NOKIA

7750 SERVICE ROUTER

7950 EXTENSIBLE ROUTING SYSTEM

SEGMENT ROUTING AND PCE USER GUIDE

RELEASE 21.5.R1

3HE 17731 AAAA TQZZA 01
Issue 01

June 2021

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

Table of Contents

1 Getting started

1.1 About this Guide

This guide describes the Nokia SR OS Segment Routing and PCE functionality.

This guide is organized into functional chapters that provide concepts and descriptions of the implementation flow.

The topics and commands described in this document apply to the following SR OS products:

- 7750 SR
- 7950 XRS

The following chassis types are available for the 7750 and 7950 SR OS routers:

- 7750 SR-12e
- 7750 SR-7/12
- 7750 SR-7s
- 7750 SR-14s
- 7750 SR-1
- 7750 SR-1s/2s
- 7950 XRS 20/20e

See the *SR OS 21.x.Rx Software Release Notes*, part number 3HE 17177 000*x* TQZZA, for a list of unsupported features by platform and chassis.

Command outputs shown in this guide are examples only; actual displays may differ depending on supported functionality and user configuration.

Note: The Segment Routing and PCE supports configuration using classic CLI and MD-CLI. This guide provides configuration examples based on classic CLI syntax only.

The SR OS CLI trees and command descriptions can be found in the following guides:

- *7450 ESS, 7750 SR, 7950 XRS, and VSR Classic CLI Command Reference Guide*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR MD-CLI Command Reference Guide*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Clear, Show, and Tools Command Reference Guide (for both MD-CLI and Classic CLI)*

Note: This guide generically covers Release 21.*x*.Rx content and may contain some content that is released in later maintenance loads. See the *SR OS 21.x.Rx Software Release Notes*, part number 3HE 17177 000*x* TQZZA, for information about features supported in each load of the Release 21.*x*.R*x* software.

2 Segment Routing with MPLS Data Plane (SR-MPLS)

This section describes:

- Segment Routing (SR) in the shortest path forwarding
- SR with Traffic Engineering (SR-TE)
- SR policies

2.1 Segment Routing in Shortest Path Forwarding

Segment routing adds to IS-IS and OSPF routing protocols the ability to perform shortest path routing and source routing using the concept of abstract segment. A segment can represent a local prefix of a node, a specific adjacency of the node (interface/next-hop), a service context, or a specific explicit path over the network. For each segment, the IGP advertises an identifier referred to as Segment ID (SID).

When segment routing is used together with MPLS data plane, the SID is a standard MPLS label. A router forwarding a packet using segment routing pushes one or more MPLS labels. This is the scope of the features described in this section.

Segment routing using MPLS labels can be used in both shortest path routing applications and in traffic engineering applications. This section focuses on the shortest path forwarding applications.

When a received IPv4 or IPv6 prefix SID is resolved, the Segment Routing module programs the Incoming Label Map (ILM) with a swap operation and also the LTN with a push operation both pointing to the primary/LFA NHLFE. An IPv4 or IPv6 SR tunnel to the prefix destination is also added to the TTM and is available for use by shortcut applications and L2/L3 services.

Segment routing introduces the remote LFA feature which expands the coverage of the LFA by computing and automatically programming SR tunnels which are used as backup next-hops. The SR shortcut tunnels terminate on a remote alternate node which provides loop-free forwarding for packets of the resolved prefixes. When the **loopfree-alternates** option is enabled in an IS-IS or OSPF instance, SR tunnels are protected with an LFA backup next-hop. If the prefix of a specific SR tunnel is not protected by the base LFA, the remote LFA automatically computes a backup next-hop using an SR tunnel if the **remote-lfa** option is also enabled in the IGP instance.

2.1.1 Configuring Segment Routing in Shortest Path

The user enables segment routing in an IGP routing instance using the following sequence of commands.

First, the user configures the global label block, referred to as Segment Routing Global Block (SRGB), which is reserved for assigning labels to segment routing prefix SIDs originated by this router. This range is carved from the system dynamic label range and is not instantiated by default:

```
config>router>mpls-labels>sr-labels start start-value end end-value
```

Next, the user enables the context to configure segment routing parameters within a given IGP instance:

```
config>router>isis>segment-routing
config>router>ospf>segment-routing
```

The key parameter is the configuration of the prefix SID index range and the offset label value which this IGP instance uses. Because each prefix SID represents a network global IP address, the SID index for a prefix must be unique network-wide. Thus, all routers in the network are expected to configure and advertise the same prefix SID index range for a given IGP instance. However, the label value used by each router to represent this prefix, that is, the label programmed in the ILM, can be local to that router by the use of an offset label, referred to as a start label:

Local Label (Prefix SID) = start-label + {SID index}

The label operation in the network becomes thus very similar to LDP when operating in the independent label distribution mode (RFC 5036) with the difference that the label value used to forward a packet to each downstream router is computed by the upstream router based on advertised prefix SID index using the above formula.

Figure 1: Packet Label Encapsulation using Segment Routing Tunnel shows an example of a router advertising its loopback address and the resulting packet label encapsulation throughout the network.

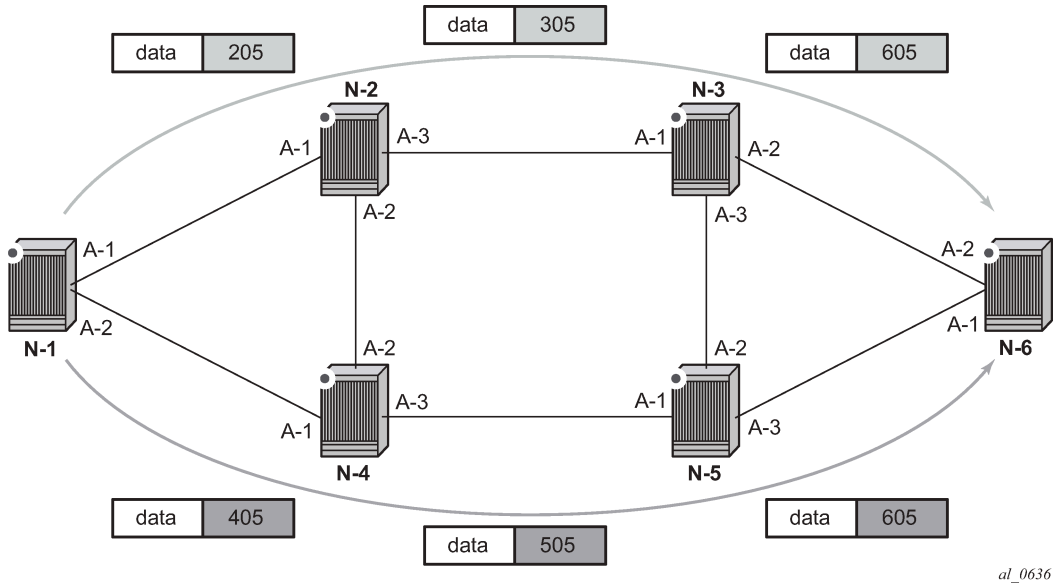

Figure 1: Packet Label Encapsulation using Segment Routing Tunnel

Router N-6 advertises loopback 10.10.10.1/32 with a prefix index of 5.Routers N-1 to N-6 are configured with the same SID index range of [1,100] and an offset label of 100 to 600 respectively. The following are the actual label values programmed by each router for the prefix of PE2:

- N-6 has a start label value of 600 and programs an ILM with label 605.
- N-3 has a start label of 300 and swaps incoming label 305 to label 605.
- N-2 has a start label of 200 and swaps incoming label 205 to label 305.

Similar operations are performed by N-4 and N-5 for the bottom path.

N-1 has an SR tunnel to N-6 with two ECMP paths. It pushes label 205 when forwarding an IP or service packet to N-6 via downstream next-hop N-2 and pushes label 405 when forwarding via downstream next-hop N-4.

The CLI for configuring the prefix SID index range and offset label value for a given IGP instance is as follows:

```
config>router>isis>segment-routing>prefix-sid-range {global | start-label label-value max-
index index-value}
config>router>ospf>segment-routing>prefix-sid-range {global | start-label label-value max-
index index-value}
```

There are two mutually-exclusive modes of operation for the prefix SID range on the router. In the global mode of operation, the user configures the global value and this IGP instance assumes the start label value as the lowest label value in the SRGB and the prefix SID index range size equal to the range size of the SRGB. After one IGP instance selected the **global** option for the prefix SID range, all IGP instances on the system are restricted to do the same.

The user must shutdown the segment routing context and delete the **prefix-sid-range** command in all IGP instances in order to change the SRGB. After the SRGB is changed, the user must re-enter the **prefix-sid-range** command again. The SRGB range change fails if an already allocated SID index/label goes out of range.

In the per-instance mode of operation, the user partitions the SRGB into non-overlapping sub-ranges among the IGP instances. The user thus configures a subset of the SRGB by specifying the start label value and the prefix SID index range size. All resulting net label values (start-label + index) must be within the SRGB or the configuration fails. Furthermore, the code checks for overlaps of the resulting net label value range across IGP instances and strictly enforces that these ranges do not overlap.

The user must shutdown the segment routing context of an IGP instance in order to change the SID index/label range of that IGP instance using the **prefix-sid-range** command. In addition, any range change fails if an already allocated SID index/label goes out of range.

The user can, however, change the SRGB on the fly as long as it does not reduce the current per-IGP instance SID index/label range defined with the **prefix-sid-range**. Otherwise, the user must shutdown the segment routing context of the IGP instance and delete and re-configure the **prefix-sid-range** command.

Finally, the user brings up segment routing on that IGP instances by un-shutting the context:

```
config>router>isis>segment-routing>no shutdown
config>router>ospf>segment-routing>no shutdown
```

This command fails if the user has not previously enabled the **router-capability** option in the IGP instance. Segment routing is a new capability and needs to be advertised to all routers in a given domain so that routers which support the capability only programs the node SID in the data path towards neighbors which support it.

```
config>router>isis>advertise-router-capability {area | as}
config>router>ospf>advertise-router-capability {link | area | as}
```

The IGP segment routing extensions are area-scoped. As a consequence, the user must configure the flooding scope to **area** in OSPF and to **area** or **as** in IS-IS, otherwise performing **no shutdown** of the segment-routing node fail.

Next, the user assigns a node SID index or label to the prefix representing the primary address of an IPv4 or IPv6 network interface of type **loopback** using one of the following commands:

```
config>router>isis>interface>ipv4-node-sid index value
config>router>ospf>interface>node-sid index value
config>router>isis>interface>ipv4-node-sid label value
config>router>ospf>interface>node-sid label value
config>router>isis>interface>ipv6-node-sid index value
config>router>isis>interface>ipv6-node-sid label value
```

Only a single node SID can be assigned to an interface. The secondary address of an IPv4 interface cannot be assigned a node SID index and does not inherit the SID of the primary IPv4 address. The same applies to the non-primary IPv6 addresses of an interface.

Above commands should fail if the network interface is not of type loopback or if the interface is defined in an IES or a VPRN context. Also, assigning the same SID index/label value to the same interface in two different IGP instances is not allowed within the same node.

Also, for OSPF the protocol version number and the instance number dictates if the node-SID index/label is for an IPv4 or IPv6 address of the interface. Specifically, the support of address families in OSPF is as follows:

- ospfv2: always IPv4 only

The value of the label or index SID is taken from the range configured for this IGP instance. When using the global mode of operation, a new segment routing module checks that the same index or label value cannot be assigned to more than one loopback interface address. When using the per-instance mode of operation, this check is not required because the index, and thus the label ranges, of the various IGP instances are not allowed to overlap.

For an individual adjacency, values for the label may be provisioned for an IS-IS or OSPF interface. If they are not provisioned, then they are dynamically allocated by the system from the dynamic label range. The following CLI commands are used:

```
config>router>isis>interface
   [no] ipv4-adjacency-sid label <value>
   [no] ipv6-adjacency-sid label <value>

config>router>ospf>area>interface
   [no] adjacency-sid label <value>
```

The *value* must correspond to a label in a reserved label block in provisioned mode referred to by the **srlb** command (see *Segment Routing Local Block* for more details of SRLBs).

A static label *value* for an adjacency SID is persistent. Therefore, the P-bit of the Flags field in the Adjacency-SID TLV advertised in the IGP is set to 1.

By default, a dynamic adjacency SID is advertised for an interface. However, if a static adjacency SID value is configured, then the dynamic adjacency SID is deleted and only the static adjacency SID used. Changing an adjacency SID from dynamic (for example, **no adjacency-sid**) to static, or vice versa, may result in traffic being dropped as the ILM is reprogrammed.

For a provisioned adjacency SID of an interface, a backup is calculated similar to a regular adjacency SID when **sid-protection** is enabled for that interface.

Provisioned adjacency SIDs are only supported on point-to-point interfaces.

2.1.2 Segment Routing Shortest in Path Forwarding with IS-IS

This section describes the segment routing shortest path forwarding with IS-IS.

2.1.2.1 IS-IS Control Protocol Changes

New TLV/sub-TLVs are defined in *draft-ietf-isis-segment-routing-extensions* and are supported in the implementation of segment routing in IS-IS. Specifically:

- the prefix SID sub-TLV
- the adjacency SID sub-TLV
- the SID/Label Binding TLV
- SR-Capabilities Sub-TLV
- SR-Algorithm Sub-TLV

This section describes the behaviors and limitations of the IS-IS support of segment routing TLV and sub-TLVs.

SR OS supports advertising the IS router capability TLV (RFC 4971) only for topology MT=0. As a result, the segment routing capability Sub-TLV can only be advertised in MT=0 which restricts the segment routing feature to MT=0.

Similarly, if prefix SID sub-TLVs for the same prefix are received in different MT numbers of the same IS-IS instance, then only the one in MT=0 is resolved. When the prefix SID index is also duplicated, an error is logged and a trap is generated, as explained in *Error and Resource Exhaustion Handling*.

I and V flags are both set to 1 when originating the SR capability sub-TLV to indicate support for processing both SR MPLS encapsulated IPv4 and IPv6 packets on its network interfaces. These flags are not checked when the sub-TLV is received. Only the SRGB range is processed.

The algorithm field is set to 0, meaning Shortest Path First (SPF) algorithm based on link metric, when originating the SR-Algorithm capability sub-TLV but is not checked when the sub-TLV is received.

Both IPv4 and IPv6 prefix and adjacency SID sub-TLVs originate within MT=0.

SR OS originates a single prefix SID sub-TLV per IS-IS IP reachability TLV and processes the first prefix SID sub-TLV only if multiple are received within the same IS-IS IP reachability TLV.

SR OS encodes the 32 bit index in the prefix SID sub-TLV. The 24 bit label is not supported.

SR OS originates a prefix SID sub-TLV with the following encoding of the flags and the following processing rules:

- The R-flag is set if the prefix SID sub-TLV, along with its corresponding IP reachability TLV, is propagated between levels. See below for more details about prefix propagation.
- The N-flag is always set because SR OS supports prefix SID of type node SID only.
- The P-Flag (no-PHP flag) is always set, meaning that the label for the prefix SID is pushed by the PHP router when forwarding to this router. The SR OS PHP router processes properly a received prefix

SID with the P-flag set to zero and uses implicit-null for the outgoing label towards the router which advertised it as long as the P-Flag is also set to 1.

- The E-flag (Explicit-Null flag) is always set to zero. An SR OS PHP router, however, processes properly a received prefix SID with the E-flag set to 1 and, when the P-flag is also set to 1, it pushes explicit-null for the outgoing label towards the router which advertised it.
- The V-flag is always set to 0 to indicate an index value for the SID.
- The L-flag is always set to 0 to indicate that the SID index value is not locally significant.
- The algorithm field is always set to zero to indicate Shortest Path First (SPF) algorithm based on link metric and is not checked on a received prefix SID sub-TLV.
- The SR OS still resolves a prefix SID sub-TLV received without the N-flag set but with the prefix length equal to 32. A trap, however, is raised by IS-IS.
- The SR OS does not resolve a prefix SID sub-TLV received with the N flag set and a prefix length different than 32. A trap is raised by IS-IS.
- The SR OS resolves a prefix SID received within a IP reachability TLV based on the following route preference:
  - SID received via L1 in a prefix SID sub-TLV part of IP reachability TLV
  - SID received via L2 in a prefix SID sub-TLV part of IP reachability TLV
- A prefix received in an IP reachability TLV is propagated, along with the prefix SID sub-TLV, by default from L1 to L2 by an L1L2 router. A router in L2 sets up an SR tunnel to the L1 router via the L1L2 router, which acts as an LSR.
- A prefix received in an IP reachability TLV is not propagated, along with the prefix SID sub-TLV, by default from L2 to L1 by an L1L2 router. If the user adds a policy to propagate the received prefix, then a router in L1 sets up an SR tunnel to the L2 router via the L1L2 router, which acts as an LSR.
- If a prefix is summarized by an ABR, the prefix SID sub-TLV is not propagated with the summarized route between levels. To propagate the node SID for a /32 prefix, route summarization must be disabled.
- SR OS propagates the prefix SID sub-TLV when exporting the prefix to another IS-IS instance; however, it does not propagate it if the prefix is exported from a different protocol. Thus, when the corresponding prefix is redistributed from another protocol such as OSPF, the prefix SID is removed.

SR OS originates an adjacency SID sub-TLV with the following encoding of the flags:

- the F-flag is set to zero to indicate the IPv4 family and is set to 1 to indicate an IPv6 family for the adjacency encapsulation
- the B-Flag is set to zero and is not processed on receipt
- the V-flag is always set to 1
- the L-flag is always set to 1
- the S-flag is set to zero as assigning adjacency SID to parallel links between neighbors is not supported. An adjacency received SID with S-Flag set is not processed.
- the weight octet is not supported and is set to all zeros

SR OS can originate the SID/Label Binding TLV as part of the Mapping Server feature (see *Segment Routing Mapping Server Prefix SID Resolution* for more information). It can process it properly if received. The following rules and limitations should be considered.

- Only the Mapping Server Prefix-SID Sub-TLV within the TLV is processed and the ILMs installed if the prefixes in the provided range are resolved.
- The range and FEC prefix fields are processed. Each FEC prefix is resolved normally, as for the prefix SID sub-TLV, meaning there must be an IP Reachability TLV received for the exact matching prefix.

- If the same prefix is advertised with both a prefix SID sub-TLV and a mapping server Prefix-SID sub-TLV. The resolution follows the following route preference:
  - SID received via L1 in a prefix SID sub-TLV part of IP reachability TLV
  - SID received via L2 in a prefix SID sub-TLV part of IP reachability TLV
  - SID received via L1 in a mapping server Prefix-SID sub-TLV
  - SID received via L2 in a mapping server Prefix-SID sub-TLV
- The entire TLV can be propagated between levels based on the settings of the S-flag. The TLV cannot be propagated between IS-IS instances (see *Segment Routing Mapping Server Prefix SID Resolution* for more information). Finally, an L1L2 router does not propagate the prefix-SID sub-TLV from the SID/Label binding TLV (received from a mapping server) into the IP Reachability TLV if the latter is propagated between levels.
- The mapping server which advertised the SID/Label Binding TLV does not need to be in the shortest path for the FEC prefix.
- If the same FEC prefix is advertised in multiple binding TLVs by different routers, the SID in the binding TLV of the first router which is reachable is used. If that router becomes unreachable, the next reachable one is used.
- No check is performed if the content of the binding TLVs from different mapping servers are consistent or not.
- Any other sub-TLV, for example, the SID/Label Sub-TLV, ERO metric and unnumbered interface ID ERO, is ignored but the user can get a dump of the octets of the received but not-supported sub-TLVs using the existing IGP **show** command.

2.1.2.2 Announcing ELC, MSD-ERLD, and MSD-BMI with IS-IS

IS-IS has the ability to announce node Entropy Label Capability (ELC), the Maximum Segment Depth (MSD) for node Entropy Readable Label Depth (ERLD) and the Maximum Segment Depth (MSD) for node Base MPLS Imposition (BMI). If needed, exporting the IS-IS extensions into BGP-LS requires no additional configuration. These extensions are standardized through draft-ietf-isis-mpls-elc-10, *Signaling Entropy Label Capability and Entropy Readable Label Depth Using IS-IS*, and RFC 8491, *Signaling Maximum SID Depth (MSD) Using IS-IS*.

The ELC, ERLD, and BMI IS-IS values are announced automatically when ISIS prefix attributes and router capabilities are announced and when entropy and segment routing is enabled on the router. The following configuration logic is used.

- ELC is automatically announced for host prefixes associated with an IPv4 or IPv6 node SID, when **segment-routing** and **segment-routing entropy-label** and **prefix-attributes-tlv** are enabled for IS-IS. Although the ELC capability is a node property, it is assigned to prefixes to allow inter-area or inter-as signaling. Consequently, the prefix-attribute TLV must be enabled accordingly within IS-IS.
- The router maximum node ERLD is announced for IS-IS when **segment-routing** and **segment-routing entropy-label** is enabled together with **advertise-router-capability**.
- The router maximum node MSD-BMI for IS-IS is announced when **segment-routing** and **advertise-router-capability** are enabled.
- Exporting ELC, MSD-ERLD, and MSD-BMI IS-IS extensions into BGP-LS encoding is enabled automatically when database-export for BGP-LS is configured.
- The announced value for maximum node MSD-ERLD and MSD-BMI can be modified to a smaller number using the **override-bmi** and **override-erld** commands. This can be useful when services (such as EVPN) or more complex link protocols (such as Q-in-Q) are deployed. Provisioning correct ERLD

and BMI values help controllers and local-cspf to construct valid segment routing label stacks to be deployed in the network.

Segment routing parameters are configured in the following contexts:

configure>router>isis>segment-routing>maximum-sid-depth

configure>router>isis>segment-routing>maximum-sid-depth>**override-bmi** *value*

configure>router>isis>segment-routing>maximum-sid-depth>**override-erld** *value*

2.1.2.3 Entropy Label for IS-IS Segment Routing

The router supports the MPLS entropy label, as specified in RFC 6790, on IS-IS segment-routed tunnels. LSR nodes in a network can load-balance labeled packets in a more granular way than by hashing on the standard label stack. Refer to the *MPLS Guide* for more information.

Announcing of Entropy Label Capability (ELC) is supported, however processing of Entropy Label Capability (ELC) signaling is not supported for IS-IS segment-routed tunnels. Instead, ELC is configured at the head end LER using the **configure router isis entropy-label override-tunnel-elc** command. This command causes the router to ignore any advertisements for ELC that may or may not be received from the network, and instead to assume that the whole domain supports entropy labels.

2.1.2.4 IPv6 Segment Routing using MPLS Encapsulation

This feature implements support for SR IPv6 tunnels in IS-IS MT=0. The user can configure a node SID for the primary IPv6 global address of a loopback interface, which then gets advertised in IS-IS. IS-IS automatically assigns and advertises an adjacency SID for each adjacency with an IPv6 neighbor. After the node SID is resolved, it is used to install an IPv6 SR-ISIS tunnel in the TTM for use by the services.

2.1.2.4.1 IS-IS MT=0 Extensions

The IS-IS MT=0 extensions consist of supporting the advertising and resolution of the prefix SID sub-TLV within the IP Reach TLV-236 (IPv6), which is defined in RFC 5308.The adjacency SID is still advertised as a sub-TLV of the Extended IS Reachability TLV 22, as defined in RFC 5305, *IS-IS Extensions for Traffic Engineering*, as in the case of an IPv4 adjacency. The router sets the V-Flag and I-Flag in the SR-Capabilities Sub-TLV to indicate that it is capable of processing SR MPLS encapsulated IPv4 and IPv6 packets on its network interfaces.

2.1.2.4.2 Service and Forwarding Contexts Supported

The service and forwarding contexts supported with the SR-ISIS IPv6 tunnels are:

- SDP of type **sr-isis** with **far-end** option using IPv6 address
- VLL, VPLS, IES/VPRN spoke-interface, R-VPLS
- Support of PW redundancy within Epipe/Ipipe VLL, Epipe spoke termination on VPLS and R-VPLS, and Epipe/Ipipe spoke termination on IES/VPRN
- IPv6 static route resolution to indirect next-hop using Segment Routing IPv6 tunnel
- Remote mirroring and L3 encap LI

2.1.2.4.3 Services Using SDP with a SR IPv6 Tunnel

The MPLS SDP of type **sr-isis** with a **far-end** option using an IPv6 address is supported. Note the SDP must have the same IPv6 **far-end** address, used by the control plane for the T-LDP session, as the prefix of the node SID of the SR IPv6 tunnel.

```
configure
        – service
            – [no] sdp sdp-id mpls
                – [no] far-end ipv6-address
                – sr-isis
                – no sr-isis
```

The **bgp-tunnel**, **lsp**, **sr-te lsp**, **sr-ospf**, and **mixed-lsp-mode** commands are blocked within the SDP configuration context when the far-end is an IPv6 address.

SDP admin groups are not supported with an SDP using an SR IPv6 tunnel, or with SR-OSPF for IPv6 tunnels, and the attempt to assign them is blocked in the CLI.

Services that use LDP control plane such as T-LDP VPLS and R-VPLS, VLL, and IES/VPRN spoke interface have the spoke SDP (PW) signaled with an IPv6 T-LDP session because the **far-end** option is configured to an IPv6 address. The spoke SDP for these services binds to an SDP that uses an SR IPv6 tunnel where the prefix matches the **far-end** address. SR OS also supports the following:

- the IPv6 PW control word with both data plane packets and VCCV OAM packets
- Hash Label and Entropy Label, with the above services
- network domains in VPLS

The PW switching feature is not supported with LDP IPv6 control planes. As a result, the CLI does not allow the user to enable the **vc-switching** option whenever one or both spoke SDPs uses an SDP that has the **far-end** configured as an IPv6 address.

L2 services that use BGP control plane such as dynamic MS-PW, BGP-AD VPLS, BGP-VPLS, BGP-VPWS, and EVPN MPLS cannot bind to an SR IPv6 tunnel because a BGP session to a BGP IPv6 peer does not support advertising an IPv6 next-hop for the L2 NLRI. As a result, these services do not auto-generate SDPs using an SR IPv6 tunnel. In addition, they skip any provisioned SDPs with **far-end** configured to an IPv6 address when the **use-provisioned-sdp** option is enabled.

SR OS also supports multi-homing with T-LDP active/standby FEC 128 spoke SDP using SR IPv6 tunnel to a VPLS/B-VPLS instance. BGP multi-homing is not supported because BGP IPv6 does not support signaling an IPv6 next-hop for the L2 NLRI.

The Shortest Path Bridging (SPB) feature works with spoke SDPs bound to an SDP that uses an SR IPv6 tunnel.

2.1.2.5 Segment Routing Mapping Server Function for IPv4 Prefixes

The mapping server feature allows the configuration and advertisement via IS-IS of the node SID index for prefixes of routers which are in the LDP domain. This is performed in the router acting as a mapping server and using a prefix-SID sub-TLV within the SID/Label binding TLV in IS-IS.

The user configures the SR mapping database in IS-IS using the following CLI command:

```
configure
        – router
            – [no] isis
                – segment-routing
                – no segment-routing
                    – mapping-server
                        – sid-map node-sid {index 0..4294967295 [range  0..65535]} prefix {{ip-
address/mask} | {ip-address}{netmask}} [set-flags {s}] [level {1|2|1/2}]
                        – no sid-map node-sid index 0..4294967295
```

The user can enter the node SID index for one prefix or a range of prefixes by specifying the first index value and, optionally, a range value. The default value for the range option is 1. Only the first prefix in a consecutive range of prefixes must be entered. The user can enter the first prefix with a mask lower than 32 and the SID/Label Binding TLV is advertised but the routers do not resolve these prefix SIDs and instead originates a trap.

The **no** form of the **sid-map** command deletes the range of node SIDs beginning with the specified index value. The **no** form of the **mapping-server** command deletes all node SID entries in the IS-IS instance.

By setting the S-flag, the user can indicate to the IS-IS routers in the rest of the network that the flooding scope of the SID/Label binding TLV is the entire domain. In that case, a router receiving the TLV advertisement should leak it between IS-IS levels. If leaked from level 2 to level 1, the D-flag must be set, after which the TLV cannot be leaked back into level 2. Otherwise, the S-flag is clear by default and the TLV must not be leaked by routers receiving the mapping server advertisement.

Note: SR OS does not leak this TLV between IS-IS instances and does not support the multi-topology SID/Label Binding TLV format. In addition, the user can specify the mapping server's flooding scope for the generated SID/Label binding TLV using the **level** option. This option allows further narrowing of the flooding scope configured under the router IS-IS level-capability for a one or more SID/Label binding TLVs if required. The default flooding scope of the mapping server is L1/L2, which can be narrowed by what is configured under the router IS-IS level-capability.

The A-flag is used to indicate that a prefix for which the mapping server prefix SID is advertised is directly attached. The M-flag is used to advertise a SID for a mirroring context to provide protection against the failure of a service node. None of these flags are supported on the mapping server; the mapping client ignores them.

Each time a prefix or a range of prefixes is configured in the SR mapping database in any routing instance, the router issues for this prefix, or range of prefixes, a prefix-SID sub-TLV within a IS-IS SID/label binding TLV in that instance. The flooding scope of the TLV from the mapping server is determined as explained above. No further check of the reachability of that prefix in the mapping server route table is performed and no check if the SID index is duplicate with some existing prefix in the local IGP instance database or if the SID index is out of range with the local SRGB.

2.1.2.5.1 Segment Routing Mapping Server Prefix SID Resolution

- IP Prefix Resolution
  - SPF calculates the next-hops, up to **max-ecmp**, to reach a destination node.
  - Each prefix inherits the next-hops of one or more destination nodes advertising it.
  - A prefix advertised by multiple nodes, all reachable with same cost, inherits up to **max-ecmp** next-hops from the advertising nodes.
  - The next-hop selection, up to **max-ecmp**, value is deterministic and is based on sorting the next-hops by:

    1. lowest next-hop router-id
    2. lowest interface index, for parallel links to same router-id

    Each next-hop keeps a reference to the destination nodes of whom it was inherited.
- Prefix SID Resolution
  - For a specified prefix, IGP selects the SID value among multiple advertised values respecting the following preference order:

    1. the local intra-area SID owned by this router
    2. the prefix SID sub-TLV advertised within an IP Reach TLV

       If multiple SIDs exist, select the SID corresponding to the destination router or the ABR with the lowest system ID that is reachable using the first next-hop of the prefix.
    3. the IS-IS SID and label binding TLV from the mapping server

       If multiple SIDs exist, select the following the preference rules in *draft-ietf-spring-conflict-resolution-05 [sid-conflict-resolution]* when applied to the SRMS entries of the conflicting SIDs. The order of these rules is as follows:

       - smallest range

       - smallest starting address

       - smallest algorithm

       - smallest starting SID

    Note: If an L1L2 router acts as a mapping server and also re-advertises the mapping server prefix SID from other mapping servers, the redistributed mapping server prefix SID is preferred by other routers resolving the prefix, which may result in not selecting the mapping server respecting these rules.

  - The selected SID is used with all ECMP next-hops step (I) towards all destination nodes or ABR nodes which advertised the prefix.
  - If duplicate prefix SIDs exist for different prefixes after the above steps, the first SID that is processed is programmed for its corresponding prefix. Subsequent SIDs cause a duplicate SID trap and are not programmed. The corresponding prefixes themselves are still resolved and programmed normally using IP next-next-hops.

- SR Tunnel Programming
  - If the prefix SID is resolved from a prefix SID sub-TLV advertised within an IP Reach TLV, one of the following applies.
    1. The SR ILM label is swapped to a SR NHLFE label as in regular SR tunnel resolution when the next-hop of the ISIS prefix is SR enabled.
    2. The SR ILM label is stitched to an LDP FEC of the same prefix when either the next-hop of the ISIS prefix is not SR enabled (no SR NHLFE) or an import policy rejects the prefix (SR NHLFE deprogrammed).

       The LDP FEC could also be resolved using the same or a different IGP instance as that of the prefix SID sub-TLV or using a static route.
  - If the prefix SID is resolved from a mapping server advertisement, one of the following applies.
    1. The SR ILM label is stitched to an LDP FEC of the same prefix, if one exists. The stitching is performed even if an import policy rejects the prefix in the local ISIS instance.

       The LDP FEC could also be resolved using a static route, a route within an IS-IS instance, or a route within an OSPF instance. The latter two can be the same as, or different from the IGP instance that advertised the mapping server prefix SID sub-TLV.
    2. Otherwise, the SR ILM label is swapped to a SR NHLFE label. This is only possible if a route is exported from another IGP instance into the local IGP instance without propagating the prefix SID sub-TLV with the route. Otherwise, the SR ILM label is swapped to a SR NHLFE label towards the stitching node.

2.1.3 Segment Routing Shortest in Path Forwarding with OSPF

This section describes the segment routing shortest path forwarding with OSPF.

2.1.3.1 OSPFv2 Control Protocol Changes

New TLV/sub-TLVs are defined in *draft-ietf-ospf-segment-routing-extensions-04* and are required for the implementation of segment routing in OSPF. Specifically:

- the prefix SID sub-TLV part of the OSPFv2 Extended Prefix TLV
- the prefix SID sub-TLVpart of the OSPFv2 Extended Prefix Range TLV
- the adjacency SID sub-TLV part of the OSPFv2 Extended Link TLV
- SID/Label Range Capability TLV
- SR-Algorithm Capability TLV

This section describes the behaviors and limitations of the OSPF support of segment routing TLV and sub-TLVs.

SR OS originates a single prefix SID sub-TLV per OSPFv2 Extended Prefix TLV and processes the first one only if multiple prefix SID sub-TLVs are received within the same OSPFv2 Extended Prefix TLV.

SR OS encodes the 32-bit index in the prefix SID sub-TLV. The 24-bit label or variable IPv6 SID is not supported.

SR OS originates a prefix SID sub-TLV with the following encoding of the flags.

- The NP-Flag is always set, meaning that the label for the prefix SID is pushed by the PHP router when forwarding to this router. SR OS PHP routers properly processes a received prefix SID with the NP-flag set to zero and uses implicit-null for the outgoing label towards the router which advertised it.
- The M-Flag is always unset because SR OS does not support originating a mapping server prefix-SID sub-TLV.
- The E-flag is always set to zero. SR OS PHP routers properly processes a received prefix SID with the E-flag set to 1, and when the NP-flag is also set to 1 it pushes explicit-null for the outgoing label towards the router which advertised it.
- The V-flag is always set to 0 to indicate an index value for the SID.
- The L-flag is always set to 0 to indicate that the SID index value is not locally significant.
- The algorithm field is always set to zero to indicate Shortest Path First (SPF) algorithm based on link metric and is not checked on a received prefix SID sub-TLV.

SR OS resolves a prefix SID received within an Extended Prefix TLV based on the following route preference:

- SID received via an intra-area route in a prefix SID sub-TLV part of Extended Prefix TLV
- SID received via an inter-area route in a prefix SID sub-TLV part of Extended Prefix TLV

SR OS originates an adjacency SID sub-TLV with the following encoding of the flags.

- The F-flag is unset to indicate the Adjacency SID refers to an adjacency with outgoing IPv4 encapsulation.
- The B-flag is set to zero and is not processed on receipt.
- The V-flag is always set.
- The L-flag is always set.
- The S-flag is not supported.
- The weight octet is not supported and is set to all zeros.

An adjacency SID is assigned to next hops over both the primary and secondary interfaces.

SR OS can originate the OSPFv2 Extended Prefix Range TLV as part of the Mapping Server feature and can process it properly if received. The following rules and limitations should be considered.

- Only the prefix SID sub-TLV within the TLV is processed and the ILMs installed if the prefixes are resolved.
- The range and address prefix fields are processed. Each prefix is resolved separately.
- If the same prefix is advertised with both a prefix SID sub-TLV in a IP reachability TLV and a mapping server Prefix-SID sub-TLV, the resolution follows the following route preference:
  - the SID received via an intra-area route in a prefix SID sub-TLV part of Extended Prefix TLV
  - the SID received via an inter-area route in a prefix SID sub-TLV part of Extended Prefix TLV
  - the SID received via an intra-area route in a prefix SID sub-TLV part of a OSPFv2 Extended Range Prefix TLV
  - the SID received via an inter-area route in a prefix SID sub-TLV part of a OSPFv2 Extended Range Prefix TLV
- No leaking of the entire TLV is performed between areas. Also, an ABR will not propagate the prefix-SID sub-TLV from the Extended Prefix Range TLV (received from a mapping server) into an Extended Prefix TLV if the latter is propagated between areas.

- The mapping server which advertised the OSPFv2 Extended Prefix Range TLV does not need to be in the shortest path for the FEC prefix.
- If the same FEC prefix is advertised in multiple OSPFv2 Extended Prefix Range TLVs by different routers, the SID in the TLV of the first router which is reachable is used. If that router becomes unreachable, the next reachable one is used.
- No check is performed to determine whether or not the contents of the OSPFv2 Extended Prefix Range TLVs received from different mapping servers are consistent.
- Any other sub-TLV, for example, the ERO metric and unnumbered interface ID ERO, is ignored but the user can get a dump of the octets of the received but not-supported sub-TLVs using the existing IGP **show** command.

SR OS supports propagation on ABR of external prefix LSA into other areas with routeType set to 3 as per *draft-ietf-ospf-segment-routing-extensions-04*.

SR OS supports propagation on ABR of external prefix LSA with route type 7 from NSSA area into other areas with route type set to 5 as per *draft-ietf-ospf-segment-routing-extensions-04*. SR OS does not support propagating of the prefix SID sub-TLV between OSPF instances.

When the user configures an OSPF import policy, the outcome of the policy applies to prefixes resolved in RTM and the corresponding tunnels in TTM. So, a prefix removed by the policy will not appear as both a route in RTM and as an SR tunnel in TTM.

2.1.3.2 OSPFv3 Control Protocol Changes

The OSPFv3 extensions consist of support for the following TLVs:

- a prefix SID that is a sub-TLV of the OSPFv3 prefix TLV. The OSPFv3 prefix TLV is a new top-level TLV of the extended prefix LSA introduced in *draft-ietf-ospf-ospfv3-lsa-extend*. The OSPFv3 instance can operate in either LSA sparse mode or extended LSA mode.

  The **config**>**router**>**extended-lsa only** command advertises the prefix SID sub-TLV in the extended LSA format in both cases.
- an adjacency SID that is a sub-TLV of the OSPFv3 router-link TLV. The OSPFv3 router-link TLV is a new top-level TLV in the extended router LSA introduced in *draft-ietf-ospf-ospfv3-lsa-extend*. The OSPFv3 instance can operate in either LSA sparse mode or extended LSA mode. The **config**>**router**>**extended-lsa only** command advertises the adjacency SID sub-TLV in the extended LSA format in both cases.
- the SR-Algorithm TLV and the SID/Label Range TLV. Both of these TLVs are part of the TLV-based OSPFv3 Router Information opaque LSA defined in RFC 7770.

2.1.3.3 Announcing ELC, MSD-ERLD and MSD-BMI with OSPF

OSPF has the ability to announce node Entropy Label Capability (ELC), the Maximum Segment Depth (MSD) for node Entropy Readable Label Depth (ERLD), and the Maximum Segment Depth (MSD) for node Base MPLS Imposition (BMI). If needed, exporting these OSPF extensions into BGP-LS requires no additional configuration. These extensions are standardized through draft-ietf-ospf-mpls-elc-12, *Signaling Entropy Label Capability and Entropy Readable Label-stack Depth Using OSPF*, and RFC 8476, *Signaling Maximum SID Depth (MSD) Using OSPF*.

The ELC, ERLD, and BMI OSPF values are announced automatically when entropy and segment routing is enabled on the router. The following configuration logic is used:

- ELC is automatically announced for host prefixes associated with a node SID when **segment-routing** and **segment-routing entropy-label** are enabled for OSPF.
- The router maximum node ERLD is announced for OSPF when **segment-routing** and **segment-routing entropy-label** is enabled together with **advertise-router-capability**.
- The router maximum node MSD-BMI for OSPF is announced when **segment-routing advertise-router-capability** are enabled.
- Exporting ELC, MSD-ERLD and MSD-BMI OSPF extensions into BGP-LS encoding occurs automatically when **database-export** for BGP-LS is configured.
- The announced value for maximum node MSD-ERLD and MSD-BMI can be modified to a smaller number using the **override-bmi** and **override-erld** commands. This can be useful when services (such as EVPN) or more complex link protocols (such as Q-in-Q) are deployed. Provisioning correct ERLD and BMI values help controllers and **local-cspf** to construct valid segment routing label stacks to be deployed in the network.

Segment routing parameters are configured in the following context:

configure>router>ospf>segment-routing

configure>router>ospf>segment-routing>**override-bmi** *value*

configure>router>ospf>segment-routing>**override-erld** *value*

2.1.3.4 Entropy Label for OSPF Segment Routing

The router supports the MPLS entropy label, as specified in RFC 6790, on OSPF segment-routed tunnels. LSR nodes in a network can load-balance labeled packets in a more granular way than by hashing on the standard label stack. Refer to the *MPLS Guide* for more information.

Announcing of Entropy Label Capability (ELC) is supported, however, processing of ELC signaling is not supported for OSPF segment-routed tunnels. Instead, ELC is configured at the head end LER using the **configure router ospf entropy-label override-tunnel-elc** command. This command causes the router to ignore any advertisements for ELC that may or may not be received from the network, and instead to assume that the whole domain supports entropy labels.

2.1.3.5 IPv6 Segment Routing using MPLS Encapsulation in OSPFv3

This feature implements support for SR IPv6 tunnels in OSPFv3 instances 0 to 31. The user can configure a node SID for the primary IPv6 global address of a loopback interface, which then gets advertised in OSPFv3. OSPFv3 automatically assigns and advertises an adjacency SID for each adjacency with an IPv6 neighbor. After the node SID is resolved, it is used to install an IPv6 SR-OSPF3 tunnel in the TTMv6 for use by the routes and services.

2.1.3.6 Segment Routing Mapping Server for IPv4 Prefixes

The mapping server feature supports the configuration and advertisement, in OSPF, of the node SID index for prefixes of routers which are in the LDP domain. This is performed in the router acting as a mapping server and using a prefix-SID sub-TLV within an OSPF Extended Prefix Range TLV.

Use the following command syntax to configure the SR mapping database in OSPF:

```
configure
        – router
            – [no]ospf
                – segment-routing
                – no segment-routing
                    – mapping-server
                        – sid-map node-sid {index value <0..4294967295> [range value
 <1..65535>]} prefix {{ip-address/mask}|{netmask}}[scope {area area-id | as}]
                        – no sid-map node-sid index value
```

The user enters the node SID index, for one prefix or a range of prefixes, by specifying the first index value and, optionally, a range value. The default value for the **range** option is 1. Only the first prefix in a consecutive range of prefixes must be entered. If the user enters the first prefix with a mask lower than 32, the OSPF Extended Prefix Range TLV is advertised but a router that receives does not resolve SID and instead originates a trap.

The **no** form of the **sid-map** command deletes the range of node SIDs beginning with the specified index value. The **no** form of the **mapping-server** command deletes all node SID entries in the OSPF instance.

Use the **scope** option to specify the mapping server's own flooding scope for the generated OSPF Extended Prefix Range TLV. There is no default value. If the scope is a specific area, then the TLV is flooded only in that area.

An ABR that propagates an intra-area OSPF Extended Prefix Range TLV flooded by the mapping server in that area into other areas sets the inter-area flag (IA-flag). The ABR also propagates the TLV if received with the inter-area flag set from other ABR nodes but only from the backbone to leaf areas and not vice-versa. However, if the identical TLV was advertised as an intra-area TLV in a leaf area, the ABR will not flood the inter-area TLV into that leaf area.

Note: SR OS does not leak the OSPF Extended Prefix Range TLV between OSPF instances.

Each time a prefix or a range of prefixes is configured in the SR mapping database in any routing instance, the router issues for this prefix, or range of prefixes, a prefix-SID sub-TLV within a OSPF Extended Prefix Range TLV in that instance. The flooding scope of the TLV from the mapping server is determined as previously explained. No further check of the reachability of that prefix in the mapping server route table is performed. Additionally, no check is performed if the SID index is a duplicate of an existing prefix in the local IGP instance database or if the SID index is out of range with the local SRGB.

2.1.3.6.1 Segment Routing Mapping Server Prefix SID Resolution

The rules for IP prefix resolution, prefix SID resolution, and SR tunnel programing are as follows:

- IP Prefix Resolution
  - SPF calculates the next hops, up to **max-ecmp**, to reach a destination node
  - Each prefix inherits the next hops of one or more destination nodes advertising it
  - A prefix advertised by multiple nodes, all reachable with same cost, will inherit up to max-ecmp next hops from the advertising nodes
  - Next hop selection, up to **max-ecmp** value, is deterministic and is based on sorting the next hops by:

    1. lowest next hop router-id
    2. lowest interface index, for parallel links to same router-id

    Each next hop keeps a reference to the destination nodes of whom it was inherited
- Prefix SID Resolution
  - For a given prefix, IGP selects the SID value among multiple advertised values respecting the following preference order:

    1. local intra-area SID owned by this router
    2. prefix SID sub-TLV advertised within a OSPF Extended Prefix TLV

    If multiple SIDs exist, select the SID corresponding to the destination router or ABR with the lowest OSPF Router-ID which is reachable via the first next hop of the prefix
    3. OSPF Extended Prefix Range TLV from mapping server

    If multiple SIDs exist, select the following, using the preference rules in *draft-ietf-spring-conflict-resolution-05* when applied to the SRMS entries of the conflicting SIDs. The order of these rules is as follows:

    - smallest range

    - smallest starting address

    - smallest algorithm

    - smallest starting SID
  - The selected SID is used with all ECMP next hops from step (I) towards all destination nodes or ABR nodes which advertised the prefix.
  - If duplicate prefix SIDs exist for different prefixes after above steps, the first SID which is processed is programmed for its corresponding prefix.

    Subsequent SIDs causes a duplicate SID trap and are not programmed. The corresponding prefixes themselves are still resolved normally using IP next-hops.

- SR Tunnel Programming
  - If the prefix SID is resolved from a prefix SID sub-TLV advertised within an OSPF Extended Prefix TLV, one of the following applies.
    1. The SR ILM label is swapped to an SR NHLFE label as in regular SR tunnel resolution when the next-hop of the OSPF prefix is SR-enabled.
    2. The SR ILM label is stitched to an LDP FEC of the same prefix when either the next-hop of the OSPF prefix is not SR enabled (no SR NHLFE) or an import policy rejects the prefix (SR NHLFE deprogrammed).

       The LDP FEC could also be resolved using the same or a different IGP instance as that of the prefix SID sub-TLV or using a static route.
  - If the prefix SID is resolved from a mapping server advertisement, one of the following applies.
    1. The SR ILM label is stitched to an LDP FEC of the same prefix, if one exists. The stitching is performed even if an import policy rejects the prefix in the local OSPF instance. The LDP FEC could also be resolved using a static route, a route within an IS-IS instance, or a route within an OSPF instance. The latter two can be the same as, or different from the IGP instance that advertised the mapping server prefix SID sub-TLV.
    2. The SR ILM label is swapped to an SR NHLFE label toward the stitching node.

2.1.4 Segment Routing with BGP

Segment routing allows a router, potentially by action of an SDN controller, to source route a packet by prepending a segment router header containing an ordered list of segment identifiers (SIDs). Each SID can be viewed as some sort of topological or service-based instruction. A SID can have a local meaning to one particular node or it can have a global meaning within the SR domain, such as the instruction to forward the packet on the ECMP-aware shortest path to reach some prefix P. With SR-MPLS each SID is an MPLS label and the complete SID list is a stack of labels in the MPLS header.

For all the routers in a network domain to have a common understanding of the meaning of a topology SID, the association of the SID with an IP prefix must be propagated by a routing protocol. Traditionally this is done by an IGP protocol, however, in some cases the meaning of a SID may need to be carried across network boundaries that extend beyond IGP protocol boundaries. For these cases, it is possible for BGP to carry the association of an SR-MPLS SID with an IP prefix. This is possible by attaching a prefix-SID BGP path attribute to an IP route belonging to a labeled-unicast address family. The prefix SID attribute attached to a labeled-unicast route for prefix P advertises a SID corresponding to the network-wide instruction to forward the packet along the ECMP-aware BGP-computed best path or paths to reach P. The prefix-SID attribute is an optional, transitive BGP path attribute with type code 40. This attribute encodes a 32-bit label-index (into the SRGB space) and may also provide details about the SRGB space of the originating router. The encoding of this BGP path attribute and its semantics are further described in *draft-ietf-idr-bgp-prefix-sid*.

An SR OS router with upgraded software that understands the prefix SID attribute can prevent it from propagating outside the segment routing domain where it is applicable, using the **block-prefix-sid** command. This BGP command removes the prefix SID attribute from all routes sent and received to and from the IBGP and EBGP peers included in the scope of the command. By default, the attribute propagates without restriction.

SR OS attaches a meaning to a prefix SID attribute only when it is attached to routes belonging to the labeled-unicast IPv4 and labeled-unicast IPv6 address families. It has no meaning when attached to other

types of routes. When attached to routes of unsupported address families the prefix SID attribute is ignored but still propagated, as with any other optional transitive attribute.

Segment routing must be administratively enabled under BGP using the following command: **config**>**router**>**bgp**>**segment-routing**>**no shutdown** for any of the following to be possible:

- For BGP to redistribute a static or IGP route for a /32 IPv4 prefix as a label-ipv4 route, or a /128 IPv6 prefix as a label-ipv6, with a prefix SID attribute, a **route-table-import** policy with an **sr-label-index** action is required.
- For BGP to add or modify the prefix SID attribute in a received label-ipv4 or label-ipv6 route, a BGP **import** policy with an **sr-label-index** action is required.
- For BGP to advertise a label-ipv4 or label-ipv6 route with an incoming datapath label based on the attached prefix SID attribute when BGP segment-routing is disabled, new label values assigned to label-ipv4 or label-ipv6 routes come from the router's dynamic label range and carry no network-wide meaning.

To enable BGP segment routing, the base router BGP instance must be associated with a **prefix-sid-range**. This command tells BGP which SRGB label block it is allowed to use (for example, to allocate labels from) and to advertise in the Originator SRGB TLV of the prefix SID attribute. The **global** parameter value indicates that BGP should use the SRGB as configured under **config**>**router**>**mpls-labels**>**sr-labels**. The **start-label** and **max-index** parameters are used to restrict the BGP prefix SID label range to a subset of the global SRGB.

Note: The **start-label** and **max-index** must reside within the global SRGB or the command fails.

This is useful in cases where partitioning of the SRGB into non-overlapping subranges dedicated to different IGP/BGP protocol instances is desired. Segment routing under BGP must be shutdown before any changes can be made to the **prefix-sid-range** command.

A unique label-index value should be assigned to each different IPv4 or IPv6 prefix that is advertised with a BGP prefix SID. If label-index N1 is assigned to a BGP-advertised prefix P1, and N1 plus the SRGB start-label creates a label value that conflicts with another SR programmed LFIB entry, then the conflict situation is addressed as follows:

1. If the conflict is with another BGP route for prefix P2 that was advertised with a prefix SID attribute, all the conflicting BGP routes (for P1 and P2) are advertised with a normal BGP-LU label from the dynamic label range.
2. If the conflict is with an IGP route, and BGP is not trying to redistribute that IGP route as a label-ipv4 or label-ipv6 route with a route-table-import policy action that uses the **prefer-igp** keyword in the **sr-label-index** command, the BGP route loses to the IGP route and it is advertised with a normal BGP-LU label from the dynamic label range.
3. If the conflict is with an IGP route, and BGP is trying to redistribute that IGP route as a label-ipv4 or label-ipv6 route with a route-table-import policy action that uses the **prefer-igp** keyword in the **sr-label-index** command, this is not considered a conflict and BGP uses the IGP-signaled label-index to derive its advertised label. This has the effect of stitching the BGP segment routing tunnel to the IGP segment routing tunnel.

Note: This use of the **prefer-igp** option is only possible when BGP segment routing is configured with the **prefix-sid-range global** command.

Any /32 label-ipv4 or /128 label-ipv6 BGP routes containing a prefix SID attribute are resolvable and usable in the same way as /32 label-ipv4 or /128 label-ipv6 routes without a prefix SID attribute. In other

words, these routes are installed in the route table and tunnel-table (unless **disable-route-table-install** or **selective-label-ipv4-install** are in effect), and they can have ECMP next hops or FRR backup next hops and be used as transport tunnels for any service that supports BGP-LU transport.

It should be noted that receiving a /32 label-ipv4 or /128 label-ipv6 route with a prefix-SID attribute does not create a tunnel in the segment-routing database; it only creates a label swap entry when the route is re-advertised with a new next hop. This means that the first SID in any SID-list of an SR policy should not be based on a BGP prefix SID; if this advice is not followed, then the SID-list may appear to be valid but the datapath is not programmed correctly. However, it is fine to use a BGP prefix SID as a non-first SID in any SR policy.

2.1.5 Segment Routing Operational Procedures

2.1.5.1 Prefix Advertisement and Resolution

After segment routing is successfully enabled in the IS-IS or OSPF instance, the router performs the following operations. See *IS-IS Control Protocol Changes*, *OSPFv2 Control Protocol Changes*, and *OSPFv3 Control Protocol Changes* for details of all TLVs and sub-TLVs for both IS-IS and OSPF protocols.

1. Advertise the Segment Routing Capability Sub-TLV to routers in all areas/levels of this IGP instance. However, only neighbors with which it established an adjacency interprets the SID/label range information and use it for calculating the label to swap to or push for a given resolved prefix SID.
2. Advertise the assigned index for each configured node SID in the new prefix SID sub-TLV with the N-flag (node-SID flag) set. Then the segment routing module programs the incoming label map (ILM) with a pop operation for each local node SID in the data path.
3. Assign and advertise automatically an adjacency SID label for each formed adjacency over a network IP interface in the new Adjacency SID sub-TLV. The following points should be considered.

   - Adjacency SID is advertised for both numbered and unnumbered network IP interface.
   - Adjacency SID is not advertised for an IES interface because access interfaces do not support MPLS.
   - The adjacency SID must be unique per instance and per adjacency. Furthermore, ISIS MT=0 can establish an adjacency for both IPv4 and IPv6 address families over the same link and in such a case a different adjacency SID is assigned to each next-hop. However, the existing IS-IS implementation assigns a single Protect-Group ID (PG-ID) to the adjacency and as such when the state machine of a BFD session tracking the IPv4 or IPv6 next-hop times out, an action is triggered for the prefixes of both address families over that adjacency.

   The segment routing module programs the incoming label map (ILM) with a swap to an implicit null label operation, for each advertised adjacency SID.
4. Resolve received prefixes and, if a prefix SID sub-TLV exists, the Segment Routing module programs the ILM with a swap operation and an LTN with a push operation, both pointing to the primary/LFA NHLFE. An SR tunnel is also added to the TTM. If a node SID resolves over an IES interface, the data path is not programmed and a trap is raised. Thus, only next-hops of an ECMP set corresponding to network IP interfaces are programmed in data path; next-hops corresponding to IES interfaces are not programmed. However, if the user configures the interface as network on one side and IES on the other side, MPLS packets for the SR tunnel received on the access side are dropped.
5. LSA filtering causes SIDs not to be sent in one direction which means some node SIDs is resolved in parts of the network upstream of the advertisement suppression.

When the user enables segment routing in a given IGP instance, the main SPF and LFA SPF are computed normally and the primary next-hop and LFA backup next-hop for a received prefix are added to RTM without the label information advertised in the prefix SID sub-TLV. In all cases, the segment routing (SR) tunnel is not added into RTM.

2.1.5.2 Error and Resource Exhaustion Handling

2.1.5.2.1 Procedure 1: Providing support of multiple topologies for the same destination prefix

The SR OS supports assigning different prefix-SID indexes and labels to the same prefix in different IGP instances. While other routers that receive these prefix SIDs programs a single route into RTM, based on the winning instance ID as per RTM route type preference, the SR OS adds two tunnels to this destination prefix in TTM. This provides for the support of multiple topologies for the same destination prefix.

For example: In two different instances (L2, IS-IS instance 1 and L1, IS-IS instance 2—see *Figure 2: Programming multiple tunnels to the same destination*), Router D has the same prefix destination, with different SIDs (SIDx and SIDy).

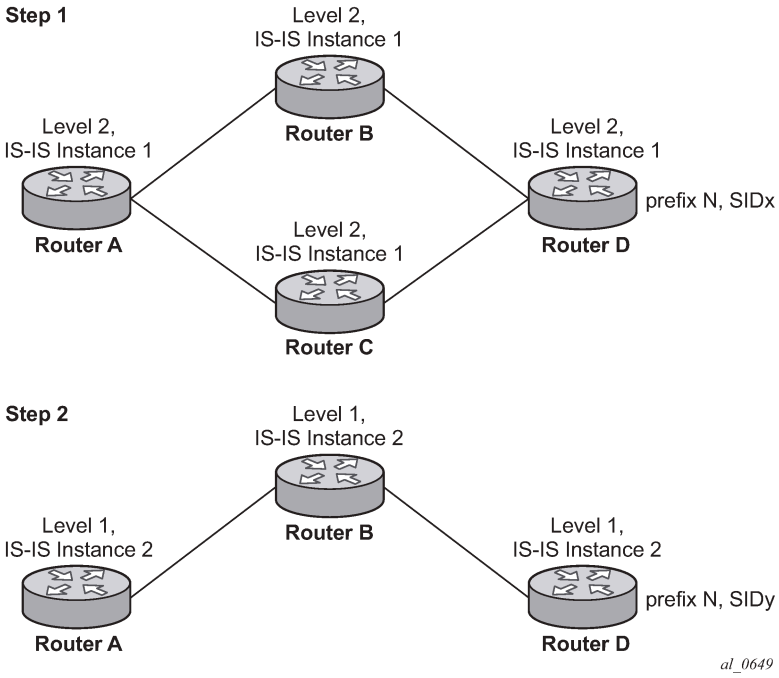

Figure 2: Programming multiple tunnels to the same destination

Assume the following route type preference in RTM and tunnel type preference in TTM are configured:

- ROUTE_PREF_ISIS_L1_INTER (RTM) 15
- ROUTE_PREF_ISIS_L2_INTER (RTM) 18
- ROUTE_PREF_ISIS_TTM 10

Note: The TTM tunnel type preference is not used by the SR module. It is put in the TTM and is used by other applications such as VPRN auto-bind and BGP shortcut to select a TTM tunnel.

Procedure

1. Router A performs the following resolution within the single IS-IS instance 1, level 2. All metrics are the same, and ECMP = 2.

   - For prefix N, the RTM entry is:
     - prefix N
     - nhop1 = B
     - nhop2 = C
     - preference 18
   - For prefix N, the SR tunnel TTM entry is:
     - tunnel-id 1: prefix N-SIDx
     - nhop1 = B
     - nhop2 = C
     - tunl-pref 10

2. Add IS-IS instance 2 (Level 1) in the same setup, but in routers A, B, and C only.

   - For prefix N, the RTM entry is:
     - prefix N
     - nhop1 = B
     - preference 15

     RTM prefers L1 route over L2 route
   - For prefix N, there are two SR tunnel entries in TTM:

     SR entry for L2:
     - tunnel-id 1: prefix N-SIDx
     - nhop1 = B
     - nhop2= C
     - tunl-pref 10

     SR entry for L1:
     - tunnel-id 2: prefix N-SIDy

2.1.5.2.2 Procedure 2: Resolving received SID indexes or labels to different routes of the same prefix within the same IGP instance

Two variations of this procedure can occur.

1. While SR OS does not allow assigning the *same* SID index or label to different routes of the same prefix within the same IGP instance, it resolves only one of the duplicate SIDs if received from another SR implementation and based on the RTM active route selection.

2. While SR OS does not allow assigning *different* SID indexes or labels to different routes of the same prefix within the same IGP instance, it resolves only one of the duplicate SIDs if received from another SR implementation and based on the RTM active route selection.

   The selected SID is used for ECMP resolution to all neighbors. If the route is inter-area and the conflicting SIDs are advertised by different ABRs, ECMP towards all ABRs uses the selected SID.

2.1.5.2.3 Procedure 3: Checking for SID error prior to programming ILM and NHLFE

If any of the following conditions are true, the router logs a trap and generates a syslog error message and does not program the ILM and NHLFE for the prefix SID.

- Received prefix SID index falls outside of the locally configured SID range.
- One or more resolved ECMP next-hops for a received prefix SID did not advertise SR Capability sub-TLV.
- Received prefix SID index falls outside the advertised SID range of one or more resolved ECMP next-hops.

2.1.5.2.4 Procedure 4: Programming ILM/NHLFE for duplicate prefix-SID indexes/labels for different prefixes

Two variations of this procedure can occur.

1. For received duplicate prefix-SID indexes or labels for different prefixes *within the same* IGP instance, the router:
   - programs ILM/NHLFE for the first one
   - logs a trap and a syslog error message
   - does not program the subsequent one in data path
2. For received duplicate prefix-SID index for different prefixes *across* IGP instances, there are two options.
   - In the global SID index range mode of operation, the resulting ILM label values is the same across the IGP instances. The router:
     – programs ILM/NHLFE for the prefix of the winning IGP instance based on the RTM route type preference
     – logs a trap and a syslog error message
     – does not program the subsequent prefix SIDs in data path
   - In per-instance SID index range mode of operation, the resulting ILM label has different values across the IGP instances. The router programs ILM/NHLFE for each prefix as expected.

2.1.5.2.5 Procedure 5: Programming ILM/NHLFE for the same prefix across IGP instances

The behavior in the case of a global SID index range is illustrated by the IS-IS example in *Figure 3: Handling of Same Prefix and SID in different IS-IS Instances*.

In global SID index range mode of operation, the resulting ILM label values is the same across the IGP instances. The router programs ILM/NHLFE for the prefix of the winning IGP instance based on the RTM

route type preference. The router logs a trap and a syslog error message, and does not program the other prefix SIDs in data path.

In per-instance SID index range mode of operation, the resulting ILM label has different values across the IGP instances. The router programs ILM/NHLFE for each prefix as expected.

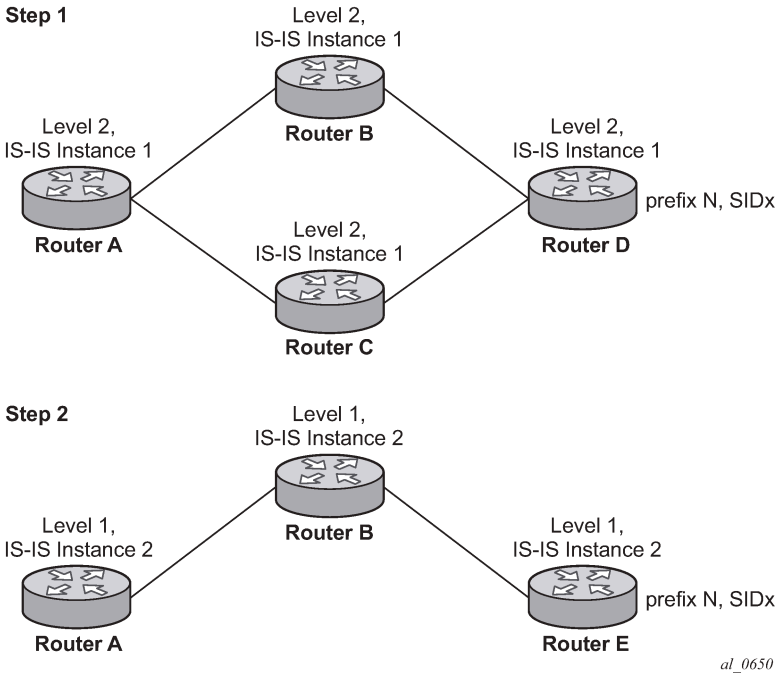

Figure 3: Handling of Same Prefix and SID in different IS-IS Instances

Assume the following route type preference in RTM and tunnel type preference in TTM are configured:

- ROUTE_PREF_ISIS_L1_INTER (RTM) 15
- ROUTE_PREF_ISIS_L2_INTER (RTM) 18
- ROUTE_PREF_ISIS_TTM 10

Note: The TTM tunnel type preference is not used by the SR module. It is put in the TTM and is used by other applications such as VPRN auto-bind and BGP shortcut to select a TTM tunnel.

1. Router A performs the following resolution within the single IS-IS instance 1, level 2. All metrics are the same, and ECMP = 2.

   - For prefix N, the RTM entry is:
     - prefix N
     - nhop1 = B
     - nhop2 = C
     - preference 18
   - For prefix N, the SR tunnel TTM entry is:
     - tunnel-id 1: prefix N-SIDx
     - nhop1 = B
     - nhop2 = C
     - tunl-pref 10
2. Add IS-IS instance 2 (Level 1) in the same setup, but in routers A, B, and E only.

   - For prefix N, the RTM entry is:
     - prefix N
     - nhop1 = B
     - preference 15

     RTM prefers L1 route over L2 route.
   - For prefix N, there is one SR tunnel entry for L2 in TTM:
     - tunnel-id 1: prefix N-SIDx
     - nhop1 = B
     - nhop2 = C
     - tunl-pref 10

2.1.5.2.6 Procedure 6: Handling ILM resource exhaustion while assigning an SID index/label

If the system exhausted an ILM resource while assigning an SID index/label to a local loopback interface, then index allocation is failed and an error is returned in CLI. In addition, the router logs a trap and generates a syslog error message.

