Network View of Example Configuration

```
       .............................................................
       :                                                           :
      ----    ----                                       ----    ----
     |PE1a|  |PE1b|.....................................|PE2a|  |PE2b|
      ----    ----                                       ----    ----
      :        :                                           :     : :
      :        :                                           :     : :
      :        :            ------.....------              :     : :
      :        :     ......|ASBR2a|   |ASBR3a|......       :     : :
      :        :     :      ------  ..:------      :       :     : :
      :        :     :              :              :       :     : :
      :        :     :      ------..:  ------      :       :     : :
      :        :     :  ...|ASBR2b|...|ASBR3b|     :       :     : :
      :        :     :  :   ------     ------      :       :     : :
      :        :     :  :                 :        :       :     : :
      :        :     :  :              ------      :       :     : :
      :        :     :  :           ..|ASBR3c|...  :       :     : :
      :        :     :  :           :  ------   :  :       : ....: :
      :  ......:     ----           :           ----       : :     :
      :  :          |PE2a|          :          |PE3b|      : :     :
      :  :           ----           :           ----       : :     :
      :  :     .......:             :             :....... : :     :
      :  :     :                   ------                : : :     :
      :  :     :                  |ASBR4b|               : : :     :
      :  :     :                   ------                : : :     :
      :  :     :           ----        :                 : : :     :
      :  :     : .........|PE4b|.....  :                 : : :     :
      :  :     : :         ----     :  :                 : : :     :
      :  :     : :                  ----                 : : :     :
      :  :     : :                 |PE4a|                : : :     :
      :  :     : :                  ----                 : : :     :
      :  :     : :                ..:  :..               : : :     :
      :  :     : :                :      :               : : :     :
      ----    ----              ----    ----             ----:   ----
    -|GW1a|--|GW1b|-          -|GW2a|--|GW2b|-         -|GW3a|--|GW3b|-
   |  ----    ----  |        |  ----    ----  |       |  ----    ----  |
   |                |        |                |       |                |
   |                |        |                |       |                |
   | Host1a  Host1b |        | Host2a  Host2b |       | Host3a  Host3b |
   |                |        |                |       |                |
   |                |        |                |       |                |
   | Dom1           |        | Dom2           |       |           Dom3 |
    ----------------          ----------------         ----------------
```

Figure 3: Topology View of Example Configuration

A node (a PCE, router, or host) that is computing a full or partial path correlates the topology information disseminated in BGP-LS with SR with the information advertised with the Tunnel Encapsulation attributes to compute that path and obtain the SIDs for the elements on that path. In order to allow a source host to compute exit points from its domain, some subset of the above information needs to be disseminated within that domain.

What is advertised external to a given AS is controlled by policy at the ASes' PEs, ASBRs, and GWs. Central control of what each node should advertise, based upon analysis of the network as a whole, is an important additional function. This and the amount of policy involved may make the use of a Route Reflector an attractive option.

The configuration of which links to other nodes and the characteristics of those links a given node advertises in BGP-LS with SR is done locally at each node and pairwise coordination between link end-points is required to ensure consistency.

Path Weighted ECMP (PWECMP) is assumed to be used by a GW for a given source domain to send all flows to a given destination domain using all paths in the backbone network to that destination domain in proportion to the minimum bandwidth on each path. PWECMP is also assumed to be used by hosts within a source domain to send flows to that domain's GWs.

7. Worked Examples

Figure 4 shows a view of the links, paths, and labels that can be assigned to part of the sample network shown in Figure 2 and Figure 3. The double-dash lines (===) indicate LSP tunnels across backbone ASes and dotted lines (...) are physical links.

At each node, a label may be assigned to each outgoing link. This is shown in Figure 4. For example, at GW1a the label L201 is assigned to the link connecting GW1a to PE1a. At PE1c, the label L302 is assigned to the link connecting PE1c to GW3b. Labels ("binding SIDs") may also be assigned to RSVP-TE LSPs. For example, at PE1a, label L202 is assigned to the RSVP-TE LSP leading from PE1a to PE1c.

At the destination domain, labels L302 and L305 are "node-SIDs"; they represent GW3b and Host3b respectively, rather than representing particular links.

