Internet Engineering Task Force (IETF)
Request for Comments: 8660
Category: Standards Track
ISSN: 2070-1721

A. Bashandy, Ed.
Arrcus
C. Filsfils, Ed.
S. Previdi
Cisco Systems, Inc.
B. Decraene
S. Litkowski
Orange
R. Shakir
Google

December 2019

Segment Routing with the MPLS Data Plane

Abstract

Segment Routing (SR) leverages the source-routing paradigm. A node steers a packet through a controlled set of instructions, called segments, by prepending the packet with an SR header. In the MPLS data plane, the SR header is instantiated through a label stack. This document specifies the forwarding behavior to allow instantiating SR over the MPLS data plane (SR-MPLS).

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8660.

Table of Contents

1. Introduction

The Segment Routing architecture [RFC8402] can be directly applied to the MPLS architecture with no change in the MPLS forwarding plane. This document specifies forwarding-plane behavior to allow Segment Routing to operate on top of the MPLS data plane (SR-MPLS). This document does not address control-plane behavior. Control-plane behavior is specified in other documents such as [RFC8665], [RFC8666], and [RFC8667].

The Segment Routing problem statement is described in [RFC7855].

Coexistence of SR over the MPLS forwarding plane with LDP [RFC5036] is specified in [RFC8661].

Policy routing and traffic engineering using Segment Routing can be found in [ROUTING-POLICY].

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. MPLS Instantiation of Segment Routing

MPLS instantiation of Segment Routing fits in the MPLS architecture as defined in [RFC3031] from both a control-plane and forwarding-plane perspective:

* From a control-plane perspective, [RFC3031] does not mandate a single signaling protocol. Segment Routing makes use of various control-plane protocols such as link-state IGPs [RFC8665] [RFC8666] [RFC8667]. The flooding mechanisms of link-state IGPs fit very well with label stacking on the ingress. A future control-layer protocol and/or policy/configuration can be used to specify the label stack.

* From a forwarding-plane perspective, Segment Routing does not require any change to the forwarding plane because Segment IDs (SIDs) are instantiated as MPLS labels, and the Segment Routing header is instantiated as a stack of MPLS labels.

We call the "MPLS Control Plane Client (MCC)" any control-plane entity installing forwarding entries in the MPLS data plane. Local configuration and policies applied on a router are examples of MCCs.

In order to have a node segment reach the node, a network operator SHOULD configure at least one node segment per routing instance, topology, or algorithm. Otherwise, the node is not reachable within the routing instance, within the topology, or along the routing algorithm, which restricts its ability to be used by an SR Policy and as a Topology Independent Loop-Free Alternate (TI-LFA).

2.1. Multiple Forwarding Behaviors for the Same Prefix

The SR architecture does not prohibit having more than one SID for the same prefix. In fact, by allowing multiple SIDs for the same prefix, it is possible to have different forwarding behaviors (such as different paths, different ECMP and Unequal-Cost Multipath (UCMP) behaviors, etc.) for the same destination.

Instantiating Segment Routing over the MPLS forwarding plane fits seamlessly with this principle. An operator may assign multiple MPLS labels or indices to the same prefix and assign different forwarding behaviors to each label/SID. The MCC in the network downloads different MPLS labels/SIDs to the FIB for different forwarding behaviors. The MCC at the entry of an SR domain or at any point in the domain can choose to apply a particular forwarding behavior to a particular packet by applying the PUSH action to that packet using the corresponding SID.

2.2. SID Representation in the MPLS Forwarding Plane

When instantiating SR over the MPLS forwarding plane, a SID is represented by an MPLS label or an index [RFC8402].

A global SID is a label, or an index that may be mapped to an MPLS label within the Segment Routing Global Block (SRGB), of the node that installs a global SID in its FIB and receives the labeled packet. Section 2.4 specifies the procedure to map a global segment

represented by an index to an MPLS label within the SRGB.

The MCC MUST ensure that any label value corresponding to any SID it installs in the forwarding plane follows the rules below:

* The label value MUST be unique within the router on which the MCC is running, i.e., the label MUST only be used to represent the SID and MUST NOT be used to represent more than one SID or for any other forwarding purpose on the router.

* The label value MUST NOT come from the range of special-purpose labels [RFC7274].

Labels allocated in this document are considered per-platform downstream allocated labels [RFC3031].

2.3. Segment Routing Global Block and Local Block

The concepts of SRGB and global SID are explained in [RFC8402]. In general, the SRGB need not be a contiguous range of labels.

For the rest of this document, the SRGB is specified by the list of MPLS label ranges [Ll(1),Lh(1)], [Ll(2),Lh(2)],..., [Ll(k),Lh(k)] where Ll(i) =< Lh(i).

The following rules apply to the list of MPLS ranges representing the SRGB:

* The list of ranges comprising the SRGB MUST NOT overlap.

* Every range in the list of ranges specifying the SRGB MUST NOT cover or overlap with a reserved label value or range [RFC7274], respectively.

* If the SRGB of a node does not conform to the structure specified in this section or to the previous two rules, the SRGB MUST be completely ignored by all routers in the routing domain, and the node MUST be treated as if it does not have an SRGB.

* The list of label ranges MUST only be used to instantiate global SIDs into the MPLS forwarding plane.

A local segment MAY be allocated from the Segment Routing Local Block (SRLB) [RFC8402] or from any unused label as long as it does not use a special-purpose label. The SRLB consists of the range of local labels reserved by the node for certain local segments. In a controller-driven network, some controllers or applications MAY use the control plane to discover the available set of Local SIDs on a particular router [ROUTING-POLICY]. The rules applicable to the SRGB are also applicable to the SRLB, except the SRGB MUST only be used to instantiate global SIDs into the MPLS forwarding plane. The recommended, minimum, or maximum size of the SRGB and/or SRLB is a matter of future study.

2.4. Mapping a SID Index to an MPLS Label

This subsection specifies how the MPLS label value is calculated given the index of a SID. The value of the index is determined by an MCC such as IS-IS [RFC8667] or OSPF [RFC8665]. This section only specifies how to map the index to an MPLS label. The calculated MPLS label is downloaded to the FIB, sent out with a forwarded packet, or both.

Consider a SID represented by the index "I". Consider an SRGB as specified in Section 2.3. The total size of the SRGB, represented by the variable "Size", is calculated according to the formula:

size = Lh(1)- Ll(1) + 1 + Lh(2)- Ll(2) + 1 + ... + Lh(k)- Ll(k) + 1

The following rules MUST be applied by the MCC when calculating the MPLS label value corresponding to the SID index value "I".