2.1.5.2.7 Procedure 7: Handling ILM/NHLFE/other IOM or CPM resource exhaustion while resolving or programming an SID index/label

If the system exhausted an ILM, NHLFE, or any other IOM or CPM resource while resolving and programming a received prefix SID or programming a local adjacency SID, then following occurs.

- The IGP instance goes into overload and a trap and syslog error message are generated.
- The segment routing module deletes the tunnel.

The user must manually clear the IGP overload condition after freeing resources. After the IGP is brought back up, it attempts to program at the next SPF all tunnels which previously failed the programming operation.

2.1.6 Segment Routing Tunnel Management

The segment routing module adds to TTM a shortest path SR tunnel entry for each resolved remote node SID prefix and programs the data path with the corresponding LTN with the push operation pointing to the primary and LFA backup NHLFEs. The LFA backup next-hop for a given prefix which was advertised with a node SID is only computed if the **loopfree-alternates** option is enabled in the IS-IS or OSPF instance. The resulting SR tunnel which is populated in TTM is automatically protected with FRR when an LFA backup next-hop exists for the prefix of the node SID.

With ECMP, a maximum of 32 primary next-hops (NHLFEs) are programmed for the same tunnel destination per IGP instance. ECMP and LFA next-hops are mutually exclusive as per existing implementation.

The default preference for shortest path SR tunnels in the TTM is set lower than LDP tunnels but higher than BGP tunnels to allow controlled migration of customers without disrupting their current deployment when they enable segment routing. The following is the setting of the default preference of the various tunnel types. This includes the preference of both SR tunnels based on shortest path (referred to as SR-ISIS and SR-OSPF).

The global default TTM preference for the tunnel types is as follows:

- ROUTE_PREF_RSVP 7
- ROUTE_PREF_SR_TE 8
- ROUTE_PREF_LDP 9
- ROUTE_PREF_OSPF_TTM 10
- ROUTE_PREF_ISIS_TTM 11
- ROUTE_PREF_BGP_TTM 12
- ROUTE_PREF_GRE 255

The default value for SR-ISIS or SR-OSPF is the same regardless if one or more IS-IS or OSPF instances programmed a tunnel for the same prefix. The selection of an SR tunnel in this case is based on lowest IGP instance ID.

The TTM preference is used in the case of BGP shortcuts, VPRN auto-bind, or BGP transport tunnel when the tunnel binding commands are configured to the **any** value which parses the TTM for tunnels in the protocol preference order. The user can choose to either go with the global TTM preference or list explicitly the tunnel types to be used. When the tunnel types are listed explicitly, the TTM preference is still used to select one type over the other. In both cases, a fallback to the next preferred tunnel type is performed if the selected one fails. Also, a reversion to a more preferred tunnel type is performed as soon as one is available. See *BGP Shortcut Using Segment Routing Tunnel*, *BGP Label Route Resolution Using Segment Routing Tunnel*, and *Service Packet Forwarding with Segment Routing* for the detailed service and shortcut binding CLI.

For SR-ISIS and SR-OSPF, the user can configure the preference of each specific IGP instance away from the above default values.

```
config>router>isis>segment-routing>tunnel-table-pref preference <1 to 255>
config>router>ospf>segment-routing>tunnel-table-pref preference <1 to 255>
```

Note: The preference of SR-TE LSP is not configurable and is the second most preferred tunnel type after RSVP-TE. This is independent if the SR-TE LSP was resolved in IS-IS or OSPF.

2.1.6.1 Tunnel MTU Determination

The MTU of an SR tunnel populated into TTM is determined as in the case of an IGP tunnel; for example, LDP LSP is based on the outgoing interface MTU minus the label stack size. Segment routing, however, supports remote LFA and TI-LFA which can program an LFA repair tunnel by adding one or more labels.

Based on the above, the user is provided with a CLI to configure the MTU of all SR tunnels within each IGP instance:

```
config>router>isis>segment-routing>tunnel-mtu bytes bytes
config>router>ospf>segment-routing>tunnel-mtu bytes bytes
```

There is no default value for this new command. If the user does not configure an SR tunnel MTU, the MTU is fully determined by IGP as explained below.

The MTU of the SR tunnel, in bytes, is then determined as follows:

*SR_Tunnel_MTU = MIN {Cfg_SR_MTU, IGP_Tunnel_MTU- (1+ frr-overhead)*4}*

Where:

- *Cfg_SR_MTU* is the MTU configured by the user for all SR tunnels within a given IGP instance using the above CLI. If no value was configured by the user, the SR tunnel MTU is fully determined by the IGP interface calculation explained next.
- *IGP_Tunnel_MTU* is the minimum of the IS-IS or OSPF interface MTU among all the ECMP paths or among the primary and LFA backup paths of this SR tunnel.
- *frr-overhead* is set the following parameters:
  - *value* of **ti-lifa [max-sr-frr-labels** labels**]** if **loopfree-alternates** and **ti-lfa** are enables in this IGP instance
  - 1 if **loopfree-alternates** and **remote-lfa** are enabled but **ti-lfa** is disabled in this IGP instance
  - Otherwise, it is set to 0

The SR tunnel MTU is dynamically updated anytime any of the above parameters used in its calculation changes. This includes when the set of the tunnel next-hops changes or the user changes the configured SR MTU or interface MTU value.

Note: The above calculated SR tunnel MTU is used for the determination of an SDP MTU and for checking the Layer 2 service MTU. For the purpose of fragmentation of IP packets forwarded in GRT or in a VPRN over an SR shortest path tunnel, the data path always deducts the worst case MTU (5 labels or 6 labels if hash label feature is enabled) from the outgoing interface MTU for the decision to fragment or not the packet. In this case, the above formula is not used.

2.1.7 Segment Routing Local Block

Some labels that are provisioned through CLI or a management interface must be allocated from the Segment Routing Local Block (SRLB). The SRLB references a reserved label block configured under

config>**router**>**mpls-labels**. Refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR MPLS Guide* for further information on reserved label blocks.

The label block to use is specified by the **srlb** command under IS-IS or OSPF:

```
config>router>isis>segment-routing
   [no] srlb <reserved-label-block-name>

config>router>ospf> segment-routing
   [no] srlb <reserved-label-block-name>
```

Provisioned labels for adjacency SIDs and adjacency SID sets must be allocated from the configured SRLB. If no SRLB is specified, or the requested label does not fall within the SRLB, or the label is already allocated, then the request is rejected.

2.1.7.1 Bundling Adjacencies in Adjacency Sets

An adjacency set is a bundle of adjacencies, represented by a common adjacency SID for the bundled set. It enables, for example, a path for an SR-TE LSP through a network to be specified while allowing the local node to spray packets across the set of links identified by a single adjacency SID.

SR OS supports both parallel adjacency sets (for example, those where adjacencies originating on one node terminate on a second, common node), and the ability to associate multiple interfaces on a specified node, irrespective of whether the far end of those interfaces' respective links terminate on the same node.

An adjacency set is created under IS-IS or OSPF using the following CLI:

```
config
   router
      isis | ospf
         segment-routing
            [no] adjacency-set <id>
               family [ipv4 | ipv6]
               parallel [no-advertise]
               no parallel
               exit
...
.              exit
            exit
config
   router
      ospf
         segment-routing
            [no] adjacency-set <id>
               parallel [no-advertise]
               no parallel
               exit
...
.
            exit
```

The **adjacency-set** *id* command specifies an adjacency set, where *id* is an unsigned integer in the range 0 to 4294967295.

In IS-IS, each adjacency set is assigned an address family, IPv4 or IPv6. The family command for IS-IS indicates the address family of the adjacency set. For OSPF, the address family of the adjacency set is implied by the OSPF version and the instance.

The **parallel** command indicates that all members of the adjacency set must terminate on the same neighboring node. The system raises a trap, when the **parallel** command is configured, if a user attempts to add an adjacency terminating on a neighboring node that differs from the existing members of the adjacency set. See *Associating an Interface with an Adjacency Set* for details about how to add interfaces to an adjacency set. In addition, the system stops advertising the adjacency set and deprograms it from TTM. **The parallel** command is enabled by default.

By default, parallel adjacency sets are advertised in the IGP. The **no-advertise** option prevents a parallel adjacency set from being advertised in the IGP; it is only advertised if the **parallel** command is configured. To prevent issues in the case of ECMP if a non-parallel adjacency set is used, coordination may be required by an external controller of the label sets for SIDs at all downstream nodes. As a result, non-parallel adjacency sets are not advertised in the IGP. The label stack below the adjacency set label must be valid at any downstream node that exposes it, even though it is sprayed over multiple downstream next-hops.

Parallel adjacency sets are programmed in TTM (unless there is an erroneous configuration of a non-parallel adjacency, as described). Non-parallel adjacency sets are not added to TTM or RTM, meaning that they cannot be used as a hop at the originating node. Parallel adjacency sets that are advertised are included in the link state database and TE database, but non-parallel adjacency sets are not included because they are not advertised.

An adjacency set with only one next hop is also advertised as an individual adjacency SID with the S flag set. However, the system does not calculate a backup for an adjacency set even if it has only one next hop.

2.1.7.1.1 Associating an Interface with an Adjacency Set

IS-IS or OSPF interfaces are associated with one or more adjacency sets using the following CLI. Both numbered and unnumbered interfaces can be assigned to the same adjacency set.

```
config
   router
      isis
         interface
           [no] adjacency-set <id>
           [no] adjacency-set <id>
           [no] adjacency-set <id>
config
   router
      ospf
         area
            interface
               [no] adjacency-set <id>
               [no] adjacency-set <id>
               [no] adjacency-set <id>
```

If an interface is assigned to an adjacency set, then a common adjacency SID value is advertised for every interface in the set, in addition to the adjacency SID corresponding to the IPv4 and or IPv6 adjacency for the interface. Each IS-IS or OSPF advertisement therefore contains two adjacency SID TLVs for an address family:

- an adjacency SID for the interface (a locally-unique value).

- an adjacency SID TLV for the adjacency set.

  This TLV is distinguished by having the S-bit (IS-IS) or G-bit (OSPF) in the flags field set to 1. Its value is the same as other adjacency SIDs in the set at that node.

By default, both the adjacency SID for an interface and the adjacency SID for a set are dynamically allocated by the system. However, it is possible for the user to configure an alternative, static value for the SID (see *Provisioning Adjacency SID Values for an Adjacency Set* for more information).

A maximum of 32 interfaces can be bound to a common adjacency set. Configuration of more than 32 interfaces is blocked by the system and a CLI error is generated.

Only point-to-point interfaces can be assigned to an adjacency set.

If a user attempts to assign an IES interface to an adjacency set, the system generates a CLI warning and segment routing does not program the association.

The IGP blocks the configuration of an adjacency set under an interface when the adjacency set has not yet been created under segment-routing.

In IS-IS, it is possible to add Layer 1, Layer 2, or a mix of Layer 1 and Layer 2 adjacencies to the same adjacency set.

2.1.7.1.2 Provisioning Adjacency SID Values for an Adjacency Set

For an adjacency set, static values are configured using the **sid** CLI command, as follows:

```
config>router>isis>segment-routing
      [no] adjacency-set <id>
         family [ipv4 | ipv6]
         [no] sid label <value>
         parallel [no-advertise]
         no parallel
         exit
      [no] adjacency-set <id>
          family [ipv4 | ipv6]
          [no] sid label <value>
          parallel [no-advertise]
          no parallel
          exit
                     ...

config>router>ospf>segment-routing
      [no] adjacency-set <id>
         [no] sid label <value>
         parallel [no-advertise]
         no parallel
         exit
      [no] adjacency-set <id>
         [no] sid label <value>
         parallel [no-advertise]
         no parallel
         exit
       ...
```

If **no sid** is configured, a dynamic value is allocated to the adjacency set. A user may change the dynamic value to specify a static SID value. Changing an adjacency set value from dynamic to a static, or vice versa, may result in traffic being dropped as the ILM is reprogrammed.

The *value* must correspond to a label in the reserved label block in provisioned mode referred to by the **srlb** command. A CLI error is generated if a user attempts to configure an invalid *value*. If a label is not configured, then the label *value* is dynamically allocated by the system from the dynamic labels range. If a static adjacency set label is configured, then the system does not advertise a dynamic adjacency set label.

A static label value for an adjacency set SID is persistent. Therefore, the P-bit of the Flags field in the Adjacency-SID TLV, referring to the adjacency set should be set to 1.

2.1.8 Loop Free Alternates

2.1.8.1 Remote LFA with Segment Routing

The user enables the remote LFA next-hop calculation by the IGP LFA SPF by appending the following new option in the existing command which enables LFA calculation:

```
config>router>isis>loopfree-alternates remote-lfa
config>router>ospf>loopfree-alternates remote-lfa
```

SPF performs the remote LFA additional computation following the regular LFA next-hop calculation when both of the following conditions are met:

- The remote-lfa option is enabled in an IGP instance.
- The LFA next hop calculation did not result in protection for one or more prefixes resolved to a given interface.

Remote LFA extends the protection coverage of LFA-FRR to any topology by automatically computing and establishing/tearing-down shortcut tunnels, also referred to as repair tunnels, to a remote LFA node which puts the packets back into the shortest without looping them back to the node which forwarded them over the repair tunnel. A repair tunnel can in theory be an RSVP LSP, an LDP-in-LDP tunnel, or an SR tunnel. In SR OS, this feature is restricted to use an SR repair tunnel to the remote LFA node.

The remote LFA algorithm for link protection is described in RFC 7490, *Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)*. Unlike the regular LFA calculation, which is calculated per prefix, the LFA algorithm for link protection is a per-link LFA SPF calculation. As such, it provides protection for all destination prefixes which share the protected link by using the neighbor on the other side of the protected link as a proxy for all these destinations. Assume the topology in *Figure 4: Remote LFA Algorithm*.

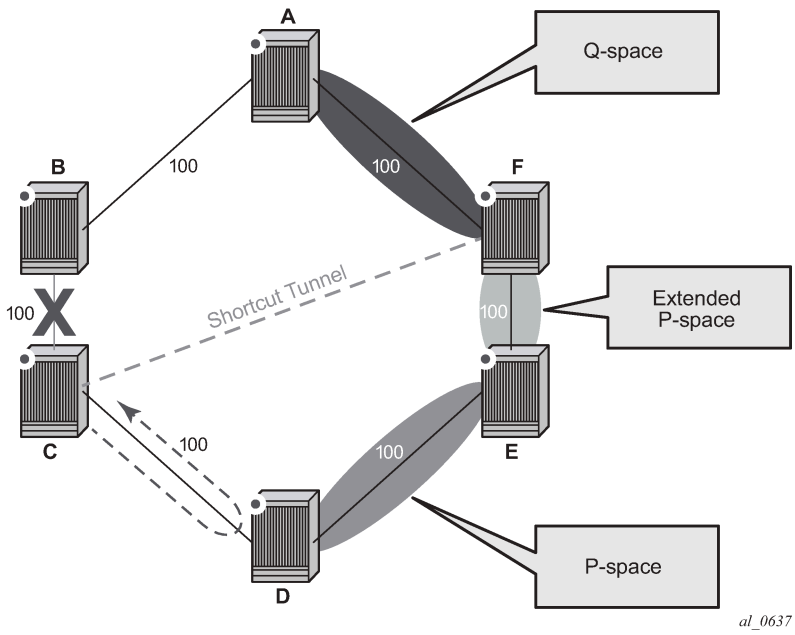

Figure 4: Remote LFA Algorithm

When the LFA SPF in node C computes the per-prefix LFA next-hop, prefixes which use link C-B as the primary next-hop has no LFA next-hop due to the ring topology. If node C used node link C-D as a back-up next-hop, node D would loop a packet back to node C. The remote LFA then runs the following algorithm, referred to as the "PQ Algorithm" in RFC 7490:

1. Compute the extended P space of Node C with respect to link C-B: set of nodes reachable from node C without any path transiting the protected link (link C-B). This yields nodes D, E, and F.

   The determination of the extended P space by node C uses the same computation as the regular LFA by running SPF on behalf of each of the neighbors of C.

   Note: RFC 7490 initially introduced the concept of P space, which would have excluded node F because, from the node C perspective, node C has a couple of ECMP paths, one of which goes via link C-B. However, because the remote LFA next-hop is activated when link C-B fails, this rule can be relaxed and node F can be included, which then yields the extended P space.

   The user can limit the search for candidate P nodes to reduce the amount of SPF calculations in topologies where many eligible P nodes can exist. A CLI command is provided to configure the maximum IGP cost from node C for a P node to be eligible:

   - **config>router>isis>loopfree-alternates remote-lfa max-pq-cost** *value*
   - **config>router>ospf>loopfree-alternates remote-lfa max-pq-cost** *value*

2. Compute the Q space of node B with respect to link C-B: set of nodes from which the destination proxy (node B) can be reached without any path transiting the protected link (link C-B).

   The Q space calculation is effectively a reverse SPF on node B. In general, one reverse SPF is run on behalf of each of C neighbors to protect all destinations resolving over the link to the neighbor. This yields nodes F and A in the example of *Figure 4: Remote LFA Algorithm*.

   The user can limit the search for candidate Q nodes to reduce the amount of SPF calculations in topologies where many eligible Q nodes can exist. The CLI above is also used to configure the maximum IGP cost from node C for a Q node to be eligible.
3. Select the best alternate node: this is the intersection of extended P and Q spaces. The best alternate node or PQ node is node F in the example of *Figure 4: Remote LFA Algorithm*. From node F onwards, traffic follows the IGP shortest path.

   If many PQ nodes exist, the lowest IGP cost from node C is used to narrow down the selection and if more than one PQ node remains, the node with lowest router-id is selected.

The details of the label stack encoding when the packet is forwarded over the remote LFA next-hop is shown in *Figure 5: Remote LFA Next-Hop in Segment Routing*.

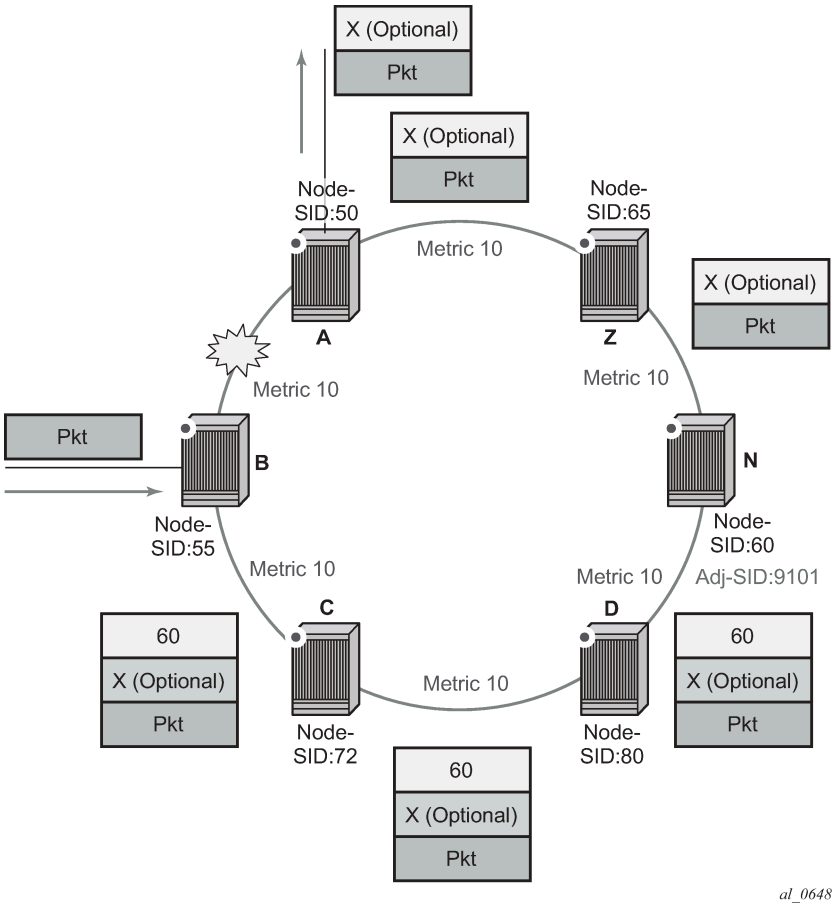

Figure 5: Remote LFA Next-Hop in Segment Routing

The label corresponding to the node SID of the PQ node is pushed on top of the original label of the SID of the resolved destination prefix. If node C has resolved multiple node SIDs corresponding to different

prefixes of the selected PQ node, it pushes the lowest node SID label on the packet when forwarded over the remote LFA backup next-hop.

If the PQ node is also the advertising router for the resolved prefix, the label stack is compressed in some cases depending on the IGP:

- In IS-IS, the label stack is always reduced to a single label, which is the label of the resolved prefix owned by the PQ node.
- In OSPF, the label stack is reduced to the single label of the resolved prefix when the PQ node advertised a single node SID in this OSPF instance. If the PQ node advertised a node SID for multiple of its loopback interfaces within this same OSPF instance, the label stack is reduced to a single label only in the case where the SID of the resolved prefix is the lowest SID value.

The following rules and limitations apply to the remote LFA implementation:

- If the user excludes a network IP interface from being used as an LFA next-hop using the CLI command **loopfree-alternate-exclude** under the interface's IS-IS or OSPF context, the interface is also excluded from being used as the outgoing interface for a remote LFA tunnel next-hop.
- As with the regular LFA algorithm, the remote LFA algorithm computes a backup next-hop to the ABR advertising an inter-area prefix and not to the destination prefix itself.

2.1.8.2 Topology Independent LFA

The Topology-Independent LFA (TI-LFA) feature improves the protection coverage of a network topology by computing and automatically instantiating a repair tunnel to a Q node which is not in the shortest path from the computing node. The repair tunnel uses the shortest path to the P node and a source-routed path from the P node to the Q node.

In addition, the TI-LFA algorithm selects the backup path that matches the post-convergence path. This helps capacity planning in the network since traffic always flows on the same path when transitioning to the FRR next-hop and then onto the new primary next-hop.

At a high level, the TI-LFA link protection algorithm searches for the closest Q node to the computing node and then selects the closest P node to this Q node, up to a maximum number of labels. This is performed on each of the post-convergence paths to each destination node or prefix D.

When the TI-LFA feature is enabled in IS-IS, it provides a TI-LFA link-protect backup path in IS-IS MT=0 for a SR-ISIS IPV4/IPv6 tunnel (node SID and adjacency SID), for a IPv4 SR-TE LSP, and for LDP IPv4 FEC when the LDP **fast-reroute backup-sr-tunnel** option is enabled.

2.1.8.2.1 TI-LFA Configuration

Users can enable TI-LFA in an IS-IS instance using one of the following command:

config>router>isis>loopfree-alternates [**remote-lfa** [**max-pq-cost** *value*]] [**ti-lfa** [**max-sr-frr-labels** *value*]]

When the **ti-lfa** option is enabled in IS-IS, it provides a TI-LFA link-protect backup path in IS-IS MT=0 for an SR-ISIS IPV4 and IPv6 tunnel (node SID and adjacency SID), for a IPv4 SR-TE LSP, and for an LDP IPv4 FEC when the LDP **fast-reroute backup-sr-tunnel** option is enabled. **See more details of the applicability of the various LFA options in** *LFA Protection Option Applicability***.**

The **max-sr-frr-labels** parameter limits the search for the LFA backup next-hop:

- 0 — The IGP LFA SPF restricts the search to TI-LFA backup next-hop which does not require a repair tunnel, meaning that P node and Q node are the same and match a neighbor. This is also the case when both P and Q node match the advertising router for a prefix.
- 1 to 3 — The IGP LFA SPF widens the search to include a repair tunnel to a P node which itself is connected to the Q nodes with a 0-to-2 hops for a total of maximum of three labels: one node SID to P node and two adjacency SIDs from P node to the Q node. If the P node is a neighbor of the computing node, its node SID is compressed and meaning that up to three adjacency SIDs can separate the P and Q nodes.
- 2 (default) — Corresponds to a repair tunnel to a non-adjacent P which is adjacent to the Q node. If the P node is a neighbor of the computing node, then the node SID of the P node is compressed and the default value of two labels corresponds to two adjacency SIDs between the P and Q nodes.

When the user attempts to change the **max-sr-frr-labels** parameter to a value that results in a change to the computed FRR overhead, then IGP checks that all SR-TE LSPs can properly account for the overhead based on the configuration of the LSP **max-sr-labels** and **additional-frr-labels** parameter values or the change is rejected.

The FRR overhead is computed by IGP and its value is set as follows:

- 0 if **segment-routing** is disabled in the IGP instance
- 0 if **segment-routing** is enabled but **remote-lfa** is disabled and **ti-lfa** is disabled
- 1 if **segment routing** is enabled and **remote-lfa** is enabled but **ti-lfa** is disabled, or if **segment-routing** is enabled and **remote-lfa** is enabled and **ti-lfa** is enabled but **ti-lfa max-sr-frr-labels** *labels* is set to 0.
- The value of **ti-lfa max-sr-frr-labels** *labels*, if **segment-routing** is enabled and **ti-lfa** is enabled regardless if **remote-lfa** is enabled or disabled.

2.1.8.2.2 TI-LFA Link-Protect Operation

This section describes TI-LFA protection behavior when the **loopfree-alternates** command is enabled with the **remote-lfa** and **ti-lfa** options as described in *TI-LFA Configuration*.

LFA Protection Option Applicability

Depending on the configured options of the **loopfree-alternates** command, the LFA SPF in an IGP instance runs algorithms in the following order:

Procedure

1. Computes a regular LFA for each node and prefix. In this step, a computed backup next-hop satisfies any applied LFA policy.

   This backup next-hop protects that specific prefix or node in the context of IP FRR, LDP FRR, SR FRR, and SR-TE FRR.

2. Follows with the TI-LFA, if the **ti-lfa** option is enabled for all prefixes and nodes, regardless of the outcome of the first step.

   A prefix or node for which a TI-LFA backup next-hop is found overrides the result from the first step in the context of LDP FRR, if the LDP **fast-reroute backup-sr-tunnel** option is enabled, in SR FRR and in SR-TE FRR.

   With SR FRR and SR-TE FRR, the TI-LFA next-hop protects the node-SID of that prefix and any adjacency-SID terminating on the node-SID of that prefix.

   The prefix or node continues to use the backup next hop found in Step *1* in the context of LDP FRR (if the LDP **fast-reroute backup-sr-tunnel** option is disabled), or in IP FRR.

3. Runs remote LFA only for the next-hop of prefixes and nodes which remain unprotected after Step *1* and Step *2* if the **remote-lfa** option is enabled. A prefix or node for which a remote LFA backup next-hop is found uses it in the context of LDP FRR, when the LDP **fast-reroute backup-sr-tunnel** option is enabled, in SR FRR and in SR-TE FRR.

To protect an adjacency SID, the LFA selection algorithm uses the following preference order:

1. Adjacency of an alternate parallel link to the same neighbor. If more than one adjacency exists, select one as follows:
   a. adjacency with the lowest metric
   b. adjacency to the neighbor with the lowest router ID (OSPF) or system ID (IS-IS), and the lowest metric
   c. with the lowest interface index and the lowest router ID (OSPF) or system ID (IS-IS)
2. An ECMP next hop to a node-SID of the same neighbor that is different from the next hop of the protected adjacency. If more than one next hop exists, select one as follows:
   a. next hop with the lowest metric
   b. next hop to the neighbor with the lowest router ID (OSPF) or system ID (IS-IS) if same lowest metric
   c. next hop to the lowest interface index if same neighbor router ID (OSPF) or system ID (IS-IS
3. LFA backup outcome of a node SID of the same neighbor, in the following preference order:
   a. TI-LFA backup
   b. LFA backup
   c. RLFA backup

TI-LFA Algorithm

At a high level, the TI-LFA link protection algorithm searches for the closest Q node to the computing node and then selects the closest P node to this Q node, up to a number of labels corresponding to the value of ti-**lfa max-sr-frr-labels *labels***, on each of the post-convergence paths to each destination node or prefix D. Consider the topology in *Figure 6: Selecting Link-Protect TI-LFA Backup Path* where router R3 computes a TI-LFA next-hop for protecting link R3-R4.

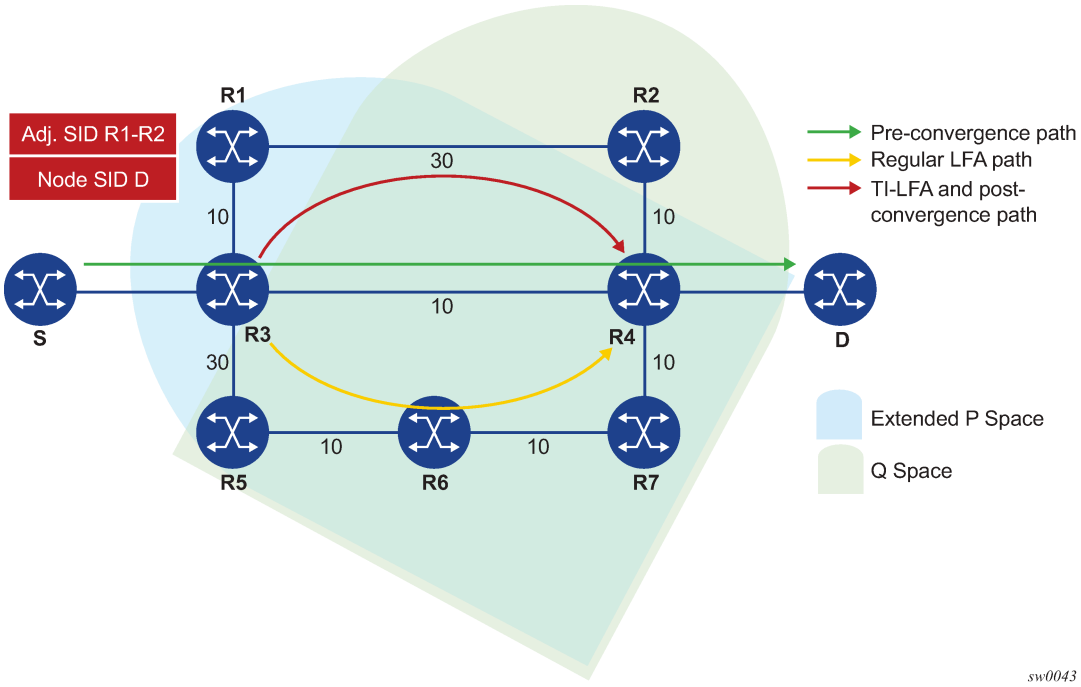

Figure 6: Selecting Link-Protect TI-LFA Backup Path

For each destination node D:

1. Compute the post-convergence SPF on the topology without the protected link.

   In *Figure 6: Selecting Link-Protect TI-LFA Backup Path*, R3 finds a single post-convergence path to destination D via R1.

   Note: The post-convergence SPF does not include IGP shortcut.

2. Compute the extended P-Space of R3 with respect to protected link R3-R4 on the post-convergence paths.

   This is the set of nodes Yi in the post-convergence paths which are reachable from R3 neighbors without any path transiting the protected link R3-R4.

   R3 computes an LFA SPF rooted at each of its neighbors within the post-convergence paths, that is, R1, using the following equation:

   ```
   Distance_opt(R1, Yi) < Distance_opt(R1, R3) + Distance_opt(R3, Yi)
   ```

   Where, Distance_opt(A,B) is the shortest distance between A and B. The extended P-space calculation yields only node R1.

3. Compute Q-space of R3 with respect to protected link R3-R4 in the post-convergence paths.

   This is the set of nodes Zi in the post-convergence paths from which the neighbor node R4 of the protected link, acting as a proxy for all destinations D, can be reached without any path transiting the protected link R3-R4.

   ```
   Distance_opt(Zi, R4) < Distance_opt(Zi, R3) + Distance_opt(R3, R4)
   ```

   The Q-space calculation yields nodes R2 and R4.

   This is the same computation of the Q-space performed by the remote LFA algorithm, except that the TI-LFA Q-space computation is performed only on the post-convergence.
4. For each post-convergence path, search for the closest Q-node and select the closest P-node to this Q-node, up to a number of labels corresponding to the value of **ti-lfa max-sr-frr-labels** *labels*.

   In the topology in *Figure 6: Selecting Link-Protect TI-LFA Backup Path*, there is a single post-convergence path, a single P-node (R1), and the closest of the two found Q-nodes to the P-Node is R2.

   R3 installs the repair tunnel to the P-Q set and includes the node-SID of R1 and the adjacency SID of the adjacency over link R1-R2 in the label stack. Note that since the P-node R1 is a neighbor of the computing node R3, the node SID of R1 is not needed and the label stack of the repair tunnel is compressed to the adjacency SID over link R1-R2 as shown in *Figure 6: Selecting Link-Protect TI-LFA Backup Path*.

   When a P-Q set is found on multiple ECMP post-convergence paths, the following selection rules are applied, in ascending order, to select a set from a single path:

   a. the lowest number of labels
   b. the next-hop to the neighbor router with the lowest **router-id** (OSPF) or **system-id** (ISIS)
   c. the next-hop corresponding to the Q node with the lowest **router-id** (OSPF) or **system-id** (ISIS)

   If multiple links with adjacency SID exist between the selected P node and the selected Q node, the following rules are used to select one of them:

   a. the adjacency SID with the lowest metric
   b. the adjacency SID with the lowest SID value if same lowest metric

TI-LFA Feature Interaction and Limitations

The following are feature interactions and limitations of the TI-LFA link protection.

- Enabling the **ti-lfa** option in an IS-IS or OSPF instance overrides the user configuration of the **loopfree-alternate-exclude** command under the interface's context in that IGP instance. In other words, the TI-LFA SPF uses that interface as a backup next-hop if it matches the post-convergence next-hop.
- Any prefix excluded from LFA protection using the **loopfree-alternates exclude prefix-policy** *prefix-policy* command under the IGP instance context is also excluded from TI-LFA.
- Since the post-convergence SPF does not use paths transiting on a node in IS-IS overload, the TI-LFA backup path automatically does not transit on the a node.
- IES interfaces are skipped in TI-LFA computation since they do not support Segment Routing with MPLS encapsulation. If the only found TI-LFA backup next-hop matches an IES interface, IGP treats

this as if there were no TI-LFA backup paths and falls back to using either a remote LFA or regular LFA backup path as per the selection rules in *LFA Protection Option Applicability*.

- The TI-LFA feature provides link-protection only. Thus, if the protected link is a broadcast interface, the TI-LFA algorithm only guarantees protection of that link and not of the Pseudo-Node (PN) corresponding to that shared subnet. In other words, if the PN is in the post-convergence path, the TI-LFA backup path may still traverse again the PN. For example, node E in *Figure 7: TI-LFA Backup Path via a Pseudo-Node* computes a TI-LFA backup path to destination D via E-C-PN-D because it is the post-convergence path when excluding link E-PN from the topology. This TI-LFA backup does not protect against the failure of the PN.

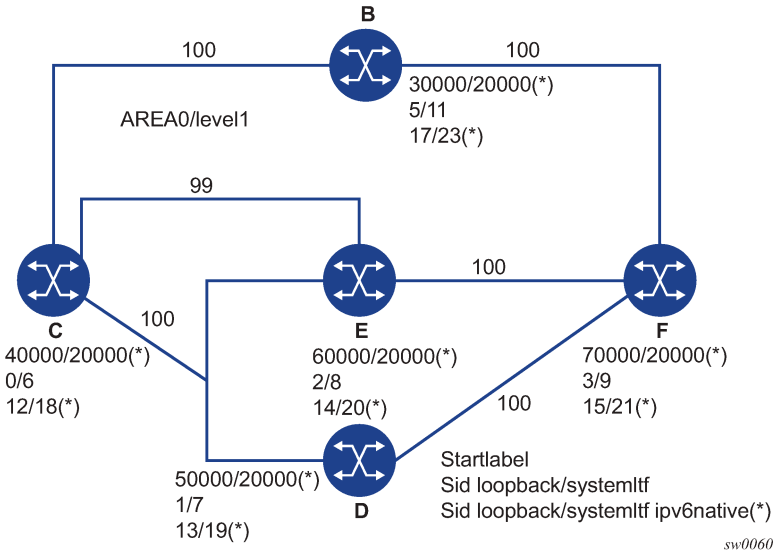

Figure 7: TI-LFA Backup Path via a Pseudo-Node

- When the computing router selects an adjacency SID among a set of parallel adjacencies between the P and Q nodes, the selection rules in**step 4 of** *TI-LFA Algorithm*are used. However, these rules may not yield the same interface the P node itself would have selected in its post-convergence SPF since the latter is based on the lowest value of the locally managed interface index.

  For example, node A in *Figure 8: Parallel Adjacencies between P and Q Nodes* computes the link-protect TI-LFA backup path for destination node E as path A-C-E, where C is the P node and E is the Q node and destination. C has a pair of adjacency SIDs with the same metric to E. Node A selects the adjacency over the P2P link C-E because it has the lowest SID value but node C may select the

interface C-PN in its post-convergence path calculation if that interface has a lower interface index than P2P link C-E.

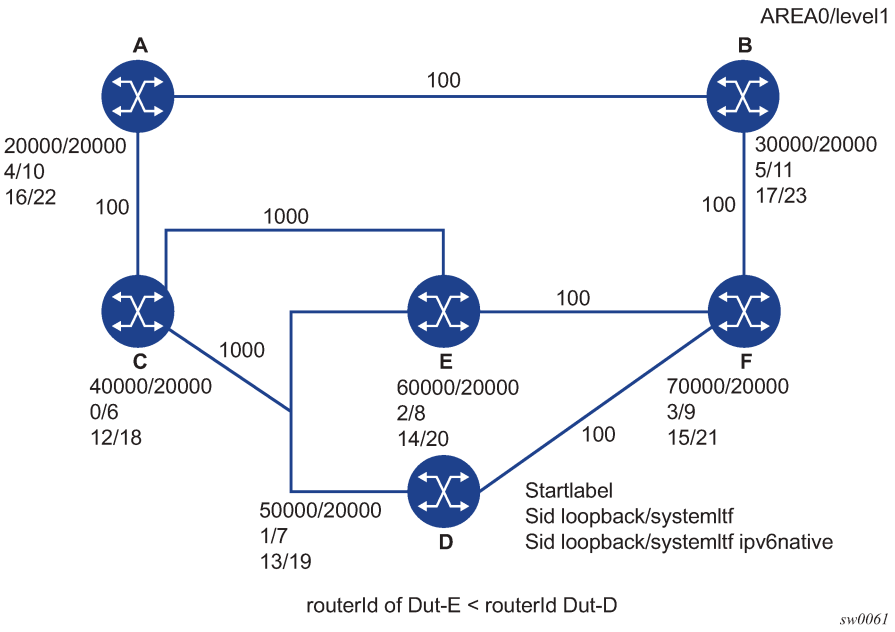

Figure 8: Parallel Adjacencies between P and Q Nodes

- When a node SID is advertised by multiple routers (anycast SID), the TI-LFA algorithm on a router which resolves the prefix of this SID computes the backup next-hop toward a single node owner of the prefix based on the rules for prefix and SID ECMP next-hop selection.

2.1.8.2.3 Data Path Support

The TI-LFA repair tunnel can have a maximum of three additional labels pushed in addition to the label of the destination node or prefix. The user can set a lower maximum value for the additional FRR labels by configuring the **ti-lfa max-sr-frr-labels** *labels CLI option*. The default value is 2.

The data path models the backup path like a SR-TE LSP and thus uses a super-NHLFE pointing to the NHLFE of the first hop in the repair tunnel. That first hop corresponds to either an adjacency SID or a node SID of the P node.

There is the special case where the P node is adjacent to the node computing the TI-LFA backup, and the Q node is the same as the P node or adjacent to the P node. In this case, the data path at the computing router pushes either zero labels or one label for the adjacency SID between P and Q nodes. The backup path uses a regular NHLFE in this case like in base LFA or remote LFA features. *Figure 6: Selecting Link-Protect TI-LFA Backup Path* shows and example of a single label in the backup NHLFE.

2.1.8.3 Node Protection Support in TI-LFA and Remote LFA

This feature extends the Remote LFA and TI-LFA features by adding support for node protection. The extensions are additions to the original link-protect LFA SPF algorithm.

When node protection is enabled, the router prefers a node-protect over a link-protect repair tunnel for a given prefix if both are found in the Remote LFA or TI-LFA SPF computations. This feature protects against the failure of a downstream node in the path of the prefix of a node SID except for the node owner of the node SID.

2.1.8.3.1 Feature Configuration

The following are the CLI commands to configure the remote LFA and TI-LFA node protection feature.

```
configure
        – router
            – [no] isis
                – [no] loopfree-alternates
                    – [no] remote-lfa [max-pq-cost 0..4294967295, default=4261412864]
                        – [no] node-protect [max-pq-nodes 1..32, default=16]
                    – [no] ti-lfa [max-sr-frr-labels 0..3, default=2]
                        – [no] node-protect
                    – exclude
                        – [no] prefix-policy prefix-policy [prefix-policy…(up to 5 max)]
                    – exit
                – exit
```

A command is added to enable node-protect calculation to both Remote LFA (**node-protect** [**max-pq-nodes** <*1..32*, default=16>]) and TI-LFA (**node-protect**).

When the **node-protect** command is enabled, the router prefers a node-protect over a link-protect repair tunnel for a given prefix if both are found in the Remote LFA or TI-LFA SPF computations. The SPF computations may only find a link-protect repair tunnel for prefixes owned by the protected node. This feature protects against the failure of a downstream node in the path of the prefix of a node SID except for the node owner of the node SID.

The parameter **max-pq-nodes** in Remote LFA controls the maximum number of candidate PQ nodes found in the LFA SPFs for which the node protection check is performed. As explained in *Remote LFA Node-Protect Operation*, the node-protect condition means the router must run the original link-protect Remote LFA algorithm plus one extra forward SPF on behalf of each PQ node found, potentially after applying the **max-pq-cost** parameter, to check if the path from the PQ node to the destination does not traverse the protected node. Setting this parameter to a lower value means the LFA SPFs uses less computation time and resources but may result in not finding a node-protect repair tunnel. The default value is 16.

2.1.8.3.2 TI-LFA Node-Protect Operation

The SR OS supports the node-protect extensions to the TI-LFA algorithm as described in *draft-bashandy-rtgwg-segment-routing-ti-lfa-05*.

Figure 9: Application of the TI-LFA Algorithm for Node Protection shows a simple topology to illustrate the operation of the node-protect in the *TI-LFA Algorithm*.

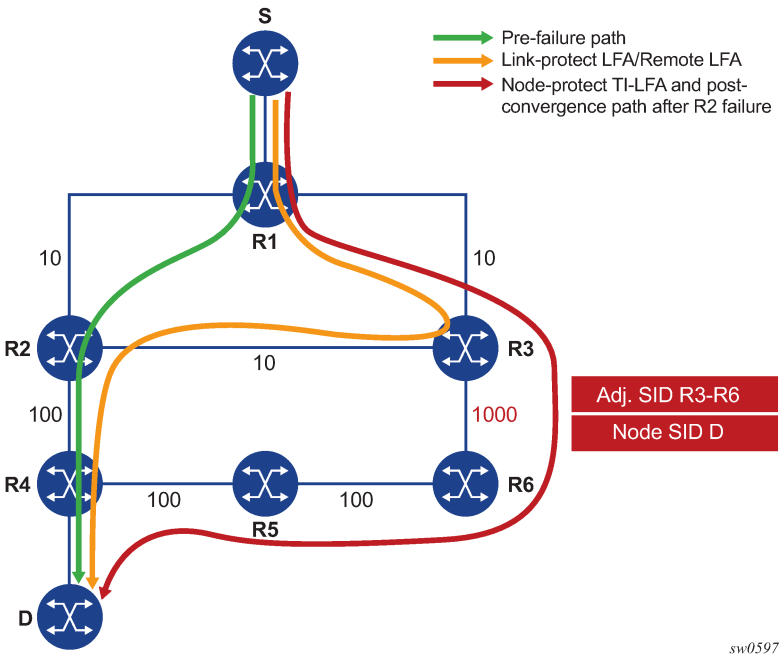

Figure 9: Application of the TI-LFA Algorithm for Node Protection

The first change is that the algorithm has to protect a node instead of a link.

The following topology computations pertain to *Figure 9: Application of the TI-LFA Algorithm for Node Protection*:

For each destination prefix D, R1 programs the TI-LFA repair tunnel (**max-sr-frr-labels=1**):

1. For prefixes other than those owned by node R2 and R3, R1 programs a node-protect repair tunnel to the P-Q pair R3-R6 by pushing the SID of adjacency R3-R6 on top of the SID for destination D and programming a next-hop of R3.
2. For prefixes owned by node R2, R1 runs the link-protect TI-LFA algorithm and programs a simple link-protect repair tunnel which consists of a backup next-hop of R3 and pushing no additional label on top of the SID for the destination prefix.
3. Prefixes owned by node R3 are not impacted by the failure of R2 since their primary next-hop is R3.

Procedure

1. Compute post-convergence SPF on the topology without the protected node.

   In *Figure 9: Application of the TI-LFA Algorithm for Node Protection*, R1 computes TI-LFA on the topology without the protected node R2 and finds a single post-convergence path to destination D via R3 and R6.

   Prefixes owned by all other nodes in the topology have a post-convergence path via R3 and R6 except for prefixes owned by node R2. The latter uses the link R3-R2 and they can only benefit from link protection.

2. Compute extended P-Space of R1 with respect to protected node R2 on the post-convergence paths.

   This is the set of nodes Yi in the post-convergence paths that are reachable from R1 neighbors, other than protected node R2, without any path transiting the protected node R2.

   R1 computes an LFA SPF rooted at each of its neighbors within the post-convergence paths, for example, R3, using the following equation:

   ```
   Distance_opt(R3, Yi) < Distance_opt(R3, R2) + Distance_opt(R2, Yi)
   ```

   Where:

   Distance_opt(A,B) is the shortest distance between A and B.

   The extended P-space calculation yields node R3 only.

3. Compute Q-space of R1 with respect to protected link R1-R2 on the post-convergence paths.

   This is the set of nodes Zi in the post-convergence paths from which node R2 can be reached without any path transiting the protected link R1-R2.

   ```
   Distance_opt(Zi, R2) < Distance_opt(Zi, R1) + Distance_opt(R1, R2)
   ```

   The reverse SPF for the Q-space calculation is the same as in the link-protect algorithm and uses the protected node R2 as the proxy for all destination prefixes. Note that if the Q-space were to be computed with respect to the protected node R2 instead of link R1-R2, a reverse SPF would have to be done to each destination D which is very costly and would not scale. Computing Q-space with respect to link R1-R2 however means the algorithm only guarantees the path from the computing node to the Q node is node-protecting. The path from the Q node to the destination Dis not guaranteed to avoid the protected node R2. The intersection of the Q-space with post-convergence path is modified in the next step to mitigate this risk.

   This step yields nodes R3, R4, R5, and R6.

4. For each post-convergence path, search for the closest Q-node to destination D and select the closest P-node to this Q-node, up to a number of labels corresponding to the value of **ti-lfa max-sr-frr-labels** *labels*.

   This step yields the following P-Q sets depending on the value of the parameter **max-sr-frr-labels**:

   - **max-sr-frr-labels**=**0**, R3 is the closest Q node to the destination D and R3 is the only P node. This case is the one which results in link protection via PQ node R3.
   - **max-sr-frr-labels**=**1**, R6 is the closest Q node to the destination D and R3 is the only P node. The repair tunnel for this case uses the SID of the adjacency over link R3-R6 and is illustrated in *Figure 9: Application of the TI-LFA Algorithm for Node Protection*.
   - **max-sr-frr-labels**=**2**, R5 is the closest Q node to the destination D and R3 is the only P node. The repair tunnel for this case uses the SIDs of the adjacencies over links R3-R6 and R6-R5.
   - **max-sr-frr-labels**=**3**, R4 is the closest Q node to the destination D and R3 is the only P node. The repair tunnel for this case uses the SIDs of the adjacencies over links R3-R6, R6-R5, and R5-R4.

   Note this step of the algorithm is modified from link protection which prefers Q nodes which are the closest to the computing router R1. This is to minimize the probability that the path from the Q node to the destination D goes via the protected node R2 as explained in step2. There is however still a probability that the found P-Q set achieves link protection only.

5. Select the P-Q Set.

   If a candidate P-Q set is found on each of the multiple ECMP post-convergence paths in step 4, the following selection rules are applied in ascending order to select a single set:

   - lowest number of labels
   - lowest next-hop router-id
   - lowest interface index if same next-hop router-id

   If multiple parallel links with adjacency SID exist between the P and Q nodes of the selected P-Q set, the following rules are used to select one of them:

   - Adjacency SID with lowest metric
   - Adjacency SID with the lowest SID value if same lowest metric

2.1.8.3.3 Remote LFA Node-Protect Operation

SR OS supports the node-protect extensions to the Remote LFA algorithm as described in RFC 8102.

Remote LFA follows a similar algorithm as TI-LFA but does not limit the scope of the calculation of the extended P-Space and of the Q-Space to the post-convergence paths.

Remote LFA adds an extra forward SPF on behalf of the PQ node to make sure that for each destination the selected PQ node does not use a path via the protected node.

Figure 10: Application of the Remote LFA Algorithm for Node Protection shows a slightly modified topology from that in *TI-LFA Feature Interaction and Limitations*. A new node R7 is added to the top ring and the metric for link R3-R6 is modified to 100.

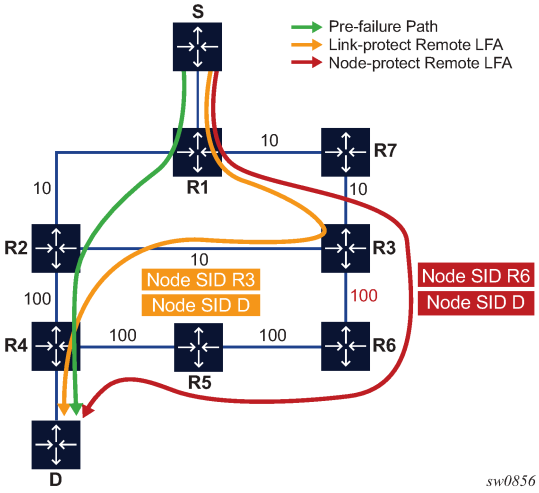

Figure 10: Application of the Remote LFA Algorithm for Node Protection

Applying the node protect remote LFA algorithm to this topology yields the following steps:

Procedure

1. Compute extended P-Space of R1 with respect to protected node R2.

   This is the set of nodes Yi which are reachable from R1 neighbors, other than protected node R2, without any path transiting the protected node R2.

   R1 computes a LFA SPF rooted at each of its neighbors, in this case, R7, using the following equation:

   ```
   Distance_opt(R7, Yi) < Distance_opt(R7, R2) + Distance_opt(R2, Yi)
   ```

   Where Distance_opt(A,B) is the shortest distance between A and B.

   Nodes R7, R3 and R6 satisfy this inequality.

2. Compute Q-space of R1 with respect to protected link R1-R2.

   This is the set of nodes Zi from which node R2 can be reached without any path transiting the protected link R1-R2.

   ```
   Distance_opt(Zi, R2) < Distance_opt(Zi, R1) + Distance_opt(R1, R2)
   ```

   The reverse SPF for the Q-space calculation is the same as in the remote LFA link-protect algorithm and uses the protected node R2 as the proxy for all destination prefixes.

   This step yields nodes R3, R4, R5, and R6.

   Therefore, the candidate PQ nodes after this step are nodes R3 and R6.

3. For each PQ node found, run a forward SPF to each destination D.

   This step is required to select only the subset of PQ nodes which does not traverse protected node R2.

   ```
   Distance_opt(PQi, D) < Distance_opt(PQi, R2) + Distance_opt(R2, D)
   ```

   Of the candidates PQ nodes R3 and R6, only PQ node R6 satisfies this inequality.

   Note this step of the algorithm is applied to the subset of candidate PQ nodes out of steps 1 and 2 and to which the parameter **max-pq-cost** was already applied. This subset is further reduced in this step by retaining the candidate PQ nodes which provide the highest coverage among all protected nodes in the topology and which number does not exceed the value of parameter **max-pq-nodes**.

   In case of multiple candidate PQ nodes out of this step, the detailed selection rules of a single PQ node from the candidate list is provided in Step*4*.

4. Select a PQ Node.

   If multiple PQ nodes satisfy the criteria in all the above steps, then R1 further selects the PQ node as follows:

   a. selects the lowest IGP cost from R1
   b. If more than one remains, R1 selects the PQ node reachable via the neighbor with the lowest router-id (OSPF) or system-id (ISIS);
   c. If more than one remains, R1 selects the PQ node with the lowest router-id (OSPF) or system-id (ISIS).

For each destination prefix D, R1 programs the remote LFA backup path:

1. For prefixes of R5, R4 or downstream of R4, R1 programs a node-protect remote LFA repair tunnel to the PQ node R6 by pushing the SID of node R6 on top of the SID for destination D and programming a next-hop of R7.
2. For prefixes owned by node R2, R1 runs the link-protect remote LFA algorithm and programs a simple link-protect repair tunnel which consists of a backup next-hop of R7 and pushing the SID of PQ node R3 on top of the SID for the destination prefix D.
3. Prefixes owned by nodes R7, R3, and R6 are not impacted by the failure of R2 since their primary next-hop is R7.

2.1.8.3.4 TI-LFA and Remote LFA Node Protection Feature Interaction and Limitations

LFA Protection Option Applicability describes the order of activation of the various LFA types on a per prefix basis: TI-LFA, followed by base LFA, followed by remote LFA.

Node protection is enabled for TI-LFA and remote LFA separately. The base LFA prefers node protection over link protection.

The order of activation of the LFA types supersedes the protection type (node versus link). Consequently, it is possible that a prefix can be programmed with a link-protect backup next-hop by the more preferred LFA type. For example, a prefix is programmed with the only link-protect backup next-hop found by the base LFA while there exists a node-protect remote LFA next-hop.

2.1.8.4 LFA Policies

2.1.8.4.1 Application of LFA Policy to a Segment Routing Node SID Tunnel

When a route next-hop policy template is applied to an interface, the LFA backup selection algorithm is extended to also apply to IPv4/IPv6 SR-ISIS, and IPv4 SR-OSPF node-SID tunnels in which a primary next hop is reachable using that interface. The extension applies to base LFA, Remote LFA (RLFA), and Topology-Independent LFA (TI-LFA).

The following general rules apply across all LFA methods.

- The LFA policy constraints admin-group (**include-group** and **exclude-group**) and SRLG (**srlg-enable**) are only checked against the outgoing interface used by the LFA/RLFA/TI-LFA backup path.
- The LFA policy parameter **protection-type** {**link** | **node**}, which controls the preference among link and node protection backup types, applies to all LFA methods.

  Base LFA automatically computes both protection types but prefers, on a prefix basis, link-protect over node-protect backup next hop by default.

  By default, RLFA and TI-LFA only perform link-protect backup path computation unless the optional command **node-protect** is enabled, in which case, the preference is reversed.

  For all three LFA methods, when the LFA policy enables a preference for link-protect or node-protect, the backup path is selected from the computed paths based on the configuration for the individual LFA method protection preference and the outcome (node-protect or link-protect) of the actual computation

within each method. Note, however, that on a per-destination prefix basis, the post-convergence constraint of TI-LFA is selected over the LFA protection type in all cases. The selection rule uses the TI-LFA backup (if one exists), even if it is of a less-preferred protection type than the one backup path computed by base LFA and RLFA.

For example, assume that an LFA policy with **protection-type=node** is applied to an ISIS interface and the **node-protect** command is enabled in both RLFA and TI-LFA contexts in this ISIS instance. If TI-LFA found a link-protect backup path for the destination prefix of a SR-ISIS tunnel, it is always selected over the base LFA node-protect and RLFA node-protect backup paths.

The outcomes of LFA policy selections for specified destination prefixes of SR tunnels are summarized in *Table 1: Outcome of LFA Policy with protection-type=node* and *Table 2: Outcome of LFA Policy with protection-type=link* .

Table 1: Outcome of LFA Policy with protection-type=node

	LFA Policy protection-type=node								
	Base LFA (LFA) Outcome								
	none			link-protect			node-protect		
	TI-LFA Outcome			TI-LFA Outcome			TI-LFA Outcome		
	none	link-protect	node-protect	none	link-protect	node-protect	none	link-protect	node-protect
RLFA Outcome									
—	—	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA
link-protect	RLFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA

	LFA Policy protection-type=node								
	Base LFA (LFA) Outcome								
	none			link-protect			node-protect		
	TI-LFA Outcome			TI-LFA Outcome			TI-LFA Outcome		
	none	link-protect	node-protect	none	link-protect	node-protect	none	link-protect	node-protect
RLFA Outcome									
node-protect	RLFA	TI-LFA	TI-LFA	RLFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA

Table 2: Outcome of LFA Policy with protection-type=link

	LFA Policy protection-type=link								
	Base LFA (LFA) Outcome								
	none			link-protect			node-protect		
	TI-LFA Outcome			TI-LFA Outcome			TI-LFA Outcome		
	none	link-protect	node-protect	none	link-protect	node-protect	none	link-protect	node-protect
RLFA Outcome									
—	—	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA
link-protect	RLFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA
node-protect	RLFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA	LFA	TI-LFA	TI-LFA

- LFA policy parameter **nh-type** {**ip** | **tunnel**}, which controls preference among the backup of type IP and type tunnel (IGP shortcut), is not applicable to RLFA and TI-LFA backup paths.

  However, the parameter applies if the LFA policy results in selecting a base LFA backup and the user-enabled resolution of SR-ISIS or SR-OSPF tunnel over IGP shortcut using RSVP-TE LSP.
- When configured on an interface, the route next-hop policy template applies to destination prefixes of:

  – IPv4 and IPv6 SR-ISIS node SID tunnels and
  – IPv4 SR-OSPF node SID tunnels
  – where the primary next hop is reachable using that interface

  The route next-hop policy template also indirectly applies to:

  – IPv4 or IPv6 SR-TE LSPs
  – IPv4 or IPv6 SR policies that use any of the previously mentioned SR tunnels as the top SID in their SID list

  Finally, the LFA policy indirectly applies to IPv4 LDP FECs when the LDP **fast-reroute backup-sr-tunnel** option is enabled and the FEC is protected with a SR tunnel.
- An LFA policy, applied to an interface cannot be selectively enabled or disabled per LFA method.
- As a result of these rules, at most one backup path will remain in each LFA method. In that case, the selection preference is as follows:

  1. TI-LFA backup IP next hop or repair tunnel
  2. Base LFA backup next hop

     This can be of type IP (default or if **nh-type** type preference set to **ip**) or of type tunnel (**nh-type** type preference is set to **tunnel** and family SRv4 or SRv6 resolves to IGP shortcut using RSVP-TE LSP).
  3. Remote LFA repair tunnel

2.1.8.4.2 Application of LFA Policy to Adjacency SID Tunnel

The modifications to TI-LFA and RLFA as described in *Application of LFA Policy to a Segment Routing Node SID Tunnel* are also applied to adjacency SID tunnel in a similar fashion.

The LFA selection algorithm for an adjacency to a neighbor is modified by applying the LFA policy of the link of the protected adjacency. It adheres to the following preference order:

1. Adjacency of an alternate parallel link to the same neighbor, determined as follows:

   a. apply admin-group and SRLG constraints of the LFA policy of the link of the protected adjacency
   b. select the adjacency with best admin-group(s) according to the preference specified in the value of the **include-group** option in the route next-hop policy template
   c. select the adjacency with lowest metric
   d. select the adjacency to the neighbor with the lowest router ID (OSPF) or system ID (IS-IS), and the lowest metric
   e. select the adjacency over the lowest interface index, and the lowest neighbor router ID (OSPF) or system ID (IS-IS)

2. ECMP next hop to a node-SID of the same neighbor, determined as follows:

   a. apply admin-group and SRLG constraints of the LFA policy of the link of the protected adjacency
   b. select the next hop with the best admin-group(s) according to the preference specified in the value of the **include-group** option in the route next-hop policy template
   c. select the next hop with lowest metric
   d. select the next hop to the neighbor with the lowest router ID (OSPF) or system ID (ISIS), and the lowest metric
   e. select the next hop over the lowest interface index, and the lowest neighbor router ID (OSPF) or system ID (IS-IS)
3. LFA backup outcome of a of a node SID of the same neighbor

   a. select a LFA backup with an outgoing link that does not conflict with the LFA policy of the link of the protected adjacency

   Note: If a different LFA policy was already applied in the computation of the LFA backup of the node SID of the neighbor, it is possible that some links to that node SID may have been eliminated before applying the LFA policy of the link of the protected adjacency.

2.1.8.4.3 Application of LFA Policy to Backup Node SID Tunnel

The backup node SID feature allows OSPF to use the path to an alternate ABR as an RLFA backup for forwarding packets of prefixes outside the local area or domain when the path to the primary ABR fails.

This feature reduces the label stack size by omitting the PQ node label if a regular RLFA algorithm is run.

The backup node SID algorithm consists of the following steps:

Procedure

1. Perform an SPF on the modified topology with the primary ABR removed.

   This action resolves the backup node SID using the path to the alternate ABR.

2. Install the ILM to use the backup node SID for transit traffic with the maximum ECMP next hops found in step*1*.

3. Use the backup node SID as an RLFA backup for prefixes outside the local area or domain. This step is modified as follows to select the backup node SID by applying the LFA policy corresponding the primary next hop of these prefixes:

   a. for each neighbor [Ni] found in step*1*, use the LFA policy to select the best next-hop interface
   b. among the remaining interfaces, use the LFA policy to select best [Ni] and select its interface

4. A backup node SID is always preferred to a regular RLFA backup. This does not change after applying the LFA policy because the main objective of the backup node SID feature is to reduce the label stack size of the backup tunnel.

2.1.8.4.4 Configuration Example of LFA Policy use in Remote LFA and TI-LFA

Figure 11: Application of LFA Policy to RLFA and TI-LFA shows a sample network topology that uses the OSPF routing protocol and in which the user assigns an SRLG ID to each group of OSPF links to represent fate-sharing among the links in the group. Assume the router **ecmp** value is set to **1**.

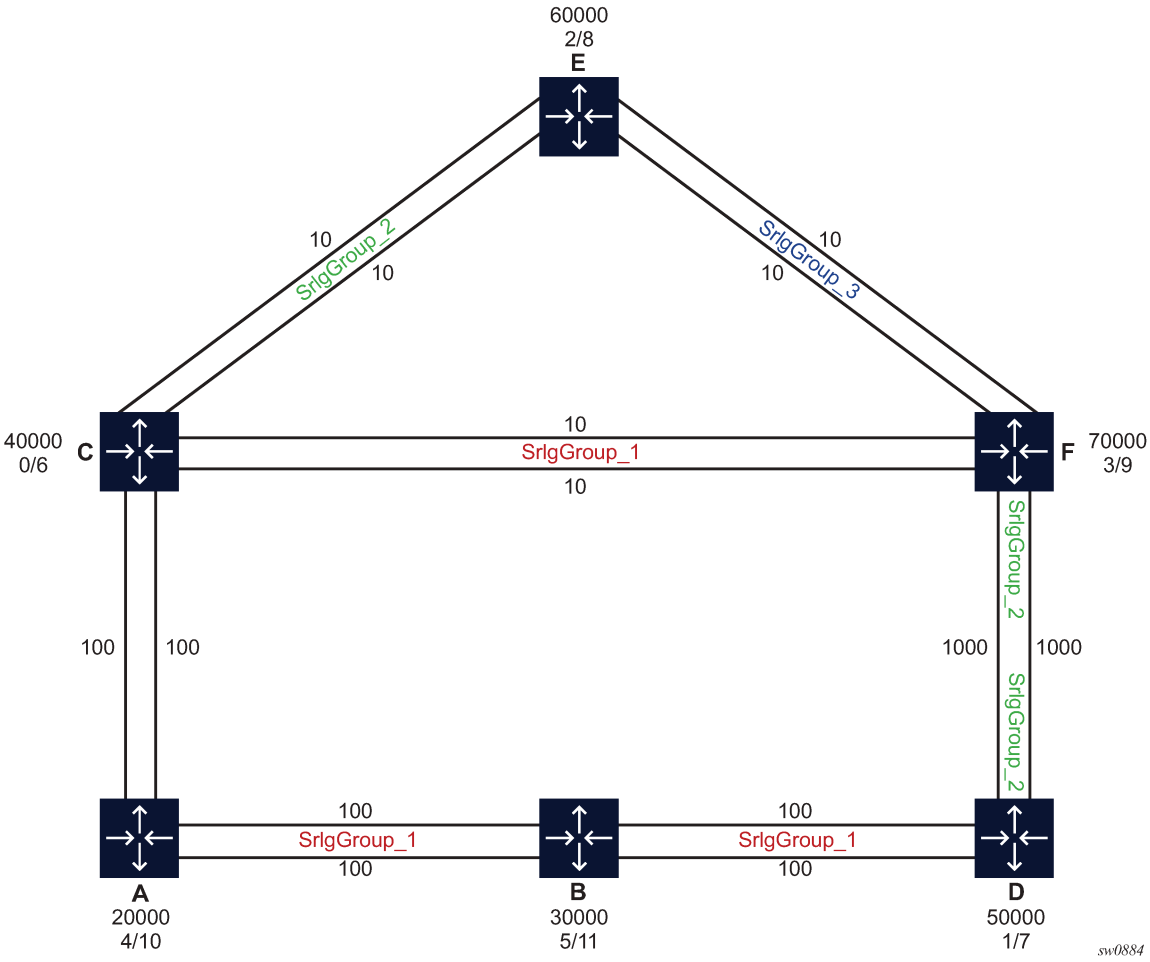

Figure 11: Application of LFA Policy to RLFA and TI-LFA

The user wants to enforce that the LFA backup computed and programmed by each node for a given destination prefix avoids the SRLG ID of the primary next hop of that prefix. To that effect, the user applies an LFA policy to each link that is used as a primary next hop to reach destination prefixes.