When a node processes a packet, the label at the top of the label stack indicates the link (or RSVP-TE LSP) on which that node is to transmit the packet. The node pops that label off the label stack before transmitting the packet on the link. However, if the top

label is a node-SID, the node processing the packet is expected to transmit the packet on whatever link it regards as the shortest path to the node represented by the label.

```
      ----        L202                                             ----
     |    |=======================================================|    |
     |PE1a|                                                       |PE1c|
     |    |=======================================================|    |
      ----        L203                                             ----
      :                                                             : :
      :     ----     L205                                     ----  : :
      :    |PE1b|============================================|PE1d| : :
      :     ----                                              ----  : :
      :      :                                                  :   : :
      :      :                                                  :   : :
      :      :    ----  L207  ------  L209  ------          L303:   : :
      :L201  :   |    |======|ASBR2a|......|      |             :   : :
      :      :   |    |       ------       |      | L210  ----  :   : :
      :      :   |PE2a|                    |ASBR3a|======|PE3b| :   : :
      :      :   |    | L208  ------  L211 |      |       ----  :   : :
      :      :   |    |======|ASBR2b|......|      |       :     :   : :
      :  L204:    ----       ------         ------     ...:     :   : :
      :      :      :                                  :        :   : :
      :  ....:      :                                  : .......:   : :
      :  :          :                                  : :          : :
      :  :          :L206                          L301: : .........: :
      :  :          :                                  : : : L304     :
      :  :      ....:                                  : : :      ....:
      :  :      :                                      : : :      : L302
      ----    ----                                     -----    ----
    -|GW1a|--|GW1b|-                                 -|GW3a |--|GW3b|-
   |  ----    ----  |                               |  -----    ----  |
   |    :      :    |                               |     :      :    |
   |L103:      :L102|                               | L303:      :L304|
   |    :      :    |                               |     :      :    |
   |   N1      N2   |                               |    N3      N4   |
   |    :..  ..:    |                               |     :  ....:    |
   | L101 :  :      |                               |     :  :        |
   |     Host1a     |                               |   Host3b (L305) |
   |                |                               |                 |
   | Dom1           |                               |            Dom3 |
    ----------------                                 -----------------
```

Figure 4: Tunnels and Labels in Example Configuration

Let's consider several different possible ways to direct a packet from Host1a in Dom1 to Host3b in Dom3.

a. Full source route imposed at source

In this case it is assumed that the entity responsible for determining an end-to-end path has access to the topologies of both domains and of the backbone network. This might happen if all of the networks are owned by the same operator in which case the information can be shared into a single database for use by an offline tool, or the information can be distributed using routing protocols such that the source host can see enough to select the path. Alternatively, the end-to-end path could be produced through cooperation between computation entities each responsible for different domains along the path.

If the path is computed externally it is pushed to the source host. Otherwise, it is computed by the source host itself.

Suppose it is desired for a packet from Host1a to travel to Host3b via the following source route:

```
Host1a->N1->GW1a->PE1a->(RSVP-TE LSP)->PE1c->GW3b->N4->Host3b
```

Host1a would impose the following label stack would be imposed (with the first label representing the top of stack), and then send the packet to N1:

```
L103, L201, L202, L302, L304, L305
```

N1 sees L103 at the top of the stack, so it pops the stack and forwards the packet to GW1a. GW1a sees L201 at the top of the stack, so it pops the stack and forwards the packet to PE1a. PE1a sees L202 at the top of the stack, so it pops the stack and forwards the packet over the RSVP-TE LSP to PE1c. As the packet travels over this LSP, its top label will be an RSVP-TE signaled label representing the LSP. That is, PE1a imposes an additional label stack entry for the tunnel LSP.

At the end of the LSP tunnel, the MPLS tunnel label will be popped, and PE1c will see L302 at the top of the stack. PE1c pops the stack and forwards the packet to GW3b. GW3b will see L304 at the top of the stack, so it pops the stack and forwards the packet to N4. Finally, N4 sees L305 at the top of the stack, so it pops the stack and forwards the packet to Host 3b. No remote visibility into Dom3.

b. It is possible that the source domain does not have visibility into the destination domain.

This occurs if the destination domain does not export its topology, but even in this case, it will export reachability information so that the source host or the path computation entity will know:

- The GWs through which the destination can be reached.

- The SID to use for the destination prefix.

Suppose we want a packet to follow the source route:

Host1a->N1->GW1a->PE1a->(RSVP-TE LSP)->PE1c->GW3b->...->Host3b

(The ellipsis indicates a part of the path that is not explicitly specified.) Thus, the label stack imposed at the source host would be:

L103, L201, L202, L302, L305

Processing is as per case a., but when the packet reaches the GW of the destination domain, it can either simply forward the packet along the shortest path to Host3b, or it can insert additional labels to direct the path to the destination.

c. Dom1 only has reachability information

The source domain (or the path computation entity) may be further restricted in its view of the network. It is possible that it knows the location of the destination in the destination domain, and knows the GWs to the destination domain that provide reachability to the destination, but that it has no view of the backbone network. This leads to the packet being forwarded in a manner similar to 'per-domain path computation' described in Section 5.6.