0 =< I < size. If index "I" does not satisfy the previous inequality, then the label cannot be calculated.

The label value corresponding to the SID index "I" is calculated as follows:

```
j = 1 , temp = 0

While temp + Lh(j)- Ll(j) < I

   temp = temp + Lh(j)- Ll(j) + 1

   j = j+1

label = I - temp + Ll(j)
```

An example for how a router calculates labels and forwards traffic based on the procedure described in this section can be found in Appendix A.1.

2.5. Incoming Label Collision

The MPLS Architecture [RFC3031] defines the term Forwarding Equivalence Class (FEC) as the set of packets with similar and/or identical characteristics that are forwarded the same way and are bound to the same MPLS incoming (local) label. In Segment Routing MPLS, a local label serves as the SID for a given FEC.

We define SR FEC [RFC8402] as one of the following:

* (Prefix, Routing Instance, Topology, Algorithm) [RFC8402], where a topology identifies a set of links with metrics. For the purpose of incoming label collision resolution, the same Topology numerical value SHOULD be used on all routers to identify the same set of links with metrics. For MCCs where the "Topology" and/or "Algorithm" fields are not defined, the numerical value of zero MUST be used for these two fields. For the purpose of incoming label collision resolution, a routing instance is identified by a single incoming label downloader to the FIB. Two MCCs running on the same router are considered different routing instances if the only way the two instances know about each other’s incoming labels is through redistribution. The numerical value used to identify a routing instance MAY be derived from other configuration or MAY be explicitly configured. If it is derived from other configuration, then the same numerical value SHOULD be derived from the same configuration as long as the configuration survives router reload. If the derived numerical value varies for the same configuration, then an implementation SHOULD make the numerical value used to identify a routing instance configurable.

* (next hop, outgoing interface), where the outgoing interface is physical or virtual.

* (number of adjacencies, list of next hops, list of outgoing interfaces IDs in ascending numerical order). This FEC represents parallel adjacencies [RFC8402].

* (Endpoint, Color). This FEC represents an SR Policy [RFC8402].

* (Mirror SID). The Mirror SID (see [RFC8402], Section 5.1) is the

IP address advertised by the advertising node to identify the Mirror SID. The IP address is encoded as specified in Section 2.5.1.

This section covers the RECOMMENDED procedure for handling the scenario where, because of an error/misconfiguration, more than one SR FEC as defined in this section maps to the same incoming MPLS label. Examples illustrating the behavior specified in this section can be found in Appendix A.2.

An incoming label collision occurs if the SIDs of the set of FECs {FEC1, FEC2, ..., FECk} map to the same incoming SR MPLS label "L1".

Suppose an anycast prefix is advertised with a Prefix-SID by some, but not all, of the nodes that advertise that prefix. If the Prefix-SID sub-TLVs result in mapping that anycast prefix to the same incoming label, then the advertisement of the Prefix-SID by some, but not all, of the advertising nodes MUST NOT be treated as a label collision.

An implementation MUST NOT allow the MCCs belonging to the same router to assign the same incoming label to more than one SR FEC.

The objective of the following steps is to deterministically install in the MPLS Incoming Label Map, also known as label FIB, a single FEC with the incoming label "L1". By "deterministically install", we mean if the set of FECs {FEC1, FEC2,..., FECk} map to the same incoming SR MPLS label "L1", then the steps below assign the same FEC to the label "L1" irrespective of the order by which the mappings of this set of FECs to the label "L1" are received. For example, first-come, first-served tiebreaking is not allowed. The remaining FECs may be installed in the IP FIB without an incoming label.

The procedure in this section relies completely on the local FEC and label database within a given router.

The collision resolution procedure is as follows:

1. Given the SIDs of the set of FECs, {FEC1, FEC2,..., FECk} map to the same MPLS label "L1".

2. Within an MCC, apply tiebreaking rules to select one FEC only, and assign the label to it. The losing FECs are handled as if no labels are attached to them. The losing FECs with algorithms other than the shortest path first [RFC8402] are not installed in the FIB.

    a. If the same set of FECs are attached to the same label "L1", then the tiebreaking rules MUST always select the same FEC irrespective of the order in which the FECs and the label "L1" are received. In other words, the tiebreaking rule MUST be deterministic.

3. If there is still collision between the FECs belonging to different MCCs, then reapply the tiebreaking rules to the remaining FECs to select one FEC only, and assign the label to that FEC.

4. Install the selected FEC into the IP FIB and its incoming label into the label FIB.

5. The remaining FECs with the default algorithm (see the Prefix-SID algorithm specification [RFC8402]) may be installed in the FIB natively, such as pure IP entries in case of Prefix FEC, without any incoming labels corresponding to their SIDs. The remaining FECs with algorithms other than the shortest path first [RFC8402]

are not installed in the FIB.

2.5.1. Tiebreaking Rules

The default tiebreaking rules are specified as follows:

1. Determine the lowest administrative distance among the competing FECs as defined in the section below. Then filter away all the competing FECs with a higher administrative distance.

2. If more than one competing FEC remains after step 1, select the smallest numerical FEC value. The numerical value of the FEC is determined according to the FEC encoding described later in this section.

These rules deterministically select which FEC to install in the MPLS forwarding plane for the given incoming label.

This document defines the default tiebreaking rules that SHOULD be implemented. An implementation MAY choose to support different tiebreaking rules and MAY use one of these instead of the default tiebreaking rules. To maximize MPLS forwarding consistency in case of a SID configuration error, the network operator MUST deploy, within an IGP flooding area, routers implementing the same tiebreaking rules.

Each FEC is assigned an administrative distance. The FEC administrative distance is encoded as an 8-bit value. The lower the value, the better the administrative distance.

The default FEC administrative distance order starting from the lowest value SHOULD be:

* Explicit SID assignment to a FEC that maps to a label outside the SRGB irrespective of the owner MCC. An explicit SID assignment is a static assignment of a label to a FEC such that the assignment survives a router reboot.

  - An example of explicit SID allocation is static assignment of a specific label to an Adj-SID.

  - An implementation of explicit SID assignment MUST guarantee collision freeness on the same router.

* Dynamic SID assignment:

  - All FEC types, except for the SR Policy, are ordered using the default administrative distance defined by the implementation.

  - The Binding SID [RFC8402] assigned to the SR Policy always has a higher default administrative distance than the default administrative distance of any other FEC type.