For instance, node F uses the top interface to node C as the primary next hop for the SR-OSPF tunnel to the SID of node C. The LFA policy states that the LFA backup must exclude outgoing interfaces which are member of SRLG ID of the interface of the primary next hop. Therefore, node F must select an LFA backup that avoids SRLG ID=**SrlgGroup_1**.

Node F enabled base LFA, remote LFA with **node-protect**, and TI-LFA with **node-protect** on the OSPF routing instance. The LFA SPF yields the following candidate LFA backup paths for the tunnel to the SID of node C:

1. Base LFA returns two backup paths: next hop over the second interface to C (cost 10) and next hop over the interface to node E (cost 20).

   After applying the LFA policy, only next hop over the interface to node E (cost 20) remains. The second interface to node C is also a member of SRLG ID=**SrlgGroup_1** and therefore the LFA next hop using it is excluded.
2. TI-LFA returns a single backup path: the next hop over the second interface to C (cost 10).

   After applying the LFA policy, no LFA backup path remains.
3. Remote LFA returns two backup paths: one backup path by PQ node C over the second interface to C (cost 10) and one by PQ node E over the interface to node E (cost 20).

   After applying the LFA policy, only the backup path by PQ node E over the interface to node E (cost 20) remains.
4. The LFA backup paths found by all three LFA methods are only link-protecting because node C is a neighbor of node F.
5. The final outcome is the selection among the LFA methods and base LFA is preferred to RLFA; therefore, next hop over the interface to node E (cost 20) is selected and programmed by node F as the backup path for the SR-OSPF tunnel to the SID of node C.
6. The adjacency from node F to node C over first interface to node C also inherits the same LFA backup path as the node SID of C since the same LFA policy applies.

The following are excerpts of the CLI configuration of node F in this specific example. The commands relevant to the LFA policy applied to link F-C as identified by arrows.

In addition, the output of show commands in node F highlights both the primary and the link-protect base LFA backup for both the node SID tunnel to C and the adjacency SID tunnel over the first interface to node C.

Because C is the termination for both its node SID and the adjacency SID tunnels from node F, only link protection can be provided as shown by the output of show command **tools**>**dump**>**router**>**ospf sr-database** (field L(R)). However, the output of the same show command for the tunnel to the SID of node D indicates the base LFA backup over the direct interface to node D is node-protecting (field Tn(R)).

```
 *A:Dut-F>config>router# info
----------------------------------------------
#----------------------------------------------------
echo "IP Configuration"
#----------------------------------------------------
        if-attribute                            <-------
            srlg-group "SrlgGroup_1" value 1    <-------
            srlg-group "SrlgGroup_2" value 2
            srlg-group "SrlgGroup_3" value 3
        exit
        route-next-hop-policy                   <-------
            begin                               <-------
            template "templateSrlgGroup_1"      <-------
                srlg-enable
            exit
            template "templateSrlgGroup_2"
                srlg-enable
            exit
            template "templateSrlgGroup_3"
                srlg-enable
            exit
            commit
        exit
```

```
        interface "DUTF_TO_DUTC.1.0"          <-------
            address 1.0.36.6/24
            secondary 51.0.36.6/24
            port 1/1/4:1
            mac 00:00:00:00:00:06
            ipv6
                address 3ffe::100:2406/120 primary-preference 1
                address 3ffe::3300:2406/120 primary-preference 2
            exit
            if-attribute                      <-------
                srlg-group "SrlgGroup_1"      <-------
            exit
            no shutdown
        exit
        interface "DUTF_TO_DUTC.2.0"          <-------
            address 2.0.36.6/24
            secondary 52.0.36.6/24
            port 1/1/4:2
            mac 00:00:00:00:00:06
            ipv6
                address 3ffe::200:2406/120 primary-preference 1
                address 3ffe::3400:2406/120 primary-preference 2
            exit
            if-attribute                      <-------
                srlg-group "SrlgGroup_1"      <-------
            exit
            no shutdown
        exit
        interface "DUTF_TO_DUTD.1.0"
            address 1.0.46.6/24
            secondary 51.0.46.6/24
            port 1/1/1:1
            mac 00:00:00:00:00:06
            ipv6
                address 3ffe::100:2e06/120 primary-preference 1
                address 3ffe::3300:2e06/120 primary-preference 2
            exit
            if-attribute
                srlg-group "SrlgGroup_2"
            exit
            no shutdown
        exit
        interface "DUTF_TO_DUTD.2.0"
            address 2.0.46.6/24
            secondary 52.0.46.6/24
            port 1/1/1:2
            mac 00:00:00:00:00:06
            ipv6
                address 3ffe::200:2e06/120 primary-preference 1
                address 3ffe::3400:2e06/120 primary-preference 2
            exit
            if-attribute
                srlg-group "SrlgGroup_2"
            exit
            no shutdown
        exit
        interface "DUTF_TO_DUTE.1.0"          <-------
            address 1.0.56.6/24
            secondary 51.0.56.6/24
            port 1/1/2:1
            mac 00:00:00:00:00:06
            ipv6
                address 3ffe::100:3806/120 primary-preference 1
                address 3ffe::3300:3806/120 primary-preference 2
            exit
            if-attribute                      <-------
```

```
                srlg-group "SrlgGroup_3"      <-------
            exit
            no shutdown
        exit
        interface "DUTF_TO_DUTE.2.0"          <-------
            address 2.0.56.6/24
            secondary 52.0.56.6/24
            port 1/1/2:2
            mac 00:00:00:00:00:06
            ipv6
                address 3ffe::200:3806/120 primary-preference 1
                address 3ffe::3400:3806/120 primary-preference 2
            exit
            if-attribute                      <-------
                srlg-group "SrlgGroup_3"      <-------
            exit
            no shutdown
        exit
        interface "loopbackF.1.0"
            address 1.0.66.6/32
            secondary 51.0.66.6/32
            loopback
            ipv6
                address 3ffe::100:4206/128 primary-preference 1
                address 3ffe::3300:4206/128 primary-preference 2
            exit
            no shutdown
        exit
        interface "loopbackF.2.0"
            address 2.0.66.6/32
            secondary 52.0.66.6/32
            loopback
            ipv6
                address 3ffe::200:4206/128 primary-preference 1
                address 3ffe::3400:4206/128 primary-preference 2
            exit
            no shutdown
        exit
        interface "system"
            address 10.20.1.6/32
            ipv6
                address 3ffe::a14:106/128
            exit
            no shutdown
        exit
        ip-fast-reroute
        router-id 10.20.1.6
#--------------------------------------------------
echo "MPLS Label Range Configuration"
#--------------------------------------------------
        mpls-labels
            sr-labels start 20000 end 80000
        exit
#--------------------------------------------------
echo "OSPFv2 Configuration"
#--------------------------------------------------
        ospf 0 10.20.1.6
            traffic-engineering
            database-export identifier 0
            advertise-router-capability area
            loopfree-alternates               <-------
                remote-lfa                    <-------
                    node-protect              <-------
                exit                          <-------
                ti-lfa max-sr-frr-labels 3    <-------
                    node-protect              <-------
```

```
            exit                            <-------
        exit                                <-------
        segment-routing
            prefix-sid-range start-label 70000 max-index 999
            egress-statistics
                adj-set
                adj-sid
                node-sid
            exit
            ingress-statistics
                adj-set
                adj-sid
                node-sid
            exit
            no shutdown
        exit
        area 0.0.0.0
            interface "system"
                node-sid index 9
                no shutdown
            exit
            interface "DUTF_TO_DUTC.1.0"       <-------
                interface-type point-to-point
                hello-interval 2
                dead-interval 10
                metric 10
                lfa-policy-map route-nh-template "templateSrlgGroup_1"  <-------
                no shutdown
            exit
            interface "DUTF_TO_DUTD.1.0"
                interface-type point-to-point
                hello-interval 2
                dead-interval 10
                metric 1000
                lfa-policy-map route-nh-template "templateSrlgGroup_2"
                no shutdown
            exit
            interface "DUTF_TO_DUTE.1.0"
                interface-type point-to-point
                hello-interval 2
                dead-interval 10
                metric 10
                lfa-policy-map route-nh-template "templateSrlgGroup_3"
                no shutdown
            exit
            interface "loopbackF.1.0"
                node-sid index 3
                no shutdown
            exit
            interface "DUTF_TO_DUTC.2.0"
                interface-type point-to-point
                hello-interval 2
                dead-interval 10
                metric 10
                lfa-policy-map route-nh-template "templateSrlgGroup_4"
                no shutdown
            exit
            interface "DUTF_TO_DUTD.2.0"
                interface-type point-to-point
                hello-interval 2
                dead-interval 10
                metric 1000
                lfa-policy-map route-nh-template "templateSrlgGroup_5"
                no shutdown
            exit
            interface "DUTF_TO_DUTE.2.0"
```

```
                    interface-type point-to-point
                    hello-interval 2
                    dead-interval 10
                    metric 10
                    lfa-policy-map route-nh-template "templateSrlgGroup_6"
                    no shutdown
                exit
                interface "loopbackF.2.0"
                    node-sid index 15
                    no shutdown
                exit
            exit
            no shutdown
        exit
-------------------------------------------
*A:Dut-F# tools dump router segment-routing tunnel
===============================================================================================================
Legend: (B) - Backup Next-hop for Fast Re-Route

        (D) - Duplicate

label stack is ordered from top-most to bottom-most

===============================================================================================================
---------------------------------------------------------------------------------------------------------------
---+
 Prefix
   |
 Sid-Type        Fwd-Type       In-Label  Prot-Inst
   |
                 Next Hop(s)                                        Out-Label(s) Interface/Tunnel-
ID |
---------------------------------------------------------------------------------------------------------------
---+
 1.0.33.3                                          <-------
 Node            Orig/Transit   70000     OSPF-0     <-------
                 1.0.36.3                                         40000       DUTF_TO_DUTC.1.0
 <-------
              (B)1.0.56.5                                         60000       DUTF_TO_DUTE.1.0
 <-------
 1.0.44.4                                          <-------
 Node            Orig/Transit   70001     OSPF-0     <-------
                 1.0.36.3                                         40001       DUTF_TO_DUTC.1.0
 <-------
              (B)1.0.46.4                                         50001       DUTF_TO_DUTD.1.0
 <-------
 1.0.55.5
 Node            Orig/Transit   70002     OSPF-0
                 1.0.56.5                                         60002       DUTF_TO_DUTE.1.0
              (B)1.0.36.3                                         40002       DUTF_TO_DUTC.1.0
 1.0.66.6
 Node            Terminating    70003     OSPF-0
 1.0.11.1
 Node            Orig/Transit   70004     OSPF-0
                 1.0.36.3                                         40004       DUTF_TO_DUTC.1.0
              (B)1.0.46.4                                         50004       DUTF_TO_DUTD.1.0
 1.0.22.2
 Node            Orig/Transit   70005     OSPF-0
                 1.0.36.3                                         40005       DUTF_TO_DUTC.1.0
              (B)1.0.46.4                                         50005       DUTF_TO_DUTD.1.0
 10.20.1.3
 Node            Orig/Transit   70006     OSPF-0
                 1.0.36.3                                         40006       DUTF_TO_DUTC.1.0
              (B)1.0.56.5                                         60006       DUTF_TO_DUTE.1.0
 10.20.1.4
 Node            Orig/Transit   70007     OSPF-0
```

```
                1.0.36.3                                                  40007       DUTF_TO_DUTC.1.0
             (B)1.0.46.4                                                  50007       DUTF_TO_DUTD.1.0
10.20.1.5
Node            Orig/Transit   70008     OSPF-0
                1.0.56.5                                                  60008       DUTF_TO_DUTE.1.0
             (B)1.0.36.3                                                  40008       DUTF_TO_DUTC.1.0
10.20.1.6
Node            Terminating    70009     OSPF-0
10.20.1.1
Node            Orig/Transit   70010     OSPF-0
                1.0.36.3                                                  40010       DUTF_TO_DUTC.1.0
             (B)1.0.46.4                                                  50010       DUTF_TO_DUTD.1.0
10.20.1.2
Node            Orig/Transit   70011     OSPF-0
                1.0.36.3                                                  40011       DUTF_TO_DUTC.1.0
             (B)1.0.46.4                                                  50011       DUTF_TO_DUTD.1.0
2.0.33.3
Node            Orig/Transit   70012     OSPF-0
                1.0.36.3                                                  40012       DUTF_TO_DUTC.1.0
             (B)1.0.56.5                                                  60012       DUTF_TO_DUTE.1.0
2.0.44.4
Node            Orig/Transit   70013     OSPF-0
                1.0.36.3                                                  40013       DUTF_TO_DUTC.1.0
             (B)1.0.46.4                                                  50013       DUTF_TO_DUTD.1.0
2.0.55.5
Node            Orig/Transit   70014     OSPF-0
                1.0.56.5                                                  60014       DUTF_TO_DUTE.1.0
             (B)1.0.36.3                                                  40014       DUTF_TO_DUTC.1.0
2.0.66.6
Node            Terminating    70015     OSPF-0
2.0.11.1
Node            Orig/Transit   70016     OSPF-0
                1.0.36.3                                                  40016       DUTF_TO_DUTC.1.0
             (B)1.0.46.4                                                  50016       DUTF_TO_DUTD.1.0
2.0.22.2
Node            Orig/Transit   70017     OSPF-0
                1.0.36.3                                                  40017       DUTF_TO_DUTC.1.0
             (B)1.0.46.4                                                  50017       DUTF_TO_DUTD.1.0
2.0.56.5
Adjacency       Transit        524282    OSPF-0
                2.0.56.5                                                  3           DUTF_TO_DUTE.2.0
             (B)1.0.56.5                                                  3           DUTF_TO_DUTE.1.0
 2.0.46.4
Adjacency       Transit        524283    OSPF-0
                2.0.46.4                                                  3           DUTF_TO_DUTD.2.0
             (B)1.0.36.3                                                  40001       DUTF_TO_DUTC.1.0
2.0.36.3
Adjacency       Transit        524284    OSPF-0
                2.0.36.3                                                  3           DUTF_TO_DUTC.2.0
             (B)1.0.36.3                                                  3           DUTF_TO_DUTC.1.0
1.0.56.5
Adjacency       Transit        524285    OSPF-0
                1.0.56.5                                                  3           DUTF_TO_DUTE.1.0
             (B)1.0.36.3                                                  40002       DUTF_TO_DUTC.1.0
1.0.46.4
Adjacency       Transit        524286    OSPF-0
                1.0.46.4                                                  3           DUTF_TO_DUTD.1.0
             (B)1.0.36.3                                                  40001       DUTF_TO_DUTC.1.0
1.0.36.3                                            <-------
Adjacency       Transit        524287    OSPF-0     <-------
                1.0.36.3                                                  3           DUTF_TO_DUTC.1.0
<----
             (B)1.0.56.5                                                  60000       DUTF_TO_DUTE.1.0
<----
-------------------------------------------------------------------------------------------------
---+
```

```
No. of Entries: 24
-------------------------------------------------------------------------------------------------
---+
*A:Dut-F#
*A:Dut-F#    tools dump router ospf sr-database
=================================================================================
Rtr Base OSPFv2 Instance 0 Segment Routing Database
=================================================================================
SID         Label St Type Prefix                                        Stitching
                                  AdvRtr           Area Flags          FRR
---------------------------------------------------------------------------------
0           70000 +R   T1 1.0.33.3/32                                      -   <-------
                           10.20.1.3         0.0.0.0 [NnP       ]  L(R)     <-------
1           70001 +R   T1 1.0.44.4/32                                      -   <-------
                           10.20.1.4         0.0.0.0 [NnP       ] Tn(R)     <-------
2           70002 +R   T1 1.0.55.5/32                                      -
                           10.20.1.5         0.0.0.0 [NnP       ]  L(R)
3           70003 +R  LT1 1.0.66.6/32                                      -
                           10.20.1.6         0.0.0.0 [NnP       ]     -
4           70004 +R   T1 1.0.11.1/32                                      -
                           10.20.1.1         0.0.0.0 [NnP       ] Tn(R)
5           70005 +R   T1 1.0.22.2/32                                      -
                           10.20.1.2         0.0.0.0 [NnP       ] Tn(R)
6           70006 +R   T1 10.20.1.3/32                                     -
                           10.20.1.3         0.0.0.0 [NnP       ]  L(R)
7           70007 +R   T1 10.20.1.4/32                                     -
                           10.20.1.4         0.0.0.0 [NnP       ] Tn(R)
8           70008 +R   T1 10.20.1.5/32                                     -
                           10.20.1.5         0.0.0.0 [NnP       ]  L(R)
9           70009 +R  LT1 10.20.1.6/32                                     -
                           10.20.1.6         0.0.0.0 [NnP       ]     -
10          70010 +R   T1 10.20.1.1/32                                     -
                           10.20.1.1         0.0.0.0 [NnP       ] Tn(R)
11          70011 +R   T1 10.20.1.2/32                                     -
                           10.20.1.2         0.0.0.0 [NnP       ] Tn(R)
12          70012 +R   T1 2.0.33.3/32                                      -
                           10.20.1.3         0.0.0.0 [NnP       ]  L(R)
13          70013 +R   T1 2.0.44.4/32                                      -
                           10.20.1.4         0.0.0.0 [NnP       ] Tn(R)
14          70014 +R   T1 2.0.55.5/32                                      -
                           10.20.1.5         0.0.0.0 [NnP       ]  L(R)
15          70015 +R  LT1 2.0.66.6/32                                      -
                           10.20.1.6         0.0.0.0 [NnP       ]     -
16          70016 +R   T1 2.0.11.1/32                                      -
                           10.20.1.1         0.0.0.0 [NnP       ] Tn(R)
17          70017 +R   T1 2.0.22.2/32                                      -
                           10.20.1.2         0.0.0.0 [NnP       ] Tn(R)
---------------------------------------------------------------------------------
No. of Entries: 18
---------------------------------------------------------------------------------
St:   R:reported  I:incomplete  W:wrong  N:not reported  F:failed
      +:SR-ack  -:no route
Type: L:local  M: mapping Srv  Tx: route type
FRR:  L:Lfa  R:RLfa  T:TiLfa  (R):Reported  (F):Failed
      Ln, Rn, Tn: FRR providing node-protection
=================================================================================
*A:Dut-F#
```

2.1.8.5 LFA Protection Using Segment Routing Backup Node SID

One of the challenges in MPLS deployments across multiple IGP areas or domains, such as in seamless MPLS design, is the provisioning of FRR local protection in access and metro domains that make use of a ring, a square, or a partial mesh topology. In order to implement IP, LDP, or SR FRR in these topologies,

the remote LFA feature must be implemented. Remote LFA provides a Segment Routing (SR) tunneled LFA next hop for an IP prefix, an LDP tunnel, or an SR tunnel. For prefixes outside of the area or domain, the access or aggregation router must push four labels: service label, BGP label for the destination PE, LDP/RSVP/SR label to reach the exit ABR/ASBR, and one label for the remote LFA next hop. Small routers deployed in these parts of the network have limited MPLS label stack size support.

Figure 12: Label Stack for Remote LFA in Ring Topology illustrates the label stack required for the primary next hop and the remote LFA next hop computed by aggregation node AGN2 for the inter-area prefix of a remote PE. For an inter-area BGP label unicast route prefix for which ABR1 is the primary exit ABR, AGN2 resolves the prefix to the transport tunnel of ABR1 and therefore, uses the remote LFA next hop of ABR1 for protection. The primary next hop uses two transport labels plus a service label. The remote LFA next hop for ABR1 uses PQ node AGN5 and pushes three transport labels plus a service label.

Seamless MPLS with Fast Restoration requires up to four labels to be pushed by AGN2, as shown in *Figure 12: Label Stack for Remote LFA in Ring Topology*.

Label Location	Label Name	Assigned By	Protocol	Use
Label 1 (Bottom)	Service (PW, VC) Label	Remote PE	MP-BGP, T-LDP	Identifies Service on Remote PE
Label 2	Inter-Area Label	ABR1	BGP-LU	Identifies Path to Remote PE
Label 3	Intra-Area	AGN1	LDP, RSVP, SR	Identifies Path to ABR1
Label 4 (Top)	R-LFA Label	AGN3	LDP, RSVP, SR	Identifies Path to AGN5

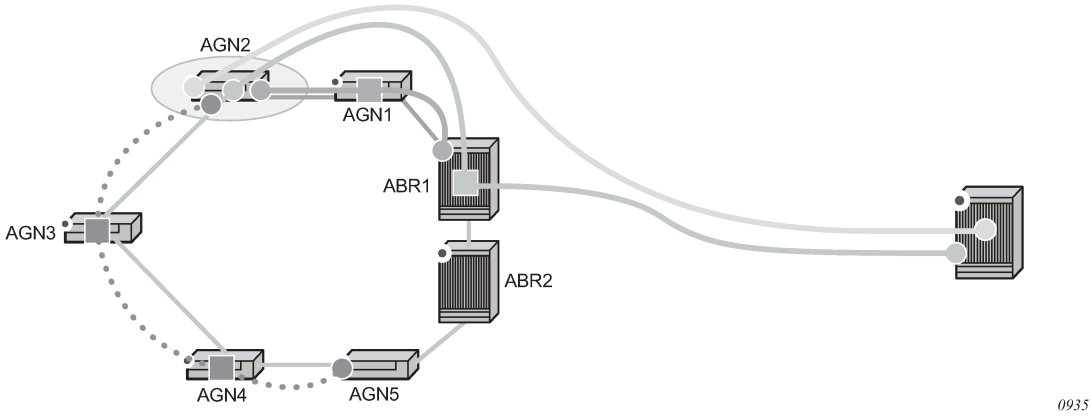

Figure 12: Label Stack for Remote LFA in Ring Topology

The objective of the LFA protection with a backup node SID feature is to reduce the label stack pushed by AGN2 for BGP label unicast inter-area prefixes. When link AGN2-AGN1 fails, packets are direct away from the failure and forwarded toward ABR2, which acts as the backup for ABR1 (and vice-versa when ABR2 is the primary exit ABR for the BGP label unicast inter-area prefix). This requires that ABR2 advertise a special label for the loopback of ABR1 that will attract packets normally destined for ABR1. These packets will be forwarded by ABR2 to ABR1 via the inter-ABR link.

As a result, AGN2 will push the label advertised by ABR2 to back up ABR1 on top of the BGP label for the remote PE and the service label. This keeps the label stack the same size for the LFA next hop to be the same size as that of the primary next hop. It is also the same size as the remote LFA next hop for the local prefix within the ring.

2.1.8.5.1 Configuring LFA Using Backup Node SID in OSPF

LFA using a backup node SID is enabled by configuring a backup node SID at an ABR/ASBR that acts as a backup to the primary exit ABR/ASBR of inter-area/inter-as routes learned as BGP labeled routes.

```
config>router>ospf>segment-routing$
        – backup-node-sid ip-prefix/prefix-length index 0..4294967295
        – backup-node-sid ip-prefix/prefix-length label 1..4294967295
```

The user can enter either a label or an index for the backup node SID.

Note: This feature only allows the configuration of a single backup node SID per OSPF instance and per ABR/ASBR. In other words, only a pair of ABR/ASBR nodes can back up each other in a an OSPF domain. Each time the user invokes the above command within the same OSPF instance, it overrides any previous configuration of the backup node SID. The same ABR/ASBR can, however, participate in multiple OSPF instances and provide a backup support within each instance.

2.1.8.5.2 Detailed Operation of LFA Protection Using Backup Node SID

As shown in *Figure 13: Backup ABR Node SID*, LFA for seamless MPLS supports environments where the boundary routers are either:

- ABR nodes that connect with Interior Border Gateway Protocol (IBGP) multiple domains, each using a different area of the same IGP instance
- ASBR nodes that connect domains running different IGP instances and use IBGP within a domain and External Border Gateway Protocol (EBGP) to the other domains

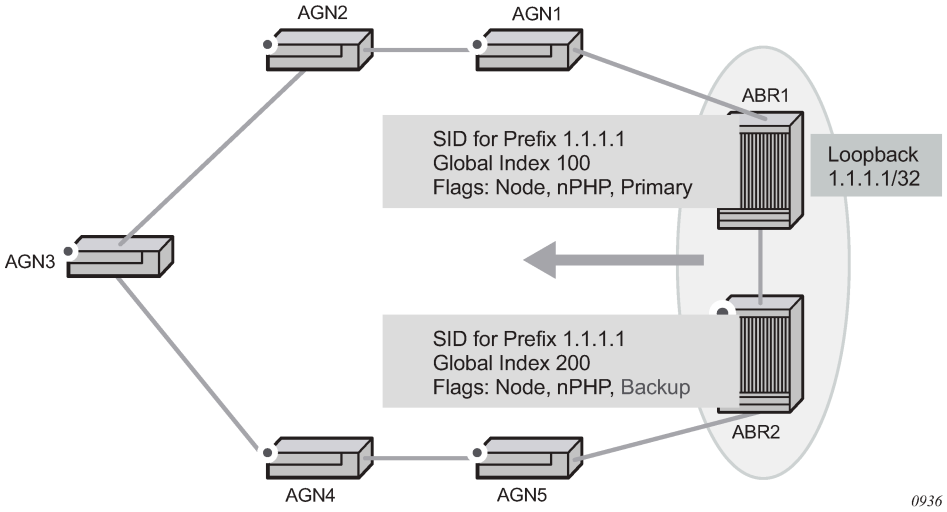

Figure 13: Backup ABR Node SID

The following steps describe the configuration and behavior of LFA Protection using Backup Node SID:

1. The user configures node SID 100 in ABR1 for its loopback prefix 1.1.1.1/32. This is the regular node SID. ABR1 advertises the prefix SID sub-TLV for this node SID in the IGP and installs the ILM using a unique label.

2. Each router receiving the prefix sub-TLV for node SID 100 resolves it as described in *Segment Routing in Shortest Path Forwarding*. Changes to the programming of the backup NHLFE of node SID 100 based on receiving the backup node SID for prefix 1.1.1.1/32 are defined in *Duplicate SID Handling*.
3. The user configures a backup node SID 200 in ABR2 for the loopback 1.1.1.1/32 of ABR1. The SID value must be different from that assigned by ABR1 for the same prefix. ABR2 installs the ILM, which performs a swap operation from the label of SID 200 to that of SID 100. The ILM must point to a direct link and next hop to reach 1.1.1.1/32 of ABR1 as its primary next hop. The IGP examines all adjacencies established in the same area as that of prefix 1.1.1.1/32 and determines which ones have ABR1 as a direct neighbor and with the best cost. If more than one adjacency has the best cost, the IGP selects the one with the lowest interface index. If there is no adjacency to reach ABR2, the prefix SID for the backup node is flushed and is not resolved. This is to prevent any other non-direct path being used to reach ABR1. As a result, any received traffic on the ILM of SID 200 traffic will be blackholed.
4. If resolved, ABR2 advertises the prefix SID sub-TLV for this backup node SID 200 and indicates in the SR Algorithm field that a modified SPF algorithm, referred to as "Backup-constrained-SPF", is required to resolve this node SID.
5. Each router receiving the prefix sub-TLV for the backup node SID 200 performs the following steps.

   The following resolution steps do not require a CLI command to be enabled.

   - The router determines which router is being backed up. This is achieved by checking the router ID owner of the prefix sub-TLV that was advertised with the same prefix but without the backup flag and which is used as the best route for the prefix. In this case, it should be ABR1. Then the router runs a modified SPF by removing node ABR1 from the topology to resolve the backup node SID 200. The primary next hop should point to the path to ABR2 in the counter clockwise direction of the ring.

     The router will not compute an LFA or a remote LFA for node SID 200 because the main SPF used a modified topology.
   - The router installs the ILM and primary NHLFE for the backup node SID.

     Only a swap label operation is configured by all routers for the backup node SID. There is no push operation, and no tunnel for the backup node SID is added into the TTM.
   - The router programs the backup node SID as the LFA backup for the SR tunnel to node SID of 1.1.1.1/32 of ABR1. In other words, each router overrides the remote LFA backup for prefix 1.1.1.1/32, which is normally PQ node AGN5.
   - If the router is adjacent to ABR1, for example AGN1, it also programs the backup node SID as the LFA backup for the protection of any adjacency SID to ABR1.
6. When node AGN2 resolves a BGP label route for an inter-area prefix for which the primary ABR exit router is ABR1, it will use the backup node SID of ABR1 as the remote LFA backup instead of the SID to the PQ node (AGN5 in this example) to save on the pushed label stack.

   AGN2 continues to resolve the prefix SID for any remote PE prefix that is summarized into the local area of AGN2 as usual. AGN2 programs a primary next hop and a remote LFA next hop. Remote LFA will use AGN5 as the PQ node and will push two labels, as it would for an intra-area prefix SID. There is no need to use the backup node SID for this prefix SID and force its backup path to go to ABR1. The backup path may exit from ABR2 if the cost from ABR2 to the destination prefix is shorter.
7. If the user excludes a link from LFA in the IGP instance (**config**>**router**>**ospf**>**area**>**interface**>**loopfree-alternate-exclude**), a backup node SID that resolves to that interface will not be used as a remote LFA backup in the same way as regular LFA or PQ remote LFA next hop behavior.

8. If the OSPF neighbor of a router is put into overload or if the metric of an OSPF interface to that neighbor is set to LSInfinity (0xFFFF), a backup node SID that resolves to that neighbor will not be used as a remote LFA backup in the same way as regular LFA or PQ remote LFA next hop behavior.
9. LFA policy is supported with a backup node SID. See *Application of LFA Policy to Backup Node SID Tunnel*.

2.1.8.5.3 Duplicate SID Handling

When the IGP issues or receives an LSA/LSP containing a prefix SID sub-TLV for a node SID or a backup node SID with a SID value that is a duplicate of an existing SID or backup node SID, the resolution in *Table 3: Handling of Duplicate SIDs* is followed.

Table 3: Handling of Duplicate SIDs

	New LSA/LSP			
Old LSA/LSP	Backup Node SID	Local Backup Node SID	Node SID	Local Node SID
Backup Node SID	Old	New	New	New
Local Backup Node SID	Old	Equal	New	New
Node SID	Old	Old	Equal/Old[1]	Equal/New[2]
Local Node SID	Old	Old	Equal/Old[1]	Equal/Old[1]

Note:

1. Equal/Old means the following.
   - If the prefix is duplicate, it is equal and no change is needed. Keep the old LSA/LSP.
   - If the prefix is not duplicate, still keep the old LSA/LSP.
2. Equal/New means the following.
   - If the prefix is duplicate, it is equal and no change is needed. Keep the old LSA/LSP.
   - If the prefix is not duplicate, pick a new prefix and use the new LSA/LSP.

2.1.8.5.4 OSPF Control Plane Extensions

All routers supporting OSPF control plane extensions must advertise support of the new algorithm "Backup-constrained-SPF" of value 2 in the SR-Algorithm TLV, which is advertised in the Router Information Opaque LSA. This is in addition to the default supported algorithm "IGP-metric-based-SPF" of

value 0. The following shows the encoding of the prefix SID sub-TLV to indicate a node SID of type backup and to indicate the modified SPF algorithm in the SR Algorithm field. The values used in the Flags field and in the Algorithm field are SR OS proprietary.

The new Algorithm (0x2) field and values are used by this feature.

```
0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |              Type             |             Length            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Flags     |   Reserved    |      MT-ID    |Algorithm (0x2)|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     SID/Index/Label (variable)                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Table 4: OSPF Control Plane Extension Fields lists OSPF control plane extension flag values.

Table 4: OSPF Control Plane Extension Fields

Field	Value
Type	2
Length	variable
Flags	1 octet field

The following flags are defined; the "B" flag is new:

```
     0  1  2  3  4  5  6  7
   +--+--+--+--+--+--+--+--+
   |  |NP|M |E |V |L | B|  |
   +--+--+--+--+--+--+--+--+
```

Table 5: OSPF Control Plane Extension Flags describes OSPF control plane extension flags.

Table 5: OSPF Control Plane Extension Flags

Flag	Description
NP-Flag	No-PHP flag If set, the penultimate hop must not pop the prefix SID before delivering the packet to the node that advertised the prefix SID.

Flag	Description
M-Flag	Mapping Server Flag If set, the SID is advertised from the Segment Routing Mapping Server functionality as described in I-D.filsfils-spring-segment-routing-ldp-interop.
E-Flag	Explicit-Null Flag If set, any upstream neighbor of the prefix SID originator must replace the prefix SID with a prefix SID having an Explicit-NULL value (0 for IPv4) before forwarding the packet.
V-Flag	Value/Index Flag If set, the prefix SID carries an absolute value. If not set, the prefix SID carries an index.
L-Flag	Local/Global Flag If set, the value/index carried by the prefix SID has local significance. If not set, then the value/index carried by this sub-TLV has global significance.
B-Flag	This flag is used by the Protection using backup node SID feature. If set, the SID is a backup SID for the prefix. This value is SR OS proprietary.
Other bits	Reserved These must be zero when sent and are ignored when received.
MT-ID	Multi-Topology ID, as defined in RFC 4915.
Algorithm	One octet identifying the algorithm the prefix SID is associated with. A value of (0x2) indicates the modified SPF algorithm, which removes from the topology the node that is backed up by the backup node SID. This value is SR OS proprietary.

Flag	Description
SID/Index/Label	Based on the V and L flags, it contains either: • a 32-bit index defining the offset in the SID/Label space advertised by this router • a 24-bit label where the 20 rightmost bits are used for encoding the label value

2.1.9 Segment Routing Data Path Support

A packet received with a label matching either a node SID or an adjacency SID is forwarded according to the ILM type and operation, as described in *Table 6: Data Path Support* .

Table 6: Data Path Support

Label type	Operation
Top label is a local node SID	Label is popped and the packet is further processed. If the popped node SID label is the bottom of stack label, the IP packet is looked up and forwarded in the appropriate FIB.
Top or next label is a remote node SID	Label is swapped to the calculated label value for the next-hop and forwarded according to the primary or backup NHLFE. With ECMP, a maximum of 32 primary next-hops (NHLFEs) are programmed for the same destination prefix and for each IGP instance. ECMP and LFA next-hops are mutually exclusive as per existing implementation.
Top or next label is an adjacency SID	Label is popped and the packet is forwarded out on the interface to the next-hop associated with this adjacency SID label. In effect, the data path operation is modeled like a swap to an implicit-null label instead of a pop.

Label type	Operation
Next label is BGP 3107 label	The packet is further processed according to the ILM operation as in current implementation. • The BGP label may be popped and the packet looked up in the appropriate FIB. • The BGP label may be swapped to another BGP label. • The BGP label may be stitched to an LDP label.
Next label is a service label	The packet is looked up and forwarded in the Layer 2 or VPRN FIB as in current implementation.

A router forwarding an IP or a service packet over an SR tunnel pushes a maximum of two transport labels with a remote LFA next-hop. This is illustrated in *Figure 14: Transport Label Stack in Shortest Path Forwarding with Segment Routing*.

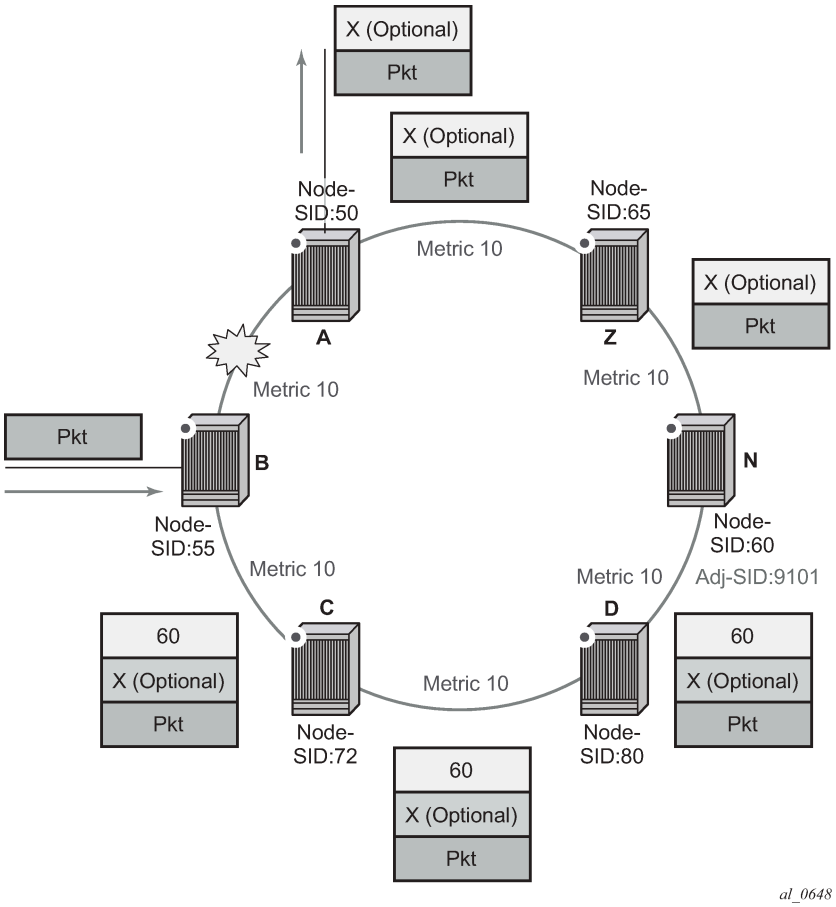

Figure 14: Transport Label Stack in Shortest Path Forwarding with Segment Routing

Assume that a VPRN service in node B forwards a packet received on a SAP to a destination VPN-IPv4 prefix X advertised by a remote PE2 via ASBR/ABR node A. Router B is in a segment routing domain while

PE2 is in an LDP domain. BGP label routes are used to distribute the PE /32 loopbacks between the two domains.

When node B forwards over the primary next-hop for prefix X, it pushes the node SID of the ASBR followed by the BGP 3107 label for PE2, followed by the service label for prefix X. When the remote LFA next-hop is activated, node B pushes one or more segment routing label: the node SID for the remote LFA backup node (node N).

When node N receives the packet while the remote LFA next-hop is activated, it pops the top segment routing label which corresponds to a local node SID. This results in popping this label and forwarding of the packet to the ASBR node over the shortest path (link N-Z).

When the ABR/ASBR node receives the packet from either node B or node Z, it pops the segment routing label which corresponds to a local node SID, then swaps the BGP label and pushes the LDP label of PE2 which is the next-hop of the BGP label route.

2.1.9.1 Hash Label and Entropy Label Support

When the **hash-label** option is enabled in a service context, hash label is always inserted at the bottom of the stack as per RFC 6391.

The LSR adds the capability to check a maximum of 16 labels in a stack. The LSR is able to hash on the IP headers when the payload below the label stack of maximum size of 16 is IPv4 or IPv6, including when a MAC header precedes it (**eth-encap-ip** option).

The Entropy Label (EL) feature, as specified in RFC 6790, is supported on RSVP, LDP, segment-routed, and BGP transport tunnels. It uses the Entropy Label Indicator (ELI) to indicate the presence of the entropy label in the label stack. The ELI, followed by the actual entropy label, is inserted immediately below the transport label for which entropy label feature is enabled. If multiple transport tunnels have the entropy label feature enabled, the ELI/EL is inserted below the lowest transport label in the stack.

The LSR hashing operates as follows:

- If the **lbl-only** hashing option is enabled, or if one of the other LSR hashing options is enabled but a IPv4 or IPv6 header is not detected below the bottom of the label stack, the LSR hashes on the EL only.
- If the **lbl-ip** option is enabled, the LSR hashes on the EL and the IP headers.
- If the **ip-only** or **eth-encap-ip** is enabled, the LSR hashes on the IP headers only.

For more information about the Hash Label and Entropy Label features, see the "MPLS Entropy Label and Hash Label" section of the *7450 ESS, 7750 SR, 7950 XRS, and VSR MPLS Guide*.

2.1.10 BGP Shortcut Using Segment Routing Tunnel

The user enables the resolution of IPv4 prefixes using SR tunnels to BGP next-hops in TTM with the following command:

```
config>router>bgp>next-hop-resolution
        – shortcut-tunnel
            – [no] family {ipv4}
                – resolution {any | disabled | filter}
                – resolution-filter
                    – [no] sr-isis
                    – [no] sr-ospf
```

```
                - [no] disallow-igp
                - exit
            - exit
        - exit
```

When **resolution** is set to **any**, any supported tunnel type in BGP shortcut context is selected following TTM preference. The following tunnel types are supported in a BGP shortcut context and in order of preference: RSVP, LDP, Segment Routing and BGP.

When the **sr-isis** or **sr-ospf** value is enabled, an SR tunnel to the BGP next-hop is selected in the TTM from the lowest preference IS-IS or OSPF instance. If many instances have the same lowest preference from the lowest numbered IS-IS or OSPF instance.

See the BGP chapter for more details.

2.1.11 BGP Label Route Resolution Using Segment Routing Tunnel

The user enables the resolution of RFC 3107 BGP label route prefixes using SR tunnels to BGP next-hops in TTM with the following command:

```
config>router>bgp>next-hop-resolution
        - labeled-routes
            - transport-tunnel
                - [no] family {label-ipv4 | label-ipv6 | vpn}
                    - resolution {any | disabled | filter}
                    - resolution-filter
                        - [no] sr-isis
                        - [no] sr-ospf
                    - exit
                - exit
            - exit
        - exit
```

When the **resolution** option is explicitly set to **disabled**, the default binding to LDP tunnel resumes. If **resolution** is set to **any**, any supported tunnel type in BGP label route context is selected following TTM preference.

The following tunnel types are supported in a BGP label route context and in order of preference: RSVP, LDP, and Segment Routing.

When the **sr-isis** or **sr-ospf** is specified using the **resolution-filter** option, a tunnel to the BGP next-hop is selected in the TTM from the lowest numbered IS-IS or OSPF instance.

See the BGP chapter for more details.

2.1.12 Service Packet Forwarding with Segment Routing

SDP subtypes of the MPLS type are available to allow service binding to an SR tunnel programmed in TTM by OSPF or IS-IS:

***A:7950 XRS-20# configure service sdp 100 mpls create**

***A:7950 XRS-20>config>service>sdp$ sr-ospf**

***A:7950 XRS-20>config>service>sdp$ sr-isis**

The SDP of type **sr-isis** or **sr-ospf** can be used with the **far-end** option. When the **sr-isis** or **sr-ospf** value is enabled, a tunnel to the far-end address is selected in the TTM from the lowest preference IS-IS or OSPF instance. If many instances have the same lowest preference from the lowest numbered IS-IS or OSPF instance. The SR-ISIS or SR-OSPF tunnel is selected at the time of the binding, following the tunnel selection rules. If a more preferred tunnel is subsequently added to the TTM, the SDP does not automatically switch to the new tunnel until the next time the SDP is being re-resolved.

The **tunnel-far-end** option is not supported. In addition, the **mixed-lsp-mode** option does not support the **sr-isis** and **sr-ospf** tunnel types.

The signaling protocol for the service labels for an SDP using an SR tunnel can be configured to static (**off**), T-LDP (**tldp**), or BGP (**bgp**).

SR tunnels can be used in VPRN and BGP EVPN with the **auto-bind-tunnel** command. See Next-Hop Resolution for more information.

Both VPN-IPv4 and VPN-IPv6 (6VPE) are supported in a VPRN or BGP EVPN service using segment routing transport tunnels with the **auto-bind-tunnel** command.

See BGP and refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 3 Services Guide: IES and VPRN* for more information about the VPRN **auto-bind-tunnel** CLI command.

2.1.13 Mirror Services and Lawful Intercept

The user can configure a spoke-SDP bound to an SR tunnel to forward mirrored packets from a mirror source to a remote mirror destination. In the configuration of the mirror destination service at the destination node, the **remote-source** command must use a spoke-sdp with VC-ID which matches the one the user configured in the mirror destination service at the mirror source node. The far-end option is not supported with an SR tunnel.

This also applies to the configuration of the mirror destination for an LI source.

Configuration at mirror source node:

```
config mirror mirror-dest 10
        – no spoke-sdp sdp-id:vc-id
        – spoke-sdp sdp-id:vc-id [create]
            – egress
                – vc-label egress-vc-label
```

Note:

- *sdp-id* matches an SDP which uses an SR tunnel
- for vc-label, both static and t-ldp egress vc labels are supported

Configuration at mirror destination node:

```
*A:7950 XRS-20# configure mirror mirror-dest 10 remote-source
        – spoke-sdp <SDP-ID>:<VC-ID> create <-- VC-ID matching that of spoke-sdp configured in
 mirror destination context at mirror source node.
            – ingress
```

```
                - vc-label <ingress-vc-label> <--- optional: both static and t-ldp ingress vc
label are supported.
             - exit
             - no shutdown
          - exit
      - exit
```

Note:

- the **far-end** command is not supported with SR tunnel at mirror destination node; user must reference a spoke-SDP using a segment routing SDP coming from mirror source node:
  - **far-end** *ip-address* [**vc-id** *vc-id*] [**ing-svc-label** *ingress-vc-label* | **tldp**] [**icb**]
  - **no far-end** *ip-address*
- for vc-label, both static and t-ldp ingress vc labels are supported

Mirroring and LI are also supported with the PW redundancy feature when the endpoint spoke-sdp, including the ICB, is using an SR tunnel. Routable Lawful Intercept Encapsulation (**config>mirror>mirror-dest>encap# layer-3-encap**) when the remote L3 destination is reachable over an SR tunnel is also supported.

2.1.14 Class-Based Forwarding for SR-ISIS over RSVP-TE LSPs

To enable CBF+ECMP for SR-ISIS over RSVP-TE:

- configure the resolution of SR over RSVP-TE LSPs as IGP shortcuts
- configure class based forwarding parameters in the MPLS context (a class forwarding policy, forwarding classes to sets associations, and RSVP-TE LSPs to forwarding sets associations)
- enable class forwarding in the segment routing context

When SR-ISIS resolves to an ECMP set of RSVP-TE LSPs and class forwarding is enabled in the segment routing context, the following behaviors apply:

- If no LSP in the full ECMP set, has been assigned with a class forwarding policy configuration, the set is considered as inconsistent from a CBF perspective. The system programs, in the forwarding path, the whole ECMP set and regular ECMP spraying occurs over the full set.
- If the ECMP set refers to more than one class forwarding policy, the set is inconsistent from a CBF perspective. The system programs, in the forwarding path, the whole ECMP set without any CBF information, and regular ECMP spraying occurs over the full set.
- In all other cases the ECMP set is considered consistent from a CBF perspective and the following rules apply:
  - If there is no default set (either user-defined or implicit) referenced in a CBF-consistent ECMP set, the system automatically selects one set as the default one. The selected set is the non-empty one with the lowest ID amongst those referenced by the LSPs of the ECMP set.
  - The system programs the data-path such that a packet which has been classified to a particular forwarding class is forwarded using the LSP(s) associated to the forwarding set which itself is associated to that forwarding class. In the event where the forwarding set is composed of multiple LSPs, the system performs ECMP over these LSPs.
  - Forwarding classes which are either not explicitly mapped to a set or which are mapped to a set for which all LSPs are down are forwarded using the default-set. The system re-elects a default set

in cases where all the LSPs of the current default-set become inactive. The system also adapts (updates data-path programming) to configuration or state changes.

– The CBF capability is available with any system profile. The number of sets is limited to four with system profile None or A, and to six with system profile B.

2.1.15 Segment Routing Traffic Statistics

This section describes capabilities and procedures applicable to IS-IS, OSPFv2, and OSPFv3.

SR OS can enable and collect SID traffic statistics on the ingress and egress data paths. Statistics can also be shown, monitored, and cleared, as well as accessed using telemetry.

IS-IS and OSPFv2 support Node SID, Adjacency SID, and Adjacency Set statistics. OSPFv3 supports Node SID and Adjacency SID statistics. The following commands are used to enter the context that allows for configuring the types of SIDs for which to collect traffic statistics.

- **configure router isis segment-routing egress-statistics**
- **configure router ospf segment-routing egress-statistics**
- **configure router ospf3 segment-routing egress-statistics**
- **configure router isis segment-routing ingress-statistics**
- **configure router ospf segment-routing ingress-statistics**
- **configure router ospf3 segment-routing ingress-statistics**

By default, statistics collection is disabled on all types of SIDs. If statistics are disabled (after having been enabled), the statistics indices that were allocated are released and the counter values are cleared.

On ingress, depending on which types of SIDs have statistics enabled, the following apply:

- The system allocates a statistic index to each programmed ILM, corresponding to the local node SID (including backup node SID) and to the local adjacency SIDs (including adjacencies advertised as set m embers).
- The system allocates a statistic index to each programmed ILM, corresponding to the received node SID advertisements.

On egress, depending on which types of SIDs have statistics enabled, the following apply:

- The system allocates a statistic index shared by the programmed NHLFEs (primary, and backup if any) corresponding to the local Adjacency SIDs and to the received Adjacency SIDs advertisements, and a statistic index shared by the primary NHLFEs (as many as members) of each adjacency set.
- The system allocates a statistic index shared by the programmed NHLFEs (one or more primaries, and backup if any) corresponding to each of the received node SID advertisements.

Note: The statistic indices constitute a finite resource. The system may not be able to allocate as many indices as needed. In this case, the system issues a notification and automatically retries to allocate statistic indices, but does not issue further notifications in case it still fails to allocate the needed statistic indices. If the system successfully allocates all the required statistic indices to IGP SIDs, then a second notification is issued to inform the user. A state variable records whether a SID has an index allocated.

Note: The allocation of statistic indices is non-deterministic. If more statistic indices are required system-wide, for example, upon a reboot, the system may not be able to re-allocate the statistic indices to the same entities as before the reboot.

2.1.16 Micro-Loop Avoidance Using Loop-Free SR Tunnels for IS-IS

Transient forwarding loops, or micro loops, occur during IGP convergence as a result of the transient inconsistency among forwarding states of the nodes of the network. The micro-loop avoidance feature supports the use of loop-free SR paths and a configurable time as a solution to the micro-loop issue for SR IS-IS SID tunnels.

2.1.16.1 Configuring Micro-loop Avoidance

The following command enables the micro-loop avoidance feature within each IGP instance:

```
config>router>isis>segm-rtng# micro-loop-avoidance
    – micro-loop-avoidance [fib-delay fib-delay]
    – no micro-loop-avoidance
```

fib-delay : [1..300] - default: 15, in 100s of milliseconds

The *fib-delay* timer should be configured to a value that corresponds to the worst-case IGP convergence in a given network domain. The default value of 1.5 seconds (1500 milliseconds) corresponds to a network with a nominal convergence time.

When this feature is disabled using the **no micro-loop-avoidance** command, any active FIB delay timer is forced to expire immediately and the new next hops are programmed for all impacted node SIDs. The feature is disabled for the next SPF runs.

When this feature is enabled, the following scenarios apply:

- IS-IS MT=0 for a SR-ISIS IPv4/IPv6 tunnel (node SID)
- IPv4 and IPv6 SR-TE LSP that use a node SID in their segment list
- IPv4 and IPv6 SR policy that use a node SID in their segment list

2.1.16.2 Micro-Loop Avoidance Algorithm Process

The SR OS micro-loop avoidance algorithm provides a loop-free mechanism in accordance with IETF *draft-bashandy-rtgwg-segment-routing-uloop*. The algorithm supports a single event on a P2P link or broadcast link with two neighbors for only the following cases:

- link addition or restoration
- link removal or failure
- link metric change

Using the algorithm, the router applies the following micro-loop avoidance process.

1. After it receives the topology updates and before the new SPF is started, the router verifies that the update corresponds to a single link event. Updates for the two directions of the link are treated as a single link event.

   If two or more link events are detected, the micro-loop avoidance procedure is aborted for this SPF and the existing behavior is maintained.

   Note: The micro-loop avoidance procedure is aborted if the subsequent link event received by an ABR is from a different area than the one that triggered the event initially. However, if the received event comes from a different IGP instance, the ABR handles it independently and triggers the micro-loop avoidance procedure, as long as it is a single event in that IGP instance.

2. The main SPF and LFA SPFs (base LFA, remote LFA, and/or TI-LFA based on the user configuration in that IGP instance) are run.
3. No action is performed for a node or a prefix if the SPF has resulted in no change to its next hop(s) and metric(s).
4. No action is performed for a node or a prefix if the SPF has resulted in a change to its next-hop(s) and/or metric(s), and the new next hops are resolved over RSVP-TE LSPs used as IGP shortcuts.

   Note: Nokia strongly recommends enabling CSPF for the RSVP-TE LSP used in IGP shortcut application. This avoids IGP churn and ensures micro-loop avoidance in the path of the RSVP control plane messages which would otherwise be generated following the convergence of IGP since the next hop in the ERO is looked up in the routing table.

5. The route is marked as micro-loop avoidance eligible for a node or a prefix if the SPF has resulted in a change to its next hop(s) or metric(s). The router performs the following:

   - for each SR node SID that uses a micro-loop avoidance eligible route with ECMP next hops, activates the common set of next hops between the previous and new SPF
   - for each SR node SID that uses a micro-loop-avoidance eligible route with a single next-hop, computes and activates a loop-free SR tunnel applicable to the specific link event

     This tunnel acts as the micro-loop avoidance primary path for the route and uses the same outgoing interface as the newly computed primary next hop.

     See *Micro-Loop Avoidance for Link Addition, Restoration, or Metric Decrease* and *Micro-Loop Avoidance for Link Removal, Failure, or Metric Increase*.
   - programs the TI-LFA, base LFA, or remote LFA backup path that protects the new primary next hop of the node SID
6. The **fib-delay** timer is started to delay the programming of the new main and LFA SPF results into the FIB.
7. The new primary next hop(s) are programmed for node SID routes that are marked eligible for the micro-loop avoidance procedure upon the expiration of the **fib-delay** timer.

Note: If a new SPF is scheduled while the **fib-delay** timer is running, the timer is forced to expire and the entire procedure is aborted.

If a CPM switchover is triggered while the **fib-delay** timer is running, the timer is forced to expire and the entire procedure is aborted.

In both cases, the next hops from the most recently run SPF are programmed for all impacted node SIDs. A subsequent event restarts the procedure at Step*1*.

2.1.16.3 Micro-Loop Avoidance for Link Addition, Restoration, or Metric Decrease

The network topology in *Figure 15: Micro-Loop Avoidance in Link Addition or Restoration* depicts an example of link addition or restoration.

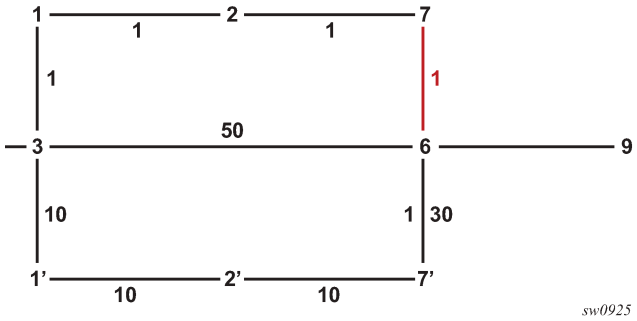

Figure 15: Micro-Loop Avoidance in Link Addition or Restoration

The micro-loop avoidance algorithm performs the following steps in the network topology example in *Figure 15: Micro-Loop Avoidance in Link Addition or Restoration*.

1. Link 7-6 is added to the topology.
2. Router 3 detects a single link addition between remote nodes 7 and 6.
3. Router 3 runs main and LFA SPFs.
   - all nodes downstream of the added link in Dijkstra tree see a next-hop change: nodes 6 and 9
   - all nodes upstream of the added link see no route change: nodes 1, 2, and 7
   - nodes 1', 2', and 7' are not using node 6 or 7 as parent node and are not impacted by the link addition event
4. For all nodes downstream from the added link, the algorithm computes and activates an SR tunnel that forces traffic to remote endpoint 6 of the added link.
   - The algorithm pushes node SID 7 and adjacency SID of link 7-6 in SR IS-IS tunnel for these nodes
5. The use of the adjacency SID of link 7-6 skips the FIB state on node 7 and traffic to all nodes downstream of 6 are not impacted by micro-loop convergence.
6. The same method applies to metric decrease of link 7-6 that causes traffic to be attracted to that link.

2.1.16.4 Micro-Loop Avoidance for Link Removal, Failure, or Metric Increase

The network topology in *Figure 16: Micro-Loop Avoidance in Link Removal or Failure* depicts an example of link removal or failure.

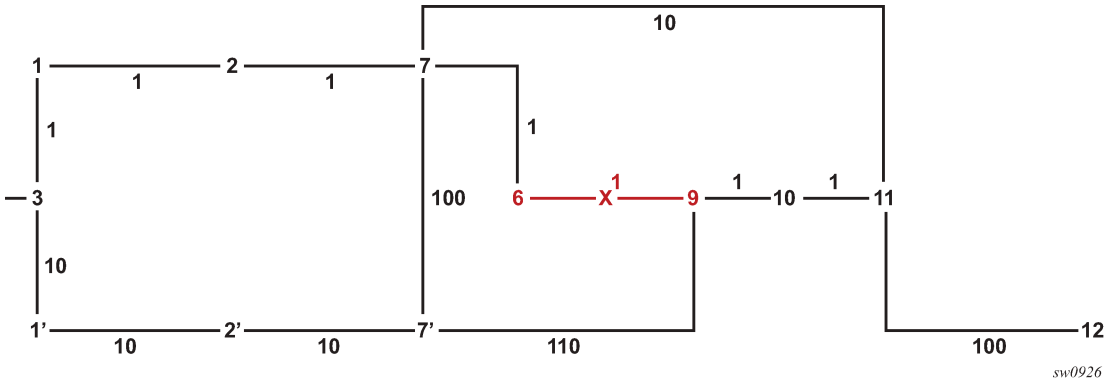

Figure 16: Micro-Loop Avoidance in Link Removal or Failure

The micro-loop avoidance algorithm performs the following steps in the network topology example in *Figure 16: Micro-Loop Avoidance in Link Removal or Failure*.

1. Link 6-9 is removed or fails.
2. Router 3 detects a single link event and runs main and LFA SPFs.
   - all nodes downstream of the removed link in the Dijkstra tree see a next-hop change: nodes 9, 10, 11, and 12
   - plus nodes 10, 11, and 12 are no longer downstream of node 9
   - all nodes upstream of the removed link see no route change: nodes 1, 2, 7, and 6
   - nodes 1', 2', and 7' are not using node 6 or 9 as parent nodes and are therefore are not impacted by the link removal event
3. For each impacted node, the algorithm computes and activates a loop-free SR tunnel to the farthest node in the shortest path that did not see a next-hop change, and then uses the adjacency SIDs to reach destination node.
   - For SR IS-IS tunnel of node 12, push SID of node 7 and then SIDs of adjacencies 7-11 and 11-12.
   - Similar to P-Q set calculation in TI-LFA, but the P node is defined as the farthest node in the shortest path to the destination in the new topology with no next-hop change.
   - The maximum number of labels used for the P-Q set is determined as follows.
     – If TI-LFA is enabled, use the value of **max-sr-frr-labels**.
     – If TI-LFA is disabled, use the value of 3 that matches the maximum value of TI-LFA parameter **max-sr-frr-labels**.
     – In both cases, this value is passed to MPLS for checking against parameter **max-sr-label** [**additional-frr-labels**] for all configured SR-TE LSPs and SR-TE LSP templates.
   - A future implementation compresses the path from the P node to the destination using an extra Q node calculation.
   - The path to the P node may travel over an RSVP-TE LSP used as an IGP shortcut. In this case, the RSVP-TE LSP must have CSPF enabled to avoid churn in IGP and to avoid micro-loop in the path of the RSVP control plane messages that are generated following the convergence of IGP, because the next hop in the ERO is looked up in the routing table.
   - When SR-LDP stitching is enabled and the path to the P node or the path between the P and Q nodes is partly on the LDP domain, no loop-free SR tunnel is programmed and IGP programs the new next hop or hops.
4. The same method applies to a metric increase of link 6-9 that causes traffic to move away from that link; for example, a metric change from 1 to 200.

2.1.17 Configuring Flexible Algorithms

2.1.17.1 Configuring IS-IS for Flexible Algorithms for SR-MPLS

IGP protocols traditionally compute best paths over the network based on the IGP metric assigned to the links. Many network deployments use RSVP-TE based or SR-based TE to enforce traffic over a path that is computed using different metrics or constraints than the shortest IGP path. The SR Flexible Algorithm (Flex-Algorithm) solution allows IGPs to compute constraint-based paths over the network. This section describes the use of SR prefix SIDs to compute a constraint topology and send packets along the constraint-based paths.

Using Flex-Algorithms can reduce the number of SR SIDs that must be imposed to send packets along a constrained path; this implementation simplifies the hardware capabilities of SR routing tunnel head-end devices.

The supported depth of the label stack is considered in an SR network when SR-TE tunnels or SR policies are deployed. In such tunnel policies, the packet source routing is based on the SR label stack pushed on the packet. The depth of the label stack that a router can push on a packet determines the complexity of the SR-TE tunnel construction that the router can support.

The SR Flex-Algorithm solution allows the creation of composed metrics based upon arbitrary parameters (for example, delay, link administrative group, cost, and so on) when using Flex-Algorithms. A network-wide set of composed topology constraints (also known as the Flexible Algorithm Definition (FAD)) creates an SR Flex-Algorithm topology. The IGP calculates the best path using constraint-based SPF and the FAD to create the best paths through the Flex-Algorithm topology.

With Flex-Algorithms, each Flex-Algorithm topology can send data flows along the most optimal constrained path toward its destination using a single label, which reduces the imposed label stack along.

Using this solution, backup path calculations (for example, Loop Free Alternate (LFA), Remote LFA (R-LFA) and Topology Independent LFA (TI-LFA)) can be constrained to the SR Flex-Algorithm topology during link failure.

Perform the following tasks to configure Flex-Algorithms using IS-IS.

1. *Configuring the Flexible Algorithm Definition* (optional)
2. *Configuring IS-IS for Flexible Algorithms for SR-MPLS*
3. *Configuring IS-IS Flex-Algorithm Prefix Node SID*
4. *Verifying Basic Flex-Algorithm Behavior*

2.1.17.1.1 Configuring the Flexible Algorithm Definition

To guarantee loop-free forwarding for paths that are computed for a specific Flex-Algorithm, all routers configured to participate in that Flex-Algorithm must agree on the FAD. The agreement ensures that routing loops and inconsistent forwarding behavior is avoided.

Each router that is configured to participate in a specific Flex-Algorithm must select the FAD based on standardized tie-breaking rules. This ensures consistent FAD selection in cases where different routers advertise different definitions for a specific Flex-Algorithm. The following tie-breaking rules apply.

- From the FAD advertisements in the area (including both locally generated advertisements and received advertisements), select the one with the highest priority value.
- If there are multiple FAD advertisements with the same priority, select one that originated from the router with the highest system ID.

A router that is not participating in a specific Flex-Algorithm is allowed to advertise the FAD for that specific Flex-Algorithm. Any change in the FAD may result in temporary disruption of traffic that is forwarded based on those Flex-Algorithm paths. The impact is similar to any other event that requires network-wide convergence.

If a node is configured to participate in a Flex-Algorithm, but the selected FAD includes calculation-type, metric-type, constraint, flag, or a sub-TLV that is not supported by the node, the node stops participation and removes any forwarding state associated with the Flex-Algorithm.

Use the following syntax to configure FADs.

```
config>router
        – flexible-algorithm-definitions
            –   flex-algo <fad-name> [create]
            –  no flex-algo <fad-name>
                – description <description-string>
                – [no] description
                – exclude
                    – admin-group <admin-group>
                    –     [no] admin-group <admin-group>
                – flags-tlv
                – [no] flags-tlv
                – include-all
                    – admin-group <admin-group>
                    – [no] admin-group <admin-group>
                – include-any
                    – admin-group <admin-group>
                    – [no] admin-group <admin-group>
                – metric-type {igp|te-metric|delay}
                – [no] metric-type
                – priority <[0..255>
                – [no] priority
                – shutdown
                – [no] shutdown
```

The following is a sample configuration output for a basic FAD:

```
router
  flexible-algorithm-definitions
    flex-algo "My128" create
      description "This-is-my-algo128"
        metric-type delay
        no shutdown
    exit
  exit
```

2.1.17.1.2 Configuring IS-IS Flex-Algorithm Participation

Up to seven Flex-Algorithms in the range 128 to 255 can be configured for IS-IS. Use the **participate** command to configure participation for the specific algorithm. If a locally configured FAD exists, advertise

this definition by using the **advertise** command. A router is not required to advertise a configured FAD to participate in a Flex-Algorithm.

If a Flex-Algorithm is enabled to participate or advertise the FAD, it is configured and active for all configured IS-IS areas.

Use the following syntax to configure Flex-Algorithms for IS-IS.