At the source host a simple label stack is imposed navigating the domain and indicating the destination GW and the destination host.

L101, L103, L302, L305

As the packet leaves the source domain, the source GW determines the PE to use to enter the backbone using nothing more than the BGP preferred route to the destination GW.

When the packet reaches the first PE it has a label stack just identifying the destination GW and host (L302, L305). The PE uses information it has about the backbone network topology and available LSPs to select an LSP tunnel, impose the tunnel label, and forward the packet.

When the packet reaches the end of the LSP tunnel, it is processed as described in case b.

d. Stitched LSPs across the backbone

A variant of all these cases arises when the packet is sent using a path that spans multiple ASes. For example, one that crosses AS2 and AS3 as shown in Figure 2.

In this case, basing the example on case a., the source host would impose the label stack:

```
L102, L206, L207, L209, L210, L301, L303, L305
```

and would then send the packet to N2.

When the packet reaches PE2a as previously described and the top label (L207) selects an LSP tunnel that leads to ASBR2a. At the end of that LSP tunnel the next label (L209) routes the packet from ASBR2a to the ASBR3a, where the next label (L210) identifies the next LSP tunnel to use. Thus, SR has been used to stitch together LSPs to make a longer path segment. As the packet emerges from the final LSP tunnel, forwarding continues as previously described.

8. Label Stack Depth Considerations

As described in Section 3.1, one of the issues with a Segment Routing approach is that the label stack can get large, for example when the source route becomes long. A mechanism to mitigate this problem is needed if the solution is to be fully applicable in all environments.

An Internet-Draft called "Segment Routing Traffic Engineering Policy using BGP" [I-D.previdi-idr-segment-routing-te-policy] introduces the concept of hierarchical source routes as a way to compress source route headers. It functions by having the egress node for a set of source routes advertise those source routes along with an explicit request that each node that is an ingress node for one or more of those source routes should advertise a binding SID for the set of source routes for which it is the ingress. (It should be noted that the set of source routes can either be advertised by the egress node as described here, or could be advertised by a controller on behalf

of the egress node.) Such an ingress node advertises its set of source routes and a binding SID as an adjacency in BGP-LS as described in Section 6. These source routes represent the weighted ECMP paths between the ingress node and the egress node. (Note also that the binding SID may be supplied by the node that advertises the source routes - the egress or the controller - or may be chosen by ingress node.)

A remote node that wishes to reach the egress node would then construct a source route consisting of the segment IDs necessary to reach one of the ingress nodes for the path it wishes to use along with the binding SID that the ingress node advertised to identify the set of paths. When the selected ingress node receives a packet with a binding SID it has advertised, it replaces the binding SID with the labels for one of its source routes to the egress node (it will choose one of the source routes in the set according to its own weighting algorithms and policy).

8.1. Worked Example

Consider the topology in Figure 4. Suppose that it is desired to construct full segment routed paths from ingress to egress, but that the resulting label stack (segment route) is too large. In this case the gateways to Dom3 (GW3a and GW3b) can advertise all of the source routes from the gateways to Dom1 (GW1a and GW1b). The gateways to Dom1 then assign binding SIDs to those source routes and advertise those SIDs into BGP-LS.

Thus, GW3b would advertise the two source routes (L201, L202, L302 and L201, L203, L302), and GW1a would advertise into BGP-LS its adjacency to GW3b along with a binding SID. Should Host1a wish to send a packet via GW1a and GW3b, it can include L103 and this binding SID in the source route. GW1a is free to choose which source route to use between itself and GW3b using its weighted ECMP algorithm.

Similarly, GW3a would advertise the following set of source routes:

- L201, L202, L304
- L201, L203, L304
- L204, L205, L303
- L206, L207, L209, L210, L301
- L206, L208, L211, L210, L301

GW1a would advertise a binding SID for the first three, and GW1b would advertise a binding SID for the other two.

9. Gateway Considerations

As described in Section 5, we define a new tunnel type, "SR tunnel", and when the GWs to a given domain advertise a route to a prefix X within the domain, they will each include a Tunnel Encapsulation attribute with multiple tunnel instances each of type "SR tunnel", one for each GW and each containing a Remote Endpoint sub-TLV with that GW's address.

In other words, each route advertised by any GW identifies all of the GWs to the same domain.

Therefore, even if only one of the routes is distributed to other ASes, it will not matter how many times the next hop changes, as the Tunnel Encapsulation attribute (and its remote endpoint sub-TLVs) will remain unchanged.