To maximize MPLS forwarding consistency, if the same FEC is advertised in more than one protocol, a user MUST ensure that the administrative distance preference between protocols is the same on all routers of the IGP flooding domain. Note that this is not really new as this already applies to IP forwarding.

The numerical sort across FECs SHOULD be performed as follows:

* Each FEC is assigned a FEC type encoded in 8 bits. The type codepoints for each SR FEC defined at the beginning of this section are as follows:

    120: (Prefix, Routing Instance, Topology, Algorithm)

130: (next hop, outgoing interface)

140: Parallel Adjacency [RFC8402]

150: SR Policy [RFC8402]

160: Mirror SID [RFC8402]

The numerical values above are mentioned to guide implementation. If other numerical values are used, then the numerical values must maintain the same greater-than ordering of the numbers mentioned here.

* The fields of each FEC are encoded as follows:

  - All fields in all FECs are encoded in big endian order.

  - The Routing Instance ID is represented by 16 bits. For routing instances that are identified by less than 16 bits, encode the Instance ID in the least significant bits while the most significant bits are set to zero.

  - The address family is represented by 8 bits, where IPv4 is encoded as 100, and IPv6 is encoded as 110. These numerical values are mentioned to guide implementations. If other numerical values are used, then the numerical value of IPv4 MUST be less than the numerical value for IPv6.

  - All addresses are represented in 128 bits as follows:

    o The IPv6 address is encoded natively.

    o The IPv4 address is encoded in the most significant bits, and the remaining bits are set to zero.

  - All prefixes are represented by (8 + 128) bits.

    o A prefix is encoded in the most significant bits, and the remaining bits are set to zero.

    o The prefix length is encoded before the prefix in an 8-bit field.

  - The Topology ID is represented by 16 bits. For routing instances that identify topologies using less than 16 bits, encode the topology ID in the least significant bits while the most significant bits are set to zero.

  - The Algorithm is encoded in a 16-bit field.

  - The Color ID is encoded using 32 bits.

* Choose the set of FECs of the smallest FEC type codepoint.

* Out of these FECs, choose the FECs with the smallest address family codepoint.

* Encode the remaining set of FECs as follows:

  - (Prefix, Routing Instance, Topology, Algorithm) is encoded as (Prefix Length, Prefix, routing_instance_id, Topology, SR Algorithm).

  - (next hop, outgoing interface) is encoded as (next hop, outgoing_interface_id).

- (number of adjacencies, list of next hops in ascending numerical order, list of outgoing interface IDs in ascending numerical order) is used to encode a parallel adjacency [RFC8402].

- (Endpoint, Color) is encoded as (Endpoint_address, Color_id).

- (IP address) is the encoding for a Mirror SID FEC. The IP address is encoded as described above in this section.

* Select the FEC with the smallest numerical value.

The numerical values mentioned in this section are for guidance only. If other numerical values are used, then the other numerical values MUST maintain the same numerical ordering among different SR FECs.

2.5.2. Redistribution between Routing Protocol Instances

The following rule SHOULD be applied when redistributing SIDs with prefixes between routing protocol instances:

* If the SRGB of the receiving instance is the same as the SRGB of the origin instance, then:

  - the index is redistributed with the route.

* Else,

  - the index is not redistributed and if the receiving instance decides to advertise an index with the redistributed route, it is the duty of the receiving instance to allocate a fresh index relative to its own SRGB. Note that in this case, the receiving instance MUST compute the local label it assigns to the route according to Section 2.4 and install it in FIB.

It is outside the scope of this document to define local node behaviors that would allow the mapping of the original index into a new index in the receiving instance via the addition of an offset or other policy means.

2.5.2.1. Illustration

```
A----IS-IS----B---OSPF----C-192.0.2.1/32 (20001)
```

Consider the simple topology above, where:

* A and B are in the IS-IS domain with SRGB = [16000-17000]

* B and C are in the OSPF domain with SRGB = [20000-21000]

* B redistributes 192.0.2.1/32 into the IS-IS domain

In this case, A learns 192.0.2.1/32 as an IP leaf connected to B, which is usual for IP prefix redistribution

However, according to the redistribution rule above, B decides not to advertise any index with 192.0.2.1/32 into IS-IS because the SRGB is not the same.

2.5.2.2. Illustration 2

Consider the example in the illustration described in Section 2.5.2.1.

When router B redistributes the prefix 192.0.2.1/32, router B decides

to allocate and advertise the same index 1 with the prefix 192.0.2.1/32.

Within the SRGB of the IS-IS domain, index 1 corresponds to the local label 16001. Hence, according to the redistribution rule above, router B programs the incoming label 16001 in its FIB to match traffic arriving from the IS-IS domain destined to the prefix 192.0.2.1/32.

2.6. Effect of Incoming Label Collision on Outgoing Label Programming

When determining what outgoing label to use, the ingress node that pushes new segments, and hence a stack of MPLS labels, MUST use, for a given FEC, the label that has been selected by the node receiving the packet with that label exposed as the top label. So in case of incoming label collision on this receiving node, the ingress node MUST resolve this collision by using this same "Incoming Label Collision resolution procedure" and by using the data of the receiving node.

In the general case, the ingress node may not have the exact same data as the receiving node, so the result may be different. This is under the responsibility of the network operator. But in a typical case, e.g., where a centralized node or a distributed link-state IGP is used, all nodes would have the same database. However, to minimize the chance of misforwarding, a FEC that loses its incoming label to the tiebreaking rules specified in Section 2.5 MUST NOT be installed in FIB with an outgoing Segment Routing label based on the SID corresponding to the lost incoming label.

Examples for the behavior specified in this section can be found in Appendix A.3.

2.7. PUSH, CONTINUE, and NEXT

PUSH, NEXT, and CONTINUE are operations applied by the forwarding plane. The specifications of these operations can be found in [RFC8402]. This subsection specifies how to implement each of these operations in the MPLS forwarding plane.

2.7.1. PUSH

As described in [RFC8402], PUSH corresponds to pushing one or more labels on top of an incoming packet then sending it out of a particular physical interface or virtual interface, such as a UDP tunnel [RFC7510] or the Layer 2 Tunneling Protocol version 3 (L2TPv3) [RFC4817], towards a particular next hop. When pushing labels onto a packet’s label stack, the Time-to-Live (TTL) field [RFC3032] [RFC3443] and the Traffic Class (TC) field [RFC3032] [RFC5462] of each label stack entry must, of course, be set. This document does not specify any set of rules for setting these fields; that is a matter of local policy. Sections 2.10 and 2.11 specify additional details about forwarding behavior.