```
config>router>isis
        – flexible-algorithms
            –   [no] flex-algo flex-algo
                – advertise fad-name
                – no advertise
                – [no] loopfree-alternates
                – [no] participate
            – [no] shutdown
```

Note: When a router participates in Flex-Algorithms, it will only advertise support for the Flex-Algorithm where the router can comply with the winning FAD, provided that at least one FAD exists for this algorithm.

The following is a sample configuration output for Flex-Algorithm participation:

```
isis 0
    flexible-algorithms
      flex-algo 128
        advertise "My128"
        participate
      exit
      no shutdown
    exit
```

The following output is an example of IS-IS router capability when a FAD is advertised:

```
*A:Dut-B# show router isis database Dut-B.00-00 detail level 2
===============================================================================
Rtr Base ISIS Instance 0 Database (detail)
===============================================================================
Displaying Level 2 database
-------------------------------------------------------------------------------
LSP ID    : Dut-B.00-00                                 Level     : L2
Sequence  : 0x94                  Checksum  : 0x4ae0   Lifetime  : 969
Version   : 1                     Pkt Type  : 20       Pkt Ver   : 1
Attributes: L1L2                  Max Area  : 3        Alloc Len : 1492
SYS ID    : 4900.0000.0002        SysID Len : 6        Used Len  : 223
TLVs :
  Supp Protocols:
    Protocols     : IPv4
  IS-Hostname   : Dut-B
  Router ID   :
    Router ID   : 10.20.1.2
Router Cap : 10.20.1.2, D:0, S:0
    TE Node Cap : B E M  P
    SR Cap: IPv4 MPLS-IPv6
       SRGB Base:20000, Range:10001
    SR Alg: metric based SPF, 128
    Node MSD Cap: BMI : 12 ERLD : 15
    FAD Sub-Tlv:
        Flex-Algorithm   : 128
```

```
        Metric-Type      : delay
        Calculation-Type : 0
        Priority         : 100
        Flags: M
```

2.1.17.1.3 Configuring IS-IS Flex-Algorithm Prefix Node SID

The prefix node SID (IPv4 and/or IPv6) must be assigned for each participating Flex-Algorithm.

The Flex-Algorithm SIDs are allocated from the label block assigned to SR and configuring a special range is not required.

Note: Flex-Algorithm node SIDs can be configured for IPv4 and/or IPv6 prefixes.

Use the following syntax to configure the prefix node SIDs for IS-IS Flex-Algorithms.

```
config>router>isis>interface
        – ipv4-node-sid
        – flex-algo
            – ipv4-node-sid index <value>
            – ipv4-node-sid label <value>
            – no ipv4-node-sid
            – ipv6-node-sid index <value>
            – ipv6-node-sid label <value>
            – no ipv6-node-sid
```

The following is a sample configuration output for Flex-Algorithm prefix node SIDs:

```
router
  mpls-labels
    sr-labels start 20000 end 30000
  exit
  interface "Loopback0"
    address 10.20.1.2/32
    loopback
    no shutdown
  exit
  isis 0
    segment-routing
      prefix-sid-range global
      no shutdown
    exit
    interface "Loopback0"
      ipv4-node-sid index 2
      passive
      flex-algo 128
        ipv4-node-sid index 12
      exit
      no shutdown
    exit
```

The following output is an example of the Level 2 database of an advertised IS-IS:

```
A:Dut-B# show router isis database Dut-B.00-00 detail level 2
===============================================================================
```

```
Rtr Base ISIS Instance 0 Database (detail)
===============================================================================
Displaying Level 2 database
-------------------------------------------------------------------------------
LSP ID    : Dut-B.00-00                                 Level     : L2
Sequence  : 0x9d                  Checksum  : 0x38e9    Lifetime  : 626
Version   : 1                     Pkt Type  : 20        Pkt Ver   : 1
Attributes: L1L2                  Max Area  : 3         Alloc Len : 1492
SYS ID    : 4900.0000.0002        SysID Len : 6         Used Len  : 223
……<snip>……
  TE IP Reach   :
    Default Metric  : 10
    Control Info:    , prefLen 30
    Prefix   : 10.10.10.0
    Default Metric  : 0
    Control Info:   S, prefLen 32
    Prefix   : 10.20.1.2
    Sub TLV   :
      Prefix-SID Index:2, Algo:0, Flags:NnP
      Prefix-SID Index:12, Algo:128, Flags:NnP
    Default Metric  : 10
    Control Info:    , prefLen 30
    Prefix   : 10.10.10.8
...<snip>...
```

2.1.17.1.4 Verifying Basic Flex-Algorithm Behavior

The creation of the segment routing Flex-Algorithm forwarding information results in the label forwarding tables on the router. On a Nokia router, it is possible to look both at the tunnel table and the routing table to understand the Flex-Algorithm path toward a destination prefix.

For example, algorithm 128 has been configured to use the delay metric, and consequently forwards traffic using the lowest delay through the network. In *Figure 17: Selecting the Lowest Delay Path*, Node B is configured with IP address 10.20.1.2/32, the A-B path has the best default IGP metric, and the A-C-B path has the best delay.

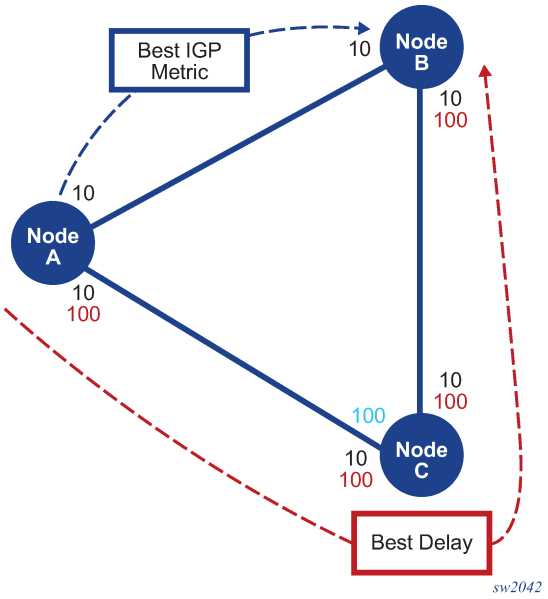

Figure 17: Selecting the Lowest Delay Path

The following output is an example of the **tunnel-table** command:

```
A:Dut-A# show router tunnel-table
===============================================================================
IPv4 Tunnel Table (Router: Base)
===============================================================================
Destination           Owner     Encap TunnelId  Pref   Nexthop        Metric
   Color
-------------------------------------------------------------------------------
10.10.10.2/32         isis (0)  MPLS  524298    11     10.10.10.2     0
10.10.10.6/32         isis (0)  MPLS  524292    11     10.10.10.6     0
10.20.1.2/32          isis (0)  MPLS  524296    11     10.10.10.2     10
10.20.1.2/32          isis (0)  MPLS  524306    11     10.10.10.6     200
10.20.1.3/32          isis (0)  MPLS  524294    11     10.10.10.6     10
10.20.1.3/32          isis (0)  MPLS  524307    11     10.10.10.6     100
-------------------------------------------------------------------------------
Flags: B = BGP or MPLS backup hop available
       L = Loop-Free Alternate (LFA) hop available
       E = Inactive best-external BGP route
       k = RIB-API or Forwarding Policy backup hop
===============================================================================
A:Dut-A#
```

The following output is an example of the **detail** option of the **tunnel-table** command:

```
A:Dut-A# show router tunnel-table 10.20.1.2/32 detail
===============================================================================
Tunnel Table (Router: Base)
===============================================================================
Destination      : 10.20.1.2/32
NextHop          : 10.10.10.2
Tunnel Flags     : entropy-label-capable
Age              : 18h21m35s
CBF Classes      : (Not Specified)
Owner            : isis (0)             Encap            : MPLS
Tunnel ID        : 524296               Preference       : 11
Tunnel Label     : 20002                Tunnel Metric    : 10
```

```
Tunnel MTU       : 1560                Max Label Stack  : 1
-------------------------------------------------------------------------------
Destination      : 10.20.1.2/32
NextHop          : 10.10.10.6
Tunnel Flags     : entropy-label-capable
Age              : 02h01m32s
CBF Classes      : (Not Specified)
Owner            : isis (0)            Encap            : MPLS
Algorithm        : 128
Tunnel ID        : 524306              Preference       : 11
Tunnel Label     : 20012               Tunnel Metric    : 200
Tunnel MTU       : 1560                Max Label Stack  : 1
-------------------------------------------------------------------------------
Number of tunnel-table entries          : 2
Number of tunnel-table entries with LFA : 0
===============================================================================
A:Dut-A#
```

The following output is an example of the route table with and without the Flex-Algorithm context:

```
A:Dut-A# show router isis routes
===============================================================================
Rtr Base ISIS Instance 0 Route Table
===============================================================================
Prefix[Flags]                     Metric     Lvl/Typ     Ver.  SysID/Hostname
  NextHop                                                MT     AdminTag/SID[F]
-------------------------------------------------------------------------------
10.10.10.0/30                     10         1/Int.      65    Dut-A
   0.0.0.0                                                 0       0
10.10.10.4/30                     10         1/Int.      42    Dut-A
0.0.0.0                                                   0       0
10.10.10.8/30                     20         2/Int.      65    Dut-B
   10.10.10.2                                              0       0
10.20.1.1/32                      0          1/Int.      42    Dut-A
   0.0.0.0                                                 0       0/1[NnP]
10.20.1.2/32                      10         2/Int.      65    Dut-B
   10.10.10.2                                              0       0/2[NnP]
10.20.1.3/32                      10         2/Int.      42    Dut-C
   10.10.10.6                                              0       0/3[NnP]
-------------------------------------------------------------------------------
No. of Routes: 6 (6 paths)
-------------------------------------------------------------------------------
Flags        : L = LFA nexthop available
SID[F]       : R  = Re-advertisement
               N  = Node-SID
               nP = no penultimate hop POP
               E  = Explicit-Null
               V  = Prefix-SID carries a value
               L  = value/index has local significance
===============================================================================
A:Dut-A#
A:Dut-A# show router isis routes flex-algo 128
===============================================================================
Rtr Base ISIS Instance 0 Flex-Algo 128 Route Table
===============================================================================
Prefix[Flags]                     Metric     Lvl/Typ     Ver.  SysID/Hostname
  NextHop                                                MT     AdminTag/SID[F]
-------------------------------------------------------------------------------
10.20.1.2/32                      200        2/Int.      82    Dut-C
   10.10.10.6                                              0       0/12[NnP]
10.20.1.3/32                      100        2/Int.      82    Dut-C
   10.10.10.6                                              0       0/13[NnP]
-------------------------------------------------------------------------------
No. of Routes: 2 (2 paths)
```

```
-------------------------------------------------------------------------------
Flags        : L = LFA nexthop available
SID[F]       : R  = Re-advertisement
               N  = Node-SID
               nP = no penultimate hop POP
               E  = Explicit-Null
               V  = Prefix-SID carries a value
               L  = value/index has local significance
===============================================================================
A:Dut-A#
```

The following output is an example of the **detail** option for the route table, with and without the Flex-Algorithm context:

```
A:Dut-A# show router isis routes 10.20.1.2 detail
===============================================================================
Rtr Base ISIS Instance 0 Route Table (detail)
===============================================================================
Prefix           : 10.20.1.2/32
Status           : Active               Level             : 2
NextHop          : 10.10.10.2
Metric           : 10                   Type              : Internal
SPF Version      : 65                   SysID/Hostname    : Dut-B
MT               : 0                    AdminTag          : 0
SID              : 2                    SID-Flags         : NnP
-------------------------------------------------------------------------------
No. of Routes: 1 (1 path)
-------------------------------------------------------------------------------
SID[F]       : R  = Re-advertisement
               N  = Node-SID
               nP = no penultimate hop POP
               E  = Explicit-Null
               V  = Prefix-SID carries a value
               L  = value/index has local significance
===============================================================================
A:Dut-A#

A:Dut-A# show router isis routes 10.20.1.2 flex-algo 128 detail
===============================================================================
Rtr Base ISIS Instance 0 Flex-Algo 128 Route Table (detail)
===============================================================================
Prefix           : 10.20.1.2/32
Status           : Active               Level             : 2
NextHop          : 10.10.10.6
Metric           : 200                  Type              : Internal
SPF Version      : 82                   SysID/Hostname    : Dut-C
MT               : 0                    AdminTag          : 0
SID              : 12                   SID-Flags         : NnP
-------------------------------------------------------------------------------
No. of Routes: 1 (1 path)
-------------------------------------------------------------------------------
SID[F]       : R  = Re-advertisement
               N  = Node-SID
               nP = no penultimate hop POP
               E  = Explicit-Null
               V  = Prefix-SID carries a value
               L  = value/index has local significance
===============================================================================
A:Dut-A#
```

2.1.17.1.5 Configuration and Usage Considerations for Flex-Algorithms

The following considerations must be taken into account when configuring and using Flex-Algorithms.

- IS-IS algorithms 128 to 255 can program only the tunnel table, while IS-IS for algorithm 0 can program both the tunnel and the IP routing tables. For operational simplicity, the **show>router>isis>routes** command displays the correct egress interface.
- To prevent the accidental creation of an overload of local FADs, the operator is only allowed to configure a maximum of 256 local FADs on a router.
- A router can participate in a maximum of seven Flex-Algorithms. Each algorithm has the capability to advertise a single locally configured FAD.
- The SR OS implementation assumes that the participation of a specific **flex-algo** command includes its participation in Flex-Algorithms in all enabled IGP areas. For example, on an IS-IS Level 1 and Level 2 capable router (default router), the same FAD participates and is advertised on both levels. To advertise a FAD only at Level 1 or Level 2, the operator should configure a Level 1-only or Level 2-only router to advertise the FAD. Alternatively, an additional **flex-algo** can be used; for example, algorithm 129 in Level 1 and algorithm 128 in Level 2.
- All Flex-Algorithm participating nodes must advertise the locally used FADs when configured and optionally advertise node participation when the winning FAD is supported.
- The winning FAD on a router is select ed based on the following tie-breaker:

  1. select the FAD with the highest priority
  2. select the FAD advertised by the highest IGP system ID
- If the local router does not support the winning FAD, the router should remove itself from the **flex-algo** topology by not advertising algorithm participation in the IS-IS router TLV capability. In such a case, no SPF is computed and any prefix SID of that **flex-algo** is removed from the associated routing and tunnel tables.
- When the FAD selects a metric type, only links that have such metric type configured are considered for the **flex-algo** topology.
- Leaking of a FAD on an ABR is not supported.
- When advertising the FAD flags-TLV, the SR OS router always sets the M-flag, which forces the IS-IS routers to use Flex-Algorithm aware metrics for inter-area routing. The enforced M-flag ensures that the best ABR, according to the Flex-Algorithm, is selected to exit the area outside the local IGP area. Without the M-flag, the wrong ABR may be selected and cause routing loops or a traffic blackhole. This handling assumes that an ABR must advertise the IS-IS Flex-Algorithm prefix metric sub-TLV when leaking prefixes and associated SIDs. Advertising the flags-TLV is optional, and is controlled through the **no flags-tlv** configuration within the Flex-Algorithm definition.

- SR OS supports the Administrative Groups (AGs) as defined in RFC 5305. The following considerations apply:

  – up to 32 link colors can be used
  – Flex-Algorithm feature reuses the existing AGs in combination with application-specific TLV extensions

    Note: Although the same AG can be used for Flex-Algorithm and LFA policies, Nokia recommends that AGs that are used for LFA policies should be avoided.

  – SR OS provides the following limited Extended Administrative Group (EAG) support for Flex-Algorithm.

    - The Nokia implementation supports only AG advertisement; EAG advertisement is not supported. The IS-IS TLV types used for an AG and an EAG are different.
    - For backward compatibility, vendors may use only the first 32 colors in the EAG.
    - If EAG is used to add a color on the links, the link attribute size can be 4 octets (or a multiple of 4 octets) long.
    - The EAG for Flex-Algorithms is forwarded for appropriate ASLA encoding in accordance with *draft-ietf-isis-te-app-14.txt*.
    - When an EAG ASLA link attribute is received, the SR OS router handles it as follows.

      SR OS provides limited EAG support and only parses EAGs that are 4 octets long. The EAG represents a traditional 4-octet AG to support backward compatibility.

      SR OS treats the ASLA-encoded EAG as opaque information when the EAG size is a multiple of 4 octets long (that is, 4, 8, and so on).

      Due to limited EAG support, a new trap is not sent if the AG and EAG link attributes are inconsistent. In such a case, the AG attributes are used in accordance with RFC 7308.
    - The receipt of a Flex-Algorithm FAD that contains an include/exclude EAG ASLA link attribute is handled as follows.

      If the SR OS router receives a FAD where the AG TLV length is 4 octets, the FAD can be used for **flex-algo** and it is treated as an AG.

      If the SR OS router receives a FAD where the AG TLV length is greater than 4 octets and bits are set to 1 in the first 4 octets only (the remaining bits are set to 0), the FAD participates assuming that the AGs have been configured as a result of EAG backward compatibility.

      If the SR OS router receives a FAD where the length of the AG TLV is greater than 4 octets and has bits set to 1 beyond the first 32 bits, the router will block this FAD. SR OS does not support EAG bits beyond the first 32 bits.

- Flex-Algorithm uses the IS-IS min/max unidirectional link delay sub-TLV as defined in RFC 8570. This delay is set through the static configuration.
- SR OS allows the user to enable and disable Flex-Algorithm Loop Free Alternate (LFA) paths. The LFA type is inherited from the algorithm 0 base topology configuration.
- Operators can protect links and nodes using the LFA fast-convergence technology. If the primary path is constrained by a specific **flex-algo** topology, the LFA SPF calculation is executed within the **flex-algo** topology. This calculation identifies the correct LFA, R-LFA or TI-LFA bounded by this topology.

Consequently, the constraints of a specific **flex-algo** topology are respected even during failure scenarios.

  - Enabling or disabling the **flex-algo** dependent LFA, R-LFA, or TI-LFA is aligned with enabling the LFA within the router **flex-algo** context.
  - A new configuration node LFA is added in the IGP parameter within the Flex-Algorithm configuration. The **shutdown** and **no shutdown** commands are also added to this node.
  - The LFA parameter allows the user to disable or enable loopfree alternates for this **flex-algo**. The *rlfa* and *tlfa* parameters are inherited from algorithm 0.
  - The Flex-Algorithm LFA exclude policy configuration is copied from the **flex-algo** 0 configuration.
  - The Flex-Algorithm aware LFA may cause additional resource consumption (for example, in memory and in CPU).
  - SR OS Flex-Algorithm support for LFA policies supported by algorithm 0, including SRLG, protection type, exclude and include groups.
- Interaction with SR-LDP mapping server

  - Flex-Algorithms are not compatible with the SR-LDP mapping server. SR OS only supports mapping-server TLV with algorithm 0.
- Interaction with SR-TE policy

  - Flex-Algorithms have no impact on how SR-TE LSPs are used. Applications that support the use of SR-TE LSPs continue to be supported. All SR-TE resolution mechanisms are supported.
  - SR-TE changes as follows as a result of Flex-Algorithm support.

    - When an SR-TE path is constructed through manual router configuration or received from the PCE, the sequence of SR-TE SIDs may include one or more Flex-Algorithm prefix node SIDs.
    - At the SR-TE head-end router, the sequenced SR-TE label stack (the sequence of SIDs) is imposed upon the payload and the packet is forwarded using the NHLFE from the top label or SID.
  - Validity of a specific SR-TE LSP is the same as without Flex-Algorithm support.
- Interaction with SR policies

  - Similar to SR-TE LSPs, SR policies are only influenced by Flex-Algorithms due to construction of the segment list. The segment list may be constructed using one or more Flex-Algorithm prefix node label SIDs. All applications capable of using SR policies will have opaque awareness if a segment list is constructed using Flex-Algorithm labels or SIDs.
- Flex-Algorithm and adjacency SID protection

  - During fast-reroute process, local repair of the links to reach the Q-node from the P-node will be determined by the sub-topology defined by the Flex-Algorithm. Therefore, the used link will consider the configured administrative group constraints.
  - However, the adj-sid backup is based upon algo=0, since adj-sids are not advertised using a Flex-Algorithm. Consequently, there is a risk to violate the Flex-Algorithm if the related link breaks while it is in use as backup for a Flex-Algorithm path. This Flex-Algorithm SLA break can be avoided when adj-sids are configured with no backup capability.
- Duplicate SID handling

  - IS-IS uses the first learned remote SID and generates a trap for duplicate entries.

- Interaction with IGP shortcut and forwarding adjacency features
  - To select the optimal shortest path within a constrained topology, Flex-Algorithm paths are carefully crafted using the constraints specified in the FAD. If the constrained topology includes logical RSVP-TE links that conceal FAD constraints, the Flex-Algorithm may send traffic wrongly over out-of-profile physical links.
  - To avoid the use of Flex-Algorithm in the range of 128 to 255, which causes data plane traffic to be sent over tunnels that hide physical link properties, the following features are not supported:
    - SR-LDP stitching
    - IGP shortcut
    - forwarding adjacency; forwarding adjacencies are not considered in the **flex-algo** topology.
- Relationship between Flex-Algorithm and algorithm 0 configuration

  A configured router with Flex-Algorithm does not have to advertise an algo 0 SID.
- Interaction of Flex-Algorithm aware nodes and FAD flags-TLV

  When Flex-Algorithms are enabled, SR OS advertises by default FAD flags-TLV in IGP to signal the mandatory use of Flex-Algorithm aware performance metrics. For correct Flex-Algorithm operation, it is expected that Flex-Algorithm aware nodes support FAD flags-TLV interpretation. For improved interoperability, it is possible to stop advertising the FAD flags-TLV using the **no flags-tlv** command when defining a flexible algorithm on SR OS.
- Flex-Algorithm for BGP services
  - BGP next-hop can be automatically resolved over an IGP Flex-Algorithms topology using the import policy action flex-algo (for BGP, BGP LU, and VPN).
  - BGP next-hop Flex-Algorithms aware autobind for BGP EVPN service is not supported.
- Flex-Algorithm and TLV encoding

  Flex-Algorithms BGP-LS export and TLV encoding is supported.

2.1.17.2 Configuring IS-IS Flex-Algorithm for SRv6

SRv6 introduces flexible algorithms to the IPv6 dataplane.

A router is provisioned with topology or algorithm-specific locators for each of the topology or algorithm pairs supported by that node. Each locator is a covering prefix for all SIDs provisioned on that router which have the matching topology or algorithm. Locators associated with flexible algorithms are not advertised in a Prefix Reachability TLV (236 or 237). However, locators associated with algorithm 0 are advertised in a Prefix Reachability TLV (236 or 237) so that legacy routers that do not support SRv6 can install a forwarding entry for algorithm 0 SRv6 traffic.

Each SRv6 locator is associated with an algorithm (either algorithm 0 or a flexible algorithm in the range of 128 to 255) and each algorithm represents a topologically-constrained forwarding construct. The M-flag within the flexible algorithm prefix metric sub-TLV is not applicable to prefixes advertised as SRv6 locators. The metric field in the locator TLV is used regardless of the M-flag in the FAD advertisement.

A router configured to participate in a flexible algorithm must use the selected FAD to compute the corresponding routing table. The available options are described below.

- Algorithm 0 (legacy routing table entries) is constructed from information advertised as a traditional IP reach TLV or as an SRv6 locator (tlv27). When IP reach TLV and SRv6 locator TLV contain conflicting information, then the IP reach TLV information is used.
- Algorithms ranging from 128 to 255 (Flex-Algorithm routing table entries) are constructed from information advertised and constructed from locators found in the SRv6 locator TLV (tlv27).

For route leaking of flexible algorithm-aware SRv6 locators between IS-IS areas, the following rules apply when a topology TLV is leaked (IP reach TLV or SRv6 locator TLV) including leaked locators and end SIDs:

- algorithm 0: this SRv6 locator route is programmed as regular IS-IS route. If an IS-IS route is readvertised and has also an SRV6 locator TLV, it will be readvertised as a regular IP reach TLV and SRv6 locator TLV.
- algorithms ranging from 128 to 255: if locator leaking is enabled, the original SRv6 locator TLV will be readvertised as a SRv6 locator TLV into the other area.

- Default locator leaking behavior between levels:
  - Level 1 to Level 2: leaking is enabled by default
  - Level 2 to Level 1: leaking is disabled by default
  - Changing the default leaking behavior requires an export policy where the **prefix-list** keyword behavior is enhanced to match upon prefixes or locators found in the routing table regardless of the associated algorithm. The **prefix-list** allows combined support for algorithm 0 locators (regular prefixes) and algorithm locators ranging from 128 to 255 (Flex-Algorithm prefixes).

```
Configure
+---router
    +---policy-options
        +---[no] policy-statement <name>
            +---from
                +---[no] level [1|2]
                +---[no] prefix-list
                +---[no] protocol
                +---[no] tag
            +---to
                 +---<snip>
```

If a locator is associated with a flexible algorithm and the LFA is enabled, then LFA paths to the locator prefix must be calculated using the flexible algorithm in the corresponding topology to guarantee that they follow the same constraints as the calculation of the primary paths. LFA paths must only use SRv6 SIDs advertised specifically for the given flexible algorithm. The LFA configuration is inherited from algorithm 0. The anycast behavior of SRv6 flexible algorithms is inherited from the standard algorithm 0 (standard SPF) SRv6 configuration.

The IS-IS neighbor advertisements are topology-specific and not algorithm-specific. Therefore, the SRv6 End.X SIDs inherit topology from the associated neighbor advertisement, but the algorithm is specified in the individual SID. All End.X SIDs are a subnet of a locator with matching topology and algorithm which is advertised by the same node in an SRv6 locator TLV. The End.X SIDs which do not meet this requirement are ignored. All End.X SIDs must find a supernet by the subnet of a locator with the matching algorithm which is advertised by the same router in an SRv6 locator TLV. The End.X SIDs which do not meet this requirement are ignored.

IS-IS protocol limitations affect enabling SRv6 flexible algorithms on a broadcast network. On a broadcast network, the LAN End.X SIDs of all neighbors for all participating flexible algorithms need to be advertised in a single LSP fragment because each IS-IS TE-NBR with all its TLV blocks must be advertised in one IS-

IS LSP fragment. The amount of information inserted by segment routing for SRv6 into the LSP fragment depends upon the number of the flexible algorithms used, the number of static or auto-end.X configured per locator, and if both SRv6 and SR-MPLS are deployed.

2.2 Segment Routing With Traffic Engineering (SR-TE)

When segment routing is used together with MPLS data plane, the SID is a standard MPLS label. A router forwarding a packet using segment routing therefore pushes one or more MPLS labels.

Segment routing using MPLS labels can be used in both shortest path routing applications (see *Segment Routing in Shortest Path Forwarding* for more information) and in traffic engineering (TE) applications, as described in this section.

An SR-TE LSP supports a primary path, with Fast Reroute (FRR) backup, and one or more secondary paths. A secondary path can be configured as standby.

SR OS implements the following computation methods for the paths of a SR-TE LSP:

- Hop-to-Label Translation — the TE-DB converts the list of hops, destination of the LSP and the strict or loose hops in the path definition, to a list of SIDs by searching the IGP instances with segment routing enabled. This method does not support TE constraints except for loose or strict hops.

  See *SR-TE LSP Path Computation Using Hop-to-Label Translation* for more details.
- Local CSPF — the LSP path TE constraints are considered in the path computation. This method implements most of the CSPF capabilities supported with RSVP-TE LSP with very few exception such as the bandwidth constraint which cannot be booked with SR-TE LSP because of the lack of a signaling protocol to establish the LSP path.

  See *SR-TE LSP Path Computation Using Local CSPF* for more details.
- Path Computation Element (PCE) — in this case the router acting as a PCE Client (PCC) requests a computation of the path of a SR-TE LSP from the PCE using the PCEP Protocol.

  See *Path Computation Element Protocol (PCEP)* for more details.
- User specified SID list — the SR-LSP feature provides the option for the user to manually configure each path of the LSP using an explicit list of SID values.

  See *SR-TE LSP Paths using Explicit SIDs* for more details.

The configured or computed path of a SR-TE LSP can use a combination of node SIDs and adjacency SIDs.

2.2.1 SR-TE MPLS Configuration Commands

The following MPLS commands and nodes are supported:

- Global MPLS-level commands and nodes:

  interface, **lsp**, **path**, **shutdown**

- LSP-level commands and nodes:

  bfd, **bgp-shortcut**, **bgp-transport-tunnel**, **cspf, exclude**, **hop-limit**, **igp-shortcut, include**, **metric**, **metric-type, path-computation-method, primary**, **retry-limit**, **retry-timer**, **revert-timer**, **shutdown**, **to**, **from**, **vprn-auto-bind**
- Both primary and secondary paths are supported with a SR-TE LSP. The following primary path level commands and nodes are supported with SR-TE LSP:

  bandwidth, **bfd**, **exclude**, **hop-limit**, **include**, **priority**, **shutdown**

  The following secondary path level commands and nodes are supported with SR-TE LSP:

  bandwidth, **bfd**, **exclude**, **hop-limit**, **include**, **path-preference**, **priority**, **shutdown**, **srlg**, **standby**

The following MPLS commands and nodes are not supported:

- Global MPLS level commands and nodes not applicable to SR-TE LSP (configuration is ignored):

  admin-group-frr, **auto-bandwidth-multipliers**, **auto-lsp**, **bypass-resignal-timer**, **cspf-on-loose-hop**, **dynamic-bypass**, **exponential-backoff-retry**, **frr-object**, **hold-timer**, **ingress-statistics**, **least-fill-min-thd**, **least-fill-reoptim-thd**, **logger-event-bundling**, **lsp-init-retry-timeout**, **lsp-template**, **max-bypass-associations**, **mbb-prefer-current-hops**, **mpls-tp**, **p2mp-resignal-timer**, **p2mp-s2l-fast-retry**, **p2p-active-path-fast-retry**, **retry-on-igp-overload**, **secondary-fast-retry-timer**, **shortcut-local-ttl-propagate**, **shortcut-transit-ttl-propagate**, **srlg-database**, **srlg-frr**, **static-lsp**, **static-lsp-fast-retry**, **user-srlg-db**
- LSP level commands and nodes not supported with SR-TE LSP (configuration blocked):

  adaptive, **adspec**, **auto-bandwidth**, **class-type**, **dest-global-id**, **dest-tunnel-number**, **exclude-node**, **fast-reroute**, **ldp-over-rsvp**, **least-fill**, **main-ct-retry-limit, p2mp-id, primary-p2mp-instance, propagate-admin-group, protect-tp-path, rsvp-resv-style, working-tp-path**
- The following primary path level commands and nodes are not supported with SR-TE LSP:

  adaptive, **backup-class-type**, **class-type**, **record**, **record-label**
- The following secondary path level commands and nodes are not supported with SR-TE LSP:

  adaptive, **class-type**, **record**, **record-label**

The user can associate an empty path or a path with strict or loose explicit hops with the paths of the SR-TE LSP using the **hop**, **primary**, and **secondary** commands.

A hop that corresponds to an adjacency SID must be identified with its far-end host IP address (next-hop) on the subnet. If the local end host IP address is provided, this hop is ignored because this router can have multiple adjacencies (next-hops) on the same subnet.

A hop that corresponds to a node SID is identified by the prefix address.

Details of processing the user configured path hops are provided in *SR-TE LSP Instantiation*.

2.2.2 SR-TE LSP Instantiation

When an SR-TE LSP is configured on the router, its path can be computed by the router or by an external TE controller referred to as a Path Computation Element (PCE). This feature works with the Nokia stateful

PCE which is part of the Network Services Platform (NSP).The SR OS supports the following modes of operations which are configurable on a per SR-TE LSP basis:

- When the path of the LSP is computed by the router acting as a PCE Client (PCC), the LSP is referred to as PCC-initiated and PCC-controlled.

  A PCC-initiated and controlled SR-TE LSP has the following characteristics:

  – Can contain strict or loose hops, or a combination of both
  – Supports both a basic hop-to-label translation and a full CSPF as a path computation method.
  – The capability exists to report an SR-TE LSP to synchronize the LSP database of a stateful PCE server using the **pce-report** option, but the LSP path cannot be updated by the PCE. In other words, the control of the LSP is maintained by the PCC
- When the path of the LSP is computed by the PCE at the request of the PCC, it is referred to as PCC-initiated and PCE-computed.

  A PCC-initiated and PCE-computed SR-TE LSP supports the Passive Stateful Mode, which enables the **path-computation-method pce** option for the SR-TE LSP so PCE can perform path computation at the request of the PCC only. The PCC retains control.

  The capability exists to report an SR-TE LSP to synchronize the LSP database of a stateful PCE server using the **pce-report** option.
- When the path of the LSP is computed and updated by the PCE following a delegation from the PCC, it is referred to as PCC-initiated and PCE-controlled.

  A PCC-initiated and PCE-controlled SR-TE LSP allows Active Stateful Mode, which enables the **pce-control** option for the SR-TE LSP so PCE can perform path computation and updates following a network event without the explicit request from the PCC. The PCC delegates full control.

The user can configure the path computation requests only (PCE-computed) or both path computation requests and path updates (PCE-controlled) to PCE for a specific LSP using the **path-computation-method pce** and **pce-control** commands.

The **path-computation-method pce** option sends the path computation request to the PCE instead of the local CSPF. When this option is enabled, the PCE acts in Passive Stateful mode for this LSP. In other words, the PCE can perform path computations for the LSP only at the request of the router. This is used in cases where the operator wants to use the PCE specific path computation algorithm instead of the local router CSPF algorithm.

The default value is **no path-computation-method**.

The user can also enable the router's full CSPF path computation method. See *SR-TE LSP Path Computation Using Local CSPF* for more details.

The **pce-control** option allows the router to delegate full control of the LSP to the PCE (PCE-controlled). Enabling it means the PCE is acting in Active Stateful mode for this LSP and allows PCE to reroute the path following a failure or to re-optimize the path and update the router without requiring the router to request it.

Note:

- The user can delegate LSPs computed by either the local CSPF or the hop-to-label translation path computation methods.

- The user can delegate LSPs which have the **path-computation-method pce** option enabled or disabled. The LSP maintains its latest active path computed by PCE or the router at the time it was delegated. The PCE will only make an update to the path at the next network event or re-optimization. The default value is **no pce-control**.
- PCE report is supported for SR-TE LSPs with more than one path. However, PCE computation and PCE control are not supported in such cases. PCE computation and PCE control are supported for SR-TE LSPs with only one path that is either primary or secondary.

In all cases, the PCC LSP database is synchronized with the PCE LSP database using the PCEP Path Computation State Report (PCRpt) message for LSPs that have the **pce-report** command enabled.

The global MPLS level **pce-report** command can be used to enable or disable PCE reporting for all SR-TE LSPs for the purpose of LSP database synchronization. This configuration is inherited by all LSPs of a given type. The PCC reports both CSPF and non-CSPF LSP. The default value is disabled (**no pce-report**). This default value controls the introduction of PCE into an existing network and allows the operator to decide if all LSP types need to be reported.

The LSP level **pce-report** command overrides the global configuration for reporting LSP to PCE. The default value is to inherit the global MPLS level value. The **inherit** value returns the LSP to inherit the global configuration for that LSP type.

Note: If PCE reporting is disabled for the LSP, either due to inheritance or due to LSP level configuration, enabling the **pce-control** option for the LSP has no effect. To help troubleshoot this situation, operational values of both the **pce-report** and **pce-control** are added to the output of the LSP path **show** command.

For more information about configuring PCC-Initiated and PCC-Controlled LSPs, see *Configuring PCC-controlled, PCE-computed, and PCE-controlled SR-TE LSPs*.

2.2.2.1 PCC-Initiated and PCC-Controlled LSP

In this mode of operation, the user configures the LSP name, primary path name and optional secondary path name with the path information in the referenced path name, entering a full or partial explicit path with all or some hops to the destination of the LSP. Each hop is specified as an address of a node or an address of the next hop of a TE link. Optionally, each hop may be specified as a SID value corresponding to the MPLS label to use on a given hop. In this case, the whole path must consist of SIDs.

To configure a primary or secondary path to always use a specific link whenever it is up, the strict hop must be entered as an address corresponding to the next-hop of an adjacency SID, or the path must consist of SID values for every hop. If the strict hop corresponds to an address of a loopback address, it is translated into an adjacency SID as explained below and therefore does not guarantee that the same specific TE link is picked.

MPLS assigns a Tunnel-ID to the SR-TE LSP and a path-ID to each new instantiation of the primary or secondary path. These IDs represent both the MBB path and the original path of a given SR-TE LSP, which both exist during the process of an MBB update of the primary path.

Note: The concept of MBB is not exactly accurate in the context of a SR-TE LSP because there is no signaling involved and, as such, the new path information immediately overrides the older one.

The router retains full control of the path of the LSP. The LSP path label stack size is checked by MPLS against the maximum value configured for the LSP after the TE-DB returns the label stack. See *Service and Shortcut Application SR-TE Label Stack Check* for more information about this check.

The ingress LER performs the following steps to resolve the user-entered path before programming it in the data path:

1. MPLS passes the path information to the TE-DB, which uses the hop-to-label translation or the full CSPF method to convert the list of hops into a label stack. The TE database returns the actual selected hop SIDs plus labels as well the configured path hop addresses which were used as the input for this conversion.
2. The ingress LER validates the first hop of the path to determine the outgoing interface and next hop where the packet is to be forwarded and programs the data path according to the following conditions.

   - If the first hop corresponds to an adjacency SID (host address of next-hop on the link's subnet), the adjacency SID label is not pushed. In other words, the ingress LER treats forwarding to a local interface as a push of an implicit-null label.
   - If the first hop is a node SID of some downstream router, then the node SID label is pushed.

   In both cases, the SR-TE LSP tracks and rides the SR shortest path tunnel of the SID of the first hop.
3. In the case where the router is configured as a PCC and has a PCEP session to a PCE, the router sends a PCRpt message to update PCE with the state of UP and the RRO object for each LSP which has the **pce-report** option enabled. PE router does not set the delegation control flag to keep LSP control. The state of the LSP is now synchronized between the router and the PCE.

2.2.2.1.1 Guidelines for PCC-Initiated and PCC-Controlled LSPs

The router supports both a full CSPF and a basic hop-to-label translation path computation methods for a SR-TE LSP. In addition, the user can configure a path for the SR-TE LSP by explicitly entering SID label values.

The ingress LER has a few ways to detect a path is down or is not optimal and take immediate action:

- Failure of the top SID detected via a local failure or an IGP network event. In this case, the LSP path goes down and MPLS will retry it.
- Timeout of the seamless BFD session when enabled on the LSP and the **failure-action** is set to the value of **failover-or-down**. In this case, the path goes down and MPLS will retry it.
- Receipt of an IGP link event in the TE database. In this case, MPLS performs an ad-hoc re-optimization of the paths of all SR-TE LSPs if the user enabled the MPLS level command **sr-te-resignal resignal-on-igp-event**. This capability only works when the path computation method is the local CSPF. It allows the ingress LER not only to detect a single remote failure event which causes packets to drop but also a network event which causes a node SID to reroute and thus forwarding packets on a potentially sub-optimal TE path.
- Performing a manual or timer based resignal of the SR-TE LSP. This applies only when the path computation method is the local CSPF. In this case, MPLS re-optimizes the paths of all SR-TE LSPs.

With both the hop-to-label path computation method and the user configured SID labels, the ingress LER does not monitor network events which affect the reachability of the adjacency SID or node SID used in the label stack of the LSP, except for the top SID. As a result, the label stack may not be updated to reflect changes in the path except when seamless BFD is used to detect failure of the path. It is therefore recommended to use this type of SR-TE LSP in the following configurations only:

- empty path
- path with a single node-SID loose-hop
- path of an LSP to a directly-connected router (single-hop LSP) with an adjacency-SID or a node-SID loose/strict hop
- strict path with hops consisting of adjacencies explicitly configured in the path as IP addresses or SID labels.

The user can also configure a SR-TE LSP with a single loose-hop using the anycast SID concept to provide LSR node protection within a given plane of the network TE topology. This is illustrated in *Figure 18: Multi-plane TE with Node Protection*. The user configures all LSRs in a given plane with the same loopback interface address, which must be different from that of the system interface and the router-id of the router, and assigns them the same node-SID index value. All routers must use the same SRGB.

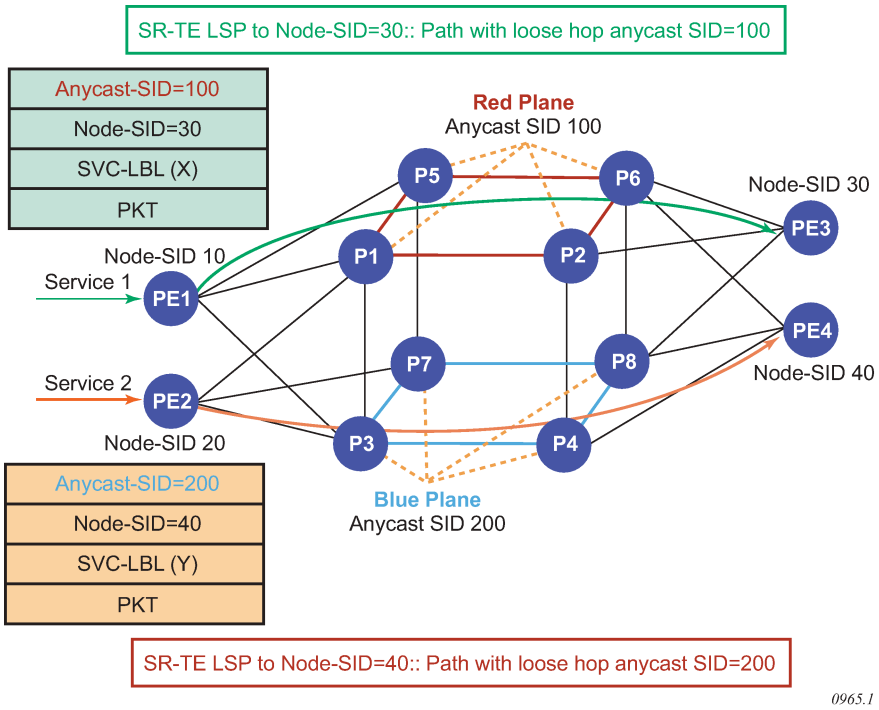

Figure 18: Multi-plane TE with Node Protection

Then user configures in a LER a SR-TE LSP to some destination and adds to its path either a loose-hop matching the anycast loopback address or the explicit label value of the anycast SID. The SR-TE LSP to any destination will hop over the closest of the LSRs owning the anycast SID because the resolution of the node-SID for that anycast loopback address uses the closest router. When that router fails, the resolution is updated to the next closest router owning the anycast SID without changing the label stack of the SR-TE LSP.

2.2.2.2 PCC-Initiated and PCE-Computed or Controlled LSP

In this mode of operation, the ingress LER uses Path Computation Element Communication Protocol (PCEP) to communicate with a PCE-based external TE controller (also referred to as the PCE). The router instantiates a PCEP session to the PCE. The router is referred to as the PCE Client (PCC).

The following PCE control modes are supported:

Passive Control Mode	In this mode, the user enables the **path-computation-method pce** command for one or more SR-TE LSPs and a PCE performs path computations at the request of the PCC.
Active Control Mode	In this mode, the user enables the **pce-control** command for an LSP, which allows the PCE to perform both path computation and periodic reoptimization of the LSP path without an explicit request from the PCC.

For the PCC to communicate with a PCE about the management of the path of a SR-TE LSP, the router implements the extensions to PCEP in support of segment routing (see the PCEP section for more information).This feature works with the Nokia stateful PCE, which is part of the Network Services Platform (NSP).

The following procedure describes configuring and programming a PCC-initiated SR-TE LSP when passive or active control is given to the PCE.

1. The SR-TE LSP configuration is created on the PE router via CLI or via OSS/SAM. The configuration dictates which PCE control mode is desired: active (**pce-control** option enabled) or passive (**path-computation-method pce** enabled and **pce-control** disabled).
2. The PCC assigns a unique PLSP-ID to the LSP. The PLSP-ID uniquely identifies the LSP on a PCEP session and must remain constant during its lifetime. PCC on the router tracks the association of {PLSP-ID, SRP-ID} to {Tunnel-ID, Path-ID} and uses the latter to communicate with MPLS about a specific path of the LSP.
3. The PE router does not validate the entered path. While the PCC can include the IRO objects for any loose or strict hop in the configured LSP path in the Path Computation Request (PCReq) message to PCE, the PCE ignores them and computes the path with the other constraints, excepting the IRO.
4. The PE router sends a PCReq message to the PCE to request a path for the LSP and includes the LSP parameters in the METRIC object, the LSPA object, and the Bandwidth object. It also includes the LSP object with the assigned PLSP-ID. At this point, the PCC does not delegate control of the LSP to the PCE.
5. PCE computes a new path, reserves the bandwidth, and returns the path in a Path Computation Reply (PCRep) message with the computed ERO in the ERO object. It also includes the LSP object with the unique PLSP-ID, the METRIC object with the computed metric value if any, and the Bandwidth object.

   Note: For the PCE to use the SRLG path diversity and admin-group constraints in the path computation, the user must configure the SRLG and admin-group membership against the MPLS interface and verify that the traffic-engineering option is enabled in IGP. This causes IGP to flood the link SRLG and admin-group membership in its participating area and for the PCE to learn it in its TE database.

6. The PE router updates the CPM and the data path with the new path.

   Up to this step, the PCC and PCE are using passive stateful PCE procedures. The next steps synchronize the LSP database of the PCC and PCE for both PCE-computed and PCE-controlled LSPs. They also initiate the active PCE stateful procedures for the PCE-controlled LSP only.
7. PE router sends a PCRpt message to update PCE with the state of UP and the RRO as confirmation, including the LSP object with the unique PLSP-ID. For a PCE-controlled LSP, the PE router also sets a delegation control flag to delegate control to the PCE. The state of the LSP is now synchronized between the router and the PCE.

8. Following a network event or re-optimization, PCE computes a new path for a PCE-Controlled LSP and returns it in a Path Computation Update (PCUpd) message with the new ERO. It includes the LSP object with the same unique PLSP-ID assigned by the PCC and the Stateful Request Parameter (SRP) object with a unique SRP-ID-number to track error and state messages specific to this new path.

   Note: If the **no pce-control** command is performed while a PCUpdate MBB is in progress on the LSP, the router aborts and removes the information and state related to the in-progress PCUpdate MBB. As the LSP is no longer controlled by the PCE, the router may take further actions depending on the state of the LSP. For example, if the LSP is up, and has FRR active or pre-emption, then the router starts a GlobalRevert or pre-emption MBB. If the LSP is down, the router starts the retry-timer to trigger setup.

9. The PE router updates the CPM and the data path with the new path.
10. The PE router sends a new PCRpt message to update PCE with the state of UP and the RRO as confirmation. The state of the LSP is now synchronized between the router and the PCE.
11. If the user makes any configuration change to the PCE-computed or PCE-controlled LSP, MPLS requests PCC to first revoke delegation in a PCRpt message (PCE-controlled only), and then MPLS and PCC follow the above steps to convey the changed constraint to PCE, which will result in a new path programmed into the data path, the LSP databases of PCC and PCE to be synchronized, and the delegation to be returned to PCE.

    In the case of an SR-TE LSP, MBB is not supported. Therefore, PCC first tears down the LSP and sends a PCRpt message to PCE with the Remove flag set to 1 before following this configuration change procedure.

Note: The preceding procedure is followed when the user performs a **no shutdown** on a PCE-controlled or PCE-computed LSP. The starting point is an administratively-down LSP with no active paths.

The following steps are followed for an LSP with an active path:

- If the user enabled the **path-computation-method pce** option on a PCC-controlled LSP which has an active path, no action is performed until the next time the router needs a path for the LSP following a network event of an LSP parameter change. At that point the procedures above are followed.
- If the user enabled the **pce-control** option on a PCC-controlled or PCE-computed LSP which has an active path, PCC will issue a PCRpt message to PCE with the state of UP and the RRO of the active path. It will set delegation control flag to delegate control to PCE. PCE will keep the active path of the LSP and will not update until the next network event or re-optimization. At that point the procedures above are followed.

The PCE supports the computation of disjoint paths for two different LSPs originating or terminating on the same or different PE routers. To indicate this constraint to PCE, the user must configure the PCE path profile ID and path group ID the LSP belongs to. These parameters are passed transparently by PCC to PCE and, so, opaque data to the router. The user can configure the path profile and path group using the **path-profile** *profile-id* [**path-group** *group-id*] command.

The association of the optional path group ID is to allow PCE determine which profile ID this path group ID must be used with. One path group ID is allowed per profile ID. The user can, however, enter the same path group ID with multiple profile IDs by executing this command multiple times. A maximum of five entries of **path-profile** [*path-group*] can be associated with the same LSP. More details of the operation of the PCE path profile are provided in the PCEP section of this guide.

2.2.3 SR-TE LSP Path Computation

For PCC-controlled SR-TE LSPs, CSPF is supported on the router using the **path-computation-method local-cspf** command. See *SR-TE LSP Path Computation Using Local CSPF* for details about the full CSPF path computation method. By default, the path is computed using the hop-to-label translation method. In the latter case, MPLS makes a request to the TE-DB to get the label corresponding to each hop entered by the user in the primary path of the SR-TE LSP. See *SR-TE LSP Path Computation Using Hop-to-Label Translation* for details of the hop-to-label translation.

The user can configure the path computation request of a CSPF-enabled SR-TE LSP to be forwarded to a PCE instead of the local router CSPF (**path-computation-method local-cspf** option enabled) by enabling the **path-computation-method pce** option, as explained in *SR-TE LSP Instantiation*. The user can further delegate the re-optimization of the LSP to the PCE by enabling the **pce-control** option. In both cases, PCE is responsible for determining the label required for each returned explicit hop and includes this in the SR-ERO.

In all cases, the user can configure the maximum number of labels which the ingress LER can push for a given SR-TE LSP by using the **max-sr-labels** command.

This command is used to set a limit on the maximum label stack size of the SR-TE LSP primary path so as to allow room to insert additional transport, service, and other labels when packets are forwarded in a given context.

```
config>router>mpls>lsp>max-sr-labels label-stack-size [additional-frr-labels labels]
```

The **max-sr-labels** *label-stack-size* value should be set to account for the desired maximum label stack of the primary path of the SR-TE LSP. Its range is 1-11 and the default value is 6.

The value in **additional-frr-labels** *labels* should be set to account for additional labels inserted by remote LFA or Topology Independent LFA (TI-LFA) for the backup next-hop of the SR-TE LSP. Its range is 0-3 labels with a default value of 1.

The sum of both label values represents the worst case transport of SR label stack size for this SR-TE LSP and is populated by MPLS in the TTM such that services and shortcut applications can check it to decide if a service can be bound or a route can be resolved to this SR-TE LSP. More details of the label stack size check and requirements in various services and shortcut applications are provided in *Service and Shortcut Application SR-TE Label Stack Check* .

The maximum label stack supported by the router is discussed in *Data Path Support* and always signaled by PCC in the PCEP Open object as part of the SR-PCE-CAPABILITY TLV. It is referred to as the Maximum Stack Depth (MSD).

In addition, the per-LSP value for the **max-sr-labels** *label-stack-size* option, if configured, is signaled by PCC to PCE in the Segment-ID (SID) Depth value in a METRIC object for both a PCE-computed LSP and a PCE-controlled LSP. PCE will compute and provide the full explicit path with TE-links specified. If there is no path with the number of hops lower than the MSD value, or the SID Depth value if signaled, a reply with no path is returned to PCC.

For a PCC-controlled LSP, if the label stack returned by the TE-D exceeds the per LSP maximum SR label stack size, the LSP is brought down.

2.2.4 SR-TE LSP Path Computation Using Hop-to-Label Translation

MPLS passes the path information to the TE-DB, which converts the list of hops into a label stack as follows:

- A loose hop with an address matching any interface (loopback or not) of a router (identified by router-ID) is *always* translated to a node SID. If the prefix matching the hop address has a node SID in the TE database, it is selected by preference. If not, the node SID of any loopback interface of the same router that owns the hop address is selected. In the latter case, the lowest IP-address of that router that has a /32 Prefix-SID is selected.
- A strict hop with an address matching any interface (loopback or not) of a router (identified by router-ID) is always translated to an adjacency SID. If the hop address matches the host address reachable in a local subnet from the previous hop, then the adjacency SID of that adjacency is selected. If the hop address matches a loopback interface, it is translated to the adjacency SID of any link from the previous hop which terminates on the router owning the loopback. The adjacency SID label of the selected link is used.

  In both cases, it is possible to have multiple matching previous hops in the case of a LAN interface. In this case, the adjacency-SID with the lowest interface address is selected.
- In addition to the IGP instance that resolved the prefix of the destination address of the LSP in the RTM, all IGP instances are scanned from the lowest to the highest instance ID, beginning with IS-IS instances and then OSPF instances. For the first instance via which all specified path hop addresses can be translated, the label stack is selected. The hop-to-SID/label translation tool does not support paths that cross area boundaries. All SID/labels of a given path are therefore taken from the same IGP area and instance.
- Unnumbered network IP interfaces, which are supported in the router's TE database, can be selected when converting the hops into an adjacency SID label when the user has entered the address of a loopback interface as a strict hop; however, the user cannot configure an unnumbered interface as a hop in the path definition.

Note: For the hop-to-label translation to operate, the user must enable TE on the network links, meaning to add the network interfaces to MPLS and RSVP. In addition, the user must enable the **traffic-engineering** option on all participating router IGP instances. Note that if any router has the **database-export** option enabled in the participating IGP instances to populate the learned IGP link state information into the TE-DB, then enabling of the **traffic-engineering** option is not required. For consistency purposes, it is recommended to have the **traffic-engineering** option always enabled.

2.2.5 SR-TE LSP Path Computation Using Local CSPF

This feature introduces full CSPF path computation for SR-TE LSP paths.

The hop-to-label translation, the local CSPF, or the PCE path computation methods for a SR-TE LSP can be user-selected with the following **path-computation-method** [**local-cspf** | **pce**] command. The **no** form of this command sets the computation method to the hop-to-label translation method, which is the default value. The **pce** option is not supported with the SR-TE LSP template.

2.2.5.1 Extending MPLS and TE Database CSPF Support to SR-TE LSP

The following are the MPLS and TE database features for extending CSPF support to SR-TE LSP:

- Supports IPv4 SR-TE LSP
- Supports local CSPF on both primary and secondary standby paths of an IPv4 SR-TE LSP
- Supports local CSPF in LSP templates of types **mesh-p2p-srte** and **one-hop-p2p-srte** of SR-TE auto-LSP
- Supports path computation in single area OSPFv2 and IS-IS IGP instances
- Computes full explicit TE paths using TE links as hops and returning a list of SIDs consisting of adjacency SIDs and parallel adjacency set SIDs. SIDs of a non-parallel adjacency set is not used in CSPF. The details of the CSPF path computation are provided in *SR-TE Specific TE-DB Changes*. Loose-hop paths, using a combination of node SID and adjacency SID, are not required.
- Uses random path selection in the presence of ECMP paths that satisfy the LSP and path constraints. Least-fill path selection is not required.
- Provides an option to reduce or compress the label stack such that the adjacency SIDs corresponding to a segment of the explicit path are replaced with a node SID whenever the constraints of the path are met by all the ECMP paths to that node SID. The details of the label reduction are provided in *SR-TE LSP Path Label Stack Reduction*.
- Uses legacy TE link attributes as in RSVP-TE LSP CSPF
- Uses timer re-optimization of all paths of the SR-TE LSP that are in the operational UP state. This differs from RSVP-TE LSP resignal timer feature which re-optimizes the active path of the LSP only.

  MPLS provides the current path of the SR-TE LSP and TE-DB updates the total IGP or TE metric of the path, checking the validity of the hops and labels as per current TE-DB link information. CSPF then calculates a new path and provides both the new and metric updated current path back to MPLS. MPLS programs the new path only if the total metric of the new computed path is different than the updated metric of the current path, or if one or more hops or labels of the current path are invalid. Otherwise, the current path is considered one of the most optimal ECMP paths and is not updated in the data path.

  Timer resignal applies only to the CSPF computation method and not to the ip-to-label computation method.
- Uses manual re-optimization of a path of the SR-TE LSP. In this case, the new computed path is always programmed even if the metric or SID list is the same.
- Supports ad-hoc re-optimization. This SR-TE LSP feature for SR-TE LSP triggers the ad-hoc resignaling of all SR-TE LSPs if one or more IGP link down events are received in TE-DB.

  Once the re-optimization is triggered, the behavior is the same as the timer-based resignal or the delay option of the manual resignal. MPLS forces the expiry of the resignal timer and asks TE-DB to re-evaluate the active paths of all SR-TE LSPs. The re-evaluation consists of updating the total IGP or TE metric of the current path, checking the validity of the hops and labels, and computing a new CSPF for each SR-TE LSP. MPLS programs the new path only if the total metric of the new computed path is different than the updated metric of the current path, or if one or more hops or labels of the current path are invalid. Otherwise, the current path is considered one of the most optimal ECMP paths and is not updated in the data path.
- Supports using unnumbered interfaces in the path computation. There is no support for configuring an unnumbered interface as a hop in the path of the LSP is not required. So, the path can be empty or include hops with the address of a system or loopback interface but path computation can return a path that uses TE links corresponding to unnumbered interfaces.
- Supports **admin-group**, **hop-count**, IGP metric, and TE-metric constraints
- Bandwidth constraint is not supported since SR-TE LSP does not have an LSR state to book bandwidth. Thus, the **bandwidth** parameter, when enabled on the LSP path, has no impact on local CSPF path calculation. However, the **bandwidth** parameter is passed to PCE when it is the selected path computation method. PCE reserves bandwidth for the SR-TE LSP path accordingly.

2.2.5.2 SR-TE Specific TE-DB Changes

With the RSVP-TE LSP feature, the TE-DB only populates OSPFv2 local and remote TE-enabled links. A TE-link is a link that has one or more TE attributes added to it in the MPLS interface context. Link TE attributes are TE metric, bandwidth, and membership in a SRLG or an Admin-Group.

The SR-TE LSP path computation supports using SR-enabled links which may or may not have TE attributes and therefore the TE-DB is enhanced with, the following changes:

- OSPFv2 is modified to pass all links, regardless if they are TE-enabled or SR-enabled, to TE-DB as currently performed by IS-IS.
- TE-DB relaxes the link back-check when performing a CSPF calculation to ensure that there is at least one link from the remote router to the local router. Since OSPFv2 advertises the remote link IP address or remote link identifier only when a link is TE-enabled, the strict check about the reverse direction of a TE-link cannot be performed if the link is SR-enabled but not TE-enabled.

As a consequence of this change, CSPF can compute an SR-TE LSP with SR-enabled links that do not have TE attributes. This means that if the user admin shuts down an interface in MPLS, an SR-TE LSP path which uses this interface will not go operationally down.

2.2.5.3 SR-TE LSP and Auto-LSP-Specific CSPF Changes

The local CSPF for an SR-TE LSP is performed in two phases. The first phase (Phase 1) computes a fully explicit path with all TE links to the destination specified as in the case of a RSVP-TE LSP.

If the user enabled label stack reduction or compression for this LSP, a second phase (Phase 2) is applied to reduce the label stack so that adjacency SIDs corresponding to a segment of the explicit path are replaced with a node SID whenever the constraints of the path are met by all the ECMP paths to that node SID. The details of the label reduction are provided in *SR-TE LSP Path Label Stack Reduction*.

The CSPF computation algorithm for the fully explicit path in the first phase remains mostly unchanged from its behavior for an RSVP-TE LSP.

The meaning of a strict and loose hop in the path of the LSP are the same as in CSPF for an RSVP-TE LSP. A strict hop means that the path from the previous hop must be a direct link. A loose hop means the path from the previous hop can traverse intermediate routers.

A loose hop may be represented by a set of back-to-back adjacency SIDs if not all paths to the node SID of that loose hop satisfy the path TE constraints. This is different from the ip-to-label path computation method where a loose hop always matches a node SID since no TE constraints are checked in the path to that loose hop.

When the label stack of the path is reduced or compressed, it is possible that a strict hop is represented by a node SID, if all the links from the previous hop satisfy the path TE constraints. This is different from the ip-to-label path computation method wherein a strict hop always matches an adjacency SID or a parallel adjacency set SID.

The first phase of CSPF returns a full explicit path with each TE link specified all the way to the destination and which label stack may contain protected adjacency SIDs, unprotected adjacency SIDs, and adjacency set SIDs. The user can influence the type of adjacency protection for the SR-TE LSP using a CLI command as explained in *SR-TE LSP Path Protection*.

The SR OS does not support the origination of a global adjacency SID. If received from a third-party router implementation, it is added into the TE database but is not used in any CSPF path computation.

2.2.5.3.1 SR-TE LSP Path Protection

Also introduced with SR-TE LSP is the indication by the user if the path of the LSP must use protected or unprotected adjacencies exclusively for all links of the path.

When SR OS routers form an IGP adjacency over a link and segment-routing context is enabled in the IGP instance, the static or dynamic label assigned to the adjacency is advertised in the link adjacency SID sub-TLV. By default, an adjacency is always eligible for LFA/RLFA/TI-LFA protection and the B-flag in the sub-TLV is set. The presence of a B-flag does not reflect the instant state of the availability of the adjacency LFA backup; it reflects that the adjacency is eligible for protection. The SR-TE LSP using the adjacency in its path still comes up if the adjacency does not have a backup programmed in the data path at that instant. Use the **configure>router>isis>interface> no sid-protection** command to disable protection. When protection is disabled, the B-flag is cleared and the adjacency is not eligible for protection by LFA/RLFA/TI-LFA.

SR OS also supports the adjacency set feature that treats a set of adjacencies as a single object and advertises a link adjacency sub-TLV for it with the S-flag (SET flag) set to 1. The adjacency set in the SR OS implementation is always unprotected, even if there is a single member link in it and therefore the B-flag is always clear. Only a parallel adjacency set, meaning that all links terminate on the same downstream router, are used by the local CSPF feature.

Be aware that the same P2P link can participate in a single adjacency and in one or more adjacency sets. Therefore, multiple SIDs can be advertised for the same link.

Third party implementations of Segment Routing may advertise two SIDs for the same adjacency: one protected with B-flag set and one unprotected with B-flag clear. SR OS can achieve the same behavior by adding a link to a single-member adjacency SET, in which case a separate SID is advertised for the SET and the B-flag is cleared while the SID for the regular adjacency over that link has its B-flag set by default. In all cases, SR OS CSPF can use all local and remote SIDs to compute a path for an SR-TE LSP based on the desired local protection property.

There are three different behaviors of CSPF introduced with SR-TE LSP with respect to local protection:

1. When the **local-sr-protection** command is not enabled (**no local-sr-protection**) or is set to **preferred**, the local CSPF prefers a protected adjacency over an unprotected adjacency whenever both exist for a TE link. This is done on a link-by-link basis after the path is computed based on the LSP path constraints. This means that the protection state of the adjacency is not used as a constraint in the path computation. It is only used to select an SID among multiple SIDs once the path is selected. Thus, the computed path can combine both types of adjacencies.

   If a parallel adjacency set exists between two routers in a path and all the member links satisfy the constraints of the path, a single protected adjacency is selected in preference to the parallel adjacency set which is selected in preference to a single unprotected adjacency.

   If multiples ECMP paths satisfy the constraints of the LSP path, one path is selected randomly and then the SID selection above applies. There is no check if the selected path has the highest number of protected adjacencies.

2. When the **local-sr-protection** command is set to a value of **mandatory**, CSPF uses it as an additional path constraint and selects protected adjacencies exclusively in computing the path of the SR-TE LSP. Adjacency sets cannot be used because they are always unprotected.

   If no path that satisfies the other LSP path constraints and consists of all TE links with protected adjacencies, the path computation returns no path.
3. Similarly, when the **local-sr-protection** command to **none**, CSPF uses it as an additional path constraint and selects unprotected adjacencies exclusively in computing the path of the SR-TE LSP.

   If a parallel adjacency set exists between two routers in a path and all the member links satisfy the constraints of the path, it is selected in preference to a single unprotected adjacency.

   If no path satisfies the other LSP path constraints and consists of all TE links with unprotected adjacencies, the path computation returns no path.

The **local-sr-protection** command impacts PCE-computed and PCE-controlled SR-TE LSP. When the **local-sr-protection** command is set to the default value **preferred**, or to the explicit value of **mandatory**, the local-protection-desired flag (L-flag) in the LSPA object in the PCReq (Request) message or in the PCRpt (Report) message is set to a value of 1.

When the **local-sr-protection** command is set to **none**, the local-protection-desired flag (L-flag) in the LSPA object is cleared. The PCE path computation checks this flag to decide if protected adjacencies are used in preference to unprotected adjacencies (L-flag set) or must not be used at all (L-flag clear) in the computation of the SR-TE LSP path.

2.2.5.3.2 SR-TE LSP Path Label Stack Reduction

The objective of the label stack reduction is twofold:

- It reduces the label stack so ingress PE routers with a lower Maximum SID Depth (MSD) can still work.
- It provides the ability to spray packets over ECMP paths to an intermediate node SID when all these paths satisfy the constraints of the SR-TE LSP path. Even if the resulting label stack is not reduced, this aspect of the feature is still useful.

If the user enables the **label-stack-reduction** command for this LSP, a second phase is applied attempting to reduce the label stack that resulted from the fully explicit path with adjacency SIDs and adjacency sets SIDs computed in the first phase.

This is to attempt a replacement of adjacency and adjacency set SIDs corresponding to a segment of the explicit path with a node SID whenever the constraints of the path are met by all the ECMP paths to that node SID.

This is the procedure followed by the label stack reduction algorithm:

1. Phase 1 of the CSPF returns up to three fully explicit ECMP paths that are eligible for label stack reduction. These paths are equal cost from the point of view of IGP metric or TE metric as configured for that SR-TE LSP.
2. Each fully explicit path of the SR-TE LSP that is computed in Phase 1 of the CSPF is split into a number of segments that are delimited by the user-configured loose or strict hops in the path of the LSP. Label stack reduction is applied to each segment separately.
3. Label stack reduction in Phase 2 consists of traversing the CSPF tree for each ECMP path returned in Phase 1 and then attempting to find the farthest node SID in a path segment that can be used to summarize the entire path up to that node SID. This requires that all links of ECMP paths are able to

reach the node SID from the current node on the CSPF tree in order to satisfy all the TE constraints of the SR-TE LSP paths. ECMP is based on the IGP metric, in this case, since this is what routers use in the data path when forwarding a packet to the node SID.

If the TE metric is enabled for the SR-TE LSP, then one of the constraints is that the TE metric must be the same value for all the IGP metric ECMP paths to the node SID.

4. CSPF in Phase 2 selects the first candidate ECMP path from Phase 1 which reduced label stack that satisfies the constraint carried in the **max-sr-labels** command.
5. The CSPF path computation in Phase 1 always avoids a loop over the same hop as is the case with the RSVP-TE LSP. In addition, the label stack reduction algorithm prevents a path from looping over the same hop due to the normal routing process. For example, it checks if the same node is involved in the ECMP paths of more than one segment of the LSP path and builds the label stack to avoid this situation.
6. During the MBB procedure of a timer or manual re-optimization of a SR-TE LSP path, the TE-DB performs additional steps as compared to the case of the initial path computation:

   - MPLS provides TE-DB with the current working path of the SR-TE LSP.
   - TE-DB updates the path's metric based on the IGP or TE link metric (if TE metric enabled for the SR-TE LSP).
     - For each adjacency SID, it verifies that the related link and SID are still in its database and that the link fulfills the LSP constraints. If so, it picks up the current metric.
     - For each node SID, it verifies that the related prefix and SID are still available, and if so, checks that all the links on the shortest IGP path to the node owning the node SID fulfill the SR-TE LSP path constraints. This is re-using the same checks detailed in Step*3* for the label compression algorithm.
   - CSPF computes a new path with or without label stack reduction as explained in Steps *1*, *2*, and *3*.
   - TE-DB returns both paths to MPLS. MPLS always programs the new path in the case of a manual re-optimization. MPLS compares the metric of the new path to the current path and if different, programs the new path in the case of a timer re-optimization.
7. TE-DB returns to MPLS the following additional information together with the reduced path ERO and label stack:

   - a list of SRLGs of each hop in the ERO represented by a node SID and that includes SRLGs used by links in all ECMP paths to reach that node SID from the previous hop.
   - the cost of each hop in the ERO represented by an adjacency SID or adjacency set SID. This corresponds to the IGP metric or TE metric (if TE is metric-enabled for the SR-TE LSP) of that link or set of links. In the case of an adjacency set, all TE metrics of the links must be the same, otherwise CSPF does not select the set.
   - the cost of each hop in the ERO represented by a node SID and this corresponds to the cumulated IGP metric or TE metric (if TE metric is enabled for the SR-TE LSP) to reach the node SID from the previous hop using the fully explicit path computed in Phase 1.
   - the total cost or computed metric of the SR-TE LSP path. This consists of the cumulated IGP metric or TE metric (if TE metric enabled for the SR-TE LSP) of all hops of the fully explicit path computed in Phase 1 of the CSPF.
8. If label stack reduction is disabled, the values of the **max-sr-labels** and the **hop-limit** commands are applied to the full explicit path in Phase 1.

   The minimum of the two values is used as a constraint in the full explicit path computation.

   If the resulting ECMP paths net hop-count in Phase 1 exceeds this minimum value no path is returned by TE-DB to MPLS

9. If label stack reduction is enabled, the values of the **max-sr-labels** and the **hop-limit** commands are both ignored in Phase 1 and only the value of the max-sr-labels is used as a constraint in Phase 2.

   If the resulting net label stack size after reduction of all candidate paths in Phase 2 exceeds the value of parameter **max-sr-labels** then no path is returned by TE-DB to MPLS.
10. The label stack reduction does not support the use of an anycast SID, a prefix SID with N-flag clear, in order to replace a segment of the SR-TE LSP path. Only a node SID is used.

2.2.5.3.3 Interaction with SR-TE LSP Path Protection

Label stack reduction is only attempted when the path protection **local-sr-protection** command is disabled or is configured to the value of **preferred**.

If **local-sr-protection** is configured to a value of **none** or **mandatory**, the command is ignored, and the fully explicit path computed out of Phase 1 is returned by the TE-DB CSPF routine to MPLS. This is because a node SID used to replace an adjacency SID or an adjacency set SID can be unprotected or protected by LFA and this is based on local configuration on each router which resolves this node SID but is not directly known in the information advertised into the TE-DB. Therefore, CSPF cannot enforce the protection constraint requested along the path to that node SID.

2.2.5.3.4 Examples of SR-TE LSP Path Label Stack Reduction

Figure 19: Label Stack Reduction in a 3-Tier Ring Topology illustrates a metro aggregation network with three levels of rings for aggregating regional traffic from edge ring routers into a PE router.

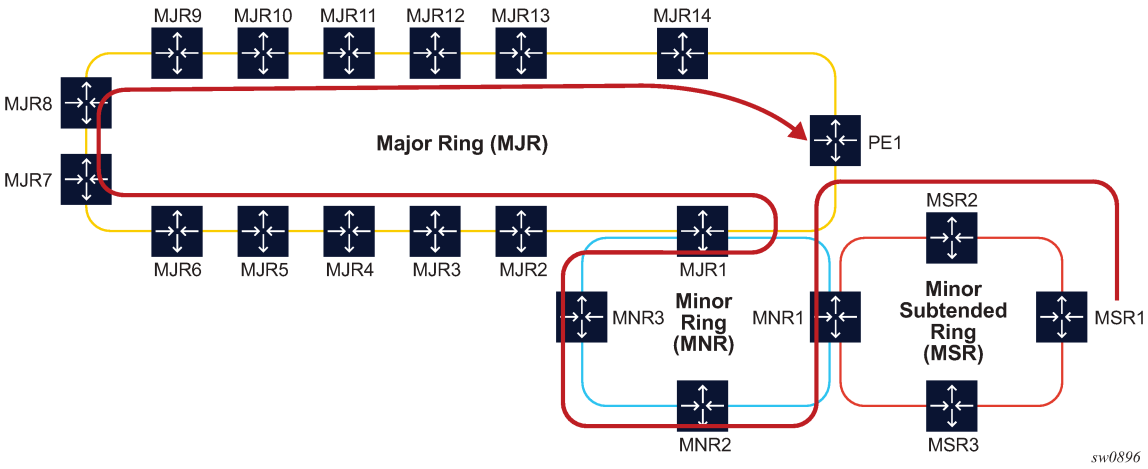

Figure 19: Label Stack Reduction in a 3-Tier Ring Topology

The path of the highlighted LSP uses admin groups to force the traffic eastwards or westwards over the 3-ring topologies such that it uses the longest path possible. Assume all links in a bottom-most ring1 have admin-group=east1 for the eastward direction and admin-group=west1 for the westward direction.

Similarly, links in middle ring2 have admin-group=east2 and admin-group=west2 and links in top-most ring3 have admin-group=east3 and admin-group=west3. To achieve the longest path shown, the LSP or path should have an include statement: include east1 west2 east3. The fully explicit path computed in Phase 1 of CSPF results in label stack of size 18.

The label stack reduction algorithm searches for the farthest node SID in that path which can replace a segment of the strict path while maintaining the stated admin-group constraints. The reduced label stack contains the SID adjacency MSR1-MSR2, the found node SIDs plus the node SID of the destination for a total of four labels to be pushed on the packet (the label for the adjacency MSR1-MSR2 is not pushed):

{N-SID MNR2, N-SID of MNR3, N-SID of MJR8, N-SID of PE1}

Figure 20: Label Stack Reduction in the Presence of ECMP Paths illustrates an example topology which creates two TE planes by applying a common admin group to all links of a given plane. There are a total of four ECMP paths to reach PE2 from PE1, two within the red plane and two within the blue plane.

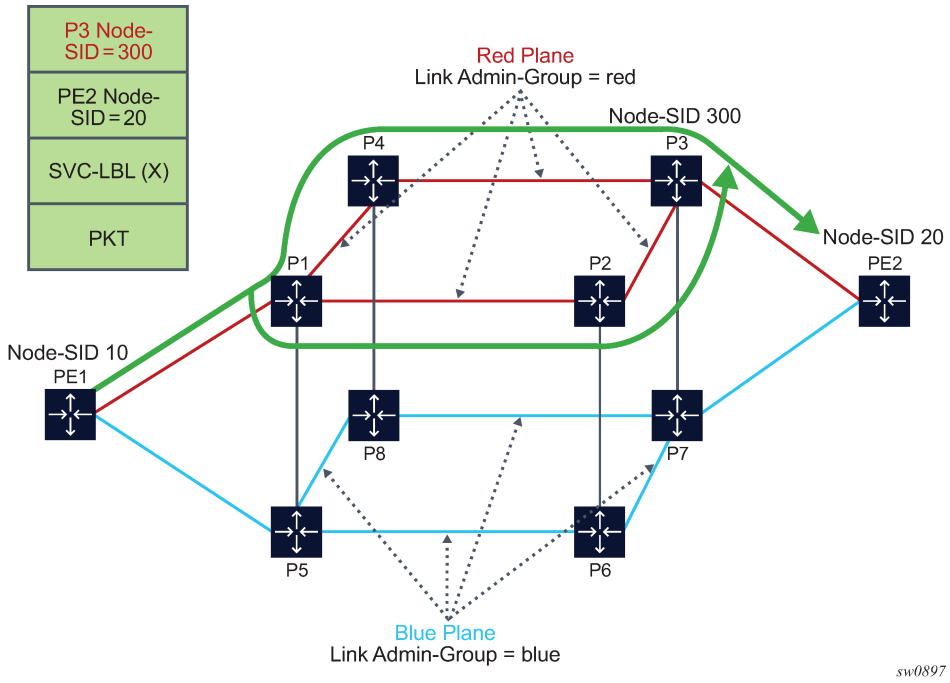

Figure 20: Label Stack Reduction in the Presence of ECMP Paths

For a SR-TE LSP from PE1 to PE2 which includes the red admin-group as a constraint, Phase 1 of CSPF results in two fully explicit paths using adjacency SID of the red TE links:

path 1 = {PE1-P1, P1-P2, P2-P3, P3-PE2}

path 2 = {PE1-P1, P1-P4, P4-P3, P3-PE2}

Phase 2 of CSPF finds node SID of P3 as the farthest hop it can reach directly from PE1 while still satisfying the 'include red' admin-group constraint. If the node SID of PE2 is used as the only SID, then traffic would also be sent over the blue links.

Then, the reduced label stack is: {P3 Node-SID=300, PE2 Node-SID=20}.

The resulting SR-TE LSP path combines the two explicit paths out of Phase 1 into a single path with ECMP support.

2.2.6 SR-TE LSP Paths using Explicit SIDs

SR OS supports the ability for SR-TE primary and secondary paths to use a configured path containing explicit SID values. The SID value for an SR-TE LSP hop is configured using the **sid-label** command under **configure**>**router**>**mpls**>**path** as follows:

```
configure router mpls
         path <name>
            [no] hop <hop-index> sid-label <sid-value>
```

Where *sid-value* specifies an MPLS label value for that hop in the path.

When SIDs are explicitly configured for a path, the user must provide all of the necessary SIDs to reach the destination. The router does not validate whether the whole label stack provided is correct other than checking that the top SID is programmed locally. A path can come up even if it contains SIDs that are invalid. The user or controller programming the path should ensure that the SIDs are correct. A path must consist of either all SIDs or all IP address hops.

A path containing SID label hops is used even if **path-computation-method** {**local-cspf** | **pce**} is configured for the LSP. That is, the path computation method configured at the LSP level is ignored when explicit SIDs are used in the path. This means that the router can bring up the path if the configured path contains SID hops even if the LSP has path computation enabled.

Note: When an LSP consists of some SID label paths and some paths under local-CSPF computation, the router cannot guarantee SRLG diversity between the CSPF paths and the SID label paths because CSPF does not know of the existence of the SID label paths because they are not listed in the TE database.

Paths containing explicit SID values can only be used by SR-TE LSPs.

2.2.7 SR-TE LSP Protection

The router supports local protection of a given segment of an SR-TE LSP, and end-to-end protection of the complete SR-TE LSP.

Each path is locally protected along the network using LFA/remote-LFA next-hop whenever possible. The protection of a node SID re-uses the LFA and remote LFA features introduced with segment routing shortest path tunnels; the protection of an adjacency SID has been added to the SR OS in the specific context of an SR-TE LSP to augment the protection level. The user must enable the **loopfree-alternates** [**remote-lfa**] option in IS-IS or OSPF.

An SR-TE LSP has state at the ingress LER only. The LSR has state for the node SID and adjacency SID, whose labels are programmed in label stack of the received packet and which represent the part of the ERO of the SR-TE LSP on this router and downstream of this router. In order to provide protection for a SR-TE LSP, each LSR node must attempt to program a link-protect or node-protect LFA next-hop in the ILM record of a node SID or of an adjacency SID and the LER node must do the same in the LTN record of the SR-TE LSP. The following are details of the behavior:

- When the ILM record is for a node SID of a downstream router which is not directly connected, the ILM of this node SID points to the backup NHLFE computed by the LFA SPF and programmed by the SR

module for this node SID. Depending on the topology and LFA policy used, this can be a link-protect or node-protect LFA next-hop.

This behavior is already supported in the SR shortest path tunnel feature at both LER and LSR. As such, an SR-TE LSP that transits at an LSR and that matches the ILM of a downstream node SID automatically takes advantage of this protection when enabled. If required, node SID protection can be disabled under the IGP instance by excluding the prefix of the node SID from LFA.

- When the ILM is for a node SID of a directly connected router, then the LFA SPF only provides link protection. The ILM or LTN record of this node SID points to the backup NHLFE of this LFA next-hop. An SR-TE LSP that transits at an LSR and that matches the ILM of a neighboring node SID automatically takes advantage of this protection when enabled.

  Note: Only link protection is possible in this case because packets matching this ILM record can either terminate on the neighboring router owning the node SID or can be forwarded to different next-hops of the neighboring router; that is, to different next-next-hops of the LSR providing the protection. The LSR providing the connection does not have context to distinguish among all possible SR-TE LSPs and, as such, can only protect the link to the neighboring router.

- When the ILM or LTN record is for an adjacency SID, it is treated as in the case of a node SID of a directly connected router (as above).

  When protecting an adjacency SID, the PLR first tries to select a parallel link to the node SID of the directly connected neighbor. That is the case when this node SID is reachable over parallel links. The selection is based on lowest interface ID. When no parallel links exist, then regular LFA/rLFA algorithms are applied to find a loopfree path to reach the node SID of the neighbor via other neighbors.

  The ILM or LTN for the adjacency SID must point to this backup NHLFE and will benefit from FRR link-protection. As a result, an SR-TE LSP that transits at an LSR and matches the ILM of a local adjacency SID automatically takes advantage of this protection when enabled.
- At the ingress LER, the LTN record points to the SR-TE LSP NHLFE, which itself will point to the NHLFE of the SR shortest path tunnel to the node SID or adjacency SID of the first hop in the ERO of the SR-TE LSP. As such, the FRR link or node protection at ingress LER is inherited directly from the SR shortest path tunnel.

When an adjacency to a neighbor fails, IGP withdraws the advertisement of the link TLV information as well as its adjacency SID sub-TLV. However, the LTN or ILM record of the adjacency SID must be kept in the data path for a sufficient period of time to allow the ingress LER to compute a new path after IGP converges. If the adjacency is restored before the timer expires, the timer is aborted as soon as the new ILM or LTN records are updated with the new primary and backup NHLFE information. By default, the ILM/LTN and NHLFE information is kept for a period of 15 seconds.

The adjacency SID hold timer is configured using the **adj-sid-hold** command, and activated when the adjacency to neighbor fails due to the following conditions:

- The network IP interface went down due a link or port failure or due to the user performing a shutdown of the port.
- The user shuts down the network IP interface in the **config**>**router** or **config**>**router**>**ospf**/**isis** context.
- The adjacency SID hold timer is not activated if the user deleted an interface in the **config**>**router**>**ospf**/**isis** context.

Note:

- The adjacency SID hold timer does not apply to the ILM or LTN of a node SID, because NHLFE information is updated in the data path as soon as IGP is converged locally and a new primary and LFA backup next-hops have been computed.
- The label information of the primary path of the adjacency SID is maintained and re-programmed if the adjacency is restored before the above timer expires. However, the backup NHLFE may change when a new LFA SPF is run while the adjacency ILM is being held by the timer running. An update to the backup NHLFE is performed immediately following the LFA SPF and may cause packets to drop.
- A new PG-ID is assigned each time an adjacency comes back up. This PG-ID is used by the ILM of the adjacency SID and the ILMs of all downstream node SIDs which resolve to the same next-hop.

While protection is enabled globally for all node SIDs and local adjacency SIDs when the user enables the **loopfree-alternates** option in ISIS or OSPF at the LER and LSR, there are applications where the user wants traffic to never divert from the strict hop computed by CSPF for a SR-TE LSP. In that case, the user can disable protection for all adjacency SIDs formed over a given network IP interface using the **sid-protection** command.

The protection state of an adjacency SID is advertised in the B-FLAG of the IS-IS or OSPF Adjacency SID sub-TLV. No mechanism exists in PCEP for the PCC to signal to PCE the constraint to use only adjacency SIDs, which are not protected. The Path Profile ID is configured in PCE with the no-protection constraint.

2.2.7.1 Local Protection

Each path may be locally protected through the network using LFA/remote-LFA nexthop whenever possible. The protection of a SID node re-uses the LFA and remote LFA features introduced with segment routing shortest path tunnels; the protection of an adjacency SID has been added to the SR OS in the specific context of an SR-TE LSP to augment the protection level. The user must enable the **loopfree-alternates remote-lfa** option in IS-IS or OSPF.

This behavior is already supported in the SR shortest path tunnel feature at both LER and LSR. As such, an SR-TE LSP that transits at an LSR and that matches the ILM of a downstream SID node automatically takes advantage of this protection when enabled. If required, SID node protection can be disabled under the IGP instance by excluding the prefix of the SID node from LFA.

2.2.7.2 End to End Protection

This section provides a brief introduction to end to end protection for SR-TE LSPs. See *Seamless BFD for SR-TE LSPs* for more detailed description of protection switching using Seamless BFD and a configured failure-action.

End-to-end protection for SR-TE LSPs is provided using secondary or standby paths. Standby paths are permanently programmed in the data path, while secondary paths are only programmed when they are activated. S-BFD is used to provide end-to-end connectivity checking. The **failure-action failover-or-down** command under the **bfd** context of the LSP configures a switchover from the currently active path to an available standby or secondary path if the S-BFD session fails on the currently active path. If S-BFD is not configured, then the router that is local to a segment can only detect failures of the top SID for that segment. End-to-end protection with S-BFD may be combined with local protection, but it is recommended that the S-BFD control packet timers be set to 1 second or more to allow sufficient time for any local protection action for a given segment to complete without triggering S-BFD to go down on the end to end LSP path.

To prevent failure between the paths of an SR-TE LSP, that is to avoid, for example, a failure of a primary path that affects its standby backup path, then disjoint paths should be configured or the **srlg** command configured on the secondary paths.

As with RSVP-TE LSPs, SR-TE standby paths support the configuration of a path preference. This value is used to select the standby path to be used when more than one available path exists.

For more details of end to end protection of SR-TE LSPs with S-BFD, see section *Seamless BFD for SR-TE LSPs*.

2.2.8 Seamless BFD for SR-TE LSPs

Seamless BFD (S-BFD) is a form of BFD that requires significantly less state and reduces the need for session bootstrapping as compared to LSP BFD. For more information, refer to "Seamless Bidirectional Forwarding Detection (S-BFD)" in *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*. S-BFD also requires centralized configuration for the reflector function, as well as a mapping at the head-end node between the remote session discriminator and the IP address for the reflector by each session. This configuration and the mapping are described in the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*. This user guide describes the application of S-BFD to SR-TE LSPs, and the LSP configuration required for this feature.

S-BFD is supported in the following SR objects or contexts:

- PCC-Initiated:
  - SR-TE LSP level
  - SR-TE primary path
  - SR-TE secondary and standby path
- PCE-Initiated SR-TE LSPs
- SR-TE auto-LSPs

2.2.8.1 Configuration of S-BFD on SR-TE LSPs

For PCC-initiated or PCC-controlled LSPs, it is possible to configure an S-BFD session under the SR-TE LSP context, the primary path context, and the SR-TE secondary path by using the **config>router>mpls>lsp**, **config>router>mpls >lsp>primary**, and **config>router>mpls>lsp>secondary** commands.

The remote discriminator value is determined by passing the "to" address of the LSP to BFD, which then matches it to a mapping table of peer IP addresses to reflector remote discriminators, that are created by the centralized configuration under the IGP (refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*). If there is no match to the "to" address of the LSP, then a BFD session is not established on the LSP or path.

Note: A remote peer IP address to discriminator mapping must exist prior to bringing an LSP administratively up.

The referenced BFD template must specify parameters consistent with an S-BFD session. For example, the endpoint type is **cpm-np** for platforms supporting a CPM P-chip, otherwise a CLI error is generated. The same BFD template can be used for both S-BFD and any other type of BFD session requested by MPLS.

If S-BFD is configured at the LSP level, then sessions are created on all paths of the LSP.

```
config>router>mpls>lsp <name> sr-te
      bfd
        [no] bfd-enable
        [no] bfd-template
        [no] wait-for-up-timer <seconds>
        exit
```

S-BFD can alternatively be configured on the primary or a specific secondary path of the LSP, as follows:

```
config>router>mpls>lsp <name> sr-te
   primary <name>
      bfd
        [no] bfd-enable
        [no] bfd-template <name>
        [no] wait-for-up-timer <seconds>
        exit
```

```
config>router>mpls>lsp <name> sr-te
   secondary <name>
      bfd
        [no] bfd-enable
        [no] bfd-template <name>
        [no] wait-for-up-timer <seconds>
        exit
      standby
```

The wait-for-up-timer is only applicable if failure action is **failover-or-down**. For more information, see *Support for BFD Failure Action with SR-TE LSPs*.

For PCE-initiated LSPs and SR-TE auto LSPs, S-BFD session parameters are specified in the LSP template. The "to" address that is used for determining the remote discriminator is derived from the far end address of the auto LSP or PCE-initiated LSP.

```
config>router>mpls
   lsp-template <name> pce-init-p2p-sr-te <default | 1...4294967295>
      bfd
        [no] bfd-enable
        [no] bfd-template
        [no] wait-for-up-timer <seconds>
```

```
config>router>mpls
   lsp-template <name> mesh-sr-te <1...4294967295>
      bfd
        [no] bfd-enable
        [no] bfd-template
        [no] wait-for-up-timer <seconds>
```

```
config>router>mpls
   lsp-template <name> p2p-sr-te <1...4294967295>
      bfd
        [no] bfd-enable
        [no] bfd-template
        [no] wait-for-up-timer <seconds>
```

2.2.8.2 Support for BFD Failure Action with SR-TE LSPs

SR OS supports the configuration of a **failure-action** of type **failover-or-down** for SR-TE LSPs. The **failure-action** command is configured at the LSP level or in the LSP template. It can be configured whether S-BFD is applied at the LSP level or the individual path level.

For LSPs with a primary path and a standby or secondary path and **failure-action** of type **failover-or-down**:

- A path is held in an operationally down state when its S-BFD session is down.
- If all paths are operationally down, then the SR-TE LSP is taken operationally down and a trap is generated.
- If S-BFD is enabled at the LSP or active path level, a switchover from the active path to an available path is triggered on failure of the S-BFD session on the active path (primary or standby).
- If S-BFD is not enabled on the active path, and this path is shut down, then a switchover is triggered.
- If S-BFD is enabled on the candidate standby or secondary path, then this path is only selected if S-BFD is up.
- An inactive standby path with S-BFD configured is only considered as available to become active if it is not operationally down, for example, its S-BFD session, is up and all other criteria for it to become operational are true. It is held in an inactive state if the S-BFD session is down.
- The system does not revert to the primary path, nor start a reversion timer when the primary path is either administratively down or operationally down, because the S-BFD session is not up or down for any other reason.

For LSPs with only one path and **failure-action** of type **failover-or-down**:

- A path is held in an operationally down state when its S-BFD session is down.
- If the path is operationally down, then the LSP is taken operationally down and a trap is generated.

Note: S-BFD and other OAM packets can still be sent on an operationally down SR-TE LSP.

2.2.8.2.1 SR-TE LSP State Changes and Failure Actions Based on S-BFD

A path is first configured with S-BFD. This path is held operationally down and not added to the TTM until BFD comes up (subject to the BFD wait time).

The BFD **wait-for-up-timer** provides a mechanism that cleans up the LSP path state at the head end in both cases where S-BFD does not come up in the first place, and where S-BFD goes from up to down. This timer is started when BFD is first enabled on a path or an existing S-BFD session transitions from up to down. When this timer expires and if S-BFD is not up, the path is torn down by removing it from the TTM and the IOM and the LSP retry timer is started.

In the S-BFD up to down case, if there is only one path, the LSP is removed immediately from the TTM when S-BFD fails, and then deprogrammed when the **wait-for-up-timer** expires.

If all the paths of an LSP are operationally down due to S-BFD, then the LSP is taken operationally down and removed from the TTM and the BFD **wait-for-up-timer** is started for each path. If one or more paths do not have S-BFD configured on them, or are otherwise not down, then the LSP is not taken operationally down.

When an existing S-BFD session fails on a path and the failure action is **failover-or-down**, the path is put into the operationally down state. This state and reason code are displayed in a **show>router>bfd>seamless-bfd** command and a trap is raised. The configured failure action is then enacted.

2.2.8.3 S-BFD Operational Considerations

A minimum control packet timer transmit interval of 10 ms can be configured. To maximize the reliability of S-BFD connectivity checking in scaled scenarios with short timers, cases where BFD can go down due to normal changes of the next hop of an LSP path at the head end must be avoided. It is therefore recommended that LFA is not configured at the head end LER when using S-BFD with sub-second timers. When the LFA is not configured, protection of the SR-TE LSP is still provided end-to-end by the combination of S-BFD connectivity checking and primary or secondary path protection.

Similar to the case of LDP and RSVP, S-BFD uses a single path for a loose hop; multiple S-BFD sessions for each of the ECMP paths or spraying of S-BFD packets across the paths is not supported. S-BFD is not down until all the ECMP paths of the loose hop go down.

Note: With very short control packet timer values in scaled scenarios, S-BFD may bounce if the next-hop that the path is currently using goes down because it takes a finite time for BFD to be updated to use another next-hop in the ECMP set.

2.2.9 Static Route Resolution using SR-TE LSP

The user can forward packets of a static route to an indirect next-hop over an SR-TE LSP programmed in TTM by configuring the following static route tunnel binding command:

```
config>router>static-route-entry {ip-prefix/prefix-length} [mcast] indirect {ip-address}
 tunnel-next-hop
        – resolution {any | disabled | filter}
        – resolution-filter
            – [no] sr-te
                – [no] [lsp name1]
                – [no] [lsp name2]
                – .
                – .
                – [no] [lsp name-N]
            – exit
        – [no] disallow-igp
        – exit
    – exit
```

The user can select the **sr-te** tunnel type and either specify a list of SR-TE LSP names to use to reach the indirect next-hop of this static route or have the SR-TE LSPs automatically select the indirect next-hop in TTM.

2.2.10 BGP Shortcuts Using SR-TE LSP

The user can forward packets of BGP prefixes over an SR-TE LSP programmed in TTM by configuring the following BGP shortcut tunnel binding command:

```
config>router>bgp>next-hop-resolution
```

```
        – shortcut-tunnel
            – [no] family {ipv4}
                – resolution {any | disabled | filter}
                – resolution-filter
                    – [no] sr-te
                – exit
            – exit
        – exit
```

2.2.11 BGP Label Route Resolution Using SR-TE LSP

The user can enable SR-TE LSP, as programmed in TTM, for resolving the next-hop of a BGP IPv4 or IPv6 (6PE) label route by enabling the following BGP transport tunnel command:

```
config>router>bgp>next-hop-res>
        – labeled-routes
            – transport-tunnel
                – [no] family {label-ipv4 | label-ipv6 | vpn}
                    – resolution {any | disabled | filter}
                    – resolution-filter
                        – [no] sr-te
                    – exit
                – exit
            – exit
```

2.2.12 Service Packet Forwarding using SR-TE LSP

An SDP sub-type of the MPLS encapsulation type allows service binding to a SR-TE LSP programmed in TTM by MPLS:

```
*A:7950 XRS-20# configure service sdp 100 mpls create
    – *A:7950 XRS-20>config>service>sdp$ sr-te-lsp lsp-name
```

The user can specify up to 16 SR-TE LSP names. The destination address of all LSPs must match that of the SDP far-end option. Service data packets are sprayed over the set of LSPs in the SDP using the same procedures as for tunnel selection in ECMP. Each SR-TE LSP can, however, have up to 32 next-hops at the ingress LER when the first segment is a node SID-based SR tunnel. Consequently, service data packet will be forwarded over one of a maximum of 16x32 next-hops. The **tunnel-far-end** option is not supported. In addition, the **mixed-lsp-mode** option does not support the **sr-te** tunnel type.

The signaling protocol for the service labels for an SDP using a SR-TE LSP can be configured to static (**off**), T-LDP (**tldp**), or BGP (**bgp**).

An SR-TE LSP can be used in VPRN auto-bind with the following commands:

```
config>service>vprn>
        – auto-bind-tunnel
            – resolution {any | disabled | filter}
            – resolution-filter
                – [no] sr-te
            – exit
        – exit
```

Both VPN-IPv4 and VPN-IPv6 (6VPE) are supported in a VPRN service using segment routing transport tunnels with the **auto-bind-tunnel** command.

This **auto-bind-tunnel** command is also supported with BGP EVPN service, as shown below:

```
config>service>vpls>bgp-evpn>mpls>
        – auto-bind-tunnel
            – resolution {any | disabled | filter}
            – resolution-filter
                – [no] sr-te
            – exit
        – exit
```

The following service contexts are supported with SR-TE LSP:

- VLL, LDP VPLS, IES/VPRN spoke-interface, R-VPLS, BGP EVPN
- BGP-AD VPLS, BGP-VPLS, BGP VPWS when the **use-provisioned-sdp** option is enabled in the binding to the PW template
- intra-AS BGP VPRN for VPN-IPv4 and VPN-IPv6 prefixes with both auto-bind and explicit SDP
- inter-AS options B and C for VPN-IPv4 and VPN-IPv6 VPRN prefix resolution
- IPv4 BGP shortcut and IPv4 BGP label route resolution
- IPv4 static route resolution
- multicast over IES/VPRN spoke interface with **spoke SDP** riding a SR-TE LSP

2.2.13 Data Path Support

The support of SR-TE in the data path requires that the ingress LER pushes a label stack where each label represents a hop, a TE link, or a node, in the ERO for the LSP path computed by the router or the PCE. However, only the label and the outgoing interface to the first strict/loose hop in the ERO factor into the forwarding decision of the ingress LER. In other words, the SR-TE LSP only needs to track the reachability of the first strict/loose hop.

This actually represents the NHLFE of the SR shortest path tunnel to the first strict/loose hop. SR OS keeps the SR shortest path tunnel to a downstream node SID or adjacency SID in the tunnel table and so its NHLFE is readily available. The rest of the label stack is not meaningful to the forwarding decision. In this document, "super NHLFE" refers to this part of the label stack because it can have a much larger size.

As a result, an SR-TE LSP is modeled in the ingress LER data path as a hierarchical LSP with the super NHLFE is tunneled over the NHLFE of the SR shortest path tunnel to the first strict/loose hop in the SR-TE LSP path ERO.

Some characteristics of this design are as follows:

- The design saves on NHLFE usage. When many SR TE LSPs are going to the same first hop, they are riding the same SR shortest path tunnel, and will consume each one super NHLFE but they are pointing to a single NHLFE, or set of NHLFEs when ECMP exists for the first strict/loose hop, of the first hop SR tunnel.

  Also, the ingress LER does not need to program a separate backup super NHLFE. Instead, the single super NHLFE will automatically begin forwarding packets over the LFA backup path of the SR tunnel to the first hop as soon as the SR tunnel LFA backup path is activated.

- When the path of a SR-TE LSP contains a maximum of two SIDs, that is the destination SID and one additional loose or strict-hop SID, the SR-TE LSP will use a hierarchy consisting of a regular NHLFE pointing to the NHLFE of top SID corresponding to the first loose or strict hop.
- If the first segment is a node SID tunnel and multiple next-hops exist, then ECMP spraying is supported at the ingress LER.
- If the first hop SR tunnel, node or adjacency SID, goes down the SR module informs MPLS that outer tunnel down and MPLS brings the SR-TE LSP down and requests SR to delete the SR-TE LSP in IOM.

The data path behavior at LSR and egress LER for an SR-TE LSP is similar to that of shortest path tunnel because there is no tunnel state in these nodes. The forwarding of the packet is based on processing the incoming label stack consisting of a node SID and/or adjacency SID label. If the ILM is for a node SID and multiple next-hops exist, then ECMP spraying is supported at the LSR.

The link-protect LFA backup next-hop for an adjacency SID can be programmed at the ingress LER and LSR nodes (as explained in *SR-TE LSP Protection*).

A maximum of 12 labels, including all transport, service, hash, and OAM labels, can be pushed. The label stack size for the SR-TE LSP can be 1 to 11 labels, with a default value of 6.

The maximum value of 11 is obtained for an SR-TE LSP whose path is not protected via FRR backup and with no entropy or hash label feature enabled when such an LSP is used as a shortcut for an IGP IPv4/IPv6 prefix or as a shortcut for BGP IPv4/IPv6. In this case, the IPv6 prefix requires pushing the IPv6 explicit-null label at the bottom of the stack. This leaves 11 labels for the SR-TE LSP.

The default value of 6 is obtained in the worst cases, such as forwarding a vprn-ping packet for an inter-AS VPN-IP prefix in Option C:

6 SR-TE labels + 1 remote LFA SR label + BGP 3107 label + ELI (RFC 6790) + EL (entropy label) + service label + OAM Router Alert label = 12 labels.

The label stack size manipulation includes the following LER and LSR roles:

LER role:

- Push up to 12 labels.
- Pop up to 8 labels of which 4 labels can be transport labels

LSR role:

- Pop up to 5 labels and swap one label for a total of 6 labels
- LSR hash of a packet with up to 16 labels

An example of the label stack pushed by the ingress LER and by a LSR acting as a PLR is illustrated in *Figure 21: SR-TE LSP Label Stack Programming*.

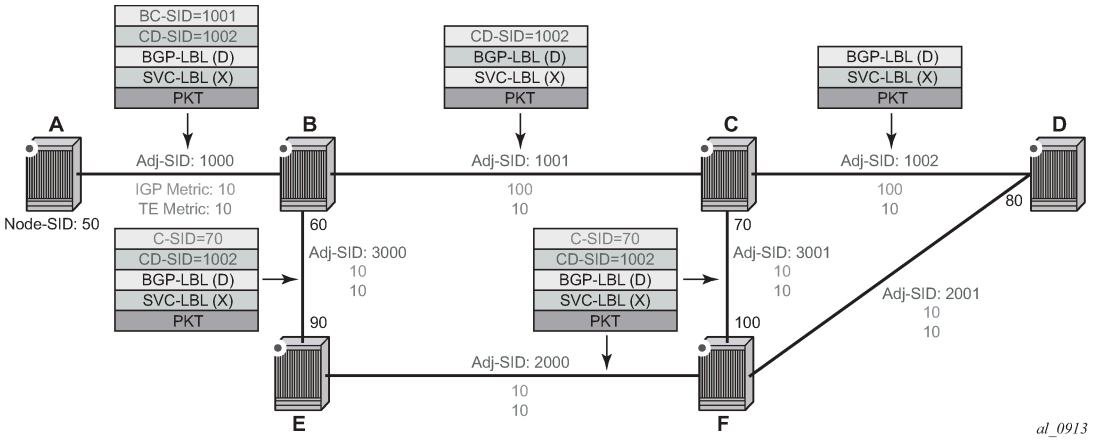

Figure 21: SR-TE LSP Label Stack Programming

On node A, the user configures an SR-TE LSP to node D with a list of explicit strict hops mapping to the adjacency SID of links: A-B, B-C, and C-D.

Ingress LER A programs a super NHLFE consisting of the label for the adjacency over link C-D and points it to the already-programmed NHLFE of the SR tunnel of its local adjacency over link A-B. The latter NHLFE has the top label and also the outgoing interface to send the packet to.

Note: SR-TE LSP does not consume a separate backup super NHLFE; it only points the single super NHLFE to the NHLFE of the SR shortest path tunnel it is riding. When the latter activates its backup NHLFE, the SR-TE LSP will automatically forward over it.

LSR Node B already programmed the primary NHLFE for the adjacency SID over link C-D and has the ILM with label 1001 point to it. In addition, node B will pre-program the link-protect LFA backup next-hop for link B-C and point the same ILM to it.

Note: There is no super NHLFE at node B as it only deals with the programming of the ILM and primary/backup NHLFE of its adjacency SIDs and its local and remote node SIDs.

VPRN service in node A forwards a packet to the VPN-IPv4 prefix X advertised by BGP peer D. *Figure 21: SR-TE LSP Label Stack Programming* shows the resulting data path at each node for the primary path and for the FRR backup path at LSR B.

2.2.13.1 SR-TE LSP Metric and MTU Settings

The MPLS module assigns a TE-LSP the maximum LSP metric value of 16777215 when the local router provides the hop-to-label translation for its path. For a TE-LSP that uses the local CSPF or the PCE for path computation (**path-computation-method pce** option enabled) by PCE and/or which has its control delegated to PCE (**pce-control** enabled), the latter will return the computed LSP IGP or TE metric in the PCReq and PCUpd messages. In both cases, the user can override the returned value by configuring an admin metric using the command **config>router>mpls>lsp>metric**.

1. The MTU setting of a SR-TE LSP is derived from the MTU of the outgoing SR shortest path tunnel it is riding, adjusted with the size of the super NHLFE label stack size.

   The following are the details of this calculation:

   SR_Tunnel_MTU = MIN {Cfg_SR_MTU, IGP_Tunnel_MTU- (1+*frr-overhead*)*4}

   Where:

   - **Cfg_SR_MTU** is the MTU configured by the user for all SR tunnels within a given IGP instance using **config**>**router**>**ospf**/**isis**>**segment-routing**>**tunnel-mtu**. If no value was configured by the user, the SR tunnel MTU is fully determined by the IGP interface calculation (explained below).
   - **IGP_Tunnel_MTU** is the minimum of the IS-IS or OSPF interface MTU among all the ECMP paths or among the primary and LFA backup paths of this SR tunnel.
   - *frr-overhead* is set to:
     - value of **ti-lfa** [**max-sr-frr-labels** *labels*] if **loopfree-alternates** and **ti-lfa** are enabled in this IGP instance
     - 1 if **loopfree-alternates** and **remote-lfa** are enabled but **ti-lfa** is disabled in this IGP instance
     - 0 for all other cases

   This calculation is performed by IGP and passed to the SR module each time it changes due to an updated resolution of the node SID.

   SR OS also provides the MTU for adjacency SID tunnel because it is needed in a SR-TE LSP if the first hop in the ERO is an adjacency SID. In that case, this calculation for SR_Tunnel_MTU, initially introduced for a node SID tunnel, is applied to get the MTU of the adjacency SID tunnel.
2. The MTU of the SR-TE LSP is derived as follows:

   SRTE_LSP_MTU= SR_Tunnel_MTU- numLabels*4

   Where:

   - **SR_Tunnel_MTU** is the MTU SR tunnel shortest path the SR-TE LSP is riding. The formula is as given above.
   - **numLabels** is the number of labels found in the super NHLFE of the SR-TE LSP. Note that at LER, the super NHLFE is pointing to the SR tunnel NHLFE, which itself has a primary and a backup NHLFEs.

   This calculation is performed by the SR module and is updated each time the SR-TE LSP path changes or the SR tunnel it is riding is updated.

   Note: The above calculated SR-TE LSP MTU is used for the determination of an SDP MTU and for checking the Layer 2 service MTU. For the purpose of fragmentation of IP packets forwarded in GRT or in a VPRN over a SR-TE LSP, the data path always deducts the worst case MTU (12 labels) from the outgoing interface MTU for the decision to fragment or not the packet. In this case, the above formula is not used.

2.2.13.2 LSR Hashing on SR-TE LSPs

The LSR supports hashing up to a maximum of 16 labels in a stack. The LSR is able to hash on the IP headers when the payload below the label stack is IPv4 or IPv6, including when a MAC header precedes

it (**ethencap-ip** option). Alternatively, it is able to hash based only on the labels in the stack, which may include the entropy label (EL) or the hash label. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR MPLS Guide* for more information about the hash label and entropy label features.

When the hash-label option is enabled in a service context, a hash label is always inserted at the bottom of the stack as per RFC 6391.

The EL feature, as specified in RFC 6790, indicates the presence of a flow on an LSP that should not be reordered during load balancing. It can be used by an LSR as input to the hash algorithm. The Entropy Label Indicator (ELI) is used to indicate the presence of the EL in the label stack. The ELI, followed by the actual EL, is inserted immediately below the transport label for which the EL feature is enabled. If multiple transport tunnels have the EL feature enabled, the ELI and EL are inserted below the lowest transport label in the stack.

The EL feature is supported with an SR-TE LSP. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR MPLS Guide* for more information.

The LSR hashing operates as follows:

- If the **lbl-only** hashing option is enabled, or if one of the other LSR hashing options is enabled but an IPv4 or IPv6 header is not detected below the bottom of the label stack, the LSR parses the label stack and hashes only on the EL or hash label.
- If the **lbl-ip** option is enabled, the LSR parses the label stack and hashes on the EL or hash label and the IP headers.
- If the **ip-only** or **eth-encap-ip** is enabled, the LSR hashes on the IP headers only.

2.2.14 SR-TE Auto-LSP

The SR-TE auto-LSP feature allows the auto-creation of an SR-TE mesh LSP and for an SR-TE one-hop LSP.

The SR-TE mesh LSP feature specifically binds an LSP template of a new type, **mesh-p2p-srte**, with one more prefix list. When the TE database discovers a router, which has a router ID matching an entry in the prefix list, it triggers MPLS to instantiate an SR-TE LSP to that router using the LSP parameters in the LSP template.

The SR-TE one-hop LSP feature specifically activates an LSP template of a new type, **one-hop-p2p-srte**. In this case, the TE database keeps track of each TE link which comes up to a directly connected IGP neighbor. It then instructs MPLS to instantiate an SR-TE LSP with the following parameters:

- the source address of the local router
- an outgoing interface matching the interface index of the TE-link
- a destination address matching the router-id of the neighbor on the TE link

In both types of SR-TE auto-LSP, the router's hop-to-label translation or local CSPF computes the label stack required to instantiate the LSP path.

Note: An SR-TE auto-LSP can be reported to a PCE but cannot be delegated or have its paths computed by PCE.

2.2.14.1 Feature Configuration

This feature introduces two new LSP template types: **one-hop-p2p-srte** and **mesh-p2p-srte**. The configuration for these commands is the same as that of the RSVP-TE auto-lsp of type **one-hop-p2p** and **mesh-p2p** respectively.

The user first creates an LSP template of the one of the following types:

- **config**>**router**>**mpls**>**lsp-template** *template-name* **mesh-p2p-srte**
- **config**>**router**>**mpls**>**lsp-template** *template-name* **one-hop-p2p-srte**

In the template, the user configures the common LSP and path level parameters or options shared by all LSPs using this template.

These new types of LSP templates contain the SR-TE LSP-specific commands as well as all other LSP or path commands common to RSVP-TE LSP and SR-TE LSP, and which are supported by the existing RSVP-TE LSP template.

Next, the user either binds the LSP template of type **mesh-p2p-srte** with one or more prefix lists using the **config**>**router**>**mpls**>**lsp-template** *template-name* **policy** *peer-prefix-policy1* [*peer-prefix-policy2*] command, or binds the LSP template of type **one-hop-p2p-srte** with the **one-hop** option using the **config**>**router**>**mpls**>**lsp-template** *template-name* **one-hop** command.

See *Configuring and Operating SR-TE* for an example configuration of the SR-TE auto-LSP creation using an LSP template of type **mesh-p2p-srte**.

2.2.14.2 Automatic Creation of an SR-TE Mesh LSP

This feature behaves the same way as the RSVP-TE auto-LSP using an LSP template of type **mesh-p2p**.

The **auto-lsp** command binds an LSP template of type **mesh-p2p-srte** with one or more prefix lists. When the TE database discovers a router that has a router ID matching an entry in the prefix list, it triggers MPLS to instantiate an SR-TE LSP to that router using the LSP parameters in the LSP template.

The prefix match can be exact or longest. Prefixes in the prefix list that do not correspond to a router ID of a destination node will never match.

The path of the LSP is that of the default path name specified in the LSP template. The hop-to-label translation tool or the local CSPF determines the node SID and adjacency SID corresponding to each loose and strict hop in the default path definition respectively.

The LSP has an auto-generated name using the following structure:

"*TemplateName-DestIpv4Address-TunnelId*"

where:

- *TemplateName* = the name of the template
- *DestIpv4Address* = the address of the destination of the auto-created LSP
- *TunnelId* = the TTM tunnel ID

In SR OS, an SR-TE LSP uses three different identifiers:

- LSP Index is used for indexing the LSP in the MIB table shared with RSVP-TE LSP. Range:
  - provisioned SR-TE LSP: 65536 to 81920
  - SR-TE auto-LSP: 81921 to 131070
- LSP Tunnel Id is used in the interaction with PCC/PCE. Range: 1 to 65536
- TTM Tunnel Id is the tunnel-ID service, shortcut, and steering applications use to bind to the LSP. Range: 655362 to 720897

The path name is that of the default path specified in the LSP template.

Note: This feature is limited to SR-TE LSP, that is controlled by the router (PCC-controlled) and which path is provided using the hop-to-label translation or the local CSPF path computation method.

2.2.14.3 Automatic Creation of an SR-TE One-Hop LSP

This feature like the RSVP-TE auto-LSP using an LSP template of **one-hop-p2p** type. Although the provisioning model and CLI syntax differ from that of a mesh LSP by the absence of a prefix list, the actual behavior is quite different. When the **one-hop-p2p** command is executed, the TE database keeps track of each TE link that comes up to a directly connected IGP neighbor. It then instructs MPLS to instantiate an SR-TE LSP with the following parameters:

- the source address of the local router
- an outgoing interface matching the interface index of the TE link
- a destination address matching the router ID of the neighbor on the TE link

In this case, the hop-to-label translation or the local CSPF returns the SID for the adjacency to the remote address of the neighbor on this link. Therefore, the **auto-lsp** command binding an LSP template of type **one-hop-p2p-srte** with the **one-hop** option results in one SR-TE LSP instantiated to the IGP neighbor for each adjacency over any interface.

Because the local router installs the adjacency SID to a link independent of whether the neighbor is SR-capable, the TE-DB finds the adjacency SID and a one-hop SR-TE LSP can still come up to such a neighbor. However, remote LFA using the neighbor's node SID will not protect the adjacency SID and so, will also not protect the one-hop SR-TE LSP because the node SID is not advertised by the neighbor.

The LSP has an auto-generated name using the following structure:

"*TemplateName-DestIpv4Address-TunnelId*"

where:

- *TemplateName* = the name of the template
- *DestIpv4Address* = the address of the destination of the auto-created LSP
- *TunnelId* = the TTM tunnel ID.

The path name is that of the default path specified in the LSP template.

Note: This feature is limited to an SR-TE LSP that is controlled by the router (PCC-controlled) and for which labels for the path are provided by the hop-to-label translation or the local CSPF path computation method.

2.2.14.4 Interaction with PCEP

A template-based SR-TE auto-LSP can only be operated as a PCC-controlled LSP. It can, however, be reported to the PCE using the **pce-report** command. It cannot be operated as a PCE-computed or PCE-controlled LSP. This is the same interaction with PCEP as that of a template-based RSVP-TE LSP.

2.2.14.5 Forwarding Contexts Supported with SR-TE Auto-LSP

The following are the forwarding contexts that can be used by an auto-LSP:

- resolution of IPv4 BGP label routes and IPv6 BGP label routes (6PE) in TTM
- resolution of IPv4 BGP route in TTM (BGP shortcut)
- resolution of IPv4 static route to indirect next-hop in TTM
- VPRN and BGP-EVPN auto-bind for both IPv4 and IPv6 prefixes

The auto-LSP is, however, not available to be used in a provisioned SDP for explicit binding by services. Therefore, an auto-LSP can also not be used directly for auto-binding of a PW template with the **use-provisioned-sdp** option in BGP-AD VPLS or FEC129 VLL service. However, an auto-binding of a PW template to an LDP LSP, which is then tunneled over an SR-TE auto-LSP is supported.

2.2.15 SR-TE LSP Traffic Statistics

As in RSVP-TE LSPs, it is possible to enable the collection of traffic statistics on SR-TE LSPs (using either a named LSP or SR-TE templates). However, traffic statistics are only available on egress of ingress LER. Also, traffic statistics cannot be recorded into an accounting file.

Unlike RSVP-TE LSP statistics, SR-TE LSP statistics are provided without any forwarding class or QoS profile distinction. However, traffic statistics are recorded and made available for each of the paths of the LSP (primary and backup). Statistic indexes are only allocated at the time the path is effectively programmed, are maintained across switch-over for primary and standbys only, and are released if egress statistics are disabled or the LSP is deleted.

Note: SR-TE LSP egress statistics are not supported on VSR.

2.2.16 SR-TE Label Stack Checks

2.2.16.1 Service and Shortcut Application SR-TE Label Stack Check

If a packet forwarded in a service or a shortcut application has resulted in the net label stack size being pushed on the packet to exceed the maximum label stack supported by the router, the packet is dropped on the egress. Each service and shortcut application on the router performs a check of the resulting net label stack after pushing all the labels required for forwarding the packet in that context.

To that effect, the MPLS module populates each SR-TE LSP in the TTM with the maximum transport label stack size, which consists of the sum of the values in **max-sr-labels** *label-stack-size* and **additional-frr-labels** *labels*.

Each service or shortcut application will then add the additional, context-specific labels, such as service label, entropy/hash label, and control-word, required to forward the packet in that context and to check that the resulting net label stack size does not exceed the maximum label stack supported by the router.

If the check succeeds, the service is bound or the prefix is resolved to the SR-TE LSP.

If the check fails, the service will not bind to this SR-TE LSP. Instead, it will either find another SR-TE LSP or another tunnel of a different type to bind to, if the user has configured the use of other tunnel types. Otherwise, the service will go down. When the service uses a SDP with one or more SR-TE LSP names, the spoke SDP bound to this SDP will remain operationally down as long as at least one SR-TE LSP fails the check. In this case, a new spoke SDP flag is displayed in the **show** output of the service: "labelStackLimitExceeded". Similarly, the prefix will not get resolved to the SR-TE LSP and will either be resolved to another SR-TE LSP or another tunnel type, or will become unresolved.

The value of **additional-frr-labels** *labels* is checked against the maximum value across all IGP instances of the parameter *frr-overhead*. This parameter is computed within a given IGP instance as described in *Table 7: frr-overhead Parameter Values* .

Table 7: frr-overhead Parameter Values

Condition	frr-overhead Parameter Value
segment-routing is disabled in the IGP instance	0
segment-routing is enabled but **remote-lfa** is disabled	0
segment-routing is enabled and **remote-lfa** is enabled	1

When the user configures or changes the configuration of **additional-frr-labels**, MPLS ensures that the new value accommodates the *frr-overhead* value across all IGP instances.

Example:

1. The user configures the **config>router>isis>loopfree-alternates remote-lfa** command.
2. The user creates a new SR-TE LSP or changes the configuration of an existing as follows: **mpls>lsp>max-sr-labels** *10* **additional-frr-labels** *0*.
3. Performing a **no shutdown** of the new LSP or changing the existing LSP configuration is blocked because the IS-IS instance enabled remote LFA, which requires one additional label on top of the 10 SR labels of the primary path of the SR-TE LSP.

If the check is successful, MPLS adds **max-sr-labels** and **additional-frr-labels** and checks that the result is lower or equal to the maximum label stack supported by the router. MPLS then populates the value of {**max-sr-labels** + **additional-frr-labels**}, along with tunnel information in TTM, and also passes **max-sr-labels** to the PCEP module.

Conversely, if the user tries a configuration change that results in a change to the computed *frr-overhead*, IGP will check that all SR-TE LSPs can properly account for the overhead or the change is rejected. On the IGP, enabling **remote-lfa** may cause the *frr-overhead* to change.

Example:

- An MPLS LSP is administratively enabled and has **mpls**>**lsp**>**max-sr-labels** *10* **additional-frr-overhead** *0* configured.
- The current configuration in IS-IS has the **loopfree-alternates** command disabled.
- The user attempts to configure

  isis>**loopfree-alternates remote-lfa**. This changes *frr-overhead* to 1.

  This configuration change will be blocked.

2.2.16.2 Control Plane Handling of Egress Label Stack Limitations

As described in *Data Path Support*, the egress IOM can push a maximum of 12 labels; however, this number may be reduced if other fields are pushed into the packets. For example, for a VPRN service, the ingress LER can send an IP VPN packet with 12 labels in the stack, including one service label, one label for OAM, and 10 transport labels. However, if entropy is configured, the number of transport labels is reduced by two (Entropy Label (EL) and Entropy Label Indicator (ELI)). Similarly, for EVPN services, the egress IOM might push specific fields that reduce the total number of supported transport labels.

To avoid silent packet drops in cases where the egress IOM cannot push the required number of labels, SR OS implements a set of procedures that prevent the system from sending packets if it is determined that the SR-TE label stack to be pushed exceeds the number of bytes that the egress IOM can put on the wire.

Table 8: Label Stack Egress IOM Restrictions on FP-based Hardware for IPVPN and EVPN Services describes the label stack egress IOM restrictions on FP-based hardware for IPVPN and EVPN services.

Table 8: Label Stack Egress IOM Restrictions on FP-based Hardware for IPVPN and EVPN Services

Features that reduce the Label Stack		Source Service Type				
		IP-VPN (VPRN)	EVPN-IFL (VPRN)	EVPN VPLS or EVPN Epipe	EVPN B-VPLS (PBB-EVPN)	EVPN-IFF (R-VPLS)
Always Computed[1]	Service Label	1	1	1	1	1
	OAM Label	1	1	0	0	0
	Control Word	0	0	1	1	1
	ESI Label	0	0	1	0	0

Features that reduce the Label Stack		Source Service Type				
		IP-VPN (VPRN)	EVPN-IFL (VPRN)	EVPN VPLS or EVPN Epipe	EVPN B-VPLS (PBB-EVPN)	EVPN-IFF (R-VPLS)
Computed if configured[2]	Hash Label (mutex with EL)	1	1	0	0	0
	Entropy EL+ELI	2	2	2	2	2
Required Labels[3]		2	2	3	2	2
Required Labels + Options[3]		4	4	5	4	4
Maximum available labels[4]		12	12	10	6	9
Maximum available transport labels without options[5]		10	10	7	4	7
Maximum available transport labels with options[5]		8	8	5	2	5

Note:

1. These rows indicate the number of labels that the system assumes are always used on a given service. For example, the system always computes two labels to be reduced from the total number of labels for VPRN services with EVPN-IFL (EVPN Interface-less model enabled).
2. These rows indicate the number of labels that the system subtracts from the total only if they are configured on the service. For example, on VPRN services with EVPN-IFL, if the user configures hash-label, the system computes one additional label. If the user configures entropy-label, the system deducts two labels instead.
3. These rows indicate the number of labels that the system deducts from the total number.
4. This row indicates a different number depending on the service type and the inner encapsulation used by each service, which reduces the maximum number of labels to push on egress. For example, while the number of labels for VPRN services is 12, the maximum number for VPLS and Epipe services is 10 (to account for space for an inner Ethernet header).
5. This row indicates the maximum SR-TE labels that the system can push when sending service packets on the wire.

The total number of labels configured in the command **max-sr-labels** *label-stack-size* [**additional-frr-labels** *labels*] must not exceed the labels indicated in the "Maximum available transport labels with/without options" rows in *Table 8: Label Stack Egress IOM Restrictions on FP-based Hardware for IPVPN and EVPN Services* . If the configured LSP labels exceed the available labels in the table, the BGP route next hop for the LSP is not resolved and the system does not even try to send packets to that LSP.

For example, for a VPRN service with EVPN-IFL where the user configures entropy-label, the maximum available transport labels is eight. If an IP Prefix route for next-hop X is received for the service and the SR-TE LSP to-X is the best tunnel to reach X, the system checks that (**max-sr-labels** + **additional-frr-labels**) is less than or equal to eight. Otherwise, the IP Prefix route is not resolved.

The same control plane check is performed for other service types, including IP shortcuts, spoke SDPs on IP interfaces, spoke SDPs on Epipes, VPLS, B-VPLS, R-VPLS, and R-VPLS in I-VPLS or PW-SAP. In all cases, the spoke SDP is brought down if the configured (**max-sr-labels** + **additional-frr-labels**) is greater than the maximum available transport labels. *Table 9: Maximum Available Transport Labels for IP Shortcuts and Spoke SDP services* indicates the maximum available transport labels for IP shortcuts and spoke SDP services.

Note: For PW-SAPs, the maximum available labels differ depending on the type of service PW-SAP used (Epipe or VPRN interface).

Table 9: Maximum Available Transport Labels for IP Shortcuts and Spoke SDP services

Features that reduce the Label Stack		Source Service Type							
		IP Shortcuts	Spoke-sdp Interface	Spoke-sdp Epipe	Spoke-sdp VPLS	Spoke-sdp B-VPLS	Spoke-sdp R-VPLS	Spoke-sdp R-VPLS I-VPLS	PW-SAP Epipe/ Interface
Always Computed[1]	Service Label	0	1	1	1	1	1	1	1/1
	OAM Label	0	1	1	1	1	0	0	0/0
	IPv6 label	1	0	0	0	0	0	0	0/0
Computed if configured[2]	Hash Label (mutex with EL)	0	1	1	1	1	1	0	0/0
	Entropy EL +ELI	2	2	2	2	2	2	2	0/0
	Control Word	0	1	1	1	1	1	1	0/0

Features that reduce the Label Stack	Source Service Type							
	IP Shortcuts	Spoke-sdp Interface	Spoke-sdp Epipe	Spoke-sdp VPLS	Spoke-sdp B-VPLS	Spoke-sdp R-VPLS	Spoke-sdp R-VPLS I-VPLS	PW-SAP Epipe/ Interface
Required Labels[3]	1	2	2	2	2	1	1	1/1
Required Labels + Options[3]	3	5	5	5	5	4	4	1/1
Maximum available labels[4]	12	9	10	10	6	8	4	10/7
Maximum available transport labels without options[5]	11	7	8	8	4	7	3	9/7
Maximum available transport labels with options[5]	9	4	5	5	1	4	0	8/6

Note:

1. Indicates the number of labels that the system assumes are always used on a specific service
2. Indicates the number of labels that the system subtracts from the total only if they are configured on the service
3. Number of labels that the system deducts from the total number
4. Indicates a different number depending on the service type and the inner encapsulation used by each service, which reduces the maximum number of labels to push on egress
5. Maximum SR-TE labels that the system can push when sending service packets on the wire

In general, the labels shown in *Table 8: Label Stack Egress IOM Restrictions on FP-based Hardware for IPVPN and EVPN Services* and *Table 9: Maximum Available Transport Labels for IP Shortcuts and Spoke SDP services* are valid for network ports that are null or dot1q encapsulated. For QinQ network ports, the available labels are deducted by one.

2.2.17 IPv6 Traffic Engineering

This feature extends the traffic engineering capability with the support of IPv6 TE links and nodes.

This feature enhances IS-IS, BGP-LS and the TE database with the additional IPv6 link TLVs and TE link TLVs and provides the following three modes of operation of the IPv4 and IPv6 traffic engineering in a network.

- Legacy Mode — This mode enables the existing traffic engineering behavior for IPv4 RSVP-TE and IPv4 SR-TE. Only the RSVP-TE attributes are advertised in the legacy TE TLVs that are used by both RSVP-TE and SR-TE LSP path computation in the TE domain routers. In addition, IPv6 SR-TE LSP path computation can now use these common attributes.
- Legacy Mode with Application Indication — This mode is intended for cases where link TE attributes are common to RSVP-TE and SR-TE applications and have the same value, but the user wants to indicate on a per-link basis which application is enabled.

  Routers in the TE domain use these attributes to compute path for IPv4 RSVP-TE LSP and IPv4/IPv6 SR-TE LSP.
- Application Specific Mode — This mode of operation is intended for future use cases where TE attributes may have different values in RSVP-TE and SR-TE applications or are specific to one application (for example, RSVP-TE 'Unreserved Bandwidth' and `Max Reservable Bandwidth' attributes).

  SR OS does not support configuring TE attributes that are specific to the SR-TE application. As a result, enabling this mode advertises the common TE attributes once using a new Application Specific Link Attributes TLV. Routers in the TE domain use these attributes to compute paths for IPv4 RSVP-TE LSP and IPv4/IPv6 SR-TE LSP.

See *IS-IS IPv4/IPv6 SR-TE and IPv4 RSVP-TE Feature Behavior* for more details on the IPv4 and IPv6 Traffic Engineering modes of operation.

The feature also adds support of IPv6 destinations to the SR-TE LSP configuration. In addition, this feature also extends the MPLS path configuration with hop indices that include IPv6 addresses.

IPv6 SR-TE LSP is supported with the hop-to-label and the local CSPF path computation methods. It requires the enabling of the IPv6 traffic engineering feature in IS-IS.

2.2.17.1 Global Configuration

In order to enable IPv6 TE on the router, a new parameter referred to as IPv6 TE Router ID must have a valid IPv6 address. The following CLI command is used to configure the parameter:

configure>**router**>**ipv6-te-router-id interface** *interface-name*

The IPv6 TE Router ID is a mandatory parameter and allows the router to be uniquely identified by other routers in an IGP TE domain as being IPv6 TE capable. IS-IS advertises this information using the IPv6 TE Router ID TLV as explained in *TE Attributes Supported in IGP and BGP-LS*.

When the command is not configured, or the **no** form of the command is configured, the value of the IPv6 TE Router ID parameter reverts to the preferred primary global unicast address of the system interface. The user can also explicitly enter the name of the system interface to achieve the same outcome.

In addition, the user can specify a different interface and the preferred primary global unicast address of that interface is used instead. Only the system or a loopback interface is allowed since the TE router ID must use the address of a stable interface.

This address must be reachable from other routers in a TE domain and the associated interface must be added to IGP for reachability. Otherwise, IS-IS withdraws the advertisement of the IPv6 TE Router ID TLV.

When configuring a new interface name for the IPv6 TE Router ID, or when the same interface begins using a new preferred primary global unicast address, IS-IS immediately floods the new value.

If the referenced system is shut down or the referenced loopback interface is deleted or is shut down, or the last IPv6 address on the interface is removed, IS-IS withdraws the advertisement of the IPv6 TE Router ID TLV.

2.2.17.2 IS-IS Configuration

In order to enable the advertisement of additional link IPv6 and TE parameters, a new **traffic-engineering-options** CLI construct is used.