9.1. Domain Gateway Auto-Discovery

To allow a given domain's GWs to auto-discover each other and to coordinate their operations, the following procedures are implemented [I-D.drake-bess-datacenter-gateway]:

- Each GW is configured with an identifier for the domain that is common across all GWs to the domain (i.e., all GWs to all domains that are connected) and unique across all domains that are connected.

- A route target [RFC4360] is attached to each GW's auto-discovery route and has its value set to the domain identifier.

- Each GW constructs an import filtering rule to import any route that carries a route target with the same domain identifier that the GW itself uses. This means that only these GWs will import those routes and that all GWs to the same domain will import each other's routes and will learn (auto-discover) the current set of active GWs for the domain.

- The auto-discovery route each GW advertises consists of the following:

  * An IPv4 or IPv6 NLRI containing one of the GW's loopback addresses (that is, with AFI/SAFI that is one of 1/1, 2/1, 1/4, 2/4).

- A Tunnel Encapsulation attribute containing the GW's encapsulation information, which at a minimum consists of an SR tunnel TLV (type to be allocated by IANA) with a Remote Endpoint sub-TLV [I-D.ietf-idr-tunnel-encaps].

To avoid the side effect of applying the Tunnel Encapsulation attribute to any packet that is addressed to the GW, the GW should use a different loopback address.

Each GW will include a Tunnel Encapsulation attribute for each GW that is active for the domain (including itself), and will include these in every route advertised externally to the domain by each GW. As the current set of active GWs changes (due to the addition of a new GW or the failure/removal of an existing GW) each externally advertised route will be re-advertised with the set of SR tunnel instances reflecting the current set of active GWs.

9.2. Relationship to BGP Link State and Egress Peer Engineering

When a remote GW receives a route to a prefix X it can use the SR tunnel instances within the contained Tunnel Encapsulation attribute to identify the GWs through which X can be reached. It uses this information to compute SR TE paths across the backbone network looking at the information advertised to it in SR BGP Link State (BGP-LS) [I-D.gredler-idr-bgp-ls-segment-routing-ext] and correlated using the domain identity. SR Egress Peer Engineering (EPE) [I-D.ietf-idr-bgpls-segment-routing-epe] can be used to supplement the information advertised in the BGP-LS.

9.3. Advertising a Domain Route Externally

When a packet destined for prefix X is sent on an SR TE path to a GW for the domain containing X, it needs to carry the receiving GW's label for X such that this label rises to the top of the stack before the GW complete its processing of the packet. To achieve this we place a prefix-SID sub-TLV for X in each SR tunnel instance in the Tunnel Encapsulation attribute in the externally advertised route for X.

Alternatively, if the GWs for a given domain are configured to allow remote GWs to perform SR TE through that domain for a prefix X, then each GW computes an SR TE path through that domain to X from each of the current active GWs and places each in an MPLS label stack sub-TLV [I-D.ietf-idr-tunnel-encaps] in the SR tunnel instance for that GW.

9.4. Encapsulations

If the GWs for a given domain are configured to allow remote GWs send them a packet in that domain's native encapsulation, then each GW will also include multiple instances of a tunnel TLV for that native encapsulation, one for each GW and each containing a remote endpoint sub-TLV with that GW's address, in externally advertised routes. A remote GW may then encapsulate a packet according to the rules defined via the sub-TLVs included in each of the tunnel TLV instances.

10. Security Considerations

TBD

11. Management Considerations

TBD

12. IANA Considerations

This document makes no requests for IANA action.

13. Acknowledgements

TBD

14. Informative References

[I-D.drake-bess-datacenter-gateway]
Drake, J., Farrel, A., Rosen, E., Patel, K., and L. Jalil, "Gateway Auto-Discovery and Route Advertisement for Segment Routing Enabled Data Center Interconnection", draft-drake-bess-datacenter-gateway-03 (work in progress), April 2017.

[I-D.gredler-idr-bgp-ls-segment-routing-ext]
Previdi, S., Psenak, P., Filsfils, C., Gredler, H., Chen, M., and j. jefftant@gmail.com, "BGP Link-State extensions for Segment Routing", draft-gredler-idr-bgp-ls-segment-routing-ext-04 (work in progress), October 2016.

[I-D.ietf-idr-bgp-prefix-sid]
Previdi, S., Filsfils, C., Lindem, A., Sreekantiah, A., and H. Gredler, "Segment Routing Prefix SID extensions for BGP", draft-ietf-idr-bgp-prefix-sid-06 (work in progress), June 2017.

[I-D.ietf-idr-bgpls-segment-routing-epe]
Previdi, S., Filsfils, C., Patel, K., Ray, S., and J. Dong, "BGP-LS extensions for Segment Routing BGP Egress Peer Engineering", draft-ietf-idr-bgpls-segment-routing-epe-13 (work in progress), June 2017.