2.7.2. CONTINUE

As described in [RFC8402], the CONTINUE operation corresponds to swapping the incoming label with an outgoing label. The value of the outgoing label is calculated as specified in Sections 2.10 and 2.11.

2.7.3. NEXT

As described in [RFC8402], NEXT corresponds to popping the topmost label. The action before and/or after the popping depends on the instruction associated with the active SID on the received packet prior to the popping. For example, suppose the active SID in the

received packet was an Adj-SID [RFC8402]; on receiving the packet, the node applies the NEXT operation, which corresponds to popping the topmost label, and then sends the packet out of the physical or virtual interface (e.g., the UDP tunnel [RFC7510] or L2TPv3 tunnel [RFC4817]) towards the next hop corresponding to the Adj-SID.

2.7.3.1. Mirror SID

If the active SID in the received packet was a Mirror SID (see [RFC8402], Section 5.1) allocated by the receiving router, the receiving router applies the NEXT operation, which corresponds to popping the topmost label, and then performs a lookup using the contents of the packet after popping the outermost label in the mirrored forwarding table. The method by which the lookup is made, and/or the actions applied to the packet after the lookup in the mirror table, depends on the contents of the packet and the mirror table. Note that the packet exposed after popping the topmost label may or may not be an MPLS packet. A Mirror SID can be viewed as a generalization of the context label in [RFC5331] because a Mirror SID does not make any assumptions about the packet underneath the top label.

2.8. MPLS Label Downloaded to the FIB for Global and Local SIDs

The label corresponding to the global SID "Si", which is represented by the global index "I" and downloaded to the FIB, is used to match packets whose active segment (and hence topmost label) is "Si". The value of this label is calculated as specified in Section 2.4.

For Local SIDs, the MCC is responsible for downloading the correct label value to the FIB. For example, an IGP with SR extensions [RFC8667] [RFC8665] downloads the MPLS label corresponding to an Adj-SID [RFC8402].

2.9. Active Segment

When instantiated in the MPLS domain, the active segment on a packet corresponds to the topmost label and is calculated according to the procedure specified in Sections 2.10 and 2.11. When arriving at a node, the topmost label corresponding to the active SID matches the MPLS label downloaded to the FIB as specified in Section 2.4.

2.10. Forwarding Behavior for Global SIDs

This section specifies the forwarding behavior, including the calculation of outgoing labels, that corresponds to a global SID when applying the PUSH, CONTINUE, and NEXT operations in the MPLS forwarding plane.

This document covers the calculation of the outgoing label for the top label only. The case where the outgoing label is not the top label and is part of a stack of labels that instantiates a routing policy or a traffic-engineering tunnel is outside the scope of this document and may be covered in other documents such as [ROUTING-POLICY].

2.10.1. Forwarding for PUSH and CONTINUE of Global SIDs

Suppose an MCC on router "R0" determines that, before sending the packet towards a neighbor "N", the PUSH or CONTINUE operation is to be applied to an incoming packet related to the global SID "Si". SID "Si" is represented by the global index "I" and owned by the router Ri. Neighbor "N" may be directly connected to "R0" through either a physical or a virtual interface (e.g., UDP tunnel [RFC7510] or L2TPv3 tunnel [RFC4817]).

The method by which the MCC on router "R0" determines that the PUSH or CONTINUE operation must be applied using the SID "Si" is beyond the scope of this document. An example of a method to determine the SID "Si" for the PUSH operation is the case where IS-IS [RFC8667] receives the Prefix-SID "Si" sub-TLV advertised with the prefix "P/m" in TLV 135, and the prefix "P/m" is the longest matching network prefix for the incoming IPv4 packet.

For the CONTINUE operation, an example of a method used to determine the SID "Si" is the case where IS-IS [RFC8667] receives the Prefix-SID "Si" sub-TLV advertised with prefix "P" in TLV 135, and the top label of the incoming packet matches the MPLS label in the FIB corresponding to the SID "Si" on router "R0".

The forwarding behavior for PUSH and CONTINUE corresponding to the SID "Si" is as follows:

* If neighbor "N" does not support SR or advertises an invalid SRGB or a SRGB that is too small for the SID "Si", then:

  - If it is possible to send the packet towards neighbor "N" using standard MPLS forwarding behavior as specified in [RFC3031] and [RFC3032], forward the packet. The method by which a router decides whether it is possible to send the packet to "N" or not is beyond the scope of this document. For example, the router "R0" can use the downstream label determined by another MCC, such as LDP [RFC5036], to send the packet.

  - Else, if there are other usable next hops, use them to forward the incoming packet. The method by which the router "R0" decides on the possibility of using other next hops is beyond the scope of this document. For example, the MCC on "R0" may chose the send an IPv4 packet without pushing any label to another next hop.

  - Otherwise, drop the packet.

* Else,

  - Calculate the outgoing label as specified in Section 2.4 using the SRGB of neighbor "N".

  - Determine the outgoing label stack

    o If the operation is PUSH:

      + Push the calculated label according to the MPLS label pushing rules specified in [RFC3032].

    o Else,

      + swap the incoming label with the calculated label according to the label-swapping rules in [RFC3031].

    o Send the packet towards neighbor "N".

2.10.2. Forwarding for the NEXT Operation for Global SIDs

As specified in Section 2.7.3, the NEXT operation corresponds to popping the topmost label. The forwarding behavior is as follows:

* Pop the topmost label

* Apply the instruction associated with the incoming label that has been popped

The action on the packet after popping the topmost label depends on the instruction associated with the incoming label as well as the contents of the packet right underneath the top label that was popped. Examples of the NEXT operation are described in Appendix A.1

2.11. Forwarding Behavior for Local SIDs

This section specifies the forwarding behavior for Local SIDs when SR is instantiated over the MPLS forwarding plane.

2.11.1. Forwarding for the PUSH Operation on Local SIDs

Suppose an MCC on router "R0" determines that the PUSH operation is to be applied to an incoming packet using the Local SID "Si" before sending the packet towards neighbor "N", which is directly connected to R0 through a physical or virtual interface such as a UDP tunnel [RFC7510] or L2TPv3 tunnel [RFC4817].