```
configure
     router
          ipv6-te-router-id interface interface-name
          no ipv6-te-router-id
          [no] isis [instance]
               traffic-engineering
               no traffic-engineering
               traffic-engineering-options
               no traffic-engineering-options
                    ipv6
                    no ipv6
                    application-link-attributes
                    no application-link-attributes
                         legacy
                         no legacy
```

The existing **traffic-engineering** command continues its role as the main command for enabling TE in an IS-IS instance. This command enables the advertisement of the IPv4 and TE link parameters using the legacy TE encoding as per RFC 5305. These parameters are used in IPv4 RSVP-TE and IPv4 SR-TE.

When the **ipv6** command under the **traffic-engineering-options** context is also enabled, then the traffic engineering behavior with IPv6 TE links is enabled. This IS-IS instance automatically advertises the new RFC 6119 IPv6 and TE TLVs and sub-TLVs as described in *TE Attributes Supported in IGP and BGP-LS*.

The **application-link-attributes** context allows the advertisement of the TE attributes of each link on a per-application basis. Two applications are supported in SR OS: RSVP-TE and SR-TE. The legacy mode of advertising TE attributes that is used in RSVP-TE is still supported but can be disabled by using the **no legacy** command that enables the per-application TE attribute advertisement for RSVP-TE as well.

Additional details of the feature behavior and the interaction of the previously mentioned CLI commands are described in *IS-IS IPv4/IPv6 SR-TE and IPv4 RSVP-TE Feature Behavior*.

2.2.17.3 MPLS Configuration

The SR-TE LSP configuration can accept an IPv6 address into the **to** and **from** parameters.

In addition, the MPLS path configuration can accept a hop index with an IPv6 address. The IPv6 address used in the **from** and **to** commands in the IPv6 SR-TE LSP, as well as the address used in the **hop** command of the path used with the IPv6 SR-TE LSP must correspond to the preferred primary global

unicast IPv6 address of a network interface or a loopback interface of the corresponding LER or LSR router. The IPv6 address can also be set to the system interface IPv6 address. Failure to follow the preceding IPv6 address guidelines for the **from**, **to** and **hop**, commands causes path computation to fail with failure code "noCspfRouteToDestination.

Link-local IPv6 address of a network interface is also not allowed in the **hop** command of the path used with the IPv6 SR-TE LSP.

All other MPLS level, LSP level, and primary or secondary path level configuration parameters available for a IPv4 SR-TE LSP are supported unless indicated otherwise.

2.2.17.4 IS-IS, BGP-LS and TE Database Extensions

IS-IS control plane extensions add support for the following RFC 6119 TLVs in IS-IS advertisements and in TE-DB.

- IPv6 interface Address TLV (ISIS_TLV_IPv6_IFACE_ADDR 0xe8)
- IPv6 Neighbor Address sub-TLV (ISIS_SUB_TLV_NBR_IPADDR6 0x0d)
- IPv6 Global Interface Address TLV (only used by ISIS in IIH PDU)
- IPv6 TE Router ID TLV
- IPv6 SRLG TLV

IS-IS also supports advertising which protocol is enabled on a given TE-link (SR-TE, RSVP-TE, or both) by using the Application Specific Link Attributes (ASLA) sub-TLV as per *draft-ietf-isis-te-app*. This causes the advertising router to send potentially different Link TE attributes for RSVP-TE and SR-TE applications and allows the router receiving the link TE attributes to know which application is enabled on the advertising router. For backward compatibility, the router continues to support the legacy mode of advertising link TE attributes, as recommended in RFC 5305, but the user can disable it.

Note: SR OS does not support configuring and advertising different link TE attribute values for RSVP-TE and SR-TE applications. The router advertises the same link TE attributes for both RSVP-TE and SR-TE applications.

See *IS-IS IPv4/IPv6 SR-TE and IPv4 RSVP-TE Feature Behavior* for more details of the behavior of the per-application TE capability.

The new TLVs and sub-TLVs are advertised in IS-IS and added into the local TE-DB when received from IS-IS neighbors. In addition, if the **database-export** command is enabled in this ISIS instance, then this information is also added in the Enhanced TE-DB.

This feature adds the following enhancements to support advertising of the TE parameters in BGP-LS routes over a IPv4 or IPv6 transport:

- Importing IPv6 TE link TLVs from a local Enhanced TE-DB into the local BGP process for exporting to other BGP peers using the BGP-LS route family that is enabled on an IPv4 or an IPv6 transport BGP session

  – RFC 6119 IPv6 and TE TLVs and sub-TLVs are carried in BGP-LS link NLRI as per RFC 7752
  – When the link TE attributes are advertised by IS-IS on a per-application basis using the ASLA TLV (ISIS TLV Type 16), then they are carried in the new BGP-LS ASLA TLV (TLV Type TBD) as per *draft-ietf-idr-bgp-ls-app-specific-attr*.
  – When a TE attribute of a given link is advertised for both RSVP-TE and SR-TE applications, there are three methods IS-IS can use. Each method results in a specific way the BGP-LS originator carries this information. These methods are summarized here but more details are provided in *IS-IS IPv4/IPv6 SR-TE and IPv4 RSVP-TE Feature Behavior*.

    - In legacy mode of operation, all TE attributes are carried in the legacy IS-IS TE TLVs and the corresponding BGP-LS link attributes TLVs as listed in *Table 10: Legacy Link TE TLV Support in TE-DB and BGP-LS*.
    - In legacy with application indication mode of operation, IGP and BGP-LS advertises the legacy TE attribute TLVs and also advertises the ASLA TLV with the legacy (L) flag set and the RSVP-TE and SR-TE application flags set. No TE sub-sub TLVs are advertised within the ASLA TLV.

      The legacy with application indication mode is intended for cases where link TE attributes are common to RSVP-TE and SR-TE applications and have the same value, but the user wants to indicate on a per-link basis which application is enabled.
    - In application specific mode of operation, the TE attribute TLVs are sent as sub-sub-TLVs within the ASLA TLV. Common attributes to RSVP-TE and SR-TE applications have the main TLV Legacy (L) flag cleared and the RSVP-TE and SR-TE application flags set. Any attribute that is specific to an application (RSVP-TE or SR-TE) is advertised in a separate ASLA TLV with the main TLV Legacy (L) flag cleared and the specific application (RSVP-TE or SR-TE) flags set.

      The application specific mode of operation is intended for future cases where TE attributes may have different values in RSVP-TE and SR-TE applications or are specific to one application (for example, the RSVP-TE 'Unreserved Bandwidth' and `Max Reservable Bandwidth' attributes).
- Exporting from the local BGP process to the local Enhanced TE-DB of IPv6 and TE link TLVs received from a BGP peer via BGP-LS route family enabled on a IPv4 or IPv6 transport BGP session
- Support of exporting of IPv6 and TE link TLVs from local Enhanced TE-DB to NSP via the CPROTO channel on the VSR-NRC

2.2.17.4.1 BGP-LS Originator Node Handling of TE Attributes

The specification of the BGP-LS originator node in support of the ASLA TLV is written with the following main objectives in mind:

1. Accommodate IGP node advertising the TE attribute in both legacy or application specific modes of operation.
2. Allow BGP-LS consumers (for example, PCE) that support the ASLA TLV to receive per-application attributes, even if the attribute values are duplicate, and easily store them per-application in the TE-DB. Also, if the BGP-LS consumers receive the legacy attributes, then they can make a determination without ambiguity that these attributes are only for RSVP-TE LSP application.
3. Continue supporting older BGP-LS consumers that rely only on the legacy attributes. This support is taken care by the backward compatibility mode described below but is not supported in SR OS.

The following are the changes needed on the BGP-LS originator node to support objectives (*1*) and (*2*). Excerpts are directly from *draft-ietf-idr-bgp-ls-app-specific-attr*:

1. *Application specific link attributes received from an IGP node using existing RSVP-TE/GMPLS encodings only (i.e. without any ASLA sub-TLV) MUST be encoded using the respective BGP-LS top-level TLVs listed in Table 1 (i.e. not within ASLA TLV). When the IGP node is also SR enabled then another copy of application specific link attributes SHOULD be also encoded as ASLA sub-TLVs with the SR application bit for them. Further rules do not apply for such IGP nodes that do not use ASLA sub-TLVs in their advertisements.*
2. *In case of IS-IS, when application specific link attributes are received from a node with the L bit set in the ASLA sub-TLV then the application specific link attributes are picked up from the legacy ISIS TLVs/sub-TLVs and MUST be encoded within the BGP-LS ASLA TLV as sub-TLVs with the application bitmask set as per the IGP ASLA sub-TLV. When the ASLA sub-TLV with the L bit set also has the RSVP-TE application bit set then the link attributes from such an ASLA sub-TLV MUST be also encoded using the respective BGP-LS top-level TLVs listed in Table 1 (i.e. not within ASLA TLV).*
3. *In case of OSPFv2/v3, when application specific link attributes are received from a node via TE LSAs then the application specific link attributes from those LSAs MUST be encoded using the respective BGP-LS TLVs listed in Table 1 (i.e. not within ASLA TLV).*
4. *Application specific link attributes received from an IGP node within its ASLA sub-TLV MUST be encoded in the BGP-LS ASLA TLV as sub-TLVs with the application bitmask set as per the IGP advertisement.*

The following are the changes needed on the BGP-LS originator node to support of objective (*3*) and which is referred to as the backward compatibility mode. Excerpts are directly from *draft-ietf-idr-bgp-ls-app-specific-attr*:

Note: The backward compatibility mode is not supported in SR OS.

1. *Application specific link attribute received in IGP ASLA sub-TLVs, corresponding to RSVP-TE or SR applications, MUST be also encoded in their existing top level TLVs (as listed in Table 1) outside of the ASLA TLV in addition to them being also advertised within the ASLA TLV*
2. *When the same application specific attribute, received in IGP ASLA sub-TLVs, has different values for RSVP-TE and SR applications then the value for RSVP-TE application SHOULD be preferred over the value for SR application for advertisement as the top level TLV (as listed in Table 1). An implementation MAY provide a knob to reverse this preference.*

2.2.17.4.2 TE Attributes Supported in IGP and BGP-LS

Table 10: Legacy Link TE TLV Support in TE-DB and BGP-LS lists the TE attributes that are advertised using the legacy link TE TLVs defined in RFC 5305 for IS-IS and in RFC 3630 for OSPF. These TE attributes are carried in BGP-LS as recommended in RFC 7752. These legacy TLVs are already supported in SR OS and in IS-IS, OSPF and BGP-LS.

To support IPv6 Traffic Engineering, the IS-IS IPv6 TE attributes (IPv6 TE Router ID and IPv6 SRLG TLV) are advertised in BGP-LS as recommended in RFC 7752.

All the above attributes can now be advertised within the ASLA TLV in IS-IS as recommended in *draft-ietf-isis-te-app* and in BGP-LS as recommended in *draft-ietf-idr-bgp-ls-app-specific-attr*. In the latter case, BGP-LS uses the same TLV type as in RFC 7752 but is included as a sub-TLV of the new BGP-LS ASLA

TLV. *Table 10: Legacy Link TE TLV Support in TE-DB and BGP-LS* lists the code points for IS-IS and BGP-LS TLVs.

Table 10: Legacy Link TE TLV Support in TE-DB and BGP-LS

Link TE TLV Description	IS-IS TLV Type (RFC 5305)	OSPF TLV Type (RFC 3630)	BGP-LS Link NLRI Link-Attribute TLV Type (RFC 7752)
Administrative group (color)	3	9	1088
Maximum link bandwidth	9	6	1089
Maximum reservable link bandwidth	10	7	1090
Unreserved bandwidth	11	8	1091
TE Default Metric	18	5	1092
SRLG	138 (RFC 4205)	16 (RFC 4203)	1096
IPv6 SRLG TLV	139 (RFC 6119)	—	1096
IPv6 TE Router ID	140 (RFC 6119)	—	1029
Application Specific Link Attributes	16 (draft-ietf-isis-te-app)	—	1122 (provisional-as per draft-ietf-idr-bgp-ls-app-specific-attr)
Application Specific SRLG TLV	238 (draft-ietf-isis-te-app)	—	1122 (provisional-as per draft-ietf-idr-bgp-ls-app-specific-attr

Table 11: Additional Link TE TLV Support in TE-DB and BGP-LS lists the TE attributes that are received from a third-party router implementation in legacy TE TLVs, or in the ASLA TLV for the RSVP-TE or SR-TE applications that are added into the local SR OSTE-DB; these are also distributed by the BGP-LS originator. However, these TLVs are not originated by a SR OS router IGP implementation.

Table 11: Additional Link TE TLV Support in TE-DB and BGP-LS

Link TE TLV Description	IS-IS TLV Type (RFC 7810)	OSPF TLV Type (RFC 7471)	BGP-LS Link NLRI Link- Attribute TLV Type (draft-ietf-idr-te-pm-bgp)
Unidirectional Link Delay	33	27	1114
Min/Max Unidirectional Link Delay	34	28	1115
Unidirectional Delay Variation	35	29	1116
Unidirectional Link Loss	36	30	1117
Unidirectional Residual Bandwidth	37	31	1118
Unidirectional Available Bandwidth	38	32	1119
Unidirectional Utilized Bandwidth	39	33	1120

Any other TE attribute received in a legacy TE TLV or in an Application Specific Link Attributes TLV is not added to the local router TE-DB and therefore, not distributed by the BGP-LS originator.

2.2.17.5 IS-IS IPv4/IPv6 SR-TE and IPv4 RSVP-TE Feature Behavior

The TE feature in IS-IS allows the advertising router to indicate to other routers in the TE domain which applications the advertising router has enabled: RSVP-TE, SR-TE, or both. As a result, a receiving router can safely prune links that are not enabled in one of the applications from the topology when computing a CSPF path in that application.

TE behavior consists of the following steps.

1. A valid IPv6 address value must exist for the system or loopback interface assigned to the **ipv6-te-router-id** command. The IPv6 address value can be either the preferred primary global unicast address of the system interface (default value) or that of a loopback interface (user configured).

   The IPv6 TE router ID is mandatory for enabling IPv6 TE and enabling the router to be uniquely identified by other routers in an IGP TE domain as being IPv6 TE capable. If a valid value does not exist, then the IPv6 and TE TLVs described in *IS-IS, BGP-LS and TE Database Extensions* are not advertised.

2. The **traffic-engineering** command enables the existing traffic engineering behavior for IPv4 RSVP-TE and IPv4 SR-TE. Enable the **rsvp** context on the router and enable **rsvp** on the interfaces in order to have IS-IS begin advertising TE attributes in the legacy TLVs. By default, the **rsvp** context is enabled as soon as the **mpls** context is enabled on the interface. If **ipv6** knob is also enabled, then the RFC 6119 IPv6 and TE link TLVs described above are advertised such that a router receiving these advertisements can compute paths for IPv6 SR-TE LSP in addition to paths for IPv4 RSVP-TE LSP and IPv4 SR-TE LSP. The receiving node cannot determine if truly IPv4 RSVP-TE, IPv4 SR-TE, or IPv6 SR-TE applications are enabled on the other routers. Legacy TE routers must assume that RSVP-TE is enabled on those remote TE links it received advertisements for.
3. When the **ipv6** command is enabled, IS-IS automatically begins advertising the RFC 6119 TLVs and sub-TLVs: IPv6 TE router ID TLV, IPv6 interface Address sub-TLV and Neighbor Address sub-TLV, or Link-Local Interface Identifiers sub-TLV if the interface has no global unicast IPv6 address. The TLVs and sub-TLVs are advertised regardless of whether TE attributes are added to the interface in the **mpls** context. The advertisement of these TLVs is only performed when the **ipv6** knob is enabled and **ipv6-routing** is enabled in this IS-IS instance and **ipv6-te-router-id** has a valid IPv6 address.

   A network IP interface is advertised with the Link-Local Interface identifiers sub-TLV if the network IP interface meets the following conditions:

   - network IP interface has link-local IPv6 address and no global unicast IPv6 address on the interface **ipv6** context
   - network IP interface has no IPv4 address and may or may not have the **unnumbered** option enabled on the interface **ipv4** context
4. The **application-link-attributes** command enables the ability to send the link TE attributes on a per-application basis and explicitly conveys that RSVP-TE or SR-TE is enabled on that link on the advertising router.

   Three modes of operation that are allowed by the **application-link-attributes** command.

   a. Legacy Mode: {**no application-link-attributes**}

   The **application-link-attributes** command is disabled by default and the **no** form matches the behavior described in list item 2. It enables the existing traffic engineering behavior for IPv4 RSVP-TE and IPv4 SR-TE. Only the RSVP-TE attributes are advertised in the legacy TE TLVs that are used by both RSVP-TE and SR-TE LSP CSPF in the TE domain routers. No separate SR-TE attributes are advertised.

   If the **ipv6** command is also enabled, then the RFC 6119 IPv6 and TE link TLVs are advertised in the legacy TLVs. A router in the TE domain receiving these advertisements can compute paths for IPv6 SR-TE LSP.

   If the user shuts down the **rsvp** context on the router or on a specific interface, the legacy TE attributes of all the MPLS interfaces or of that specific MPLS interface are not advertised. Routers

can still compute SR-TE LSPs using those links but LSP path TE constraints are not enforced since the links appear in the TE Database as if they did not have TE parameters.

Table 10: Legacy Link TE TLV Support in TE-DB and BGP-LS shows the encoding of the legacy TE TLVs in both IS-IS and BGP-LS.

b. Legacy Mode with Application Indication: {**application-link-attributes** + **legacy**}

The legacy with application indication mode is intended for cases where link TE attributes are common to RSVP-TE and SR-TE applications and have the same value, but the user wants to indicate on a per-link basis which application is enabled.

IS-IS continues to advertise the legacy TE attributes for both RSVP-TE and SR-TE application and includes the new Application Specific Link Attributes TLV with the application flag set to RSVP-TE and/or SR-TE but without the sub-sub-TLVs. IS-IS also advertises the Application Specific SRLG TLV with the application flag set to RSVP-TE and/or SR-TE but without the actual values of the SRLGs.

Routers in the TE domain use these attributes to compute CSPF for IPv4 RSVP-TE LSP and IPv4 SR-TE LSP.

If the **ipv6** command is also enabled, then the RFC 6119 IPv6 and TE TLVs are advertised. A router in the TE domain that receives these advertisements can compute paths for IPv6 SR-TE LSP.

Note: The **segment-routing** command must be enabled in the IS-IS instance or the flag for the SR-TE application will not be set in the Application Specific Link Attributes TLV or in the Application Specific SRLG TLV.

To disable advertising of RSVP-TE attributes, shut down the **rsvp** context on the router. Note, however, doing so reverts to advertising the link SR-TE attributes using the Application Specific Link Attributes TLV and the TE sub-sub-TLVs as shown in *Table 12: Details of Link TE Advertisement Methods*. If legacy attributes were used, legacy routers wrongly interpret that this router enabled RSVP and may signal RSVP-TE LSP paths using its links.

Table 10: Legacy Link TE TLV Support in TE-DB and BGP-LS lists the code points for IS-IS and BGP-LS legacy TLVs.

The following excerpt from the Link State Database (LSDB) shows the advertisement of TE parameters for a link with both RSVP-TE and SR-TE applications enabled.

```
TE IS Nbrs   :
   Nbr   : Dut-A.00
   Default Metric  : 10
   Sub TLV Len     : 124
   IF Addr   : 10.10.2.3
   IPv6 Addr : 3ffe::10:10:2:3
   Nbr IP    : 10.10.2.1
   Nbr IPv6  : 3ffe::10:10:2:1
   MaxLink BW: 100000 kbps
   Resvble BW: 500000 kbps
   Unresvd BW:
       BW[0] : 500000 kbps
       BW[1] : 500000 kbps
       BW[2] : 500000 kbps
       BW[3] : 500000 kbps
       BW[4] : 500000 kbps
       BW[5] : 500000 kbps
       BW[6] : 500000 kbps
```

```
        BW[7] : 500000 kbps
    Admin Grp : 0x1
    TE Metric : 123
    TE APP LINK ATTR    :
         SABML-flags:Legacy SABM-flags:RSVP-TE SR-TE
    Adj-SID: Flags:v4VL Weight:0 Label:524287
    Adj-SID: Flags:v6BVL Weight:0 Label:524284
    TE SRLGs     :
       SRLGs : Dut-A.00
       Lcl Addr  : 10.10.2.3
       Rem Addr  : 10.10.2.1
       Num SRLGs       : 1
             1003
    TE APP SRLGs     :
        Nbr : Dut-A.00
        SABML-flags:Legacy SABM-flags: SR-TE
        IF Addr   : 10.10.2.3
        Nbr IP    : 10.10.2.1
```

c. Application Specific Mode: {**application-link-attributes**} or {**application-link-attributes** + **no legacy**}

The application specific mode of operation is intended for future use cases where TE attributes may have different values in RSVP-TE and SR-TE applications (this capability is not supported in SR OS) or are specific to one application (for example, RSVP-TE 'Unreserved Bandwidth' and `Max Reservable Bandwidth' attributes).

IS-IS advertises the TE attributes that are common to RSVP-TE and SR-TE applications in the sub-sub-TLVs of the new ASLA sub-TLV. IS-IS also advertises the link SRLG values in the Application

Specific SRLG TLV. In both cases, the application flags for RSVP-TE and SR-TE are also set in the sub-TLV.

IS-IS begins to advertise the TE attributes that are specific to the RSVP-TE application separately in the sub-sub-TLVs of the new application attribute sub-TLV. The application flag for RSVP-TE is also set in the sub-TLV.

SR OS does not support configuring and advertising TE attributes that are specific to the SR-TE application.

Common value RSVP-TE and SR-TE TE attributes are combined in the same application attribute sub-TLV with both application flags set, while the non-common value TE attributes are sent in their own application attribute sub-TLV with the corresponding application flag set.

Figure 22: Attribute Mapping per Application shows an excerpt from the Link State Database (LSDB). Attributes in green font are common to both RSVP-TE and SR-TE applications and are combined, while the attribute in red font is specific to RSVP-TE application and is sent separately.

```
TE IS Nbrs   :
     Nbr   : Dut-A.00
     Default Metric  : 100
     Sub TLV Len     : 111
     IF Addr   : 1.0.13.3
     IPv6 Addr : 3ffe::102:606
     Nbr IP    : 1.0.13.1
     Adj-SID: Flags:v4BVL Weight:0 Label:524285
     Adj-SID: Flags:v6BVL Weight:0 Label:524284
     SABML-flags:Non-Legacy SABM-flags:RSVP-TE SR-TE
         MaxLink BW: 99999997 kbps
         Admin Grp : 0x0
         TE Metric : 100
     SABML-flags:Non-Legacy SABM-flags:RSVP-TE
         Resvble BW: 99999997 kbps
         Unresvd BW:
             BW[0] : 99999997 kbps
             BW[1] : 99999997 kbps
             BW[2] : 99999997 kbps
             BW[3] : 99999997 kbps
             BW[4] : 99999997 kbps
             BW[5] : 99999997 kbps
             BW[6] : 99999997 kbps
             BW[7] : 99999997 kbps
TE APP SRLGs    :
    Nbr : Dut-A.00
    SABML-flags:Non-Legacy SABM-flags:RSVP-TE SR-TE
    IF Addr   : 1.0.13.3
    Nbr IP    : 1.0.13.1
    Num SRLGs : 1
    SRLGs     : 1
```

sw0973

Figure 22: Attribute Mapping per Application

Routers in the TE domain use these attributes to compute CSPF for IPv4 SR-TE LSP and IPv4 SR-TE LSPs. If the **ipv6** command is also enabled, then the RFC 6119 IPv6 TLVs are advertised. A router in the TE domain receiving these advertisements can compute paths for IPv6 SR-TE LSP.

Note: The **segment-routing** command must be enabled in the IS-IS instance or the common TE attribute will not be advertised for the SR-TE application.

In order to disable advertising of RSVP-TE attributes, shut down the **rsvp** context on the router.

Table 12: Details of Link TE Advertisement Methods summarizes the IS-IS link TE parameter advertisement details for the three modes of operation of the IS-IS advertisement.

Table 12: Details of Link TE Advertisement Methods

IGP Traffic Engineering Options		Link TE Advertisement Details		
		RSVP-TE (rsvp enabled on interface)	SR-TE (segment-routing enabled in IGP instance)	RSVP-TE and SR-TE (rsvp enabled on interface and segment-routing enabled in IGP instance)
Legacy Mode: **no application-link-attributes**		Legacy TE TLVs	—	Legacy TE TLVs
Legacy Mode with Application Indication: {**application-link-attributes** + **legacy**}	**rsvp** disabled on router (**rsvp** operationally down on all interfaces)	—	Legacy TE TLVs ASLA TLV -Flags: {Legacy=0, SR-TE=1}; TE sub-sub-TLVs	Legacy TE TLVs ASLA TLV -Flags: {Legacy=1, RSVP-TE=0, SR-TE=1}
	rsvp enabled on router	Legacy TE TLVs ASLA TLV -Flags: {Legacy=1, RSVP-TE=1}	Legacy TE TLVs ASLA TLV -Flags: {Legacy=1, SR-TE=1}	Legacy TE TLVs ASLA TLV -Flags: {Legacy=1, RSVP-TE=1, SR-TE=1}

IGP Traffic Engineering Options	Link TE Advertisement Details		
	RSVP-TE (rsvp enabled on interface)	SR-TE (segment-routing enabled in IGP instance)	RSVP-TE and SR-TE (rsvp enabled on interface and segment-routing enabled in IGP instance)
Application Specific Mode: {**application-link-attributes**} or {**application-link-attributes** + **no legacy**}	ASLA TLV - Flags: {Legacy=0, RSVP-TE=1}; TE sub-sub-TLVs	ASLA TLV -Flags: {Legacy=0, SR-TE=1}; TE sub-sub-TLVs	ASLA TLV -Flags: {Legacy=0, RSVP-TE=1; SR-TE=1}; TE sub-sub-TLVs (common attributes) ASLA TLV -Flags: {Legacy=0, RSVP-TE=1}; TE sub-sub-TLVs (RSVP-TE specific attributes; e.g., Unreserved BW and Resvble BW) ASLA TLV -Flags: {Legacy=0, SR-TE=1}; TE sub-sub-TLVs (SR-TE specific attributes; not supported in SR OS 19.10.R1)

2.2.17.6 IPv6 SR-TE LSP Support in MPLS

This feature is supported with the hop-to-label, the local CSPF, and the PCE (PCC-initiated an PCE-initiated) path computation methods.

All capabilities of a IPv4 provisioned SR-TE LSP are supported with a IPv6 SR-TE LSP unless indicated otherwise. There are, however, some important differences with an IPv4 SR-TE LSP which are explained below.

The IPv6 address used in the **from** and **to** commands in the IPv6 SR-TE LSP, as well as the address used in the **hop** command of the path used with the IPv6 SR-TE LSP must correspond to the preferred primary global unicast IPv6 address of a network interface or a loopback interface of the corresponding LER or LSR router. The IPv6 address can also be set to the system interface IPv6 address. Failure to follow the preceding IPv6 address guidelines for the **from**, **to** and **hop**, commands causes path computation to fail

with failure code "noCspfRouteToDestination. A Link-Local IPv6 address of a network interface is also not allowed in the **hop** command of the path used with the IPv6 SR-TE LSP. The configuration fails.

A TE link with no global unicast IPv6 address and only a link local IPv6 address can however be used in the path computation by the local CSPF. The address shown in the 'Computed Hops' and in the 'Actual Hops' fields of the output of the path **show** command uses the neighbor's IPv6 TE router ID and the Link-Local Interface Identifier. The exceptions are if the interface is of type broadcast or is of type point-to-point but also has a local IPv4 address. Only the neighbor's IPv6 TE router ID is shown as the Link-Local Interface Identifiers sub-TLV is not advertised in these situations.

The global MPLS IPv4 state UP value requires that the system interface be in the admin UP state and to have a valid IPv4 address.

The global MPLS IPv6 state UP value requires that the interface used for the IPv6 TE router ID be in admin UP state and to have a valid preferred primary IPv6 global unicast address.

The TE interface MPLS IPv4 state UP value requires the interface be in the admin UP state in the **router** context and the global MPLS IPv4 state be in UP state.

The TE interface MPLS IPv6 state UP value requires the interface be in the admin UP state in the **router** context and the global MPLS IPv6 state be in UP state.

2.2.17.6.1 IPv6 SR-TE auto-LSP

This feature provides for the auto-creation of an IPv6 SR-TE mesh LSP and for a IPv6 SR-TE one-hop LSP.

The SR-TE mesh LSP feature specifically binds an LSP template of type **mesh-p2p-srte** with one or more IPv6 prefix lists. When the Traffic Engineering database discovers a router, which has an IPv6 TE router ID matching an entry in the prefix list, it triggers MPLS to instantiate an SR-TE LSP to that router using the LSP parameters in the LSP template.

The SR-TE one-hop LSP feature specifically activates a LSP template of type **one-hop-p2p-srte**. In this case, the TE database keeps track of each TE link which comes up to a directly connected IGP TE neighbor. It then instructs MPLS to instantiate a SR-TE LSP with the following parameters:

- the source IPv6 address of the local router
- an outgoing interface matching the interface index of the TE-link
- a destination address matching the IPv6 TE router-id of the neighbor on the TE link

A new **family** CLI leaf is added to the LSP template configuration and must be set to the **ipv6** value. By default, this command is set to the **ipv4** value for backward compatibility. When establishing both IPv4 and IPv6 SR-TE mesh auto-LSPs with the same parameters and constraints, a separate LSP template of type **mesh-p2p-srte** must be configured for each address family with the **family** CLI leaf set to the IPv4 or IPv6 value. SR-TE one-hop auto-LSPs can only be established for either IPv4 or IPv6 family, but not both. The **family** leaf in the LSP template of type **one-hop-p2p-srte** should be set to the desired IP family value.

Note: A IPv6 SR-TE auto-LSP can be reported to a PCE but cannot be delegated or have its paths computed by PCE.

All capabilities of an IPv4 SR-TE auto-LSP are supported with a IPv6 SR-TE auto-LSP unless indicated otherwise.

2.2.18 OSPF Link TE Attribute Reuse

This section describes the support of OSPF application specific TE link attributes.

2.2.18.1 OSPF Application Specific TE Link Attributes

Existing OSPFv2 TE-related link attribute advertisement (for example, bandwidth) definitions are used in RSVP-TE deployments (refer to *draft-ietf-spring-segment-routing-policy-07.txt* for more information). Since the definition of the original RSVP-TE use case, additional applications (for example, Segment Routing Traffic Engineering (SR-TE)) that may use the link attribute advertisement have also been defined.

This usage has introduced ambiguity in deployments that include a mix of RSVP-TE and SR-TE support. For example, it is not possible to unambiguously indicate the specific advertisements used by RSVP-TE and SR-TE. Although this may not be an issue for fully congruent topologies, any incongruence causes ambiguity. An additional issue arises in cases where both applications are supported on a link but the link attribute values associated with each application differ. Advertisements without OSPFv2 application specific TE link attributes do not support the advertisement of application specific values for the same attribute on a specific link.

CLI syntax:

```
Config
   router
      ospf
         traffic-engineering-options
               sr-te {legacy|application-specific-link-attributes}
               no sr-te
```

The **traffic-engineering-options** command enables the context to configure advertisement of the TE attributes of each link on a per-application basis. Two applications are supported in SR OS: RSVP-TE and SR-TE.

The **legacy** mode of advertising TE attributes that is used in RSVP-TE is still supported. In addition, the following configuration options are allowed:

- no sr-te

  This option advertises the TE information for RSVP links using TE Opaque LSAs. The **no** form is the default value.
- sr-te legacy

  This option advertises the TE information for MPLS-enabled SR links using TE Opaque LSAs.

  Note: The operator should not use the **sr-te legacy** option if the network has both RSVP-TE and SR-TE, and the links are not congruent.

- sr-te application-specific-link-attributes

  This option advertises the TE information for MPLS-enabled SR links using the new Application Specific Link Attributes (ASLA) TLVs.

The IETF Draft *draft-ietf-ospf-te-link-attr-reuse-14.txt* defines a subset of all possible TE extensions and TE Metric Extensions that can be encoded within Application Specific Link sub TLVs. *Table 13: Nokia Support for ASLA Extended Link TLV Encoding* describes the relevant values for SR OS.

Table 13: Nokia Support for ASLA Extended Link TLV Encoding

OSPFv2 Extended Link TLV Sub-TLVs (RFC7684)				
IANA	Attribute Type	TE-DB [1]	SR OS sub-TLV of Extended Link TLV [2]	SR OS Nested sub-TLV of ASLA Extended Link TLV encoding[3]
10	ASLA	✓	✓	—
11	Shared Risk Link Group	✓	—	✓
12	Unidirectional Link Delay	✓	—	—
13	Min/Max Unidirectional Link Delay	✓	—	—
14	Unidirectional Delay Variation	✓	—	—
15	Unidirectional Link Loss	✓	—	—
16	Unidirectional Residual Bandwidth	✓	—	—
17	Unidirectional Available Bandwidth	✓	—	—
18	Unidirectional Utilized Bandwidth	✓	—	—
19	Administrative Group	✓	—	Y
20	Extended Administrative Group	✓	—	—

OSPFv2 Extended Link TLV Sub-TLVs (RFC7684)				
22	TE Metric	✓	—	✓
23	Maximum Link Bandwidth	✓	✓	—

Notes:

1. Support to include the attributes from received LSA's into Nokia TE-DB and export into BGP-LS. Refer to *draft-ietf-idr-bgp-ls-app-specific-attr-02*.txt for more information.
2. Node support to encode the link attribute as sub-TLV in an OSPFv2 Extended Link TLV.
3. Node support to encode the link attribute as sub-TLV in an OSPFv2 Application Specific Extended Link sub-TLV.

The solution proposed in the OSPF Link Traffic Engineering Attribute Reuse Draft (*draft-ietf-ospf-te-link-attr-reuse-14.txt*) assumes that OSPF does not need to move all RSVP-TE attributes from the TE Opaque LSA into the Extended Link LSA. For, RSVP-TE, consequently, there is no significant modification and it can continue to be advertised using existing OSPF TLVs. For SR-TE and future applications, the ASLA TLVs may be used. Alternatively, existing TE Opaque LSAs could be used through configuration. *Table 14: Configuration Considerations for TE Opaque LSAs* describes the possible configurations for TE Opaque LSAs.

Table 14: Configuration Considerations for TE Opaque LSAs

Interior Gateway Protocol Configuration	ospf>traffic-engineering <20.7	ospf>traffic-engineering ospf>te-opts>no sr-te	ospf>traffic-engineering ospf>te-opts>sr-te legacy	ospf>traffic-engineering ospf>te-opts>sr-te application-link-attribute
Interface config	—	—	—	—
MPLS + RSVP	TE-Opaque	TE-Opaque	TE-Opaque	TE-Opaque
MPLS + SR	—	—	TE-Opaque [1]	ASLA (SR-TE)
MPLS + RSVP + SR	TE-Opaque	TE-Opaque	TE-Opaque	TE-Opaque (RSVP) + ASLA (SRTE)

Note:

1. If the local router interface is configured with MPLS+SR, and RSVP-TE is deployed on remote servers, the remote routers will wrongly conclude that the link is RSVP-enabled.

2.2.19 Configuring and Operating SR-TE

This section provides information on the configuration and operation of the Segment Routing with Traffic Engineering (SR-TE) LSP.

2.2.19.1 SR-TE Configuration Prerequisites

To configure SR-TE, the user must first configure prerequisite parameters.

First, configure the label space partition for the Segment Routing Global Block (SRGB) for all participating routers in the segment routing domain by using the **mpls-labels**>**sr-labels** command.

```
mpls-labels
        – sr-labels start 200000 end 200400
    – exit
```

Enable segment routing, traffic engineering, and advertisement of router capability in all participating IGP instances in all participating routers by using the **traffic-engineering**, **advertise-router-capability**, and **segment-routing** commands.

```
ospf 0
        – traffic-engineering
        – advertise-router-capability area
        – loopfree-alternates remote-lfa
        – area 0.0.0.202
            – stub
                – no summaries
            – exit
            – interface "system"
                – node-sid index 194
                – no shutdown
            – exit
            – interface "toSim199"
                – interface-type point-to-point
                – no shutdown
            – exit
            – interface "toSim213"
                – interface-type point-to-point
                – no shutdown
            – exit
            – interface "toSim219"
                – interface-type point-to-point
                – metric 2000
                – no shutdown
            – exit
        – exit
        – segment-routing
            – prefix-sid-range global
            – no shutdown
        – exit
        – no shutdown
    – exit
```

Configure an segment routing tunnel MTU for the IGP instance, if required, by using the **tunnel-mtu** command.

```
prefix-sid-range global
    – tunnel-mtu 1500
```

```
    – no shutdown
```

Assign a node SID to each loopback interface that a router would use as the destination of a segment routing tunnel by using the **node-sid** command.

```
ospf 0
        – area 0.0.0.202
            – interface "system"
                – node-sid index 194
                – no shutdown
            – exit
```

2.2.19.2 SR-TE LSP Configuration Overview

An SR-TE LSP can be configured as a label switched path (LSP) using the existing CLI command hierarchy under the MPLS context and specifying the new **sr-te** LSP type.

```
config>router>mpls>lsp lsp-name | mpls-tp src-tunnel-num | sr-te
```

As for an RSVP LSP, the user can configure a primary path.

Use the following CLI syntax to associate an empty path or a path with strict or loose explicit hops with the primary paths of the SR-TE LSP:

```
config>router>mpls>path>hop hop-index ip-address {strict | loose}
    – config>router>mpls>lsp>primary path-name
```

2.2.19.3 Configuring Path Computation and Control for SR-TE LSP

Use the following syntax to configure the path computation requests only (PCE-computed) or both path computation requests and path updates (PCE-controlled) to PCE for a specific LSP:

```
config>router>mpls>lsp>path-computation-method pce
    – config>router>mpls>lsp>pce-control
```

The PCC LSP database is synchronized with the PCE LSP database using the PCEP PCRpt (PCE Report) message for LSPs that have the following commands enabled:

```
config>router>mpls>pce-report sr-te {enable | disable}
    – config>router>mpls>lsp>pce-report {enable | disable | inherit}
```

2.2.19.3.1 Configuring Path Profile and Group for PCC-Initiated and PCE-Computed/Controlled LSP

The PCE supports the computation of disjoint paths for two different LSPs originating or terminating on the same or different PE routers. To indicate this constraint to PCE, the user must configure the PCE path profile ID and path group ID the LSP belongs to. These parameters are passed transparently by PCC to

PCE and are thus opaque data to the router. Use the following syntax to configure the path profile and path group:

```
config>router>mpls>lsp>path-profile profile-id [path-group group-id]
```

The association of the optional path group ID is to allow PCE determine which profile ID this path group ID must be used with. One path group ID is allowed per profile ID. The user can, however, enter the same path group ID with multiple profile IDs by executing this command multiple times. A maximum of five entries of **path-profile** [*path-group*] can be associated with the same LSP. More details of the operation of the PCE path profile are provided in the PCEP section of this guide.

2.2.19.4 Configuring SR-TE LSP Label Stack Size

Use the following syntax to configure the maximum number of labels which the ingress LER can push for a given SR-TE LSP:

```
config>router>mpls>lsp>max-sr-labels label-stack-size
```

This command allows the user to reduce the SR-TE LSP label stack size by accounting for additional transport, service, and other labels when packets are forwarded in a given context. See *Data Path Support* for more information about label stack size requirements in various forwarding contexts. If the CSPF on the PCE or the router's hop-to-label translation could not find a path that meets the maximum SR label stack, the SR-TE LSP will remain on its current path or will remain down if it has no path. The range is 1-10 labels with a default value of 6.

2.2.19.5 Configuring Adjacency SID Parameters

Configure the adjacency hold timer for the LFA or remote LFA backup next-hop of an adjacency SID.

Use the following syntax to configure the length of the interval during which LTN or ILM records of an adjacency SID are kept:

```
config>router>ospf>segment-routing>adj-sid-hold seconds[1..300, default 15]
    – config>router>isis>segment-routing>adj-sid-hold seconds[1..300, default 15]
```

```
adj-sid-hold 15
    – no entropy-label-capability
    – prefix-sid-range global
    – no tunnel-table-pref
    – no tunnel-mtu
    – no backup-node-sid
    – no shutdown
```

While protection is enabled globally for all node SIDs and local adjacency SIDs when the user enables the **loopfree-alternates** option in ISIS or OSPF at the LER and LSR, there are applications where the user wants traffic to never divert from the strict hop computed by CSPF for a SR-TE LSP. In that case, use the following syntax to disable protection for all adjacency SIDs formed over a given network IP interface:

```
config>router>ospf>area>if>no sid-protection
```

```
    – config>router>isis>if>no sid-protection
```

```
node-sid index 194
    – no sid-protection
    – no shutdown
```

2.2.19.6 Configuring PCC-controlled, PCE-computed, and PCE-controlled SR-TE LSPs

The following example shows the configuration of PCEP PCC parameters on LER routers that require peering with the PCE server:

```
keepalive 30
    – dead-timer 120
    – no local-address
    – unknown-message-rate 10
    – report-path-constraints
    – peer 192.168.48.226
        – no shutdown
    – exit
    – no shutdown
```

The following example shows the configuration of a PCC-controlled SR-TE LSP that is not reported to PCE:

```
lsp "to-SanFrancisco" sr-te
        – to 192.168.48.211
        – path-computation-method local-cspf
        – pce-report disable
        – metric 10
        – primary "loose-anycast"
        – exit
        – no shutdown
    – exit
```

The following example shows the configuration of a PCC-controlled SR-TE LSP that is reported to PCE:

```
lsp "to-SanFrancisco" sr-te
        – to 192.168.48.211
        – path-computation-method local-cspf
        – pce-report enable
        – metric 10
        – primary "loose-anycast"
        – exit
        – no shutdown
    – exit
```

The following example shows the configuration of a PCE-computed SR-TE LSP that is reported to PCE:

```
lsp "to-SanFrancisco" sr-te
        – to 192.168.48.211
        – path-computation-method local-cspf
        – pce-report enable
        – metric 10
        – primary "loose-anycast"
        – exit
        – no shutdown
```

```
    – exit
```

The following example shows the configuration of a PCE-controlled SR-TE LSP with no PCE path profile:

```
lsp "from Reno to Atlanta no Profile" sr-te
        – to 192.168.48.224
        – path-computation-method local-cspf
        – pce-report enable
        – pce-control
        – primary "empty"
        – exit
        – no shutdown
    – exit
```

The following example shows the configuration of a PCE-controlled SR-TE LSP with a PCE path profile and a maximum label stack set to a non-default value:

```
lsp "from Reno to Atlanta no Profile" sr-te
        – to 192.168.48.224
        – max-sr-labels 8 additional-frr-labels 1
        – path-computation-method pce
        – pce-report enable
        – pce-control
        – path-profile 10 path-group 2
        – primary "empty"
            – bandwidth 15
        – exit
        – no shutdown
    – exit
```

2.2.19.7 Configuring a Mesh of SR-TE Auto-LSPs

The following shows the detailed configuration for the creation of a mesh of SR-TE auto-LSPs. The network uses IS-IS with the backbone area being in Level 2 and the leaf areas being in Level 1.

The NSP is used for network discovery only and the NRC-P learns the network topology using BGP-LS.

Figure 23: Multi-level IS-IS Topology in the NSP GUI shows the view of the multi-level IS-IS topology in the NSP GUI. The backbone L2 area is highlighted in green.

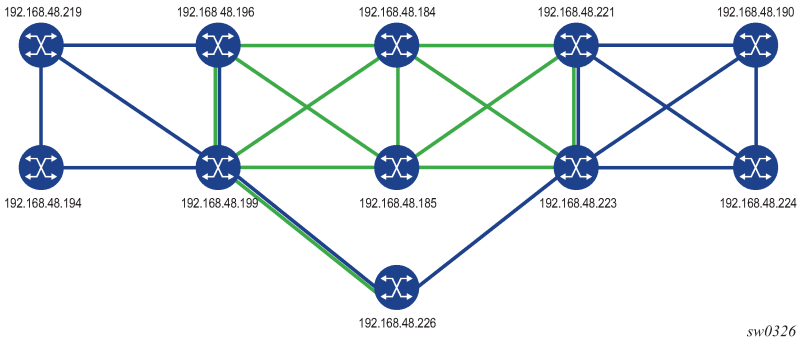

Figure 23: Multi-level IS-IS Topology in the NSP GUI

The mesh of SR-TE auto-LSPs is created in the backbone area and originates on an ABR node with address 192.168.48.199 (Phoenix 199). The LSP template uses a default path that includes an anycast

SID prefix corresponding to a transit routers 192.168.48.184 (Dallas 184) and 192.168.48.185 (Houston 185).

The following is the configuration of transit router Dallas 184, which shows the creation of a loopback interface with the anycast prefix and the assignment of a SID to it. The same configuration must be performed on the transit router Houston 185. See lines marked with an asterisk (*).

```
*A:Dallas 184>config>router# info
----------------------------------------------
echo "IP Configuration"
#----------------------------------------------
        if-attribute
            admin-group "olive" value 20
            admin-group "top" value 10
            srlg-group "top" value 10
        exit
        interface "anycast-sid"                                                 *
            address 192.168.48.99/32                                            *
            loopback                                                            *
            no shutdown                                                         *
        exit
        interface "system"
            address 192.168.48.184/32
            no shutdown
        exit
        interface "toJun164"
            address 10.19.2.184/24
            port 1/1/4:10
            no shutdown
        exit
        interface "toSim185"
            address 10.0.3.184/24
            port 1/1/2
            no shutdown
        exit
        interface "toSim198"
            address 10.0.2.184/24
            port 1/1/3
            if-attribute
                admin-group "olive"
            exit
            no shutdown
        exit
        interface "toSim199"
            address 10.0.13.184/24
            port 1/1/5
            no shutdown
        exit
        interface "toSim221"
            address 10.0.4.184/24
            port 1/1/1
            no shutdown
        exit
        interface "toSim223"
            address 10.0.14.184/24
            port 1/1/6
            no shutdown
        exit
#----------------------------------------------

*A:Dallas 184>config>router>isis# info
-----------------------------------------
            level-capability level-2
            area-id 49.0000
```

```
            database-export identifier 10 bgp-ls-identifier 10
            traffic-engineering
            advertise-router-capability area
            level 2
                wide-metrics-only
            exit
            interface "system"
                ipv4-node-sid index 384
                no shutdown
            exit
            interface "toSim198"
                interface-type point-to-point
                no shutdown
            exit
            interface "toSim185"
                interface-type point-to-point
                no shutdown
            exit
            interface "toSim221"
                interface-type point-to-point
                no shutdown
            exit
            interface "toSim199"
                interface-type point-to-point
                level 2
                    metric 100
                exit
                no shutdown
            exit
            interface "toSim223"
                interface-type point-to-point
                level 2
                    metric 100
                exit
                no shutdown
            exit
            interface "anycast-sid"                                              *
                ipv4-node-sid index 99                                           *
                no shutdown                                                      *
            exit
            segment-routing
                prefix-sid-range global
                no shutdown
            exit
            no shutdown
-----------------------------------------------
```

In the ingress LER Phoenix 199 router, the anycast SID is learned from both transit routers, but is currently resolved in IS-IS to transit router Houston 185. See lines marked with an asterisk (*).

```
*A:Phoenix 199# show router isis prefix-sids
===============================================================================
Rtr Base ISIS Instance 0 Prefix/SID Table
===============================================================================
Prefix                          SID        Lvl/Typ    SRMS   AdvRtr
                                                       MT     Flags
-------------------------------------------------------------------------------
192.168.48.194/32                399        1/Int.      N     Reno 194
                                                         0       NnP
192.168.48.194/32                399        2/Int.      N     Salt Lake 198
                                                         0       RNnP
192.168.48.194/32                399        2/Int.      N     Phoenix 199
                                                         0       RNnP
192.168.48.99/32                 99         2/Int.      N     Dallas 184         *
```

```
                                                    0       NnP            *
192.168.48.99/32                99         2/Int.      N    Houston 185    *
                                                    0       NnP            *
192.168.48.184/32               384        2/Int.      N    Dallas 184
                                                    0       NnP
192.168.48.185/32               385        2/Int.      N    Houston 185
                                                    0       NnP
192.168.48.190/32               390        2/Int.      N    Chicago 221
                                                    0       RNnP
192.168.48.190/32               390        2/Int.      N    St Louis 223
                                                    0       RNnP
192.168.48.194/32               394        1/Int.      N    Reno 194
                                                    0       NnP
192.168.48.194/32               394        2/Int.      N    Salt Lake 198
                                                    0       RNnP
192.168.48.194/32               394        2/Int.      N    Phoenix 199
                                                    0       RNnP
192.168.48.198/32               398        1/Int.      N    Salt Lake 198
                                                    0       NnP
192.168.48.198/32               398        2/Int.      N    Salt Lake 198
                                                    0       NnP
192.168.48.198/32               398        2/Int.      N    Phoenix 199
                                                    0       RNnP
192.168.48.199/32               399        2/Int.      N    Salt Lake 198
                                                    0       RNnP
192.168.48.199/32               399        1/Int.      N    Phoenix 199
                                                    0       NnP
192.168.48.199/32               399        2/Int.      N    Phoenix 199
                                                    0       NnP
192.168.48.219/32               319        2/Int.      N    Salt Lake 198
                                                    0       RNnP
192.168.48.219/32               319        2/Int.      N    Phoenix 199
                                                    0       RNnP
192.168.48.219/32               319        1/Int.      N    Las Vegas 219
                                                    0       NnP
192.168.48.221/32               321        2/Int.      N    Chicago 221
                                                    0       NnP
192.168.48.221/32               321        2/Int.      N    St Louis 223
                                                    0       RNnP
192.168.48.223/32               323        2/Int.      N    Chicago 221
                                                    0       RNnP
192.168.48.223/32               323        2/Int.      N    St Louis 223
                                                    0       NnP
192.168.48.224/32               324        2/Int.      N    Chicago 221
                                                    0       RNnP
192.168.48.224/32               324        2/Int.      N    St Louis 223
                                                    0       RNnP
192.168.48.226/32               326        2/Int.      N    PCE Server 226
                                                    0       NnP
3ffe::a14:194/128               294        1/Int.      N    Reno 194
                                                    0       NnP
3ffe::a14:194/128               294        2/Int.      N    Phoenix 199
                                                    0       RNnP
3ffe::a14:199/128               299        1/Int.      N    Phoenix 199
                                                    0       NnP
3ffe::a14:199/128               299        2/Int.      N    Phoenix 199
                                                    0       NnP
-------------------------------------------------------------------------------
No. of Prefix/SIDs: 32 (15 unique)
-------------------------------------------------------------------------------
SRMS : Y/N  = prefix SID advertised by SR Mapping Server (Y) or not (N)
       S    = SRMS prefix SID is selected to be programmed
Flags: R    = Re-advertisement
       N    = Node-SID
       nP   = no penultimate hop POP
       E    = Explicit-Null
```

```
       V    = Prefix-SID carries a value
       L    = value/index has local significance
===============================================================================

*A:Phoenix 199# tools dump router segment-routing tunnel
===============================================================================
Legend: (B) - Backup Next-hop for Fast Re-Route
        (D) - Duplicate
===============================================================================
-------------------------------------------------------------------------------+
 Prefix                                                                        |
 Sid-Type        Fwd-Type       In-Label  Prot-Inst                            |
                 Next Hop(s)                      Out-Label(s) Interface/Tunnel-ID |
-------------------------------------------------------------------------------+
 192.168.48.99                                                                 *
 Node            Orig/Transit   200099    ISIS-0                               *
                 10.0.5.185                       200099      toSim185         *
 3ffe::a14:194
 Node            Orig/Transit   200294    ISIS-0
                 fe80::62c2:ffff:fe00:0           200294      toSim194
 3ffe::a14:199
 Node            Terminating    200299    ISIS-0
 192.168.48.219
 Node            Orig/Transit   200319    ISIS-0
                 10.202.5.194                     200319      toSim194
 192.168.48.221
 Node            Orig/Transit   200321    ISIS-0
                 10.0.5.185                       200321      toSim185
 192.168.48.223
 Node            Orig/Transit   200323    ISIS-0
                 10.0.5.185                       200323      toSim185
 192.168.48.224
 Node            Orig/Transit   200324    ISIS-0
                 10.0.5.185                       200324      toSim185
 192.168.48.226
 Node            Orig/Transit   200326    ISIS-0
                 10.0.1.2                         100326      toSim226PCEServer
 192.168.48.184
 Node            Orig/Transit   200384    ISIS-0
                 10.0.5.185                       200384      toSim185
 192.168.48.185
 Node            Orig/Transit   200385    ISIS-0
                 10.0.5.185                       200385      toSim185
 192.168.48.190
 Node            Orig/Transit   200390    ISIS-0
                 10.0.5.185                       200390      toSim185
 192.168.48.194
 Node            Orig/Transit   200394    ISIS-0
                 10.202.5.194                     200394      toSim194
 192.168.48.198
 Node            Orig/Transit   200398    ISIS-0
                 10.0.9.198                       100398      toSim198
 192.168.48.199
 Node            Terminating    200399    ISIS-0
 10.0.9.198
 Adjacency       Transit        262122    ISIS-0
                 10.0.9.198                       3           toSim198
 10.202.1.219
 Adjacency       Transit        262124    ISIS-0
                 10.202.1.219                     3           toSim219
 10.0.5.185
 Adjacency       Transit        262133    ISIS-0
                 10.0.5.185                       3           toSim185
 fe80::62c2:ffff:fe00:0
 Adjacency       Transit        262134    ISIS-0
                 fe80::62c2:ffff:fe00:0           3           toSim194
```

```
 10.0.1.2
 Adjacency       Transit        262137    ISIS-0
                 10.0.1.2                         3           toSim226PCEServer
 10.0.13.184
 Adjacency       Transit        262138    ISIS-0
                 10.0.13.184                      3           toSim184
 10.0.2.2
 Adjacency       Transit        262139    ISIS-0
                 10.0.2.2                         3           toSim226PCEserver202
 10.202.5.194
 Adjacency       Transit        262141    ISIS-0
                 10.202.5.194                     3           toSim194
-------------------------------------------------------------------------------
No. of Entries: 22
-------------------------------------------------------------------------------
```

Next, a policy must be configured to add the list of prefixes to which the ingress LER Phoenix 199 must auto-create SR-TE LSPs.

```
*A:Phoenix 199>config>router>policy-options# info
----------------------------------------------
            prefix-list "sr-te-level2"
                prefix 192.168.48.198/32 exact
                prefix 192.168.48.221/32 exact
                prefix 192.168.48.223/32 exact
            exit
            policy-statement "sr-te-auto-lsp"
                entry 10
                    from
                        prefix-list "sr-te-level2"
                    exit
                    action accept
                    exit
                exit
                default-action drop
                exit
            exit
----------------------------------------------
```

Then, an LSP template of type **mesh-p2p-srte** must be configured, which uses a path with a loose-hop corresponding to anycast-SID prefix of the transit routers. The LSP template is then bound to the policy containing the prefix list. See lines marked with an asterisk (*).

```
*A:Phoenix 199>config>router>mpls# info
----------------------------------------------
            cspf-on-loose-hop
            interface "system"
                no shutdown
            exit
            interface "toESS195"
                no shutdown
            exit
            interface "toSim184"
                no shutdown
            exit
            interface "toSim185"
                admin-group "bottom"
                srlg-group "bottom"
                no shutdown
            exit
            interface "toSim194"
                admin-group "bottom"
```

```
                srlg-group "bottom"
                no shutdown
            exit
            interface "toSim198"
                no shutdown
            exit
            interface "toSim219"
                no shutdown
            exit
            path "loose-anycast-sid"                                          *
                hop 1 192.168.48.99 loose                                     *
                no shutdown                                                   *
            exit                                                              *
            lsp-template "sr-te-level2-mesh" mesh-p2p-srte                    *
                default-path "loose-anycast-sid"                              *
                max-sr-labels 8 additional-frr-labels 2                       *
                pce-report enable                                             *
                no shutdown                                                   *
            exit                                                              *
            auto-lsp lsp-template "sr-te-level2-mesh" policy "sr-te-auto-lsp" *
            no shutdown                                                       *
-------------------------------------------
```

One SR-TE LSP should be automatically created to each destination matching the prefix in the policy as soon as the router with the router ID matching the address of the prefix appears in the TE database.

The following shows the three SR-TE auto-LSPs created. See lines marked with an asterisk (*).

```
*A:Phoenix 199# show router mpls sr-te-lsp
===============================================================================
MPLS SR-TE LSPs (Originating)
===============================================================================
LSP Name                            To               Tun     Protect   Adm  Opr
                                                     Id      Path
-------------------------------------------------------------------------------
Phoenix-SL-1                        192.168.48.223    1       N/A       Up   Up
Phoenix-SL-2-Profile                192.168.48.223    2       N/A       Up   Up
Phoenix-SL-3-Profile                192.168.48.223    3       N/A       Up   Up
Phoenix-SL-4-Profile                192.168.48.223    4       N/A       Up   Up
Phoenix-SL-1-Profile                192.168.48.223    5       N/A       Up   Up
Phoenix-SL-2                        192.168.48.223    6       N/A       Up   Up
Phoenix-SL-3                        192.168.48.223    7       N/A       Up   Up
Phoenix-SL-4                        192.168.48.223    8       N/A       Up   Up
sr-te-level2-mesh-192.168.48.198-   192.168.48.198    61442   N/A       Up   Up     *
716803                                                                            *
sr-te-level2-mesh-192.168.48.221-   192.168.48.221    61443   N/A       Up   Up     *
716804                                                                            *
sr-te-level2-mesh-192.168.48.223-   192.168.48.223    61444   N/A       Up   Up     *
716805                                                                            *
-------------------------------------------------------------------------------
LSPs : 17
===============================================================================
```

The auto-generated name uses the syntax convention "*TemplateName-DestIpv4Address-TunnelId*", as explained in *Automatic Creation of an SR-TE Mesh LSP*. The tunnel ID used in the name is the TTM tunnel ID, not the MPLS LSP tunnel ID. See lines marked with an asterisk (*).

```
*A:Phoenix 199# show router mpls sr-te-lsp "sr-te-level2-mesh-192.168.48.223-
716805" detail
===============================================================================
MPLS SR-TE LSPs (Originating) (Detail)
===============================================================================
```

```
-------------------------------------------------------------------------------
Type : Originating
-------------------------------------------------------------------------------
LSP Name        : sr-te-level2-mesh-192.168.48.223-716805
LSP Type        : MeshP2PSrTe               LSP Tunnel ID        : 61444           *
LSP Index       : 126979                    TTM Tunnel Id        : 716805          *
From            : 192.168.48.199            To                   : 192.168.48.2*
Adm State       : Up                        Oper State           : Up
LSP Up Time     : 0d 00:02:12               LSP Down Time        : 0d 00:00:00
Transitions     : 3                         Path Changes         : 3
Retry Limit     : 0                         Retry Timer          : 30 sec
CSPF            : Enabled
Metric          : N/A                       Use TE metric        : Disabled
Include Grps    :                           Exclude Grps         :
None                                           None
VprnAutoBind    : Enabled
IGP Shortcut    : Enabled                   BGP Shortcut         : Enabled
IGP LFA         : Disabled                  IGP Rel Metric       : Disabled
BGPTransTun     : Enabled
Oper Metric     : 16777215
PCE Report      : Enabled
PCE Compute     : Disabled                  PCE Control          : Disabled
Max SR Labels   : 8                         Additional FRR Labels: 2
Path Profile    :
None
Primary(a)      : loose-anycast-sid         Up Time              : 0d 00:02:12
Bandwidth       : 0 Mbps
===============================================================================
```

These SR-TE auto-LSPs are also added into the tunnel table to be used by services and shortcut applications. See lines marked with an asterisk (*).

```
*A:Phoenix 199# show router tunnel-table
===============================================================================
IPv4 Tunnel Table (Router: Base)
===============================================================================
Destination       Owner     Encap TunnelId  Pref     Nexthop        Metric
-------------------------------------------------------------------------------
10.0.5.185/32     isis (0)  MPLS  524370    11       10.0.5.185     0
10.0.9.198/32     isis (0)  MPLS  524368    11       10.0.9.198     0
10.0.13.184/32    isis (0)  MPLS  524340    11       10.0.13.184    0
10.202.1.219/32   isis (0)  MPLS  524333    11       10.202.1.219   0
10.202.5.194/32   isis (0)  MPLS  524355    11       10.202.5.194   0
10.0.1.2/32       isis (0)  MPLS  524364    11       11.0.1.2       0
10.0.2.2/32       isis (0)  MPLS  524363    11       11.0.2.2       0
192.168.48.99/32  isis (0)  MPLS  524294    11       10.0.5.185     10
192.168.48.184/32 ldp       MPLS  65605     9        10.0.5.185     20
192.168.48.184/32 isis (0)  MPLS  524341    11       10.0.5.185     20
192.168.48.185/32 ldp       MPLS  65602     9        10.0.5.185     10
192.168.48.185/32 isis (0)  MPLS  524371    11       10.0.5.185     10
192.168.48.190/32 ldp       MPLS  65606     9        10.0.5.185     40
192.168.48.190/32 isis (0)  MPLS  524362    11       10.0.5.185     40
192.168.48.194/32 ldp       MPLS  65577     9        10.202.5.194   10
192.168.48.194/32 isis (0)  MPLS  524331    11       10.202.5.194   10
192.168.48.198/32 sr-te     MPLS  716803    8        192.168.48.99  16777215      *
192.168.48.198/32 ldp       MPLS  65601     9        10.0.9.198     10
192.168.48.198/32 isis (0)  MPLS  524369    11       10.0.9.198     10
192.168.48.219/32 ldp       MPLS  65579     9        10.202.5.194   20
192.168.48.219/32 isis (0)  MPLS  524334    11       10.202.5.194   20
192.168.48.221/32 sr-te     MPLS  716804    8        192.168.48.99  16777215      *
192.168.48.221/32 ldp       MPLS  65607     9        10.0.5.185     30
192.168.48.221/32 isis (0)  MPLS  524358    11       10.0.5.185     30
192.168.48.223/32 sr-te     MPLS  655362    8        10.0.13.184    200
192.168.48.223/32 sr-te     MPLS  655363    8        10.0.13.184    200
```

```
192.168.48.223/32 sr-te     MPLS  655364    8        10.0.5.185     40
192.168.48.223/32 sr-te     MPLS  655365    8        10.0.13.184    120
192.168.48.223/32 sr-te     MPLS  655366    8        10.0.5.185     120
192.168.48.223/32 sr-te     MPLS  655367    8        10.0.13.184    120
192.168.48.223/32 sr-te     MPLS  655368    8        10.0.13.184    200
192.168.48.223/32 sr-te     MPLS  655369    8        10.0.5.185     40
192.168.48.223/32 sr-te     MPLS  716805    8        192.168.48.99  16777215      *
192.168.48.223/32 ldp       MPLS  65603     9        10.0.5.185     20
192.168.48.223/32 isis (0)  MPLS  524306    11       10.0.5.185     20
192.168.48.224/32 ldp       MPLS  65604     9        10.0.5.185     30
192.168.48.224/32 isis (0)  MPLS  524361    11       10.0.5.185     30
192.168.48.226/32 isis (0)  MPLS  524365    11       11.0.1.2       65534
-------------------------------------------------------------------------------
Flags: B = BGP backup route available
       E = inactive best-external BGP route
===============================================================================
```

The details of the path of one of the SR-TE auto-LSPs now show the ERO transiting through the anycast SID of router Houston 185. See lines marked with an asterisk (*).