[I-D.ietf-idr-tunnel-encaps]
Rosen, E., Patel, K., and G. Velde, "The BGP Tunnel Encapsulation Attribute", draft-ietf-idr-tunnel-encaps-06 (work in progress), June 2017.

[I-D.ietf-isis-segment-routing-extensions]
Previdi, S., Filsfils, C., Bashandy, A., Gredler, H., Litkowski, S., Decraene, B., and j. jefftant@gmail.com, "IS-IS Extensions for Segment Routing", draft-ietf-isis-segment-routing-extensions-13 (work in progress), June 2017.

[I-D.ietf-mpls-rfc3107bis]
Rosen, E., "Using BGP to Bind MPLS Labels to Address Prefixes", draft-ietf-mpls-rfc3107bis-02 (work in progress), May 2017.

[I-D.ietf-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-ietf-ospf-segment-routing-extensions-17 (work in progress), June 2017.

[I-D.ietf-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-ietf-pce-pce-initiated-lsp-10 (work in progress), June 2017.

[I-D.ietf-pce-segment-routing]
Sivabalan, S., Filsfils, C., Tantsura, J., Henderickx, W., and J. Hardwick, "PCEP Extensions for Segment Routing", draft-ietf-pce-segment-routing-09 (work in progress), April 2017.

[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-21 (work in progress), June 2017.

[I-D.ietf-spring-segment-routing]
Filsfils, C., Previdi, S., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", draft-ietf-spring-segment-routing-12 (work in progress), June 2017.

[I-D.ietf-spring-segment-routing-mpls]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with MPLS data plane", draft-ietf-spring-segment-routing-mpls-10 (work in progress), June 2017.

[I-D.previdi-idr-segment-routing-te-policy]
Previdi, S., Filsfils, C., Mattes, P., Rosen, E., and S. Lin, "Advertising Segment Routing Policies in BGP", draft-previdi-idr-segment-routing-te-policy-07 (work in progress), June 2017.

[I-D.sivabalan-pce-binding-label-sid]
Sivabalan, S., Filsfils, C., Previdi, S., Tantsura, J., Hardwick, J., and M. Nanduri, "Carrying Binding Label/Segment-ID in PCE-based Networks.", draft-sivabalan-pce-binding-label-sid-02 (work in progress), October 2016.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <http://www.rfc-editor.org/info/rfc4360>.

[RFC5152] Vasseur, JP., Ed., Ayyangar, A., Ed., and R. Zhang, "A Per-Domain Path Computation Method for Establishing Inter-Domain Traffic Engineering (TE) Label Switched Paths (LSPs)", RFC 5152, DOI 10.17487/RFC5152, February 2008, <http://www.rfc-editor.org/info/rfc5152>.

[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>.

[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>.

[RFC5520] Bradford, R., Ed., Vasseur, JP., and A. Farrel, "Preserving Topology Confidentiality in Inter-Domain Path Computation Using a Path-Key-Based Mechanism", RFC 5520, DOI 10.17487/RFC5520, April 2009, <http://www.rfc-editor.org/info/rfc5520>.

[RFC6805] King, D., Ed. and A. Farrel, Ed., "The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS", RFC 6805, DOI 10.17487/RFC6805, November 2012, <http://www.rfc-editor.org/info/rfc6805>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <http://www.rfc-editor.org/info/rfc7752>.

[RFC7855] Previdi, S., Ed., Filsfils, C., Ed., Decraene, B., Litkowski, S., Horneffer, M., and R. Shakir, "Source Packet Routing in Networking (SPRING) Problem Statement and Requirements", RFC 7855, DOI 10.17487/RFC7855, May 2016, <http://www.rfc-editor.org/info/rfc7855>.

[RFC7911] Walton, D., Retana, A., Chen, E., and J. Scudder, "Advertisement of Multiple Paths in BGP", RFC 7911, DOI 10.17487/RFC7911, July 2016, <http://www.rfc-editor.org/info/rfc7911>.

[RFC7926] Farrel, A., Ed., Drake, J., Bitar, N., Swallow, G., Ceccarelli, D., and X. Zhang, "Problem Statement and Architecture for Information Exchange between Interconnected Traffic-Engineered Networks", BCP 206, RFC 7926, DOI 10.17487/RFC7926, July 2016, <http://www.rfc-editor.org/info/rfc7926>.

Authors' Addresses

Adrian Farrel
Juniper Networks

Email: afarrel@juniper.net

John Drake
Juniper Networks

Email: jdrake@juniper.net