An example of such a Local SID is an Adj-SID allocated and advertised by IS-IS [RFC8667]. The method by which the MCC on "R0" determines that the PUSH operation is to be applied to the incoming packet is beyond the scope of this document. An example of such a method is the backup path used to protect against a failure using TI-LFA [FAST-REROUTE].

As mentioned in [RFC8402], a Local SID is specified by an MPLS label. Hence, the PUSH operation for a Local SID is identical to the label push operation using any MPLS label [RFC3031]. The forwarding action after pushing the MPLS label corresponding to the Local SID is also determined by the MCC. For example, if the PUSH operation was done to forward a packet over a backup path calculated using TI-LFA, then the forwarding action may be sending the packet to a certain neighbor that will in turn continue to forward the packet along the backup path.

2.11.2. Forwarding for the CONTINUE Operation for Local SIDs

A Local SID on router "R0" corresponds to a local label. In such a scenario, the outgoing label towards next hop "N" is determined by the MCC running on the router "R0", and the forwarding behavior for the CONTINUE operation is identical to the swap operation on an MPLS label [RFC3031].

2.11.3. Outgoing Label for the NEXT Operation for Local SIDs

The NEXT operation for Local SIDs is identical to the NEXT operation for global SIDs as specified in Section 2.10.2.

3. IANA Considerations

This document has no IANA actions.

4. Manageability Considerations

This document describes the applicability of Segment Routing over the MPLS data plane. Segment Routing does not introduce any change in the MPLS data plane. Manageability considerations described in [RFC8402] apply to the MPLS data plane when used with Segment Routing. SR Operations, Administration, and Maintenance (OAM) use cases for the MPLS data plane are defined in [RFC8403]. SR OAM procedures for the MPLS data plane are defined in [RFC8287].

5. Security Considerations

This document does not introduce additional security requirements and mechanisms other than the ones described in [RFC8402].

6. References

6.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, DOI 10.17487/RFC3031, January 2001, <https://www.rfc-editor.org/info/rfc3031>.

[RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", RFC 3032, DOI 10.17487/RFC3032, January 2001, <https://www.rfc-editor.org/info/rfc3032>.

[RFC3443] Agarwal, P. and B. Akyol, "Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks", RFC 3443, DOI 10.17487/RFC3443, January 2003, <https://www.rfc-editor.org/info/rfc3443>.

[RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, DOI 10.17487/RFC5462, February 2009, <https://www.rfc-editor.org/info/rfc5462>.

[RFC7274] Kompella, K., Andersson, L., and A. Farrel, "Allocating and Retiring Special-Purpose MPLS Labels", RFC 7274, DOI 10.17487/RFC7274, June 2014, <https://www.rfc-editor.org/info/rfc7274>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

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

6.2. Informative References

[FAST-REROUTE] Litkowski, S., Bashandy, A., Filsfils, C., Decraene, B., Francois, P., Voyer, D., Clad, F., and P. Camarillo, "Topology Independent Fast Reroute using Segment Routing", Work in Progress, Internet-Draft, draft-ietf-rtgwg-segment-routing-ti-lfa-01, 5 March 2019, <https://tools.ietf.org/html/draft-ietf-rtgwg-segment-routing-ti-lfa-01>.

[RFC4817] Townsley, M., Pignataro, C., Wainner, S., Seely, T., and J. Young, "Encapsulation of MPLS over Layer 2 Tunneling Protocol Version 3", RFC 4817, DOI 10.17487/RFC4817, March 2007, <https://www.rfc-editor.org/info/rfc4817>.

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

[RFC5331] Aggarwal, R., Rekhter, Y., and E. Rosen, "MPLS Upstream Label Assignment and Context-Specific Label Space",

RFC 5331, DOI 10.17487/RFC5331, August 2008,
<https://www.rfc-editor.org/info/rfc5331>.

[RFC7510] Xu, X., Sheth, N., Yong, L., Callon, R., and D. Black, "Encapsulating MPLS in UDP", RFC 7510, DOI 10.17487/RFC7510, April 2015, <https://www.rfc-editor.org/info/rfc7510>.

[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, <https://www.rfc-editor.org/info/rfc7855>.

[RFC8287] Kumar, N., Ed., Pignataro, C., Ed., Swallow, G., Akiya, N., Kini, S., and M. Chen, "Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes", RFC 8287, DOI 10.17487/RFC8287, December 2017, <https://www.rfc-editor.org/info/rfc8287>.

[RFC8403] Geib, R., Ed., Filsfils, C., Pignataro, C., Ed., and N. Kumar, "A Scalable and Topology-Aware MPLS Data-Plane Monitoring System", RFC 8403, DOI 10.17487/RFC8403, July 2018, <https://www.rfc-editor.org/info/rfc8403>.

[RFC8661] Bashandy, A., Ed., Filsfils, C., Ed., Previdi, S., Decraene, B., and S. Litkowski, "Segment Routing MPLS Interworking with LDP", RFC 8661, DOI 10.17487/RFC8661, December 2019, <https://www.rfc-editor.org/info/rfC8661>.

[RFC8665] Psenak, P., Ed., Previdi, S., Ed., Filsfils, C., Gredler, H., Shakir, R., Henderickx, W., and J. Tantsura, "OSPF Extensions for Segment Routing", RFC 8665, DOI 10.17487/RFC8665, December 2019, <https://www.rfc-editor.org/info/rfc8665>.

[RFC8666] Psenak, P., Ed. and S. Previdi, Ed., "OSPFv3 Extensions for Segment Routing", RFC 8666, DOI 10.17487/RFC8666, December 2019, <https://www.rfc-editor.org/info/rfc8666>.

[RFC8667] Previdi, S., Ed., Ginsberg, L., Ed., Filsfils, C., Bashandy, A., Gredler, H., and B. Decraene, "IS-IS Extensions for Segment Routing", RFC 8667, DOI 10.17487/RFC8667, December 2019, <https://www.rfc-editor.org/info/rfc8667>.

[ROUTING-POLICY] Filsfils, C., Sivabalan, S., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", Work in Progress, Internet-Draft, draft-ietf-spring-segment-routing-policy-05, 17 November 2019, <https://tools.ietf.org/html/draft-ietf-spring-segment-routing-policy-05>.

Appendix A. Examples

A.1. IGP Segment Examples

Consider the network diagram of Figure 1 and the IP addresses and IGP segment allocations of Figure 2. Assume that the network is running IS-IS with SR extensions [RFC8667], and all links have the same metric. The following examples can be constructed.