```
*A:Phoenix 199# show router mpls sr-te-lsp "sr-te-level2-mesh-192.168.48.223-
716805" path detail
===============================================================================
MPLS SR-TE LSP sr-te-level2-mesh-192.168.48.223-716805 Path  (Detail)
===============================================================================
Legend :
    S      - Strict                      L      - Loose
    A-SID  - Adjacency SID               N-SID  - Node SID
    +      - Inherited
===============================================================================
-------------------------------------------------------------------------------
SR-TE LSP sr-te-level2-mesh-192.168.48.223-716805 Path loose-anycast-sid
-------------------------------------------------------------------------------
LSP Name         : sr-te-level2-mesh-192.168.48.223-716805
Path LSP ID      : 20480
From             : 192.168.48.199       To                  : 192.168.48.223
Admin State      : Up                   Oper State          : Up
Path Name        : loose-anycast-sid    Path Type           : Primary
Path Admin       : Up                   Path Oper           : Up
Path Up Time     : 0d 02:30:28          Path Down Time      : 0d 00:00:00
Retry Limit      : 0                    Retry Timer         : 30 sec
Retry Attempt    : 1                    Next Retry In       : 0 sec
CSPF             : Enabled              Oper CSPF           : Enabled
Bandwidth        : No Reservation       Oper Bandwidth      : 0 Mbps
Hop Limit        : 255                  Oper HopLimit       : 255
Setup Priority   : 7                    Oper Setup Priority  : 7
Hold Priority    : 0                    Oper Hold Priority   : 0
Inter-area       : N/A
PCE Updt ID      : 0                    PCE Updt State      : None
PCE Upd Fail Code: noError
PCE Report       : Enabled              Oper PCE Report     : Disabled
PCE Control      : Disabled             Oper PCE Control    : Disabled
PCE Compute      : Disabled
Include Groups   :                      Oper Include Groups  :
None                                           None
Exclude Groups   :                      Oper Exclude Groups  :
None                                           None
IGP/TE Metric    : 16777215             Oper Metric         : 16777215
Oper MTU         : 1492                 Path Trans          : 1
Failure Code     : noError
Failure Node     : n/a
Explicit Hops    :
    192.168.48.99(L)
Actual Hops      :
```

```
    192.168.48.99 (192.168.48.185)(N-SID)          Record Label        : 200099    *
 -> 192.168.48.223 (192.168.48.223)(N-SID)         Record Label        : 200323    *
===============================================================================
```

2.3 Segment Routing Policies

The concept of a Segment Routing (SR) policy is described by the IETF draft *draft-ietf-spring-segment-routing-policy*. A segment-routing policy specifies a source-routed path from a head-end router to a network endpoint, and the traffic flows that are steered to that source-routed path. A segment-routing policy intended for use by a particular head-end router can be statically configured on that router or advertised to it in the form of a BGP route.

The following terms are important to understanding the structure of a segment routing policy and the relationship between one policy and another.

- Segment-routing policy — a policy identified by the tuple of (head-end router, endpoint and color). Each segment routing policy is associated with a set of one or more candidate paths, one of which is selected to implement the segment routing policy and installed in the dataplane. Certain properties of the segment routing policy come from the currently selected path - for example, binding SID, segment list(s), and so on.
- Endpoint — the far-end router that is the destination of the source-routed path. The endpoint may be null (all-zero IP address) if no specific far-end router is targeted by the policy.
- Color — a property of a segment routing policy that determines the sets of traffic flows that are steered by the policy.
- Path — a set of one or more segment lists that are explicitly or statically configured or dynamically signaled. If a path becomes active then traffic matching the segment routing policy is load-balanced across the segment lists of the path in an equal, unequal, or weighted distribution. Each path is associated with:
  - a protocol origin (BGP or static)
  - a preference value
  - a binding SID value
  - a validation state (valid or invalid)
- Binding SID — a SID value that opaquely represents a segment routing policy (or more specifically, its selected path) to upstream routers. BSIDs provide isolation or decoupling between different source-routed domains and improve overall network scalability. Usually, all candidate paths of a segment routing policy are assigned the same BSID.

  These concepts are illustrated by the following example. Suppose there is a network of 7 nodes as shown in *Figure 24: Network Example with 2 Segment Routing Policies* and there are two classes of traffic (blue and green) to be transported between node1 and node 7. There is a segment routing policy

for the blue traffic between node1 and node7 and another segment routing policy for the green traffic between these same two nodes.

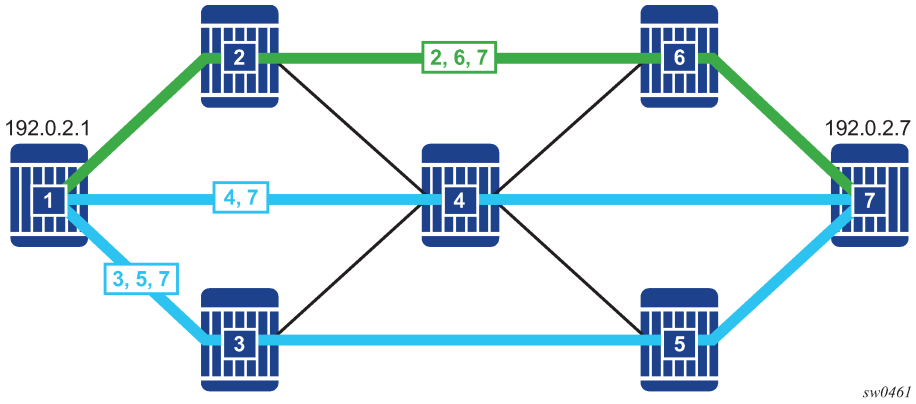

Figure 24: Network Example with 2 Segment Routing Policies

The two segment routing policies that are involved in this example and the associated relationships are depicted in *Figure 25: Relationship Between Segment Routing Policies and Paths*.

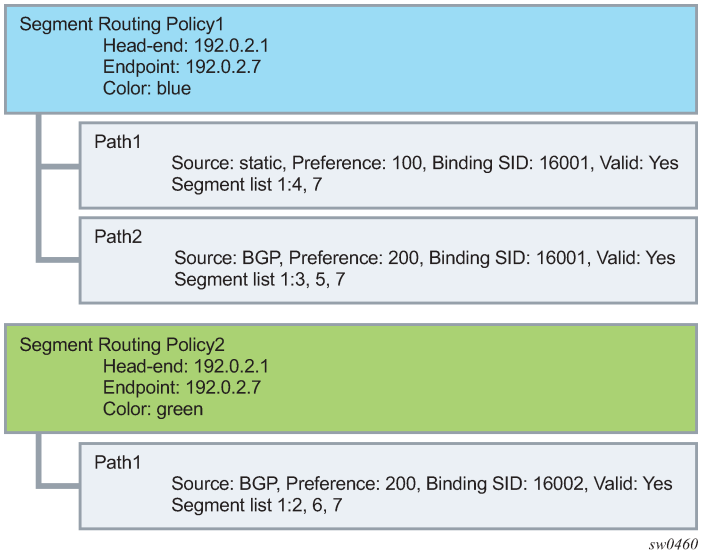

Figure 25: Relationship Between Segment Routing Policies and Paths

2.3.1 Statically-Configured Segment Routing Policies

A segment routing policy is statically configured on the router using one of the supported management interfaces. In the Nokia data model, static policies are configured under **config>router>segment-routing>sr-policies**.

There are two types of static policies: local and non-local. A static policy is local when its **head-end** parameter is configured with the value *local*. This means that the policy is intended for use by the router where the static policy is configured. Local static policies are imported into the local segment routing database for further processing. If the local segment routing database chooses a local static policy as the

best path for a particular (color, endpoint) then the associated path and its segment lists will be installed into the tunnel table (for next-hop resolution) and as a BSID-indexed MPLS label entry.

A static policy is non-local when its **head-end** parameter is set to any IPv4 address (even an IPv4 address that is associated with the local router, which is a configuration that should generally be avoided). A non-local policy is intended for use by a different router than the one where the policy is configured. Non-local policies are not installed in the local segment routing database and do not affect the forwarding state of the router where they are configured. In order to advertise non-local policies to the target router, either directly (over a single BGP session) or indirectly (using other intermediate routers, such as BGP route reflectors), the static non-local policies must be imported into the BGP RIB and then re-advertised as BGP routes. In order to import static non-local policies into BGP, you must configure the **sr-policy-import** command under **config>router>bgp**. In order to advertise BGP routes containing segment routing policies, you must add the **sr-policy-ipv4** or the **sr-policy-ipv6** family to the configuration of a BGP neighbor or group (or the entire base router BGP instance) so that the capability is negotiated with other routers.

Local and non-local static policies have the same configurable attributes. The function and rules associated with each attribute are:

- **shutdown** — used to administratively enable or disable the static policy.
- **binding-sid** — used to associate a binding SID with the static policy in the form of an MPLS label in the range 32 to 1048575. This is a mandatory parameter. The binding SID must be an available label in the **reserved-label-block** associated with segment routing policies, otherwise the policy cannot be activated.
- **color** — used to associate a color with the static policy. This is a mandatory parameter.
- **distinguisher** — used to uniquely identify a non-local static policy when it is a re-advertised as a BGP route. The value is copied into the BGP NLRI field. A unique distinguisher ensures that BGP does not suppress BGP routes for the same (color, endpoint) but targeted to different head-end routers. This is mandatory for non-local policies but optional in local policies.
- **endpoint** — used to identify the endpoint IPv4 or IPv6 address associated with the static policy. A value of 0.0.0.0 or 0::0 is permitted and interpreted as a null endpoint. This is a mandatory parameter.

  Note: When a non-local SR policy with either an IPv4 or IPv6 endpoint is selected for advertisement, the **head-end** parameter supports an IPv4 address only. This is converted into an IPv4-address-specific RT extended community (0x4102) in the advertised route in the BGP Update message.

- **head-end** — used to identify the router that is the targeted node for installing the policy. This is a mandatory parameter. The value *local* must be used when the target is the local router itself. Otherwise, any valid IPv4 address is allowed, and the policy is considered non-local. When a non-local static policy is re-advertised as a BGP route, the configured head-end address is embedded in an IPv4-address-specific route-target extended community that is automatically added to the BGP route.
- **preference** — used to indicate the degree of preference of the policy if the local segment routing database has other policies (static or BGP) for the same (color, endpoint). In order for a path to be selected as the active path for a (color, endpoint), it must have the highest preference value amongst all the candidate paths.

The following are configuration rules related to the previously described attributes:

1. Every static local policy must have a unique combination of **color**, **endpoint**, and **preference**.
2. Every static non-local policy must have a unique distinguisher.

Each static policy (local and non-local) must include, in its configuration, at least one segment-list containing at least one segment. Each static-policy can have up to 32 segment-lists, each containing up

to 11 segments. Each segment-list can be assigned a weight to influence the share of traffic that it carries compared to other segment-lists of the same policy. The default weight is 1.

The segment routing policy draft standard allows a segment-list to be configured (and signaled) with a mix of different segment types. When the head-end router attempts to install such a segment routing policy, it must resolve all of the segments into a stack of MPLS labels. In the current SR OS implementation this complexity is avoided by requiring that all (configured and signaled) segments must already be provided in the form of MPLS label values. In terms of the draft standard, this means that only type-1 segments are supported.

2.3.2 BGP Signaled Segment Routing Policies

The base router BGP instance is configured to send and receive BGP routes containing segment routing policies. In order to exchange routes belonging to the (AFI=1, SAFI=73) or (AFI=2, SAFI=73) address family with a particular base router BGP neighbor, the family configuration that applies to that neighbor must include the **sr-policy-ipv4** or the **sr-policy-ipv6** keyword respectively.

When BGP receives an **sr-policy-ipv4** route (AFI=1, SAFI=73) or a **sr-policy-ipv6 route** (AFI=2, SAFI=73) from a peer, it runs its standard BGP best path selection algorithm to choose the best path for each NLRI combination of distinguisher, endpoint, and color. If the best path is targeted to this router as head-end, BGP extracts the segment routing policy details into the local segment routing database. A BGP segment routing policy route is deemed to be targeted to this router as the head-end if either:

- it has no route-target extended community and a NO-ADVERTISE standard community
- it has an IPv4 address-specific route-target extended community with an IPv4 address matching the system IPv4 address of this router

An **sr-policy-ipv4** or a **sr-policy-ipv6** route can be received from either an IBGP or EBGP peer but it is never propagated to an EBGP peer. An **sr-policy-ipv4** or a **sr-policy-ipv6** route can be reflected to route reflector clients if this is allowed (a NO_ADVERTISE community is not attached) and the router does not consider itself the head-end of the policy.

Note: A BGP segment routing policy route is considered malformed, and triggers error-handling procedures such as session reset or treat-as-withdraw, if it does not have at least one segment-list TLV with at least one segment TLV.

2.3.3 Segment Routing Policy Path Selection and Tie-Breaking

Segment Routing policies (static and BGP) for which the local router is head-end are processed by the local segment routing database. For each (color, endpoint) combination, the database must validate each candidate path and choose one to be the active path. The steps of this process are outlined in the Segment Routing policy validation and selection process.

Procedure

1. Is the path missing a binding SID in the form of an MPLS label?

   - Yes: This path is invalid and cannot be used.
   - No: Go to next step

2. Does the path have any segment-list containing a segment type not equal to 1 (an MPLS label)?

   - Yes: This path is invalid and cannot be used.
   - No: Go to next step

3. Are all segment-lists of the path invalid?

   A segment-list is invalid if it is empty, if the first SID cannot be resolved to a set of one or more next-hops, or if the weight is 0.

   - Yes: This path is invalid and cannot be used.
   - No: Go to next step

   At this step the router attempts to resolve the first segment of each segment-list to a set of one or more next-hops and outgoing labels. It does so by looking for a matching SID in the segment routing module, which must correspond to one of the following:

   - SR-ISIS or SR-OSPF node SID
   - SR-IS or SR-OSPF adjacency SID
   - SR-IS or SR-OSPF adjacency-set SID (parallel or non-parallel set)

   Note: The label value in the first segment of the segment-list is matched against ILM label values that the local router has assigned to node-SIDs, adjacency-SIDs, and adjacency-set SIDs. The matched ILM entry may not program a swap to the same label value encoded in the segment routing policy - for example, in the case of an adjacency SID, or a node-SID reachable through a next-hop using a different SRGB base.

4. Is the binding-SID an available label in the reserved-label-block range?

   - Yes: Go to next step.
   - No: This path is invalid and cannot be used.

5. Is there another path that has reached this step that has a higher preference value?

   - Yes: This path loses the tie-break and cannot be used.
   - No: Go to next step.

6. Is there a static path?

   - Yes: Select the static path as the active path because the protocol-origin value associated with static paths (30) is higher than the protocol-origin value associated with BGP learned paths (20).
   - No: Go to next step.

7. Is there a BGP path with a lower originator value?

   The originator is a 160-bit numerical value formed by the concatenation of a 32-bit ASN and a 128-bit peer address (with IPv4 addresses encoded in the lowest 32 bits.)

   - Yes: This path loses the tie-break and cannot be used.

8. Is there another BGP path with a higher distinguisher value?

   - Yes: Select the BGP path with the highest distinguisher value.

2.3.4 Resolving BGP Routes to Segment Routing Policy Tunnels

When a statically configured or BGP signaled segment routing policy is selected to be the active path for a (**color**, **endpoint**) combination, the corresponding path and its segment lists are programmed into the tunnel table of the router. An IPv4 tunnel of type **sr-policy** (**endpoint** parameter is an IPv4 address) is programmed into the IPv4 tunnel table (TTMv4). Similarly, an IPv6 tunnel of type **sr-policy** (**endpoint** parameter is an IPv6 address) is programmed into the IPv6 tunnel table (TTMv6). The resulting tunnel entries can be used to resolve the following types of BGP routes:

- Unlabeled IPv4 routes
- Unlabeled IPv6 routes
- Label-unicast IPv4 routes
- Label-unicast IPv6 (6PE) routes
- VPN IPv4 and IPv6 routes
- EVPN routes

Specifically, an IPv4 tunnel of type **sr-policy** can be used to resolve:

- an IPv4 or the IPv4-mapped IPv6 next hop of the following route families:

  ipv4, **ipv6**, **vpn-ipv4**, **vpn-ipv6**, **label-ipv4**, **label-ipv6**, **evpn**
- the IPv6 next hop of the following route families:

  ipv6, **label-ipv4** and **label-ipv6** (SR policy with **endpoint**=0.0.0.0 only).

An IPv6 tunnel of type **sr-policy** can be used to resolve:

- the IPv6 next hop of the following route families:

  ipv4, **ipv6**, **vpn-ipv4**, **vpn-ipv6**, **label-ipv4**, **label-ipv6**, **evpn**
- the IPv4 next hop of the following route families:

  ipv4 and **label-ipv4** (SR policy with **endpoint**=0::0 only).
- the IPv4-mapped IPv6 next hop of the following route families:

  label-ipv6 (SR policy with **endpoint**=0::0 only).

2.3.4.1 Resolving Unlabeled IPv4 BGP Routes to Segment Routing Policy Tunnels

For an unlabeled IPv4 BGP route to be resolved by an SR policy:

- A color-extended community must be attached to the IPv4 route
- The base instance BGP next-hop-resolution configuration of **shortcut-tunnel>family ipv4** must allow SR policy tunnels

Note: Contrary to section 8.8.2 of *draft-filsfils-segment-routing-05*, BGP only resolves a route with multiple color-extended communities to an SR policy using the color-extended community with the highest value.

As an example, if under these conditions, there is an IPv4 route with a color-extended community (value C) and BGP next-hop address N. The order of resolution is as follows:

Procedure

1. If there is an SR policy in TTMv4 for which end-point = BGP next-hop address and color = Cn, then use this tunnel to resolve the BGP next hop.
2. If no SR policy is found in the previous step and the Cn color-extended community has its color-only (CO) bits set to '01' or '10', then try to find in TTMv4 an SR policy for which endpoint = null (0.0.0.0) and color = Cn. If there is such a policy, use it to resolve the BGP next hop.
3. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10', then try to find in TTMv6 an SR policy for which endpoint = null (0::0) and color = Cn. If there is such a policy, use it to resolve the BGP next hop.
4. If no SR policy is found in the previous steps but there is a non-SR policy tunnel in TTMv4 that is allowed by the resolution options and for which endpoint = BGP next-hop address (and for which the admin-tag meets the admin-tag-policy requirements applied to the BGP route, if applicable) then use this tunnel to resolve the BGP next hop if it has the highest TTM preference.
5. Otherwise, fall back to IGP, unless the **disallow-igp** option is configured.

2.3.4.2 Resolving Unlabeled IPv6 BGP Routes to Segment Routing Policy Tunnels

For an unlabeled IPv6 BGP route to be resolved by an SR policy:

- A color-extended community must be attached to the IPv6 route.
- The base instance BGP next-hop-resolution configuration of **shortcut-tunnel>family ipv6** must allow SR policy tunnels.

Note:

- Contrary to section 8.8.2 of draft-filsfils-segment-routing-05, BGP only resolves a route with multiple color-extended communities to an SR policy using the color-extended community with the highest value.
- For AFI2/SAFI1 routes, an IPv6 explicit null label should be always be pushed at the bottom of the stack if the policy endpoint is IPv4.

As an example, if under these conditions, there is an IPv6 route with a color-extended community (value C) and BGP next-hop address N. The order of resolution is as follows:

Procedure

1. If there is an SR policy in TTMv6 for which endpoint = the BGP next-hop address and color = Cn, then use this tunnel to resolve the BGP next hop.
2. If no SR policy is found in the previous step and the Cn color-extended community has its CO bits set to '01' or '10', then try to find a SR policy in TTMv6 for which endpoint = null (0::0) and color = Cn. If there is such a policy, use it to resolve the BGP next hop.
3. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' and there is an SR policy in TTMv4 for which endpoint = null (0.0.0.0) and color = Cn, then use this tunnel to resolve the BGP next hop.

4. If no SR policy is found in the previous steps but there is a non-SR policy tunnel in TTMv6 that is allowed by the resolution options and for which endpoint = BGP next-hop address (and for which the admin-tag meets the admin-tag-policy requirements applied to the BGP route, if applicable), then use this tunnel to resolve the BGP next hop if it has the highest TTM preference.

5. Otherwise, fall back to IGP, unless the **disallow-igp** option is configured.

2.3.4.3 Resolving Label-IPv4 BGP Routes to Segment Routing Policy Tunnels

For a label-unicast IPv4 BGP route to be resolved by an SR policy:

- A color-extended community must be attached to the label-IPv4 route.
- The base instance BGP next-hop-resolution configuration of **labeled-routes>transport-tunnel>family label-ipv4** must allow SR policy tunnels.

Note: Contrary to section 8.8.2 of draft-filsfils-segment-routing-05, BGP only resolves a route with multiple color-extended communities to an SR policy using the color-extended community with the highest value.

For example, if under these conditions, there is a label-IPv4 route with a color-extended community (value C) and BGP next-hop address N. The order of resolution is as follows.

Procedure

1. If there is an interface route that can resolve the BGP next hop, then use the direct route.

2. If **allow-static** is configured and there is a static route that can resolve the BGP next hop, then use the static route.

3. If there is no interface route or static route available or allowed to resolve the BGP next hop and the next hop is IPv4 then:
   a. Look for an SR policy in TTMv4 for which end-point = BGP next-hop address and color = Cn.

      If there is such an SR policy then try to use it to resolve the BGP next hop. If the selected SR policy has any segment-list with more than {11- **max-sr-frr-labels** under the IGPs} labels or segments, then the label-IPv4 route is unresolved.
   b. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv4 for which endpoint = null (0.0.0.0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop. If the selected SR policy has any segment-list with more than {11- **max-sr-frr-labels** under the IGPs} labels or segments, then the label-IPv4 route will be unresolved.
   c. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv6 for which endpoint = null (0::0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop. If the selected SR policy has any segment-list with more than {11- **max-sr-frr-labels** under the IGPs} labels or segments, then the label-IPv4 route is unresolved.

4. If there is no interface route or static route that is available or allowed to resolve the BGP next hop and the next hop is IPv6 then:
   a. Look for an SR policy in TTMv6 for which end-point = BGP next-hop address and color = Cn.

      If there is such an SR policy then try to use it to resolve the BGP next hop. If the selected SR policy has any segment-list with more than {11- **max-sr-frr-labels** under the IGPs} labels or segments, then the label-IPv4 route is unresolved.
   b. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv6 for which endpoint = null (0::0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop. If the selected SR policy has any segment-list with more than {11- **max-sr-frr-labels** under the IGPs} labels or segments, then the label-IPv4 route is unresolved.
   c. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv4 for which endpoint = null (0.0.0.0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop. If the selected SR policy has any segment-list with more than {11- **max-sr-frr-labels** under the IGPs} labels or segments, then the label-IPv4 route is unresolved.
5. If no SR policy is found in the previous steps but there is a non-SR policy tunnel in TTMv4 (next hop is IPv4) or TTMv6 (next hop is IPv6) that is allowed by the resolution options and for which endpoint = BGP next-hop address (and for which the admin-tag meets the admin-tag-policy requirements applied to the BGP route, if applicable), then use this tunnel to resolve the BGP next hop if it has the highest TTM preference.

2.3.4.4 Resolving Label-IPv6 BGP Routes to Segment Routing Policy Tunnels

For a label-unicast IPv6 BGP route to be resolved by an SR policy:

- A color-extended community must be attached to the label-IPv6 route.
- The base instance BGP next-hop-resolution configuration of **labeled-routes> transport-tunnel>family label-ipv6** must allow SR policy tunnels.

Note: Contrary to section 8.8.2 of draft-filsfils-segment-routing-05, BGP only resolves a route with multiple color-extended communities to an SR policy using the color-extended community with the highest value.

For example, if under these conditions, there is a label-IPv6 route with a color- extended community (value C) and BGP next-hop address N. The order of resolution is as follows.

Procedure

1. If there is an interface route that can resolve the BGP next hop, then use the direct route.
2. If **allow-static** is configured and there is a static route that can resolve the BGP next hop, then use the static route.

3. If there is no interface route or static route available or allowed to resolve the BGP next hop and the next hop is IPv6 then:
   a. Look for an SR policy in TTMv6 for which end-point = BGP next-hop address and color = Cn.

      If there is such an SR policy then try to use it to resolve the BGP next hop.
   b. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv6 for which endpoint = null (0::0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop.
   c. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv4 for which endpoint = null (0.0.0.0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop.

4. If there is no interface route or static route that is available or allowed to resolve the BGP next hop and the next hop is IPv4-mapped-IPv6 then:
   a. Look for an SR policy in TTMv4 for which end-point = BGP next-hop address and color = Cn.

      If there is such an SR policy then try to use it to resolve the BGP next hop.
   b. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv4 for which endpoint = null (0.0.0.0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop.
   c. If no SR policy is found in the previous steps and the Cn color-extended community has its CO bits set to '01' or '10' then try to find an SR policy in TTMv6 for which endpoint = null (0::0) and color = Cn.

      If there is such a policy, use it to resolve the BGP next hop.

5. If no SR policy is found in the previous steps but there is a non-SR-policy tunnel in TTMv6 (next hop is IPv6) or in TTMv4 (next hop is IPv4-mapped-IPv6) that is allowed by the resolution options and for which endpoint = BGP next-hop address (and for which the admin-tag meets the admin-tag-policy requirements applied to the BGP route, if applicable) then use this tunnel to resolve the BGP next hop if it has the highest TTM preference.

2.3.4.5 Resolving EVPN-MPLS Routes to Segment Routing Policy Tunnels

The next-hop resolution for all EVPN-VXLAN routes and for EVPN-MPLS routes without a color-extended community is unchanged by this feature.

When the resolution options associated with the **auto-bind-tunnel** configuration of an EVPN-MPLS service (vpls, b-vpls, r-vpls or E-pipe) allow **sr-policy** tunnels from TTM, then the next-hop resolution of EVPN-MPLS routes (RT-1 per-EVI, RT-2, RT-3 and RT-5) with one or more color-extended communities C1, C2, .. Cn (Cn = highest value) is based on the following rules.

Note: Contrary to section 8.8.2 of draft-filsfils-segment-routing-05, BGP only resolves a route with multiple color-extended communities to an SR policy using the color-extended community with the highest value.

Procedure

1. If the next hop is IPv6 and there is an SR policy in TTMv6 for which end-point = BGP next-hop address and color = Cn, then use this tunnel to resolve the BGP next hop.

2. Otherwise, if the next hop is IPv4 or IPv4-mapped-IPv6 and there is an SR policy in TTMv4 for which end-point = BGP next-hop address (or the IPv4 address extracted from the IPv4-mapped IPv6 BGP next-hop address) and color = Cn, then use this tunnel to resolve the BGP next hop.

3. If no SR policy is found in the previous steps but there is a non-SR policy tunnel in TTMv4 (next hop is IPv4 or IPv4-mapped-IPv6) or TTMv6 (next hop is IPv6) that is allowed by the resolution options and for which endpoint = BGP next-hop address, then use this tunnel to resolve the BGP next hop if it has the highest TTM preference.

2.3.4.6 VPRN Auto-Bind-Tunnel Using Segment Routing Policy Tunnels

When the resolution options associated with the **auto-bind-tunnel** configuration of VPRN service allow **sr-policy** tunnels from TTM, next-hop resolution of VPN-IPv4 and VPN-IPv6 routes that are imported into the VPRN and have one or more color-extended communities C1, C2, .. Cn (Cn = highest value) is based on the following rules.

Note: Contrary to section 8.8.2 of draft-filsfils-segment-routing-05, BGP only resolves a route with multiple color-extended communities to an SR policy using the color-extended community with the highest value.

Procedure

1. If the next hop is IPv6 and there is an SR policy in TTMv6 for which end-point = BGP next-hop address and color = Cn, then use this tunnel to resolve the BGP next hop.

2. Otherwise, if the next hop is IPv4 or IPv4-mapped-IPv6 and there is an SR policy in TTMv4 for which end-point = BGP next-hop address (or the IPv4 address extracted from the IPv4-mapped IPv6 BGP next-hop address in the case of VPN-IPv6 routes) and color = Cn, then use this tunnel to resolve the BGP next hop.

3. If no SR policy is found in the previous step but there is a non-SR policy tunnel in TTMv4 (next hop is IPv4 or IPv4-mapped-IPv6) or TTMv6 (next hop is IPv6) that is allowed by the resolution options and for which endpoint = BGP next-hop address, then use this tunnel to resolve the BGP next hop if it has the highest TTM preference.

2.3.5 Seamless BFD and End-to-End Protection for SR Policies

2.3.5.1 Introduction

This feature reuses of the capabilities of SR-TE LSPs to SR policy, so that operators wishing to use SR policies to enable more flexible and dynamic policy-based routing can benefit from network-based data path monitoring and fast protection switching.

Seamless BFD (S-BFD) is a form of BFD that requires significantly less state and reduces the need for session bootstrapping as compared to LSP BFD. Refer to *Seamless Bidirectional Forwarding Detection (S-BFD)* in the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*. S-BFD requires centralized configuration of a reflector function, as well as a mapping at the head-end node between the remote session discriminator and the IP address for the reflector by each session. This configuration and the mapping are described in the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*.

This section describes the application of S-BFD to SR-Polices and the configuration required for this feature. See *Seamless BFD for SR-TE LSPs* for details of the application of S-BFD to SR-TE LSPs.

S-BFD provides a connectivity check for the data path of a segment list in an SR policy, and can determine whether the segment list is up. In addition, the router also supports two protection modes for an SR policy that are distinguished by the data path programming characteristics and whether uniform failover is required between segment lists in the same SR policy candidate path (ECMP protected mode), or between the programmed candidate paths (linear mode). These protection modes are driven by the S-BFD session state on the programmed segment lists of an SR policy.

2.3.5.1.1 ECMP Protected Mode

ECMP protected mode programs all segment lists of the top-two candidate paths of an SR policy in the IOM. ECMP protected mode allows establishment of S-BFD on all of those segment lists. All of the segment lists of a specified candidate path are in the same protection group, but different candidate paths are not in the same protection group. Switchover between candidate paths is triggered by the control plane. A segment list is only included in the ECMP set of segment lists if its S-BFD session is up (user traffic is forwarded on a segment list whose S-BFD session is down). See *Figure 26: ECMP Protected SR Policy with S-BFD*.

Figure 26: ECMP Protected SR Policy with S-BFD

Figure 27: Example Application of ECMP Protected Mode with S-BFD depicts an application for S-BFD on SR policies with ECMP protected mode. Here, an SR policy is programmed at R1 by the Nokia NSP with two segment lists from R1 to R11. One segment list is using R4/R5/R7R9, and the other segment list is using R2/R3/R6/R8 and R10. These segment lists are using diverse paths and traffic that is sprayed across both of them according to the configured hashing algorithm. Separate S-BFD sessions are run on each segment list and allow the rapid detection of data path failures along the whole segment list path. R1 is able to rapidly remove a segment list from the ECMP set if S-BFD goes down, and is also able to failover to a backup SR policy (not shown) (or fall back to a less preferred LSP) if more than a certain number of the S-BFD sessions go down.

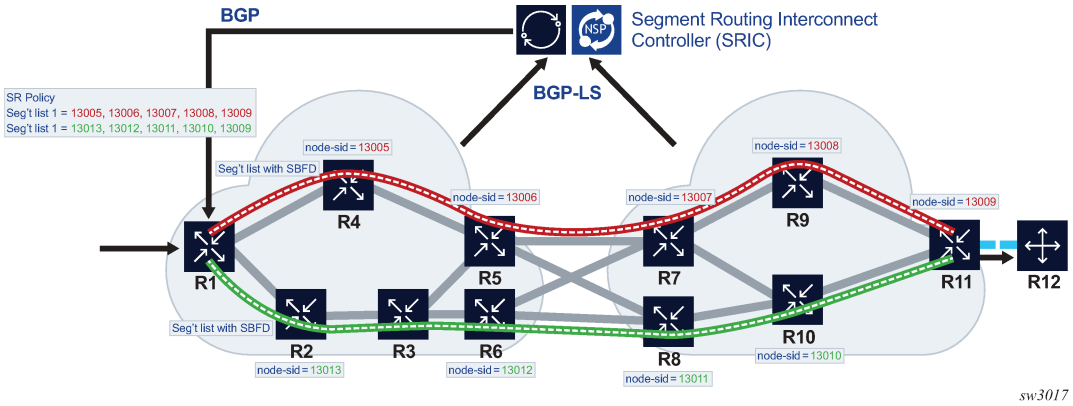

Figure 27: Example Application of ECMP Protected Mode with S-BFD

2.3.5.1.2 Linear Mode

This mode is termed linear because it is similar in operation to traditional 1-for-1 linear protection switching. It is intended to allow one or more backup paths to protect a primary path, with fast failover between candidate paths. Uniform failover is supported between candidate paths of the same SR policy. Only one segment list from each of the top-three preference candidate paths is programmed in the IOM. All of the programmed candidate paths of a specified SR policy are in the same protection group. See *Figure 28: Linear Protected SR Policy with S-BFD*.

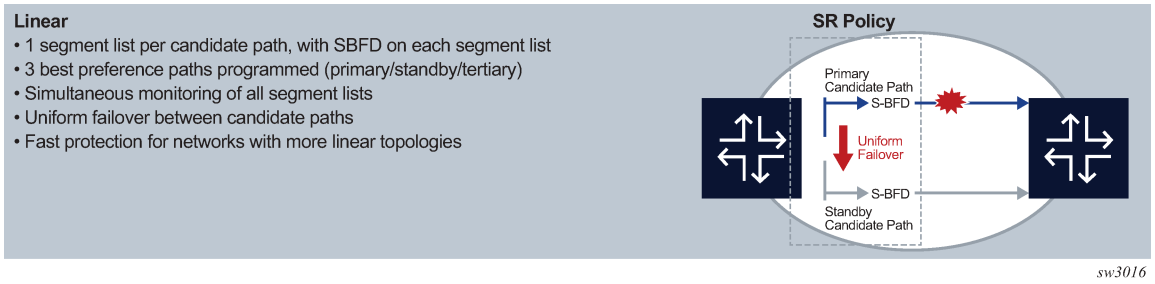

Figure 28: Linear Protected SR Policy with S-BFD

2.3.5.2 Detailed Description

This section describes the S-BFD for SR policies, support for primary and backup candidate paths, and configuration steps for S-BFD and protection for SR policies.

2.3.5.2.1 S-BFD for SR Policies

S-BFD is supported on segment lists for both static SR policies and BGP SR policies by binding a maintenance policy containing an S-BFD configuration to an imported SR policy route or a static SR policy. S-BFD packets are encapsulated on the SR policy segment lists in the same way as for SR-TE LSP paths. As in the case of SR-TE LSPs, the discriminator of the local node as well as a mapping to the far-end reflector node discriminators is first required. BFD sets the remote discriminator at the initiator of the S-BFD session based on a lookup in the S-BFD reflector discriminator using the endpoint address of the SR

policy candidate path. A candidate path of an SR policy is only treated as available if the number of up S-BFD sessions equals or exceeds a configurable threshold.

Note: When an SR policy candidate path is first programmed, a 3 second initialization hold timer is triggered. This allows the establishment of all the S-BFD sessions for all programmed paths before it decides which candidate path to activate among the eligible ones (eligible means number of segment lists with S-BFD sessions in up-state that is higher or equal to a configured threshold).

Since this is set to 3 seconds, it is recommended that the transmit and receive control packet timers are set to no more than 1 second with a maximum multiplier of 3 for S-BFD sessions.

S-BFD control packet timers, that are configurable down to 10ms, are supported for specific SR OS platforms with CPM network processor support.

The router supports an uncontrolled return path for S-BFD packets on SR policies. By default, the BFD reply packet from the reflector node is routed out-of-band to the head end of the SR policy.

2.3.5.2.2 Support for Primary and Backup Candidate Paths

End-to-end protection of static and BGP SR policies is supported using ECMP-protected or linear mode.

If an SR policy for a specified {headend, color, endpoint} is imported (by BGP) or configured (in the static case) and is selected for use, then the best (highest) preference candidate path is treated as the primary path while the next preference candidate preference policy is treated as the standby path. In linear mode, if a third path is present, then this is treated as a tertiary standby path. All of the valid segment lists for these are programmed in the IOM and made available for forwarding S-BFD packets, subject to a limitation in linear mode of one segment list per candidate path. In ECMP protected mode, the two best preference candidate paths are programmed in the IOM (up to 32 segment lists per path), while in linear mode, the three best preference candidate paths are programmed in the IOM (one segment list per candidate path).

In each case, segment lists of the best preference path are initially programmed as forwarding NHLEs while the others are programmed as non-forwarding. If the maximum number of programmed paths for a specified mode has been reached (for example, two for ECMP protected mode, and three for linear mode), and a consistent new path is received with a better preference than the existing active path, then this new path is only considered if or when the route for one of the current programmed paths is withdrawn or deleted. However, if the maximum number of programmed paths for the mode has not been reached, then the new path is programmed and any configured revert timer is started. The system switches to that better preference path immediately or when the revert timer expires (as applicable).

Failover is supported between the currently active path and the next best preference path if the currently active path is down due to S-BFD. Similar to the case of SR-TE LSPs, by default, if ECMP protected or linear mode is configured, the system switches back to the primary (best preference) SR policy path as soon as it recovers. This can happen when the number of up S-BFD sessions equals or exceeds a threshold and a hold-down timer has expired. However, it is possible to configure a revert timer to control reversion to the primary path.

All candidate paths of an SR policy must have the same binding SID when one of these two modes is applied.

2.3.5.2.3 Configuration of S-BFD and Protection for SR Policies

S-BFD and protection for SR Policies is configured using the following steps.

Refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide* for detailed information on Steps *1* and *2*. See *Configuration of SR Policy S-BFD and Mode Parameters*, *Application of S-BFD and Protection Parameters to Static SR-Policies*, and *Application of S-BFD and Protection Parameters to BGP SR-Policies* for details on Steps *3* through *5*.

Procedure

1. Configure an S-BFD reflector and mapping parameters for the remote reflector under the **configure>router>bfd>seamless-bfd** context.
2. Configure one or more BFD templates defining the BFD session parameters under the **configure>router>bfd** context.
3. Configure protection and BFD parameters that are applied to SR policies in a named maintenance policy under the **configure>router>segment-routing** context.
4. For static SR policies, apply a named maintenance policy to the static SR policy under the **configure>router>segment-routing>sr-policies>static-policy** context.
5. For dynamic BGP SR policies, configure a policy statement entry to match on a specific route or a set of routes of type **sr-policy-ipv4** with an **action** of **accept** and applying a named SR maintenance policy to them.

Configuration of SR Policy S-BFD and Mode Parameters

S-BFD and protection mode parameters are configured in a named maintenance policy. This is applied to SR policy paths that are imported by BGP as a policy statement action or by binding to a static SR policy configuration.

Maintenance policies are configured as follows:

```
configure>router>segment-routing>
   maintenance-policy <name>
      bfd-enable
      bfd-template <name>
      mode {linear | ecmp-protected}
      threshold <number>
      revert-timer <timer-value>
      hold-down-timer <timer-value>
      no shutdown
```

The **bfd-enable** command enables or disables BFD on all of the segment lists of the candidate path that are installed in the data path.

The **bfd-template** command refers to a named BFD template that must exist on the system.

The **mode** command specifies how to program the data path and how to behave if the number of BFD sessions that are up is less than the threshold and the hold-down timer has expired. All of the paths in the set must have the same mode (see *SR Policy Route and Candidate Path Parameter Consistency*). All of the allowed segment lists of the SR policy path are programmed in the IOM. The default mode is none.

In both the **linear** mode and **ecmp-protected** modes, if two or more SR policy paths with the same {headend, color, endpoint} have the same mode, then the highest preference path is treated as an effective primary path while the next highest path preference is treated as the standby path. If a third path is present in the **linear** mode, then this is treated as a tertiary path and also programmed in the IOM.

In the **ecmp-protected** mode, all the segment lists of the top two best preference paths are programmed it the IOM. However, in **linear** mode, the lowest index segment list of each of the top three preference paths is programmed in the IOM and linear protection is supported between that set. All of the segment lists of the programmed paths are made available for forwarding S-BFD packets.

If the currently active path becomes unavailable due to S-BFD, the system fails over to the next best preference candidate path that is available. If all programmed candidate paths are unavailable, then the SR policy is marked as down in TTM.

The **linear** mode supports uniform failover between candidate paths (policy routes) of the same SR policy. If **linear** mode is configured then the following rules apply.

- Only one segment list is allowed per SR policy path. If more than one is configured, then only the lowest index segment list is programmed in the data path.
- The top-3 best preference valid SR policy paths belonging to the same SR policy are programmed in the IOM and are assigned to the same protection group. Uniform failover is supported between these paths.

The **threshold** command configures the minimum number of S-BFD sessions that must be up to consider the SR policy candidate path to be up. If it is below this number, the SR policy candidate path is marked as BFD degraded by the system. The threshold parameter is only valid in the **ecmp-protected** mode (a threshold of 1 is implicit in the **linear** mode).

If the **revert-timer** command is configured, then the router starts a revert timer when the primary path recovers (for example, after the number of S-BFD sessions that are up is ≥ threshold and the hold-down timer has expired) and switches back when the timer expires. If **no revert-timer** is configured, then the system reverts to the primary path for the policy when it is restored.

If a secondary or tertiary path is currently active, and the revert timer is started (due to recovery of the primary path), but the secondary path subsequently goes down due to the number of up S-BFD sessions being less than the threshold, and no other better preference standby path is available, then the router reverts immediately to the primary path. However, if a better preference standby path is available and up, the revert timer is not canceled and the system switches to the better preference standby path and switches back to the primary path when the revert timer expires. If the hold-down timer is currently active on a better-preference path, then the system immediately switches to the primary path. If the system needs to switch to the primary path but the hold-down timer is still active on the primary path, the system cancels the timer and switches immediately.

The **hold-down-timer** command is intended to prevent bouncing of the SR policy path state if one or more BFD sessions associated with segment lists flap and cause the threshold to be repeatedly crossed in a short period of time. The hold-down timer is started when the number of S-BFD sessions that are up drops below the threshold. The SR policy path is not considered to be up again until the hold-down timer has expired and the number of S-BFD sessions that are up equals or exceeds the threshold.

A maintenance policy can only be deleted or a value changed if the maintenance policy is administratively disabled (shutdown). A maintenance policy can only be enabled if the **bfd-enable**, **bfd-template** and **mode** commands are configured. All associated SR policy paths are deleted from the IOM if a maintenance template is shutdown.

Application of S-BFD and Protection Parameters to Static SR-Policies

A named maintenance policy is applied to a static SR policy using the **maintenance-policy** command as follows:

```
config router segment-routing sr-policies
         static-policy <name>
            head-end local
            binding-sid <number>
            maintenance-policy <name>
            ...
```

A maintenance policy can only be configured if the static SR policy **head-end** is set to **local**. Policies with an IP address that is not local to the node are not programmed in the SR database and cannot have S-BFD sessions established on them by this node because they are not the head end for the SR policy path.

S-BFD needs an endpoint address for the session so that the S-BFD reflector discriminator can be looked-up as a part of the session addressing. A maintenance policy cannot be configured on an SR policy with a null endpoint.

Application of S-BFD and Protection Parameters to BGP SR-Policies

S-BFD and protection parameters can be applied to matching imported SR policy routes. Match criteria in the route import policy for the color, endpoint and route distinguisher of a policy enable matching on a specific SR policy route for **family sr-policy-v4** and **sr-policy-v6** types.

Note: For routes with the same matching distinguisher, only those with the best criteria are pushed to the SR database.

For example, matching a unique SR policy requires the following fully qualified set of match criteria:

```
configure router policy-options
   policy-statement <name>
      entry <id>
         from family sr-policy-ipv4
         from distinguisher <rd-value>
         from color <color>
         from endpoint <ip-address>
```

However, users may only require more general match criteria (for example, to apply the same maintenance template to all imported SR policy IPv4 routes, irrespective of color or endpoint).

An SR policy maintenance template is applied to matching SR policy routes using the **sr-maintenance-policy action** commands.

```
configure policy-options
   policy-statement <name>
      entry <id>
         from family sr-policy-ipv4
         ...
         action accept
           sr-maintenance-policy <name>
```

Maintenance policy statements are applicable as actions on a specific entry or as the default action.

The named SR maintenance policy must exist on the system when the commit is executed for the routing policy. If parameterization of actions is used and the named SR maintenance policy exists, the router still validates.

A change in policy options action deletes all programmed paths for that route and based on the new action, re-downloads applicable routes to the IOM.

SR Policy Route and Candidate Path Parameter Consistency

An SR policy consists of a set of one or more candidate paths. Each candidate path may be described by an SR policy route, that may be a static SR policy that is configured under the **config>router>segment-routing>sr-policies** context, or a dynamic route imported by BGP. The router checks the consistency of the following BFD and protection parameters across all of the SR policy routes for a specified SR policy.

{Maintenance-policy existence}

```
    bfd-enable
    bfd-template <name>
    mode {linear | ecmp-protected}
    revert-timer <timer-value>
```

Maintenance-policy existence covers the case where the existing programmed route is an SR policy with no maintenance policy, and the new route has a maintenance policy, and vice-versa.

Consistency is enforced across all of the static SR policy candidate paths and dynamic SR policy routes that make up a segment routing policy. Since SR policy routes or paths are imported sequentially and cannot be considered together, inconsistencies are handled as follows:

```
First policy route imported/configured:
Check: valid set of parameters
Action: If OK, program in data path and activate

Second policy route imported/configured:
Check: valid set of parameters, consistency with existing activated policy route
Action If OK, program in data path and activate, else hold in CPM but do not program

Third policy route imported/configured:
Check: valid set of parameters, consistency with existing activated policy route (s)
Action If OK, program in data path and activate, else hold in CPM but do not program
```

Inconsistent policy routes (paths) are only programmed if their parameters are valid and any programmed routes for that SR policy are deleted.

By using the same maintenance policy for all of the SR policy's routes, inconsistencies between the BFD and protection parameters of SR policy routes belonging to a specified SR policy can be avoided.

2.3.6 Traffic Statistics for Segment Routing Policies

SR policies provide the ability to collect statistics for ingress and egress traffic. In both cases, traffic statistics are collected without any forwarding class or QoS distinction.

Traffic statistics collection is enabled as follows:

- **config>router>segment-routing>sr-policies>ingress-statistics**

  Ingress — Ingress traffic collection only applies to **binding-sid** SR policies as the statistic index is attached to the ILM entry for that label. The traffic statistics provide traffic for all the instances that share the binding SID. The statistic index is released and statistics are lost when ingress traffic statistics are disabled for that binding SID, or the last instance of a policy using that label is removed from the database.
- **config>router>segment-routing>sr-policies>egress-statistics**

  Egress — Egress traffic statistics are collected globally, for all policies at the same time. Both static and signaled policies are subject to traffic statistics collection. Statistic indexes are allocated per segment list, which allows for a fine grain monitoring of traffic evolution. Also, statistic indexes are only allocated at the time the segment list is effectively programmed. However, the system allocates at most 32 statistic indexes across all the instances of a given policy. Therefore, in the case where an instance of a policy is deprogrammed and a more preferred instance is programmed, the system behaves as follows:

  – If the segment list IDs of the preferred instance are different from any of the segment list IDs of any previously programmed instance, the system allocates new statistic indexes. While that condition holds, the statistics associated with a segment list of an instance strictly reflect the traffic that used that segment list in that instance.
  – If some of the segment list IDs of the preferred instance are equal to any of the segment list IDs of any previously programmed instance, the system reuses the indexes of the preferred instance and keeps the associated counter value and increment. In this case, the traffic statistics provided per segment list not only reflect the traffic that used that segment list in that instance. It incorporates counter values of at least another segment-list in another instance of that policy.

In all cases, the aggregate values provided across all instances truly reflect traffic over the various instances of the policy.

Statistic indexes are not released at deprogramming time. They are, however, released when all the instances of a policy are removed from the database, or when the **egress-statistics** command is disabled.

3 Segment Routing with IPv6 Data Plane (SRv6)

3.1 Introduction to Segment Routing with IPv6 Data Plane (SRv6)

Segment Routing steers packets by encoding the packet-processing instructions for each intermediate and destination router directly in the packet header.

The datapath pushes a list of instructions in the form of Segment Identifiers (SIDs) onto the packet, to forward the payload packet directly to the destination using the shortest path or using source routing via one or more transit routers. Each router that terminates a SID in the segment list performs the instructions related to that SID.

Segment Routing standards specify the following two methods for programming the datapath of the encapsulated packet:

- Segment Routing MPLS (SR-MPLS) — The SID is encoded in 32 bits and programmed as an MPLS label; provides a tunnel to both IPv4 and IPv6 destinations. See *Segment Routing with MPLS Data Plane (SR-MPLS)*.
- Segment Routing IPv6 (SRv6) — The SID is encoded in 128 bits and programmed as an IPv6 address; provides a tunnel to IPv6 destinations

SRv6 datapath encapsulation models each SID using a 128-bit address. In shortest path routing, the destination SID is encoded in the Destination Address (DA) field of the outer IPv6 header. In source routing, the SIDs of the nodes the packet must visit are encoded as a SID list in a Segment Routing Header (SRH) that is a new type of routing header compliant with RFC 8200. The next SID in a segment list to which the packet is to be forwarded is copied from the SRH into the DA field of the outer IPv6 header.

SRv6 provides more than IPv6 transport with shortest path and source-routing capabilities; it provides a framework for programmability of IPv6 networks that takes advantage of the large IPv6 address space.

Figure 29: SRv6 SRH format and fields shows the SRv6 SRH format and fields (excerpt from RFC 8986).

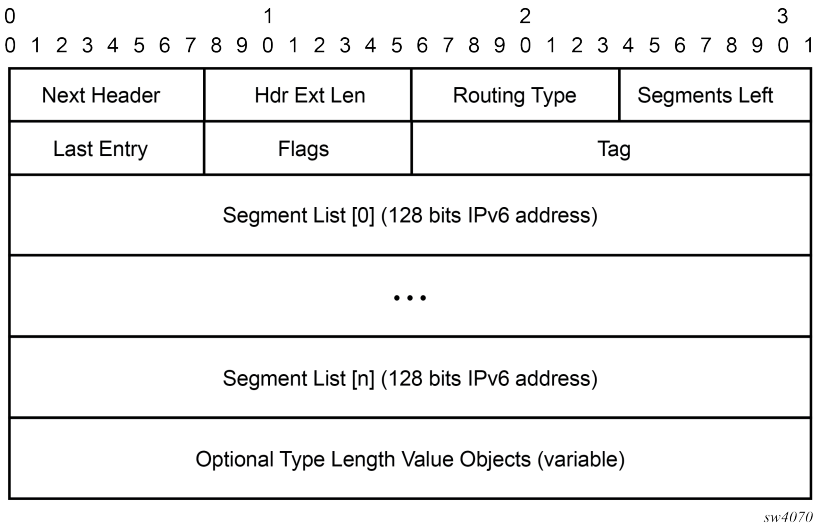

Figure 29: SRv6 SRH format and fields

Table 15: SRv6 Field descriptions

Field Name	Description
Next Header	Defined in *RFC8200, Section 4.4*
Hdr Ext Len	Defined in *RFC8200, Section 4.4*
Routing Type	4
Segments Left	Defined in *RFC8200, Section 4.4*
Last Entry	Contains the index (zero based), in the Segment List, of the last element of the Segment List
Flags	*RFC8754, Section 8.1* creates an IANA registry for new flags to be defined. The following flags are defined, as shown in Figure - TBD

Field Name	Description
Tag	Tag a packet as part of a class or group of packets; for example, packets sharing the same set of properties. When Tag is not used at the source, it must be set to zero on transmission. When Tag is not used during SRH processing, it is ignored. Tag is not used when processing the SID; as defined in *RFC8754, Section 4.3.1*. It may be used when processing other SIDs that are not defined in this document. The allocation and use of tag is outside the scope of this document.
Segment List[0..n]	128-bit IPv6 addresses representing the nth segment in the Segment List. The Segment List is encoded starting from the last segment of the SR policy. That is, the first element of the Segment List (Segment List[0]) contains the last segment of the SR policy, the second element contains the penultimate segment of the SR policy, and so on.
TLV	Type Length Value (TLV); see *RFC8754, Section 2.1*, and *Figure 31: SRv6 SID encoding*.

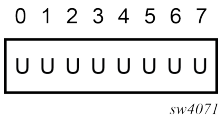

Figure 30: 8-bits of flags

Table 16: Flag descriptions

Flag	Description
U	Unused and for future use. Must be 0 on transmission and ignored on receipt.

Figure 31: SRv6 SID encoding shows the SRv6 Segment Identifier (SID) encoding format and fields.

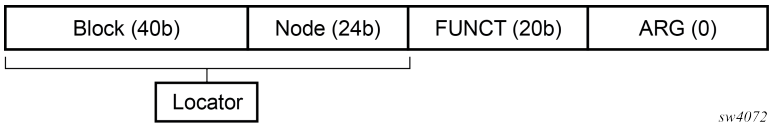

Figure 31: SRv6 SID encoding

The 128-bit address of an SRv6 SID is split into a three-field structure: {LOCATOR:FUNCTION:ARGUMENT). The size of these fields is configurable. This structure is used to encode both the transport, or reachability, information in the LOCATOR field, and SID function information in the FUNCTION field.

- LOCATOR field — encodes the transport or reachability information
- FUNCTION field — encodes the node SID function (End SID), an adjacency SID function (End.X SID), or a service function that is the equivalent of a service label in SR-MPLS
- ARGUMENT field — can be used to carry limited service or application metadata; typical use is to encode a value that identifies the source Ethernet Segment for EVPN services that require multi-homing or Etree procedures

Figure 32: SRv6 Data and Control Plane Operation shows the operation of the data and control planes when an IP-VPN route is resolved to an SRv6 tunnel.

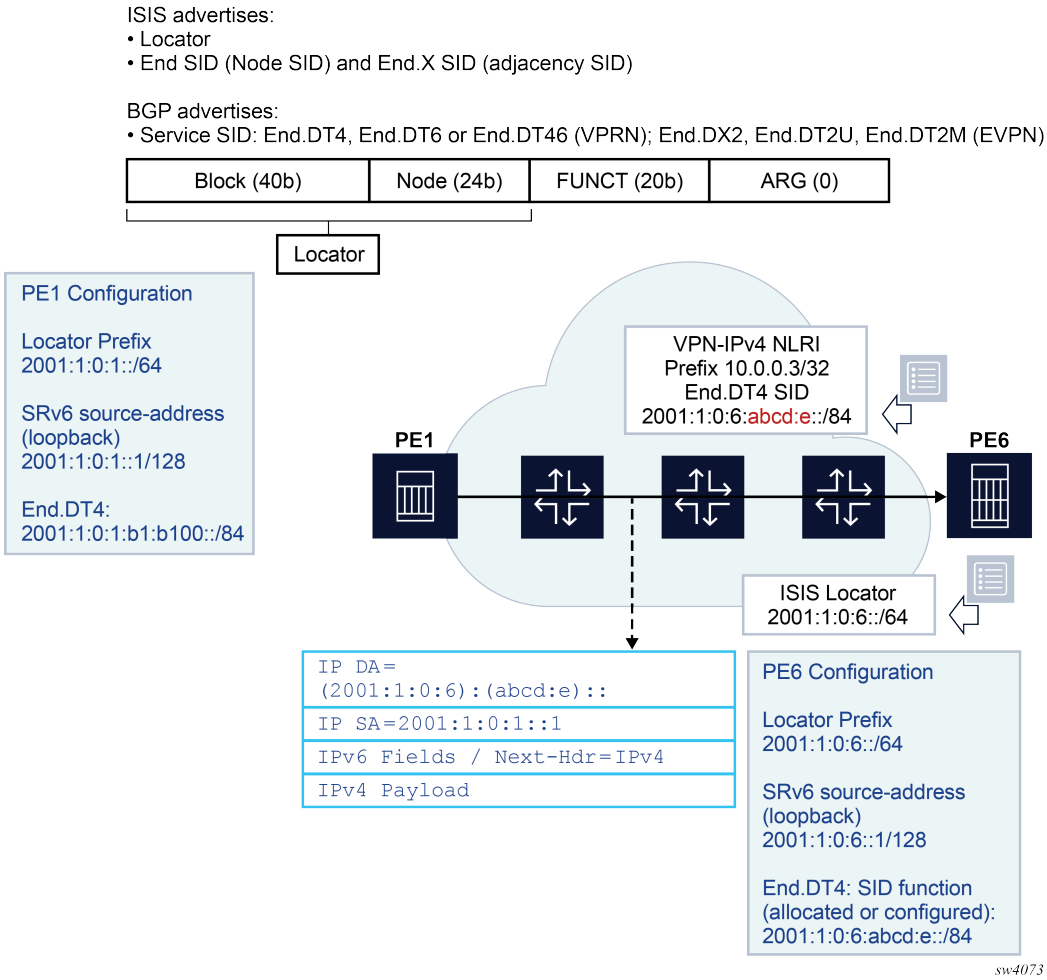

Figure 32: SRv6 Data and Control Plane Operation

The PE6 egress router advertises in IS-IS the locator route that contains its locator prefix and optionally a local node SID (End). It also advertises the SID of each adjacency (End.X) to its IS-IS neighbors.

The locator prefix provides the route to reach PE6 and is used by other routers to forward an SRv6 packet destined to any SID owned by PE6. Other routers use the End and End.X SIDs to create the repair tunnel for the Remote LFA and TI-LFA backup paths.

In BGP, PE6 advertises a VPN-IPv4 route and includes the service SID End.DT4 (equivalent in SR-MPLS to the service label in the label per-VRF model). In SRv6, the difference is the End.DT4 SID contains both the function value that identifies the specific VRF-ID in PE6 and the locator prefix that provides the reachability to router PE6.

The PE1 router resolves the received VPN-IPv4 route by validating the next hop and checking the reachability of the locator prefix of PE6 in the routing table. When PE1 receives an IPv4 packet from a CE node, it pushes an outer IPv6 header that contains the End.DT4 SID in the DA field and looks up the address in the routing table. The packet is then forwarded to one of the next hops of the route of the locator prefix of PE6.

3.2 Configuring the SRv6 Locator and SIDs

This section describes configuration of the SRv6 locator.

An SRv6 SID is a 128-bit IPv6 address which follows the structure defined by RFC 8986:

SRv6 SID={LOCATOR:FUNCTION:ARGUMENT}.

The user must configure the main SRv6 subnet for this node. This is the locator and is essentially an IPv6 subnet (prefix and length) that provides reachability (longest prefix match) to all the SIDs originated by this node. The prefix part is encoded in the LOCATOR field of the SRv6 SID.

The locator is further subdivided into a SID block and a node ID. For example, locator 3FFE:0:0:A1::/64 has a SID block of 3FFE:0 and node ID of 0:A1.

All nodes participating in a given SRv6 domain must draw their locator and SIDs from the same SID block. In the previous example, the SID block is subnet 3FFE:0::/32.

The following is the CLI structure for the configuration of the SRv6 locator.

```
configure
+--router
   +--segment-routing
      +--segment-routing-v6
         +--origination-fpe* <fpe>
         +--source-address <ipv6-address>
         +--locator* <locator-name>
            +--admin-state (enable|disable)
            +--termination-fpe <fpe>
            +--algorithm <0, 128-255>
            +--prefix
               +--ip-prefix <ipv6-address/prefix-length>
            +--block-length <0-96>
            +--function-length <20-96>
            +--static-function
                      +--max-entries <integer>
                      +--label-block <block-name>
```

One locator is required for algorithm 0 and one for each IGP flexible algorithm (128-255). The same locator can be shared by multiple IGP instances for the same algorithm number.

The locator prefix, example 3FFE:0:0:A1::/64, is advertised in the SRv6 Locator TLV in IS-IS in both algorithm 0 and any configured flexible algorithm number as defined in *draft-ietf-lsr-isis-srv6-extensions*. It is also advertised as a prefix in IP Reach TLV (ISIS TLV 236) in algorithm 0, so the routers that do not support SRv6 can still route the packet to the next hop of the locator and eventually to its destination node.

The FUNCTION field is user-configurable, but the ARGUMENT field is set to all zeros and is not configurable. The ARGUMENT length must be signaled as zero in ISIS End and End.X SIDs and in BGP service SIDs. Also, the sum of the LOCATOR and FUNCTION lengths must be less than or equal to 128.

Within algorithm 0 and each IGP flexible algorithm, the locator function (FUNCTION field) assigns the value of the End SID and End.X SID, that corresponds to the node SID and adjacency or adjacency SET SID respectively.

The locator function also assigns the value of the service SIDs owned by this node and advertised in the BGP control plane (End.DT4, End.DT6, End.DT46, and End.DX2).

The FUNCTION field can be subdivided into a static and a dynamic subrange. The user can draw from the static subrange to manually assign an SRv6 SID to a node, a local adjacency, or a service. IS-IS and BGP can draw from the dynamic subrange to assign a SID to a local adjacency or a service. The SID of an adjacency to a IS-IS neighbor, over a broadcast interface (LAN End.X), is always dynamically assigned and is not configurable.

The following CLI enables the allocation of a SRv6 SID function value. Manual allocation of a static function value is supported with the node (End SID), adjacency over a P2P interace (End.X SID), and a service SID. Auto-allocation is supported with an adjacency over a P2P interace (End.X SID) and a service SID.

```
configure
+--router
+---segment-routing
|   +---segment-routing-v6
|   |   +--- base-routing-instance
|   |   |   |   locator <locator-name>
|   |   |   |   +---function
|   |   |   |   |   |   end <integer>
|   |   |   |   |   |   +---srh-mode <psp | usp>
|   |   |   |   |   +---end-x-auto-allocate <psp|usp> <protected|unprotected>
|   |   |   |   |   +---end-x <integer>
|   |   |   |   |   |   +---interface <name>
|   |   |   |   |   |   +---srh-mode <psp | usp>
|   |   |   |   |   |   +---protection <protected|unprotected>
|   |   |   |   |   +---end-dt4 [<integer>]
|   |   |   |   |   +---end-dt6 [<integer>]
|   |   |   |   |   +---end-dt46 [<integer>]
```

3.3 IS-IS Control Plane Extensions

The following CLI enables SRv6 in the IS-IS instance and assigns a locator to each algorithm (zero or flexible algorithm).