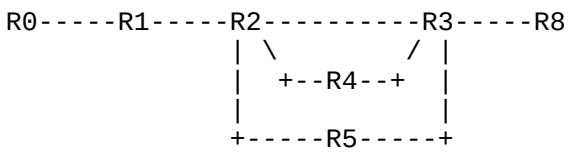

Figure 1: IGP Segments -- Illustration

```
+-----------------------------------------------------------+
| IP addresses allocated by the operator:                   |
|                      192.0.2.1/32 as a loopback of R1     |
|                      192.0.2.2/32 as a loopback of R2     |
|                      192.0.2.3/32 as a loopback of R3     |
|                      192.0.2.4/32 as a loopback of R4     |
|                      192.0.2.5/32 as a loopback of R5     |
|                      192.0.2.8/32 as a loopback of R8     |
|              198.51.100.9/32 as an anycast loopback of R4 |
|              198.51.100.9/32 as an anycast loopback of R5 |
|                                                           |
| SRGB defined by the operator as [1000,5000]               |
|                                                           |
| Global IGP SID indices allocated by the operator:         |
|                      1 allocated to 192.0.2.1/32          |
|                      2 allocated to 192.0.2.2/32          |
|                      3 allocated to 192.0.2.3/32          |
|                      4 allocated to 192.0.2.4/32          |
|                      8 allocated to 192.0.2.8/32          |
|                   1009 allocated to 198.51.100.9/32       |
|                                                           |
| Local IGP SID allocated dynamically by R2                 |
|                     for its "north" adjacency to R3: 9001 |
|                     for its "east" adjacency to R3 : 9002 |
|                     for its "south" adjacency to R3: 9003 |
|                     for its only adjacency to R4   : 9004 |
|                     for its only adjacency to R1   : 9005 |
+-----------------------------------------------------------+
```

Figure 2: IGP Address and Segment Allocation -- Illustration

Suppose R1 wants to send IPv4 packet P1 to R8. In this case, R1 needs to apply the PUSH operation to the IPv4 packet.

Remember that the SID index "8" is a global IGP segment attached to the IP prefix 192.0.2.8/32. Its semantic is global within the IGP domain: any router forwards a packet received with active segment 8 to the next hop along the ECMP-aware shortest path to the related prefix.

R2 is the next hop along the shortest path towards R8. By applying the steps in Section 2.8, the outgoing label downloaded to R1’s FIB corresponding to the global SID index "8" is 1008 because the SRGB of R2 = [1000,5000] as shown in Figure 2.

Because the packet is IPv4, R1 applies the PUSH operation using the label value 1008 as specified in Section 2.10.1. The resulting MPLS header will have the "S" bit [RFC3032] set because it is followed directly by an IPv4 packet.

The packet arrives at router R2. Because top label 1008 corresponds to the IGP SID index "8", which is the Prefix-SID attached to the prefix 192.0.2.8/32 owned by Node R8, the instruction associated with the SID is "forward the packet using one of the ECMP interfaces or next hops along the shortest path(s) towards R8". Because R2 is not the penultimate hop, R2 applies the CONTINUE operation to the packet and sends it to R3 using one of the two links connected to R3 with top label 1008 as specified in Section 2.10.1.

R3 receives the packet with top label 1008. Because top label 1008 corresponds to the IGP SID index "8", which is the Prefix-SID attached to the prefix 192.0.2.8/32 owned by Node R8, the instruction associated with the SID is "send the packet using one of the ECMP interfaces and next hops along the shortest path towards R8". Because R3 is the penultimate hop, we assume that R3 performs penultimate hop popping, which corresponds to the NEXT operation; the packet is then sent to R8. The NEXT operation results in popping the outer label and sending the packet as a pure IPv4 packet to R8.

In conclusion, the path followed by P1 is R1-R2--R3-R8. The ECMP awareness ensures that the traffic is load-shared between any ECMP path; in this case, it’s the two links between R2 and R3.

A.2. Incoming Label Collision Examples

This section outlines several examples to illustrate the handling of label collision described in Section 2.5.

For the examples in this section, we assume that Node A has the following:

* OSPF default admin distance for implementation=50

* IS-IS default admin distance for implementation=60

A.2.1. Example 1

The following example illustrates incoming label collision resolution for the same FEC type using MCC administrative distance.

FEC1:

Node A receives an OSPF Prefix-SID Advertisement from Node B for 198.51.100.5/32 with index=5. Assuming that OSPF SRGB on Node A = [1000,1999], the incoming label is 1005.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.105/32 with index=5. Assuming that IS-IS SRGB on Node A = [1000,1999], the incoming label is 1005.

FEC1 and FEC2 both use dynamic SID assignment. Since neither of the FECs are of type ’SR Policy’, we use the default admin distances of 50 and 60 to break the tie. So FEC1 wins.

A.2.2. Example 2

The following example Illustrates incoming label collision resolution for different FEC types using the MCC administrative distance.

FEC1:

Node A receives an OSPF Prefix-SID Advertisement from Node B for 198.51.100.6/32 with index=6. Assuming that OSPF SRGB on Node A = [1000,1999], the incoming label on Node A corresponding to 198.51.100.6/32 is 1006.

FEC2:

IS-IS on Node A assigns label 1006 to the globally significant Adj-SID (i.e., when advertised, the L-Flag is clear in the Adj-SID sub-TLV as described in [RFC8667]). Hence, the incoming label corresponding to this Adj-SID is 1006. Assume Node A allocates this

Adj-SID dynamically, and it may differ across router reboots.

FEC1 and FEC2 both use dynamic SID assignment. Since neither of the FECs are of type 'SR Policy', we use the default admin distances of 50 and 60 to break the tie. So FEC1 wins.

A.2.3. Example 3

The following example illustrates incoming label collision resolution based on preferring static over dynamic SID assignment.

FEC1:

OSPF on Node A receives a Prefix-SID Advertisement from Node B for 198.51.100.7/32 with index=7. Assuming that the OSPF SRGB on Node A = [1000,1999], the incoming label corresponding to 198.51.100.7/32 is 1007.

FEC2:

The operator on Node A configures IS-IS on Node A to assign label 1007 to the globally significant Adj-SID (i.e., when advertised, the L-Flag is clear in the Adj-SID sub-TLV as described in [RFC8667]).

Node A assigns this Adj-SID explicitly via configuration, so the Adj-SID survives router reboots.

FEC1 uses dynamic SID assignment, while FEC2 uses explicit SID assignment. So FEC2 wins.

A.2.4. Example 4

The following example illustrates incoming label collision resolution using FEC type default administrative distance.

FEC1:

OSPF on Node A receives a Prefix-SID Advertisement from Node B for 198.51.100.8/32 with index=8. Assuming that OSPF SRGB on Node A = [1000,1999], the incoming label corresponding to 198.51.100.8/32 is 1008.

FEC2:

Suppose the SR Policy Advertisement from the controller to Node A for the policy identified by (Endpoint = 192.0.2.208, color = 100) that consists of SID-List=<S1, S2> assigns the globally significant Binding-SID label 1008.

From the point of view of Node A, FEC1 and FEC2 both use dynamic SID assignment. Based on the default administrative distance outlined in Section 2.5.1, the Binding SID has a higher administrative distance than the Prefix-SID; hence, FEC1 wins.

A.2.5. Example 5

The following example illustrates incoming label collision resolution based on FEC type preference.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.110/32 with index=10. Assuming that the IS-IS SRGB on Node A = [1000,1999], the incoming label corresponding to 203.0.113.110/32 is 1010.

FEC2:

IS-IS on Node A assigns label 1010 to the globally significant Adj-SID (i.e., when advertised, the L-Flag is clear in the Adj-SID sub-TLV as described in [RFC8667]).

Node A allocates this Adj-SID dynamically, and it may differ across router reboots. Hence, both FEC1 and FEC2 both use dynamic SID assignment.

Since both FECs are from the same MCC, they have the same default admin distance. So we compare the FEC type codepoints. FEC1 has FEC type codepoint=120, while FEC2 has FEC type codepoint=130. Therefore, FEC1 wins.

A.2.6. Example 6

The following example illustrates incoming label collision resolution based on address family preference.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.111/32 with index=11. Assuming that the IS-IS SRGB on Node A = [1000,1999], the incoming label on Node A for 203.0.113.111/32 is 1011.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 2001:DB8:1000::11/128 with index=11. Assuming that the IS-IS SRGB on Node A = [1000,1999], the incoming label on Node A for 2001:DB8:1000::11/128 is 1011.

FEC1 and FEC2 both use dynamic SID assignment. Since both FECs are from the same MCC, they have the same default admin distance. So we compare the FEC type codepoints. Both FECs have FEC type codepoint=120. So we compare the address family. Since IPv4 is preferred over IPv6, FEC1 wins.

A.2.7. Example 7

The following example illustrates incoming label collision resolution based on prefix length.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.112/32 with index=12. Assuming that IS-IS SRGB on Node A = [1000,1999], the incoming label for 203.0.113.112/32 on Node A is 1012.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.128/30 with index=12. Assuming that the IS-IS SRGB on Node A = [1000,1999], the incoming label for 203.0.113.128/30 on Node A is 1012.

FEC1 and FEC2 both use dynamic SID assignment. Since both FECs are from the same MCC, they have the same default admin distance. So we compare the FEC type codepoints. Both FECs have FEC type codepoint=120. So we compare the address family. Both are a part of the IPv4 address family, so we compare the prefix length. FEC1 has prefix length=32, and FEC2 has prefix length=30, so FEC2 wins.

A.2.8. Example 8

The following example illustrates incoming label collision resolution based on the numerical value of the FECs.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.113/32 with index=13. Assuming that IS-IS SRGB on Node A = [1000,1999], the incoming label for 203.0.113.113/32 on Node A is 1013.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.213/32 with index=13. Assuming that IS-IS SRGB on Node A = [1000,1999], the incoming label for 203.0.113.213/32 on Node A is 1013.

FEC1 and FEC2 both use dynamic SID assignment. Since both FECs are from the same MCC, they have the same default admin distance. So we compare the FEC type codepoints. Both FECs have FEC type codepoint=120. So we compare the address family. Both are a part of the IPv4 address family, so we compare the prefix length. Prefix lengths are the same, so we compare the prefix. FEC1 has the lower prefix, so FEC1 wins.

A.2.9. Example 9

The following example illustrates incoming label collision resolution based on the Routing Instance ID.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.114/32 with index=14. Assume that this IS-IS instance on Node A has Routing Instance ID = 1000 and SRGB = [1000,1999]. Hence, the incoming label for 203.0.113.114/32 on Node A is 1014.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.114/32 with index=14. Assume that this is another instance of IS-IS on Node A but Routing Instance ID = 2000 is different and SRGB = [1000,1999] is the same. Hence, the incoming label for 203.0.113.114/32 on Node A is 1014.

These two FECs match all the way through the prefix length and prefix. So the Routing Instance ID breaks the tie, and FEC1 wins.

A.2.10. Example 10

The following example illustrates incoming label collision resolution based on the topology ID.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.115/32 with index=15. Assume that this IS-IS instance on Node A has Routing Instance ID = 1000. Assume that the prefix advertisement of 203.0.113.115/32 was received in the IS-IS Multi-topology advertisement with ID = 50. If the IS-IS SRGB for this routing instance on Node A = [1000,1999], then the incoming label of 203.0.113.115/32 for topology 50 on Node A is 1015.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.115/32 with index=15. Assume that it has the same Routing Instance ID = 1000, but 203.0.113.115/32 was advertised with IS-IS Multi-topology ID = 40, which is different. If the IS-IS SRGB on Node A = [1000,1999], then the incoming label of 203.0.113.115/32 for topology 40 on Node A is also 1015.

Since these two FECs match all the way through the prefix length, prefix, and Routing Instance ID, we compare the IS-IS Multi-topology ID, so FEC2 wins.

A.2.11. Example 11

The following example illustrates incoming label collision for resolution based on the algorithm ID.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.116/32 with index=16. Assume that IS-IS on Node A has Routing Instance ID = 1000. Assume that Node B advertised 203.0.113.116/32 with IS-IS Multi-topology ID = 50 and SR algorithm = 0. Assume that the IS-IS SRGB on Node A = [1000,1999]. Hence, the incoming label corresponding to this advertisement of 203.0.113.116/32 is 1016.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.116/32 with index=16. Assume that it is the same IS-IS instance on Node A with Routing Instance ID = 1000. Also assume that Node C advertised 203.0.113.116/32 with IS-IS Multi-topology ID = 50 but with SR algorithm = 22. Since it is the same routing instance, the SRGB on Node A = [1000,1999]. Hence, the incoming label corresponding to this advertisement of 203.0.113.116/32 by Node C is also 1016.

Since these two FECs match all the way through in terms of the prefix length, prefix, Routing Instance ID, and Multi-topology ID, we compare the SR algorithm IDs, so FEC1 wins.

A.2.12. Example 12

The following example illustrates incoming label collision resolution based on the FEC numerical value, independent of how the SID is assigned to the colliding FECs.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.117/32 with index=17. Assume that the IS-IS SRGB on Node A = [1000,1999]; thus, the incoming label is 1017.

FEC2:

Suppose there is an IS-IS Mapping Server Advertisement (SID / Label Binding TLV) from Node D that has range = 100 and prefix = 203.0.113.1/32. Suppose this Mapping Server Advertisement generates 100 mappings, one of which maps 203.0.113.17/32 to index=17. Assuming that it is the same IS-IS instance, the SRGB = [1000,1999] and hence the incoming label for 1017.

Even though FEC1 comes from a normal Prefix-SID Advertisement and FEC2 is generated from a Mapping Server Advertisement, it is not used as a tiebreaking parameter. Both FECs use dynamic SID assignment,

are from the same MCC, and have the same FEC type codepoint=120. Their prefix lengths are the same as well. FEC2 wins based on its lower numerical prefix value, since 203.0.113.17 is less than 203.0.113.117.

A.2.13. Example 13

The following example illustrates incoming label collision resolution based on address family preference.

FEC1:

SR Policy Advertisement from the controller to Node A. Endpoint address=2001:DB8:3000::100, color=100, SID-List=<S1, S2>, and the Binding-SID label=1020.

FEC2:

SR Policy Advertisement from controller to Node A. Endpoint address=192.0.2.60, color=100, SID-List=<S3, S4>, and the Binding-SID label=1020.

The FEC tiebreakers match, and they have the same FEC type codepoint=140. Thus, FEC2 wins based on the IPv4 address family being preferred over IPv6.

A.2.14. Example 14

The following example illustrates incoming label resolution based on the numerical value of the policy endpoint.

FEC1:

SR Policy Advertisement from the controller to Node A. Endpoint address=192.0.2.70, color=100, SID-List=<S1, S2>, and Binding-SID label=1021.

FEC2:

SR Policy Advertisement from the controller to Node A. Endpoint address=192.0.2.71, color=100, SID-List=<S3, S4>, and Binding-SID label=1021.

The FEC tiebreakers match, and they have the same address family. Thus, FEC1 wins by having the lower numerical endpoint address value.

A.3. Examples for the Effect of Incoming Label Collision on an Outgoing Label

This section presents examples to illustrate the effect of incoming label collision on the selection of the outgoing label as described in Section 2.6.

A.3.1. Example 1

The following example illustrates the effect of incoming label resolution on the outgoing label.

FEC1:

IS-IS on Node A receives a Prefix-SID Advertisement from Node B for 203.0.113.122/32 with index=22. Assuming that the IS-IS SRGB on Node A = [1000,1999], the corresponding incoming label is 1022.

FEC2:

IS-IS on Node A receives a Prefix-SID Advertisement from Node C for 203.0.113.222/32 with index=22. Assuming that the IS-IS SRGB on Node A = [1000,1999], the corresponding incoming label is 1022.

FEC1 wins based on the lowest numerical prefix value. This means that Node A installs a transit MPLS forwarding entry to swap incoming label 1022 with outgoing label N and to use outgoing interface I. N is determined by the index associated with FEC1 (index=22) and the SRGB advertised by the next-hop node on the shortest path to reach 203.0.113.122/32.

Node A will generally also install an imposition MPLS forwarding entry corresponding to FEC1 for incoming prefix=203.0.113.122/32 pushing outgoing label N, and using outgoing interface I.

The rule in Section 2.6 means Node A MUST NOT install an ingress MPLS forwarding entry corresponding to FEC2 (the losing FEC, which would be for prefix 203.0.113.222/32).

A.3.2. Example 2

The following example illustrates the effect of incoming label collision resolution on outgoing label programming on Node A.

FEC1:

SR Policy Advertisement from the controller to Node A. Endpoint address=192.0.2.80, color=100, SID-List=<S1, S2>, and Binding-SID label=1023.

FEC2:

SR Policy Advertisement from controller to Node A. Endpoint address=192.0.2.81, color=100, SID-List=<S3, S4>, and Binding-SID label=1023.

FEC1 wins by having the lower numerical endpoint address value. This means that Node A installs a transit MPLS forwarding entry to swap incoming label=1023 with outgoing labels, and the outgoing interface is determined by the SID-List for FEC1.

In this example, we assume that Node A receives two BGP/VPN routes:

* R1 with VPN label=V1, BGP next hop = 192.0.2.80, and color=100

* R2 with VPN label=V2, BGP next hop = 192.0.2.81, and color=100

We also assume that Node A has a BGP policy that matches color=100 and allows its usage as Service Level Agreement (SLA) steering information. In this case, Node A will install a VPN route with label stack = <S1,S2,V1> (corresponding to FEC1).

The rule described in Section 2.6 means that Node A MUST NOT install a VPN route with label stack = <S3,S4,V1> (corresponding to FEC2.)

Acknowledgements

The authors would like to thank Les Ginsberg, Chris Bowers, Himanshu Shah, Adrian Farrel, Alexander Vainshtein, Przemyslaw Krol, Darren Dukes, Zafar Ali, and Martin Vigoureux for their valuable comments on this document.

Contributors

The following contributors have substantially helped the definition and editing of the content of this document:

Martin Horneffer
Deutsche Telekom
Email: Martin.Horneffer@telekom.de

Wim Henderickx
Nokia
Email: wim.henderickx@nokia.com

Jeff Tantsura
Email: jefftant@gmail.com

Edward Crabbe
Email: edward.crabbe@gmail.com

Igor Milojevic
Email: milojevicigor@gmail.com

Saku Ytti
Email: saku@ytti.fi

Authors' Addresses

Ahmed Bashandy (editor)
Arrcus

Email: abashandy.ietf@gmail.com

Clarence Filsfils (editor)
Cisco Systems, Inc.
Brussels
Belgium

Email: cfilsfil@cisco.com

Stefano Previdi
Cisco Systems, Inc.
Italy

Email: stefano@previdi.net

Bruno Decraene
Orange
France

Email: bruno.decraene@orange.com

Stephane Litkowski
Orange
France

Email: slitkows.ietf@gmail.com

Rob Shakir
Google
United States of America

Email: robjs@google.com