```
configure
+--router
+---isis <0..31>
|   +---segment-routing-v6
|   |   +---locator <locator-name>
```

```
|   |   |   +--level-capability <level-1|level-2|level-1/2>
|   |   |   +---level <1|2>
|   |   |   |   +---metric <1..16777215>
|   |   |   +---tag <1..4294967295>
|   |   +---shutdown
|   |   |   no shutdown
```

The IS-IS control plane extensions in support of SRv6 are defined in *draft-ietf-lsr-isis-srv6-extensions*.

The IS-IS control plane advertises the SRv6 capabilities sub-TLV and the SRv6 Locator TLV. This includes the End function sub-TLV (equivalent to the prefix SID sub-TLV in SR-MPLS) and the End.X function sub-TLV (equivalent to the adjacency SID sub-TLV for a P2P link and a LAN in SR-MPLS).

The weight field in the End.X or LAN End.X sub-TLVs is not filled in on transmit and is ignored on receipt of the link TLV.

Table 17: SRv6 IS-IS TLVs desribes the supported IS-IS SRv6 TLVs in SR OS.

Table 17: SRv6 IS-IS TLVs

SRv6 TLV/Sub-TLV	Codepoint	IS-IS Context TLV	Description	SR OS Support
SRv6 Capabilities sub-TLV	25	Router CAPABILITY TLV (242)	Indicates SRv6 support	Yes
SR-Algorithm sub-TLV	19	Router CAPABILITY TLV (242)	Indicates base algorithm 0 and Flex-Alogo 128-255 support	Yes
Maximum Segments Left MSD Type sub-TLV	41	Router CAPABILITY TLV (242)	Indicates how deep a node terminating current segment can process the SRH (Segments-Left field) to move the next SID to outer IPv6 header DA field	Yes (advertised value = 11 SIDs) Received TLV is displayed in Link State DB but is not used for any purpose.

SRv6 TLV/Sub-TLV	Codepoint	IS-IS Context TLV	Description	SR OS Support
Maximum End Pop MSD Type sub-TLV	42	Router CAPABILITY TLV (242)	Maximum number of SIDs in a SRH when a node removes SRH (PSP or USP modes of SRH)	Yes (advertised value = 11 SIDs) Received TLV is displayed in Link State DB but is not used for any purpose.
Maximum H.Encaps MSD type sub-TLV	44	Router CAPABILITY TLV (242)	Indicates maximum number of SIDs in an SRH a router can push when forwarding a IP or L2 packet over a SRv6 policy	Yes (advertised value = 1 SID) Received TLV is displayed in Link State DB but is not used for any purpose.
Maximum End D MSD Type Sub-TLV	45	Router CAPABILITY TLV (242)	Maximum number of SIDs in a SRH when a node removes SRH and performs the End.DX2/4/6 or End.DT4/6 function (USP mode of SRH)	Yes (advertised value = 8 SIDs) Received TLV is displayed in Link State DB but is not used for any purpose.
SRv6 Locator TLV	27	Is a top Level IS-IS TLV	Advertises the locator prefix configured on this node to terminate SIDs in algorithm 0 and flex-algo 128-255	Yes

SRv6 TLV/Sub-TLV	Codepoint	IS-IS Context TLV	Description	SR OS Support
SRv6 End SID sub-TLV	5	SRv6 Locator TLV	Advertises the SID for the endpoint or End function (equivalent of prefix SID sub-TLV in SR-MPLS)	Yes
Prefix Attribute Flags Sub-TLV	4	SRv6 Locator TLV (also in IP Reach TLV 236)	Provides attributes of a prefix which is leaked between IS-IS levels	Yes
SRv6 End.X SID sub-TLV	43	Top level Extended IS reachability TLV (22)	Advertises the SID for the adjacency over a P2P link (equivalent of adjacency SID sub-TLV for P2P link in SR-MPLS)	Yes
SRv6 LAN End.X SID sub-TLV	44	Top level Extended IS reachability TLV (22)	Advertises the SID for the adjacency over a LAN (equivalent of adjacency SID sub-TLV for LAN in SR-MPLS)	Yes
SRv6 SID Structure Sub-Sub-TLV	5	SRv6 End SID Sub-TLV, SRv6 End.X SID Sub-TLV, SRv6 LAN End.X SID Sub-TLV	Provides the length of each field (Block, Locator, Function, and Argument) of the SRv6 SID it is advertised with	No SR OS does not advertise this sub-sub-TLV. If received from other vendor's implementation, it is not displayed in Link-State database and is also not propagated with the locator TLV.

When both SR-MPLS and SRv6 are enabled on the same IS-IS instance, an MPLS node SID cannot be configured for a prefix of the locator or an End SID. This is because the SRv6 locator subnet cannot be added to a network interface and an MPLS node SID is configurable against a network interface only.

However, both the SRv6 locator route and the SR-MPLS tunnel are programmed if IS-IS receives from a third-party router implementation, a /128 prefix that has both a locator TLV and a prefix SID TLV (with node flag enabled). If the prefix SID is for a subnet larger than /128, only the locator route is programmed and the SR-MPLS tunnel is not.

Each SID function encoded in a SRv6 SID has its own endpoint behavior codepoint as listed in *Table 18: SRv6 SID Function Endpoint Behavior Codepoints*.

Note: SRv6 standards provide the flexibility to advertise transport and service SIDs in both IS-IS and BGP. In SR OS, the IS-IS control plane only advertises the transport SID functions shown in *Table 18: SRv6 SID Function Endpoint Behavior Codepoints* and only uses the transport SID for building the LFA repair tunnels.

SR OS advertises service SID functions in the BGP control plane only as described in BGP Service control plane extensions.

For the SRH processing and removal at the SID termination , the following modes of operation are associated with the termination of the End or End.X SID. These modes are sometimes referred to as SID flavours and IS-IS assigns a unique codepoint for each mode of the same End or End.X SID.

Basic or unflavored mode	The router that terminates an End or End.X SID and the **Segments-Left** field in the received packet is 0 before decrementing, keeps the SRH in the packet, and processes the packet identified by the next-header in the SRH.Typically, the next-header indicates another SRH and the packet is then forwarded based on the lookup of that next-SID; SR OS does not support this mode.
Ultimate SRH Popping (USP) mode	The egress PE terminates the last End or End.X segment in the outer IPv6 header and then processes the service SID in the SRH. SR OS supports this mode.
Penultimate SRH Popping (PSP) mode	The router that terminates the End or End.X segment before the last in the segment list, meaning the **Segments-Left** field before decrementing has a value of 1, removes the SRH on behalf of the egress PE. SR OS supports this mode.
PSP&USP mode	This is a combination of both the USP and PSP modes. The router that terminates the End or End.X segment applies the corresponding behavior for value 0 and 1 of the **Segments-Left** field. SR OS does not support this mode.
Ultimate Segment Decapsulation (USD) mode	This is a variant of the USP in which the node skips the SRH and moves directly to process the next header indicated in the SRH. SR OS does not support this mode.

Table 18: SRv6 SID Function Endpoint Behavior Codepoints

SID Function Endpoint Behavior	Codepoint RFC 8986	SID Type: End.SID	SID Type: End.X SID	SID Type: LAN End.X SID	Advertising Protocol	Supported
End (PSP, USP, USD)	1-4, 28-31	Yes	No	No	IS-IS	Yes IS-IS only {PSP value=2, USP value=3}
End.X (PSP, USP, USD)	5-8, 32-35	No	Yes	Yes	IS-IS	Yes IS-IS only {PSP value=6, USP value=7} }
End.T (PSP, USP, USD)	9-12, 36-39	Yes	No	No	IS-IS	No *(1)*
End.DX6	16	No	Yes	Yes	BGP or IS-IS	No *(1)*
End.DX4	17	No	Yes	Yes	BGP or IS-IS	No *(1)*
End.DT6	18	Yes	No	No	BGP or static	Yes*(1)* BGP only.
End.DT4	19	Yes	No	No	BGP or static	Yes*(1)* BGP only.
End.DT46	20	Yes	No	No	BGP or static	Yes*(1)* BGP only.

SID Function Endpoint Behavior	Codepoint RFC 8986	SID Type: End.SID	SID Type: End.X SID	SID Type: LAN End.X SID	Advertising Protocol	Supported
End.DX2	21	Yes	No	No	BGP or static	No *(1)* BGP only.

Note:

1. IS-IS saves SID sub-TLVs for endpoint behavior values that it does not support if received from a third-party implementation. However, it only uses End and End.X endpoint behaviors in RLFA and TI-LFA.

   BGP advertises the supported endpoint behaviors End.DT4, End.DT6, End.DT46, and End.DX2 and accepts any behavior codepoint as long as the NLRI type is supported.

3.4 BGP Service Control Plane Extensions

This section provides an overview of the BGP service control plane extensions.

3.4.1 Overview of the BGP Requirements

The BGP service control plane required extensions are specified in *draft-ietf-bess-srv6-services*. BGP requires some changes in the IPv6, VPN-IPv4, VPN-IPv6 and EVPN family routes so that the egress PE can signal the following End programming behaviors to the ingress PE:

- Layer-3 SRv6 Service SIDs

  End.DT4 — a VPRN (or GRT) route-table lookup; signaled by VPN-IPv4 or EVPN-IFL IPv4 prefix routes (also by IPv4)

  End.DT6 — a VPRN (or GRT) route-table IPv6 lookup; signaled by VPN-IPv6 or EVPN-IFL IPv6 prefix routes (also by IPv6)

  End.DT46 — a VPRN route-table lookup for IPv4 or IPv6 prefixes; signaled by VPN-IPv4/6 or EVPN-IFL IPv4/6 prefix routes

- Layer-2 SRv6 Service SIDs

End.DX2	Layer 2 decapsulation and cross-connect to an Epipe egress SAP; signaled by A-D per EVI routes.

Note: End.DX2 is not supported in Release 21.5.

Figure 33: End.DX2 behavior for EVPN-VPWS shows an example for the End.DX2 behavior for EVPN-VPWS services.

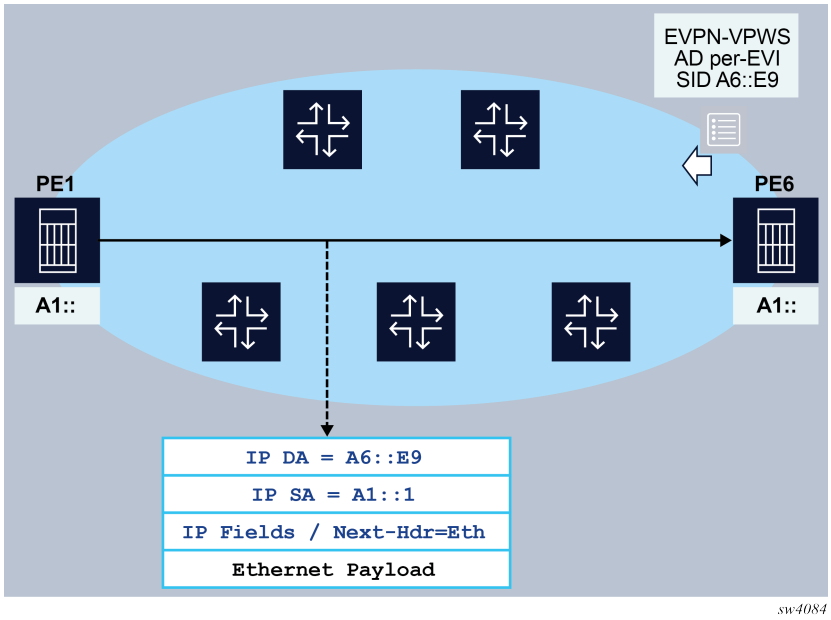

Figure 33: End.DX2 behavior for EVPN-VPWS

The ingress and egress PEs behave as follows:

- The egress PE (PE6) advertises an A-D per EVI route with the SRv6 Service SID that identifies the End.DX2 behavior. The service SID includes the configured locator in the Epipe (A6::), as well as the allocated function (E9), that identifies the Epipe at the egress PE.
- The ingress PE (PE1) imports the A-D per EVI route and creates an EVPN destination in the corresponding Epipe to A6::E9.
- When PE1 receives frames at the access SAP, it encapsulates the frames into an SRv6 packet, using the configured IP SA. The IP DA is the EVPN destination SID.
- Shortest path forwarding is considered in the example shown in *Figure 33: End.DX2 behavior for EVPN-VPWS*, and therefore the EVPN destination SID is encoded in the IP DA. If TI-LFA is required, PE1 modifies the encapsulation to include an SRH and additional SIDs.
- When the SRv6 packet arrives at PE6, the SID encoded in the IP DA identifies the packet for termination on PE6, and the Epipe for decapsulation and forwarding.

Similar procedures are followed for the other required services.

3.4.2 BGP Extensions

The following BGP extensions are required as per *draft-ietf-bess-srv6-services*:

- SRv6 Service TLV:
  - SRv6 Service TLV encoded in the BGP Prefix-SID attribute
  - SRv6 SID Information Sub-TLV (SRv6 Service Sub-TLV type 1) encoded in the SRv6 Service TLV
  - SRv6 SID Structure Sub-Sub-TLV (SRv6 Service Data Sub-Sub-TLV type 1)
- Transposition of 20 bits of the FUNCTION to the Label field of the NLRI

The BGP extensions are applied to the following routes by setting the behavior field in the SRv6 Services TLV is set as per *RFC 8986*.

- VPN-IPv4
- VPN-IPv6
- IPv4
- IPv6

3.4.3 Advertising SRv6 Service TLVs

The EVPN, VPN-IPv4, VPN-IPv6, IPv4 and IPv6 routes for the SRv6-enabled services are advertised along with the SRv6 Service TLV. The TLV format is described in *draft-ietf-bess-srv6-services* and shown in *Figure 34: SRv6 Service TLV format*.

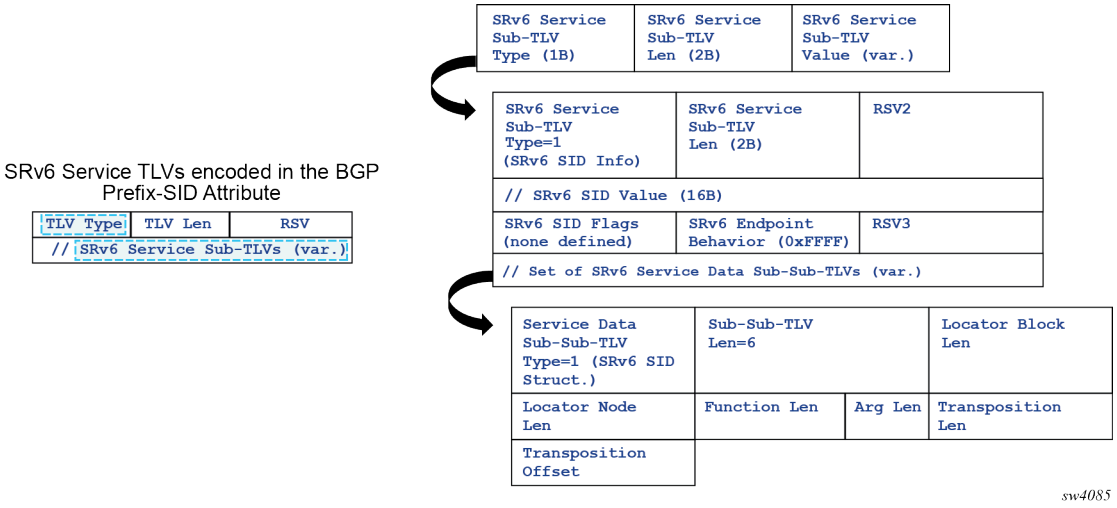

Figure 34: SRv6 Service TLV format

The SRv6 Service TLV encoded in the BGP Prefix-SID attribute can have two different types:

- Type 5 is used for Layer 3 service SIDs or the SIDs signaled for VPRN services with VPN-IP or EVPN-IFL routes. Layer 3 service SIDs are also supported for the base router along with IPv4/6 routes.
- Type 6 is used for Layer 2 service SIDs or the SIDs signaled for Epipe or VPLS services. Type 6 is not supported in Release 21.5.

The SRv6 Service TLV may contain an unordered list of sub-TLVs, but currently the SRv6 Service TLV is advertised with only one sub-TLV – the SRv6 SID Info sub-TLV (type 1). This sub-TLV encodes the following information:

- SID value — the entire 128-bit SID allocated to the service. This includes the locator configured for the service and the allocated FUNCTION (which can be dynamically allocated or statically configured on the service). The ARGUMENT is always 0.
- SID Flags — all zero.
- Endpoint behavior — encodes the behavior as in RFC 8986, in decimal values. The following are relevant for SR OS:
  - 18 – End.dt6
  - 19 – End.dt4
  - 20 – End.dt46
- One SID Structure Sub-Sub-TLV (Service Data Sub-Sub-TLV type 1).

The SID Structure Sub-Sub-TLV is always included in routes with label fields and always uses the following values when advertised:

- Locator Block Length — encodes the length of the block configured in the locator for the service
- Locator Node Length — the length of the node configured in the locator for the service
- Function Length — configurable in the range 20..96
- Argument Length — 0
- Transposition Length (TL) — 20 for EVPN and VPN-IP routes. For IP routes in the base router, the Transposition Length is always 0.
- For EVPN and VPN-IP routes:
  - If Function Length equals to 20, the Transposition Offset (TO) value equals the prefix length configured in the locator
  - If Function Length is greater than 20, the TO value equals ((prefix length configured in the locator) + (FunLength -20))
- For the base router, the TO value is always 0

3.4.4 Transposition procedures when advertising service routes

The purpose of the SID Structure Sub-Sub-TLV is twofold:

- Advertise the structure of the SRv6 SID used in the service, including the length of the Locator block, node, function and argument. This is important for future support of microSIDs and avoids overlapping issues with future values of the argument.
- Support transposition procedures for efficient service route packing. The FUNCTION is transposed into the label field in the route's NLRI. Because the rest of the SID is common for routes of the same type in the service, this transposition operation supports efficient packing of routes into the same BGP update.

Figure 35: Transposition of the FUNCTION into the NLRI shows how the FUNCTION part of the SID is transposed. SRv6 in Epipe services is not supported in Release 21.5, but the example illustrates how transposition works, and it would be similar for VPN-IP routes in Release 21.5.

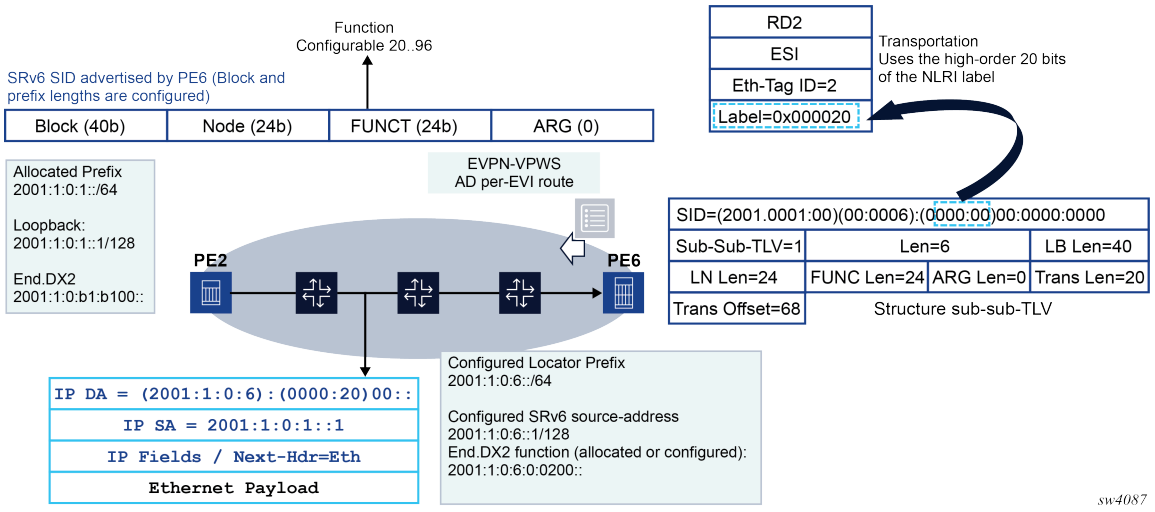

Figure 35: Transposition of the FUNCTION into the NLRI

In the *Figure 35: Transposition of the FUNCTION into the NLRI* example, PE6 is configured with an Epipe that uses a configured locator with LB length = 40 bits and LN length = 24 bits. The FUNC length is set at 24, and 20 bits are always transposed into the NLRI (non-configurable). Based on the example in *Figure 35: Transposition of the FUNCTION into the NLRI*, the following rules apply:

- On reception, the router can build any SID out of the received route, irrespective of transposition, as long as the lengths are correctly encoded.
- On transmission, the system performs a transposition for VPN-IP and EVPN service routes as follows:
- If Function Length is greater than 20 in the Locator configuration, the function bits are put at the right-most bits of the L bits. For example, if LB LEN is 40 bits and the LN Len is 24 bits:

  – If Function Length = 20, the entire function is transposed into the label field, and the following is signaled in the route:

    Length [LBL, LNL, FL, AL] : [40, 24, 20, 0]

    TL:20, TO:64

```
*A:PE-4>config>router>segment-routing>srv6>locator# info
----------------------------------------------
                        shutdown
                        block-length 40
                        termination-fpe 1
                        prefix
                            ip-prefix cafe:1:0:4::/64
                        exit
                        static-function
                            max-entries 10
                        exit
----------------------------------------------
*A:PE-4>config>router>segment-routing>srv6>locator# no shutdown

3 2021/01/19 08:21:16.827 UTC MINOR: DEBUG #2001 Base Peer 1: 2001:db8::3
"Peer 1: 2001:db8::3: UPDATE
Peer 1: 2001:db8::3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 102
    Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
```

```
        Address Family VPN_IPV4
        NextHop len 12 NextHop 192.0.2.4
        10.0.0.3/32 RD 192.0.2.4:20 Label 524254
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:20
    Flag: 0xc0 Type: 40 Len: 37 Prefix-SID-attr:
       SRv6 Services TLV (37 bytes):-
         Type: SRV6 L3 Service TLV (5)
         Length: 34 bytes, Reserved: 0x0
         SRv6 Service Information
         Service Information sub-TLV Type 1
            Type: 1 Len: 30 Rsvd1: 0x0
            SRv6 SID: cafe:1:0:4::
            SID Flags: 0x0 Endpoint Behavior: 0x14 Rsvd2: 0x0
            SRv6 SID Sub-Sub-TLV
               Type: 1 Len: 6
               BL:40 NL:24 FL:20 AL0 TL:20 TO:64
```

– If Function Length = 32, part of the function is transposed into the label field, and the following is signaled in the route:

  Length [LBL, LNL, FL, AL] : [40, 24, 20, 0]

  TL:20, TO:76

–

```
*A:PE-4>config>router>segment-routing>srv6>locator# function-length 32
*A:PE-4>config>router>segment-routing>srv6>locator# info
----------------------------------------------
                        shutdown
                        block-length 40
                        function-length 32
                        termination-fpe 1
                        prefix
                            ip-prefix cafe:1:0:4::/64
                        exit
                        static-function
                            max-entries 10
                        exit
----------------------------------------------
*A:PE-4>config>router>segment-routing>srv6>locator# no shutdown

8 2021/01/19 08:27:09.318 UTC MINOR: DEBUG #2001 Base Peer 1: 2001:db8::3
"Peer 1: 2001:db8::3: UPDATE
Peer 1: 2001:db8::3 - Send BGP UPDATE:
    Withdrawn Length = 0
    Total Path Attr Length = 102
    Flag: 0x90 Type: 14 Len: 33 Multiprotocol Reachable NLRI:
        Address Family VPN_IPV4
        NextHop len 12 NextHop 192.0.2.4
        10.0.0.3/32 RD 192.0.2.4:20 Label 524254
    Flag: 0x40 Type: 1 Len: 1 Origin: 0
    Flag: 0x40 Type: 2 Len: 0 AS Path:
    Flag: 0x40 Type: 5 Len: 4 Local Preference: 100
    Flag: 0xc0 Type: 16 Len: 8 Extended Community:
        target:64500:20
    Flag: 0xc0 Type: 40 Len: 37 Prefix-SID-attr:
       SRv6 Services TLV (37 bytes):-
         Type: SRV6 L3 Service TLV (5)
         Length: 34 bytes, Reserved: 0x0
         SRv6 Service Information
         Service Information sub-TLV Type 1
            Type: 1 Len: 30 Rsvd1: 0x0
```

```
            SRv6 SID: cafe:1:0:4::
            SID Flags: 0x0 Endpoint Behavior: 0x14 Rsvd2: 0x0
            SRv6 SID Sub-Sub-TLV
               Type: 1 Len: 6
               BL:40 NL:24 FL:32 AL0 TL:20 TO:76
```

- The label field of the NLRI (VPN-IP and EVPN routes) encodes the FUNCTION that is dynamically or statically allocated for the service.

With the transposition procedure, multiple NLRIs with the same common SRv6 SID (minus the function) can be packed into the same BGP update, as is done for regular VPN-IP or EVPN route for MPLS tunnels. The packing benefit is illustrated in *Figure 36: Transposition and route packing* .

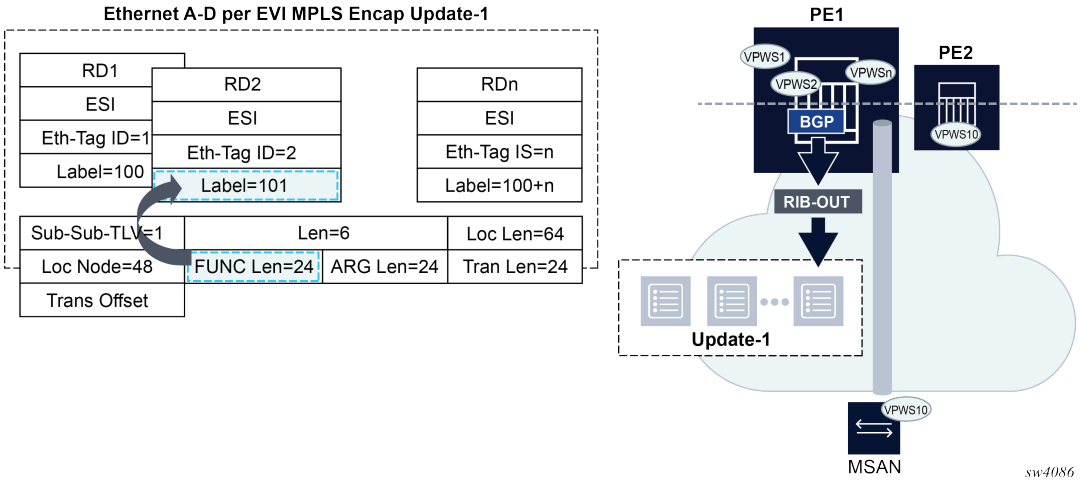

Figure 36: Transposition and route packing

Note: The transposition procedures do not apply to service SIDs in the base router, advertised via IPv4 and IPv6 families.

3.4.5 Supported Service Routes for SRv6

This section lists the service routes that are supported for SRv6.

- VPRN services configured for SRv6:
  - VPN-IPv4 routes
  - VPN-IPv6 routes
- Base router:
  - IPv6 routes if configured for SRv6 in IPv6 family
  - IPv4 routes if configured for SRv6 in IPv4 family

3.4.6 BGP Next-Hop for SRv6 Service Routes

As specified in *draft-ietf-bess-srv6-services*, the egress PE may set the next hop to any of its IPv6 addresses. When the IPv6 address value is not covered by the SRv6 Locator from which the SRv6 Service

SID is allocated, the ingress PE performs reachability checks for the SRv6 Service SID in addition to the BGP next-hop reachability procedures.

Next hop and locator resolution considerations:

- On reception of a BGP SRv6 service route, both, locator and next hop are resolved independently in the route table.
- For base instance routes (not Service routes), **no ignore-received-srv6-tlvs** triggers the independent resolution of the next-hop and the locator reachability (and collates their states that will drive route programming). The locator state is considered only if the **no ignore-received-srv6-tlvs** command is configured.
- Whether the router does next-hop-self or next-hop-unchanged does not affect the rib-in processing and reachability (since the next-hop behavior is a rib-out parameter).
- In case a received route has a resolved next-hop but unresolved locator, the show router bgp routes commands will show ‘valid/best’ but not ‘used’ in the route flags. The command **show router bgp next-hop** displays the locator and the resolution of the locator:

```
*A:PE-4# show router bgp next-hop 192.0.2.3 detail vpn-ipv4
===============================================================================
 BGP Router ID:192.0.2.4        AS:64500       Local AS:64500
===============================================================================

===============================================================================
BGP VPN Next Hop
===============================================================================
-------------------------------------------------------------------------------
VPN Next Hop          : 192.0.2.3
Autobind              : gre
Labels                : --
Admin-tag-policy      : --
Strict-tunnel-tagging : N
Color                 : --
Locator               : cafe:1:0:3::/64
-------------------------------------------------------------------------------
Resolving Prefix : 192.0.2.3/32
Preference       : 18                   Metric           : 10
Reference Count  : 6                    Owner            : GRE
Fib Programmed   : Y
Resolved Next Hop: 192.168.34.1
Egress Label     : n/a                  TunnelId         : 4294967293
Locator State    : Resolved
-------------------------------------------------------------------------------
Next Hops : 1
===============================================================================
```

3.5 Route Table, FIB Table and Tunnel Table Support

The following are the tables and information needed to process a SRv6 packet at service origination, service termination, and transit router roles.

3.5.1 RTM and FIB

SRv6 locator and SID resolution is performed in the RTM and forwarding of all SRv6 packets is performed in the FIB.

The TTM is used to save details of the SRv6 tunnel but is not used directly to forward user or CPM originated packets.

The RTM and FIB are programmed with the routes of the local and remote locators, the local End.X SIDs, and the local End SIDs.

When a policy is applied to export SRv6 routes from RTM to another IS-IS instance, only the IP Reach TLV and the locator TLV, along with the End SID sub-TLVs, are advertised by the receiving ISIS instance. Local End, End.X, and LAN End.X routes are not exported nor advertised as separate routes.

1. Remote locator (route owner = IS-IS).

   All routers in the SRv6 domain populate a resolved remote locator prefix received in the SRv6 Locator TLV in the RTM and FIB.

   A SRv6 packet is always forwarded out in the datapath using the FIB.

   For algorithm 0, if the same prefix is advertised with the IP reach prefix TLV and the SRv6 Locator TLV. A single route entry is however programmed in RTM and FIB.

   The prefix of an IGP flexible algorithm locator TLV is never advertised with an IP reach prefix TLV. Therefore the route of the locator TLV is programmed in RTM and FIB.

   a. remote locator with up to 64 ECMP next hops

      IS-IS models a remote locator prefix with two or more ECMP next hops as a regular IGP route with IP next hops.

      RTM programs the route into the FIB.

      IS-IS does not create an SRv6 tunnel entry for the locator prefix in the TTM.

   b. Remote locator with primary or backup next hops

      IS-IS models a remote locator prefix with a primary next hop or a primary and LFA backup next hop pair as an IGP route with a tunneled next hop using a protected NHLFE with a hardware PG-ID. This provides uniform failover.

      RTM programs the route into the FIB. IS-IS creates an SRv6 tunnel for the locator prefix using the SR module and the IS-IS route in the RTM and FIB points to the tunnel ID of this tunnel. The tunnel entry is added to the TTM when LFA is enabled.

2. Local Locator (route owner = SRv6)

   All routers in the SRv6 domain populate a route entry in the RTM and FIB to terminate packets destined to the local locator. This is modeled like any other local route but with the SRv6-specific route owner.

3. Local adjacency SID (route owner = IS-IS)

   All routers in the SRv6 domain populate a route entry in the RTM and regular FIB for each local End.X and LAN End.X adjacency SID with a primary and backup next hops.

   The RTM, FIB, and SR module entries are modeled exactly like a remote locator prefix with primary and backup next hops.
4. Local End SID (route owner = SRv6)

   All routers in the SRv6 domain populate a route entry in the RTM and FIB to terminate packets destined to each local End SID. This is modeled like any other local route but with the SRv6-specific route owner.

3.5.2 TTM

The tunnel table is not used directly in the SRv6 locator resolution, in SID resolution, or in packet forwarding. All resolution is performed in the RTM and forwarding is performed in the FIB.

When LFA is enabled in the IS-IS instance, a remote locator or a local adjacency that has a primary next-hop or a pair of primary and backup next-hops creates as an entry in TTM (ROUTE_OWNER_SRV6_ISIS). The entries in the RTM and FIB point to the tunnel ID of this tunnel. The TTM entry is however not used directly for forwarding SRv6 packets.

3.5.3 Users of SRv6 RTM Routes

The SRv6 locator and adjacency routes in RTM can be used to forward the following user and CPM originated packets:

- user packets
- ICMPv6 echo request and echo reply packets as explained in *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*
- UDP traceroute packets as explained in *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*

Forwarding and terminating of any other CPM originated packets are not supported. Specifically, received management protocol and control plane protocol packets that are encapsulated in SRv6 are dropped.

If the user configures the address of a BGP neighbor, an LDP peer, or an RSVP-TE LSP destination, to match a locator prefix or a SID, packets are forwarded over the SRv6 tunnel but are dropped at the destination router.

3.6 Datapath Support

This section describes the details of the packet processing in data path on ingress PE, egress PE, and transit P router roles.

3.6.1 Service Origination and Termination Roles

The SRv6 processing is performed in a specialized SRv6 FPE. The origination or termination of SRv6 on services require each the configuration of a dedicated SRv6 FPE and thus cannot share the same FPE. A single FPE can be configured for SRv6 origination. One or more FPEs can be configured for SRv6 termination. Transit SRv6 routers do not need SRv6 FPEs.

Figure 37: Packet Walkthrough Showing Both Origination and Termination on Different SRv6 FPEs shows a couple of SRv6 FPEs that are used to originate and terminate a VPRN service with SRv6 encapsulation.

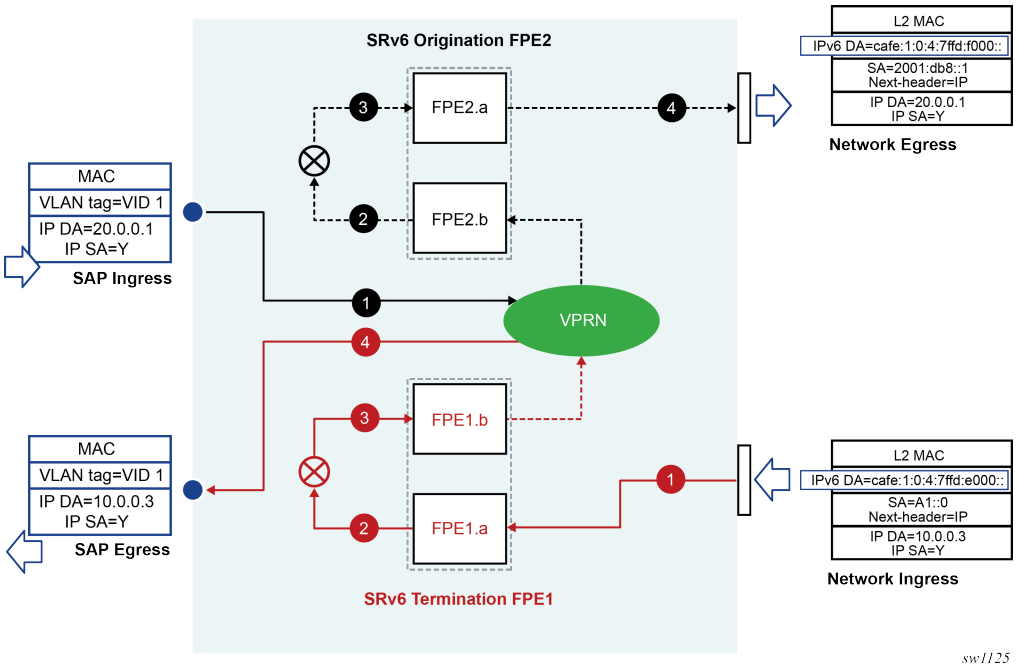

Figure 37: Packet Walkthrough Showing Both Origination and Termination on Different SRv6 FPEs

At the Ingress PE

The SRv6 FPE egress datapath receives the L2 or L3 service packet and pushes the SRv6 encapsulation header for the primary path or the backup path.

- The **hop-limit** field in the outer IPv6 header of the SRv6 tunnel is set to 255 for all transit IPv4, IPv6, and Ethernet packets encapsulated into SRv6. The hop-limit for OAM packets originated by the CPM on the router is set according to the specific OAM probe. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide* for more details.
- The SRv6 FPE ingress datapath does the lookup on the outer DA field and forwards the packet to one of the candidate egress network IP interfaces based on the flow label and or SA/DA fields of the outer IPv6 packet header. See *Using Flow Label in Load-balancing of IPv6 and SRv6 Encapsulated Packets* for more details on the spraying of SRv6 packets.

The following diagram describes details of the datapath processing of a service packet that is originated on a VPRN with SRv6 encapsulation.

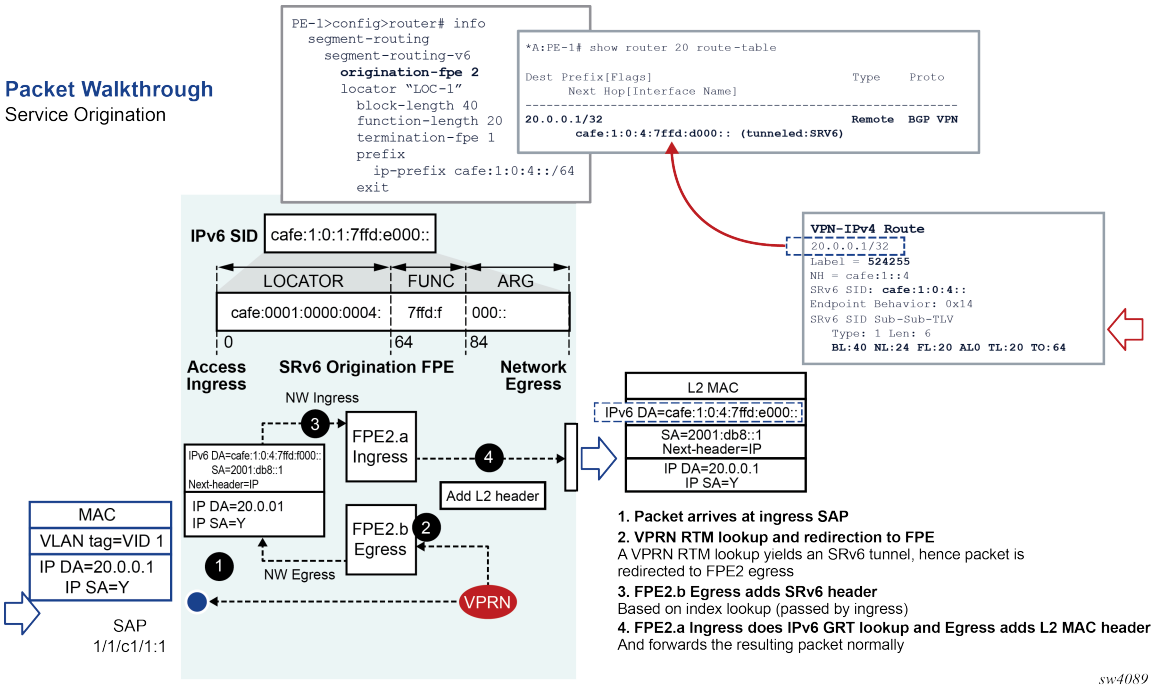

Figure 38: Walkthrough SRv6 Datapath at Service Origination Node

At the Egress PE

1. The following procedure is common to the transit router role and the service termination router role.

   On the ingress IP network interface, the SRv6 feature concurrently performs a couple of IPv6 address lookups on a received IPv6 packet: a first (longest prefix match) lookup checks if the address in the outer header DA field matches either the SRv6 local locator subnet, a local End function or local End.X function. This first lookup is for the current SID. A second lookup is performed on the next-SID in the SRH (when the IPv6 packet has an SRH). The SRv6 feature reads next-SID using the index value after decrementing the **Segments-Left** field.

   The subsequent processing depends on the outcome of the first lookup:

   a. If the match is on the locator only:

      i. if the payload type is IPv4, IPv6, or Ethernet, the packet is forwarded to the SRv6 FPE for potential service function processing; see *Service Origination and Termination Roles* for further details.

      The payload type refers to the value of the last next-header field in the processing chain of the packet. This could be the next-header field of the outer IPv6 packet, if there is no SRH. This could also be the next-header field of the active SRH (**Segments-Left**=1) for which the last SID matches the locator.

      ii. If the payload type indicates any other protocol, including ICMPv6 (ICMP ping packet) and UDP (potential traceroute message when hop-limit field has a value of 1), the packet is redirected to the CPM for further processing. Protocol matching ICMPv6 ping and UDP traceroute have

their packets processed as described in *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*. Other protocol packets are dropped.

b. If the match is on a specific local End function and the next SID lookup is not a local locator, the packet is processed as per the transit router role for these functions as detailed in *Transit Router Role with or without Segment Termination*.

c. If the match is on a specific local End function and the next SID resulted in a match on a local locator, the packet is processed as per the above parent bullet item with the next-header field used in the processing is that of the SRH.

d. If the match is on a specific local End.X function, regardless of the next SID match outcome the packet is processed as per the transit router role for these functions; see *Transit Router Role with or without Segment Termination*.

e. If the match is on a regular IPv6 route or there is no match, the packet is forwarded or dropped. For forwarded packets, the destination address could match the locator prefix or a regular IPv6 prefix of a remote node.

2. When the match is on the locator entry of the FIB, the egress SRv6 FPE datapath receives the SRv6 encapsulated packet and performs the detailed processing of the specific SID function as per *https://tools.ietf.org/html/rfc8986*. It then removes the SRv6 encapsulation headers, including SRH if any, and inserts a service label, with the value derived from the FUNCTION field value, into the inner service packet.

a. The egress SRv6 FPE decrements and propagates, into the forwarded inner packet's IPv4 TTL field or the IPv6 hop-limit field, the minimum of the incoming outer header hop-limit and inner header hop-limit (or TTL) values.

b. The ingress SRv6 FPE does an ILM lookup on the service label and forwards the packet to the service context for further processing.

The following diagram describes details of the datapath processing of a service packet that is terminated on a VPRN with SRv6 encapsulation.

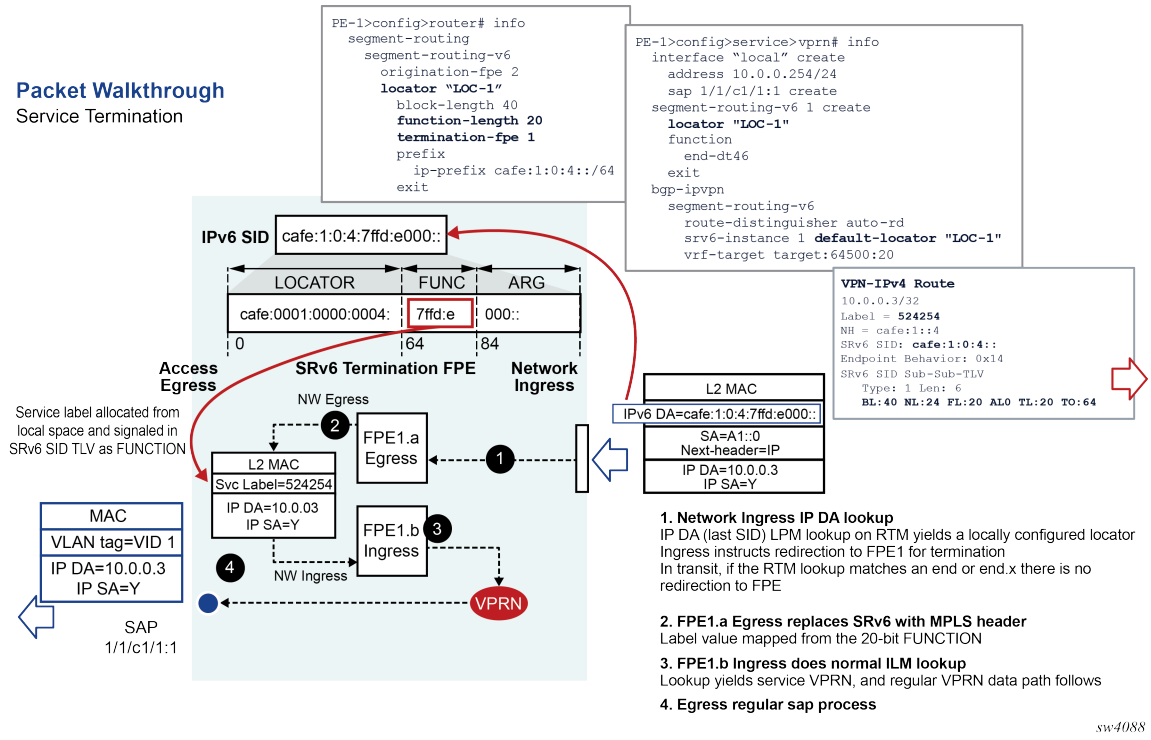

Figure 39: Walkthrough SRv6 Datapath at Service Termination Node

3.6.2 Transit Router Role with or without Segment Termination

The transit router role does not require the use of an SRv6 FPE.

The following steps summarize the packet processing for the transit router role. For more information about the specific processing of the SID function see *https://tools.ietf.org/html/rfc8986*.

1. The procedure in this step is common to a transit router role and the service termination router role.

   On the ingress IP network interface, the SRv6 feature concurrently performs a couple of IPv6 address lookups on a received IPv6 packet: a first (longest prefix match) lookup checks if the address in the outer header DA field matches either the SRv6 local locator subnet, a local End function or local End.X function. This first lookup is for the current SID. A second lookup is performed on the next-SID in the SRH when the IPv6 packet has an SRH. The next-SID is read using the index value after decrementing the **Segments-Left** field.

   The subsequent processing depends on the outcome of the first lookup:

   a. If the match is on the locator only:

      i. if the payload type is IPv4, IPv6, or Ethernet, the packet is forwarded to the SRv6 FPE for potential service function processing. See *Service Origination and Termination Roles* for further details.

         The payload type refers to the value of the last next-header field in the processing chain of the packet. This could be the next-header field of the outer IPv6 packet, if there is no SRH.

This could also be the next-header field of the active SRH (**Segments-Left**=1) which last SID matches the locator.

ii. If the payload type indicates any other protocol, including ICMPv6 (ICMP ping packet) and UDP (potential traceroute message when hop-limit field has a value of 1), the packet is redirected to the CPM for further processing.

The CPM processes packets of protocol matching ICMPv6 ping and UDP traceroute; see the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide*. The CPM drops packets of other protocols.

The CPM generates a specific ICMPv6 message to the address in the SA field of the processed or dropped packet depending on the protocol type and the result of the match of the address in the DA field of the packet. These ICMPv6 reply messages are summarized in *Table 19: ICMPv6 Reply Messages to Extracted SRv6 Packets*.

Table 19: ICMPv6 Reply Messages to Extracted SRv6 Packets

Protocol	**Destination IP Address Match Result**	**ICMPv6 Reply** (Type/Code)
ICMP echo request / reply (See *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide* for ICMPv6 Ping support in SRv6)	locator prefix [function | any arg]	echo reply / ping successful
UDP / TCP (See *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and Diagnostics Guide* for UDP Traceroute support in SRv6)	locator prefix [function | any arg]	dest unreachable, port unreachable
Any other protocol	locator prefix [function | any arg]	dest unreachable, communication prohibited

Protocol	Destination IP Address Match Result	ICMPv6 Reply (Type/Code)
All protocols including above	locator prefix | unsupported function	dest unreachable, communication prohibited

b. If the match is on a specific local End function and the next SID lookup is not a local locator, the packet is processed as per the transit router role for these functions as described in (*2*) below.

c. If the match is on a specific local End function and the next SID resulted in a match on a local locator, the packet is processed as per step (1.a) above with the next-header field used in the processing is that of the SRH.

d. If the match is on a specific local End.X function, regardless of next SID match outcome, the packet is processed as per the transit router role for these functions as described in (2) below.

e. If the match is on a regular IPv6 route or there is no match, the packet is forwarded or dropped. If the packet is forwarded, the destination address could match the locator prefix or a regular IPv6 prefix of a remote node.

2. If the match is on a local End or End.X SID, the SID termination processing is performed on the packet.

a. If the End or End.X SID is the last SID in the packet encapsulation, meaning there is no SRH or there is only expired SRHs (**Segments-Left** =0), the packet is sent to the CPM for further processing.

Note: The CPM processes ICMPv6 ping packets and UDP traceroute packets but drops any other protocol type. See the *7450 ESS, 7750 SR, 7950 XRS, and VSR OAM and*

Diagnostics Guide for more information about the processing of ICPMv6 echo request and reply packets and UPD traceroute packets.

b. If the next-header in the IPv6 header is an SRH, the **Segments-Left** field is zero, and the next-header in the SRH is another SRH, the current SRH is removed and the remaining steps are applied on the next SRH.

c. If the **Segments-Left** field is 1 and the SRH mode of the terminated SID is PSP, the SRH is removed. Otherwise, the **Segments-Left** field is decremented and used to read and copy the next SID into the DA field of the outer IPv6 header.

d. Decrement the incoming outer IPv6 header hop-limit and write it into the outgoing packet's outer IPv6 header hop-limit field.

e. If the first SID lookup of the current SID in the FIB matched an End function, use the outcome of the second SID lookup of the next SID to forward the packet to the next hop of the next SID (in the DA field of the outer IPv6 header).

f. If the first SID lookup of the current SID in the FIB matched an End.X function, override the outcome of the second SID lookup of the next SID with the set of next hops of the adjacency and forward the packet.

g. If both the current and the next SIDs match a local End or End.X SID, the packet is forwarded as indicated in *Table 20: Forwarding Behavior for Back-to-back Local SIDs*.

Table 20: Forwarding Behavior for Back-to-back Local SIDs

Current SID Match	Next SID Match	Forwarding Action
End	End	Packet is extracted to the CPM which drops it. An ICMPv6 packet (type: dest unreachable, code: communication prohibited) is sent to the address in the SA field
End	End.X	Packet is forwarded over the adjacency of the next SID to the downstream neighbor, which forwards it back to the current node for the next-next SID processing
End.X	End	Packet is forwarded over the adjacency of the current SID to the downstream neighbor, which forwards it back to the current node for the next SID processing
End.X	End.X	Packet is forwarded over the adjacency of the current SID to the downstream neighbor, which forwards it back to the current node for the next SID processing

3.6.3 Using Flow Label in Load-balancing of IPv6 and SRv6 Encapsulated Packets

When a service is bound to an SRv6 tunnel, the service packets are first forwarded to the egress network interface of the SRv6 origination FPE to build and push the SRv6 encapsulation. The packets are then handed in to the ingress network interface of the SRv6 origination FPE which sprays the packets over the ECMP next hops of the SRv6 tunnel and LAG links of the outgoing network interfaces.

The default hash calculation on the ingress service SAP or interface is based on the existing hash procedures of an IPv4, IPv6, or an Ethernet packet. For IPv6 service packets, an option is provided to include the packet's Flow Label field, when not zero, and to hash on the triplet {SA, DA, Flow Label}. The **flow-label-load-balancing** command is used to enable this behavior on an access or network interface.

The SRv6 origination FPE egress network interface copies the output of the hash on the inner packet headers into the flow label field of the outer IPv6 header that it pushes on the SRv6 encapsulated packet. This is regardless of whether the flow label is used or not in the computation of the hash on the service packet.

The SRv6 origination FPE ingress network interface does not require the **flow-label-load-balancing** command to be enabled. All SRv6 packets are automatically sprayed to the ECMP next hops of the SRv6 tunnel and LAG links of the outgoing network interfaces using a hash on the triplet {SA, DA, Flow Label} in the SRv6 packet's outer IPv6 header.

On a transit router, the hashing of SRv6 encapsulated packets can also use the Flow Label field in the outer IPv6 header to provide more entropy to the load-balancing process of SRv6 packets. The **flow-label-load-balancing** command can be configured on a network interface to hash on the triplet {SA, DA, Flow Label}. By default, a transit router only hashes on the tuple {SA, DA} in the header of a received IPv6 packet with a non-zero flow label field, including when the packet is SRv6. The description of the **flow-label-load-balancing** command and the detailed behavior of the hash feature based on the IPv6 packet flow label field and its general application to access and network interfaces is described in section *IPv6 Flow Label Load Balancing* of the *7450 ESS, 7750 SR, 7950 XRS, and VSR Interface Configuration Guide* .

3.6.4 Interaction with other Datapath Features

The following describes the interaction of the SRv6 feature with other datapath features.

- When SRv6 is enabled on the base router, datapath enables forwarding and receiving SRv6 encapsulated packets on all network interfaces of the router. The datapath, however, drops an SRv6 encapsulated packet if received from or needs forwarding to an access interface (IES, VPRN). A service packet received on either a network interface or an access interface can be encapsulated into a SRv6 packet and forwarded to a network interface.
- SRv6 feature performs concurrently a couple of IPv6 address lookups on a packet received with an SRH. The first lookup is for the current SID in the DA field in the header of the received SRv6 packet and the second is for the next SID in the SRH.

  The datapath repurposes the IPv6 unicast RPF (IPv6 uRPF) check for the next SID lookup, which means the IPv6 uRPF feature cannot be performed on all IPv6 packets received on that interface. CLI enforces this interaction and so SRv6 cannot be enabled in the base router

context (**config**>**router**>**segment-routing**>**segment-routing-v6**) if the IPv6 uPRF check (**config**>**router**>**if**>**ipv6**>**urpf**) is enabled on one or more network interfaces.

Conversely, the IPv6 uRPF check cannot be enabled on a network interface if SRv6 is enabled in the base router context.

Note: SRv6 does not impact uRPF checks of IPv4 packets received on a network interface.

- When SRv6 is enabled in one or more IGP instances, a transit router cannot check the SA field in the outer IPv6 header of a SRv6 encapsulated packet received on a network interface and that also has an SRH header. Normally an IPv6 packet which uses 0::0 or a link-local address format should be dropped. All other IPv6 packets, including a SRv6 encapsulated packet that does not have an SRH header, are checked for these two situations.
- Policy Based Routing (PBR) is allowed on flows of packets of an SRv6 tunnel. In other words, the user can apply an ACL filter on a network interface which matches on the outer SA and DA fields of the SRv6 packet and execute an action such as a redirection.

  The operator must ensure that SRv6 matching packets are directed to a router that can process and forward the SRv6 packets.

  Note: Redirecting an SRv6 packet even to an SRv6-capable router is not recommended because the processing of the SID list in the SRH can create loops for any of the SIDs in the outer DA and SRH.

3.7 LFA Support

LFA, remote LFA, and TI-LFA are supported in the following router roles:

- service originating role
- transit role with segment termination
- transit role without segment termination

The backup is computed and programmed for each remote SRv6 End SID, service SID (End.DT4 SID, End.DT6 SID, and End.D46), as well as for each local End.X or LAN End.X SID.

A base LFA backup path or a TI-LFA backup path that uses a direct IP next hop (not a repair tunnel), requires configuring a next hop that is different from the primary path and does not modify the SID list pushed on the primary path.

When the Remote LFA or TI-LFA backup path uses a repair tunnel (source routed or not), the additional SIDs of the repair tunnel must be inserted into the packet when the backup is activated. This requires the insertion of a LFA dedicated SRH into the packet. The SRV6 behavior is referred to as H.Insert.Red and is described in *https://tools.ietf.org/html/draft-filsfils-spring-srv6-net-pgm-insertion-04*. The application of this behavior to the LFA repair tunnel is described in *https://tools.ietf.org/html/draft-voyer-6man-extension-header-insertion-10*.

Figure 40: LFA Repair Tunnel Packet Encoding illustrates the packet encoding for the primary and remote LFA backup path using a dedicated reduced SRH for the repair tunnel.

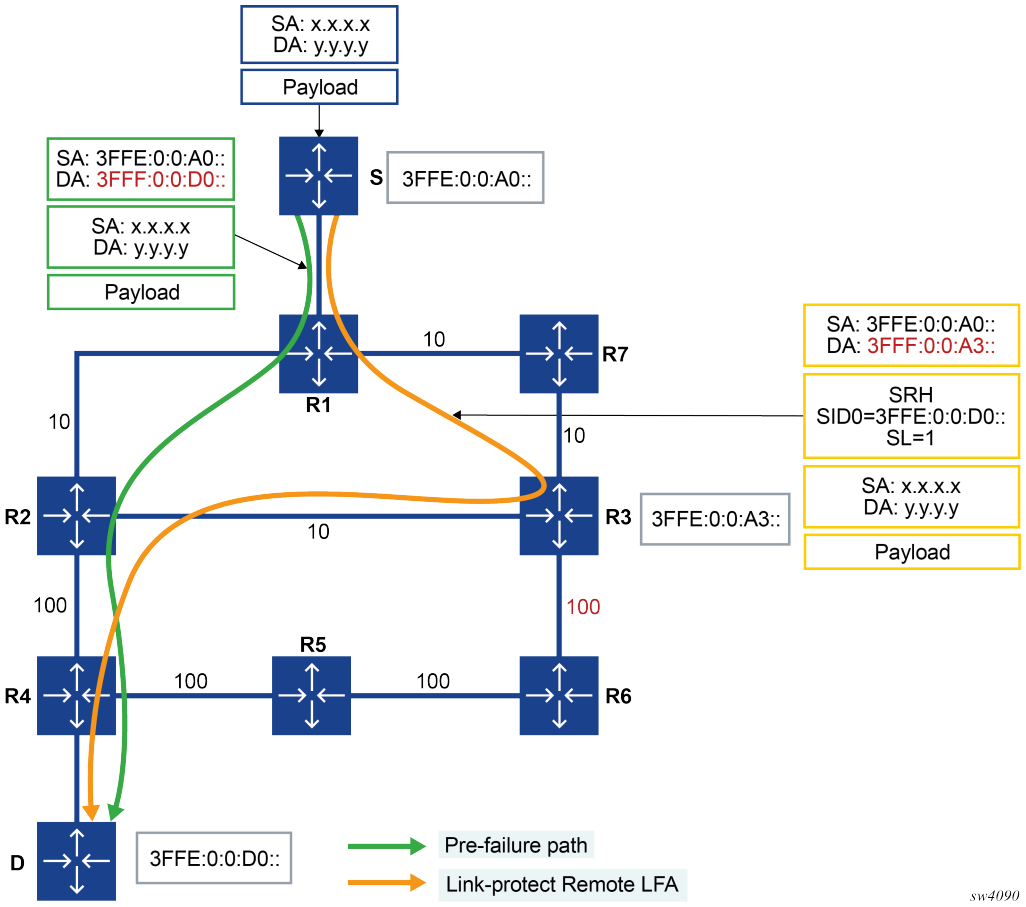

Figure 40: LFA Repair Tunnel Packet Encoding

The LFA backup path for the prefix of a remote End SID, End.DT4 SID, End.DT6 SID, or End.D46 SID is programmed in RTM and in the FIB with the entry of the prefix of the locator of that SID. This is because forwarding to these SIDs is based on looking up the remote locator they were derived from.

The LFA backup for a local End.X or a local LAN End.X is programmed in RTM and the FIB with the specific entry corresponding to this SID.

When the LFA backup is a direct IP link, the encapsulation of the packet is not changed. The packet is forwarded out of the backup path next hop interface.

When the LFA backup is a tunnel, the SRv6 feature on FP4 platforms can insert one additional SID for an RLFA or TI-LFA PQ node. This provides sufficient coverage in the following repair tunnel cases:

- The TI-LFA backup is a direct neighbor and the packet is forwarded on the backup outgoing interface with no additional SIDs.
- The P and Q nodes are the same node if it is a neighbor of the computing node or not. An End SID of the PQ node is inserted into the SID list.
- The P and Q nodes are connected using a single adjacency with the P node a neighbor of the computing node or not. An End.X SID of the P-Q adjacency is inserted into the SID list. Because an End.X adjacency SID is taken from a locator globally routable prefix, this case requires a single SID instead of a couple of SIDs with SR-MPLS.

3.7.1 IS-IS Procedures

The base LFA, remote LFA, and TI-LFA features operate on SRv6 tunnel the same way as with SR-MPLS tunnels.

The user must configure the **loopfree-alternates** command in the IS-IS instance to enable the base LFA and the **remote-lfa** or **ti-lfa** commands to enable remote LFA and/or TI-LFA.

If SR-MPLS is enabled on this IS-IS instance (**no shutdown** of the **router isis segment-routing** context), the TI-LFA algorithm makes use of the value of parameter **max-sr-frr-labels** for limiting the SID list in the backup path computation of both SR-ISIS and SRv6-ISIS tunnels. The resulting backup path is programmed for the SR-ISIS tunnel. If after compressing it, the resulting SID list has more than 1 SID, the backup path is not programmed for the SRv6-ISIS tunnel.

If SR-MPLS is disabled, the TI-LFA algorithm uses a SID list value of 1 in the backup path computation of the SRv6-ISIS tunnel.

The following is a description of the SRv6-ISIS tunnel backup path computation. A compression of the SIDs is applied to minimize the SID list of the computed backup path.

When two or more End.X SIDs of the same SRH processing type exist, IS-IS prefers the SID with protection enabled and selects the SID with lowest function value from the protected or unprotected SID subset.

1. Backup path of a remote locator

   a. TI-LFA:

      IS-IS adds the End.X SID of the P-Q set adjacency to the repair tunnel. The datapath encodes this SID into the DA field of the outer IPv6 header and moves the received value in the DA field into the LFA SRH. The protected locator prefix backup next hop in the RTM points to the tunnel ID of the locator prefix of the P-Q set adjacency.

   b. Base LFA:

      If an alternate equal-cost parallel link exists to the LFA neighbor, IS-IS programs it as a regular IP next hop of the protected locator.

      If the alternate parallel link to the LFA neighbor is not equal-cost, IS-IS programs a null SID repair tunnel. The datapath does not push an LFA SRH in this case. The protected locator prefix backup next hop in the RTM points to the tunnel ID of the local adjacency over the outgoing link to the LFA neighbor.

   c. Remote LFA:

      IS-IS adds the End SID of the PQ node to the repair tunnel. The datapath encodes this SID into the DA field of the outer IPv6 header and moves the received value of the DA field into the LFA SRH. The protected locator prefix backup next hop in the RTM points to the tunnel ID of the locator prefix of the PQ node.

      Note: If P node is a neighbor of the computing node, the SID list is empty and the backup next hop of the protected locator prefix in the RTM points to the tunnel ID of the local adjacency over the outgoing link to the LFA neighbor.

2. Backup path of a local adjacency

   a. TI-LFA

      IS-IS computes the repair tunnel by using the End.X SID of the adjacency between the P and Q nodes, plus an End SID of the neighbor node on the other side of the protected adjacency. This repair tunnel is programmed after compressing the SID list to the adjacency SID of the link between the P node and the neighbor node, only if the Q node is the same as the neighbor node on the other side of the protected adjacency. The datapath encodes that adjacency SID into the DA field of the outer IPv6 header and moves the received value of the DA field into the LFA SRH.

   b. Base LFA

      IS-IS attempts first to program a null SID repair tunnel by using the adjacency of an alternate parallel link to the neighbor on the other side of the protected adjacency. The datapath does not push an LFA SRH in this case. The protected locator prefix backup next hop in the RTM points to the tunnel ID of the local adjacency over the outgoing link to the LFA neighbor.

      If the first option fails, IS-IS programs a repair tunnel by using the End SID of the neighbor on the other side of the protected adjacency. The datapath encodes that End SID of the neighbor into the DA field of the outer IPv6 header and moves the received value of the DA field into the LFA SRH.

   c. Remote LFA

      IS-IS computes the repair tunnel by using the End SID of the PQ node and an End SID of the neighbor node on the other side of the protected adjacency. This repair tunnel is programmed after compressing the SID list to the adjacency SID of the link between the PQ node and the neighbor node, only if the PQ node is one hop from the node on the other side of the protected adjacency. The datapath encodes that adjacency SID into the DA field of the outer IPv6 header and moves the received value of the DA field into the LFA SRH.

Note: IS-IS prefers a PSP over a USP SID when selecting the additional PQ node End SID or the P-Q set adjacency End.X SID to the remote LFA or TI-LFA repair tunnel. In general, if a third-party implementation signals other SRH modes, IS-IS selects the mode in the ascending order of the SRH mode codepoint for the SID. For example, node SIDs have the following order:

- End 0x01
- End_PSP 0x02
- End_USP 0x03
- End_PSP_USP 0x04

When two or more End SIDs of the same SRH processing type exist, IS-IS selects the SID with the lowest function value.

3.7.2 Datapath Procedures

The datapath procedures for the service origination router role, as described in *Service Origination and Termination Roles*, are modified as follows when the LFA backup is a repair tunnel.

1. The top SID of the primary path is copied into the LFA SRH as the only SID in the SID list. The **Segments-Left** field of the LFA SRH is set to 1.
2. The End SID of the PQ node or the End.X SID of the P-Q adjacency is copied in the DA field of the outer IPv6 header.

3. If the PQ node is the same as the node that owns the top SID of the primary path, the LFA SRH insertion must be skipped. This is not a datapath procedure per se, but IGP compresses the SID list of the backup path to look the same as that of the primary path.

The datapath procedures for the transit router role, as described in *Transit Router Role with or without Segment Termination*, are modified as follows when the LFA backup is a repair tunnel.

1. The next SID, read from the original SRH after decrementing **Segments-Left** field, or from the DA field if no SRH, is copied into the LFA SRH as the only SID in the SID list. The **Segments-Left** field of the LFA SRH is set to 1.
2. The SID of the PQ node is copied in the DA field of the outer IPv6 header.
3. If the PQ node is the same as the node that owns the next SID, the LFA SRH insertion must be skipped. This is not a datapath procedure per se, but IGP compresses the SID list of the backup path to look the same as that of the primary path.

3.8 SRv6 Tunnel Metric and MTU Settings

IGP sets the metric of the SRv6 remote locator prefix route or that of a local adjacency SID route to the metric of the computed path of the corresponding route.

The metric of a local End SID route is set to 0; similar to any local route.

The metric of a BGP IPv4/IPv6 or VPN-IPv4/VPN-IPv6 route resolved to a SRv6 tunnel inherits the value of the locator prefix route metric.

The user should configure the network interfaces at the ingress PE and at transit P routers with an MTU value that accounts for the fixed IPv6 header (40 bytes) and the additional LFA SRHs (24 bytes each).

The SRv6 origination FPE interface (interface-b) at ingress PE requires special attention. The datapath accounts for the fixed 40-byte IPv6 header when checking for fragmenting IPv4 packets (DF=0) or dropping IPv4 (DF=1) and IPv6 packets that are intended to be forwarded over an SRv6 tunnel. If LFA is enabled in IS-IS, the LFA overhead is not accounted for, and therefore the configured MTU for the SRv6 origination FPE interface must account for it.

In addition, there are network deployments where it is not possible to modify the network interface MTU or to set all network interfaces to the same value. In that case, the user must configure the SRv6 origination FPE interface MTU to reflect this worst case MTU in the network, accounting fixed and variable LFA overhead. The following is the CLI to use for this purpose.

```
configure
+--fwd-path-ext
   +--fpe <fpe-id> [create]
         +--srv6 <origination | termination>
            +--interface-b
                +--mtu <1280-9786>
```

The **show router base icmp6** command has a global count for received “Packets Too Big” that captures the dropped packets on network interfaces, including SRv6 origination FPE interface, caused by MTU violation.

3.8.1 MTU Configuration Examples

The following are examples of MTU settings of the SRv6 origination FPE interface (interface-b).

- All default values

  By default, LFA is disabled in IS-IS. The network interface default MTU is 9786 bytes based on default FP4 Ethernet MTU of 9800 bytes.

  The SRv6 origination FPE interface MTU defaults to the same value of 9786 bytes; there is no need for further adjustments.
- All default values and RLFA or TI-LFA enabled

  The user must adjust the SRv6 origination FPE interface MTU from the first bullet item case to account for the 24 bytes introduced by LFA on the FP4 hardware.

  SRv6 origination FPE Interface MTU is 9786 - 24 = 9762 bytes.
- A remote constrained network interface and RLFA or TI-LFA enabled

  Assume a specific remote network interface in the SRv6 network is limited to a maximum value of 1500 bytes. The user must adjust the SRv6 origination FPE interface MTU in all the ingress PEs from the first bullet item case, to account for that constrained value, plus the 24 bytes introduced by LFA repair tunnel (Users can adjust for as many LFA SRHs they wish. The following example assumes 1 LFA SRH).

  SRv6 origination FPE Interface MTU = 1500 - 24 = 1476 bytes.

3.9 Service Extensions

This section describes the SRv6 forwarding path extension for the origination and termination of SRv6 services.

3.9.1 SRv6 Forwarding Path Extension

The SRv6 FPE type is required for the termination and origination of SRv6 services. The following guidelines apply for the SRv6 FPE:

- an internal or external Port Cross-Connect (PXC) can be used for the SRv6 FPE
- the SRv6 origination or termination FPE cannot be shared with other applications, but the same physical ports can be used when configuring PXC ports for multiple FPEs of different applications.

  As an example of how two FPEs can share the same physical port (hence the same bandwidth), define two PXC ports, both sharing the same underlaying physical port; for instance:

  – FPE 1 is associated to PXC-1 and FPE 2 is associated to PXC-2, where PXC-1 and PXC-2 are both assigned to port 1/1/1
- Some considerations about the SRv6 termination FPE follow:

  – it is configured per locator
  – multiple locators can optionally use the same or different FPE
  – received SRv6 traffic for a given (local) locator is redirected to the SRv6 termination FPE interface-a

- Some considerations about the SRv6 origination FPE follow:
  - there is only one SRv6 origination FPE supported per system
  - the SRv6 origination and termination FPEs are always different
- SRv6 FPE redundancy and load-balancing:
  - each FPE can use a LAG composed of as many PXC ports as needed (there is no specific limitation in the number of PXC members per LAG)
  - LAG members can be PXC ports in the same or a different card

The following CLI is required to create the FPE of type **srv6** origination or termination and apply it to a given locator. All locators may be associated to the same or a different FPE.

```
configure
+-- fpe
|   +-- application
|   |   +-- srv6
|   |   |   +-- interface-a
|   |   |   |   +-- qos
|   |   |   |       +-- network-policy
|   |   |   +-- interface-b
|   |   |   |   +-- mtu
|   |   |   |   +-- qos
|   |   |   |       +-- network-policy
|   |   |   +-- type <origination|termination>
```

```
configure
+--router
|    +--segment-routing
|    |    +--segment-routing-v6
|    |    |    +--origination-fpe <fpe>
|    |    |    +--source-address <ipv6-address>
|    |    |    +--locator <locator-name>
|    |    |    |    +--termination-fpe <fpe>
```

3.9.2 SRv6 VPRN Services

VPRN services support SRv6 End.DT4, End.DT6, and End.DT46 behaviors. VPRNs support IPv4 and IPv6 routes that are advertised in VPN-IPv4 and VPN-IPv6 families. The following CLI configures a VPRN for SRv6 with the VPN-IP families.

```
configure
+--service
|   +---vprn <service-id>
|   |   +---segment-routing-v6 <instance-id>
|   |   |   +---locator <locator-name>
|   |   |   |   +---function
|   |   |   |   |   +---end-dt4 <integer>
|   |   |   |   |   +---end-dt6 <integer>
|   |   |   |   |   +---end-dt46 <integer>
|   |   +---bgp-ipvpn
|   |   |   +---segment-routing-v6 <bgp-instance-id>
|   |   |   |   +---srv6-instance <id> default-locator <name>
|   |   |   |   +---source-address <ipv6-address>
|   |   |   |   +---route-distinguisher <rd>
|   |   |   |   +---vrf-export
|   |   |   |   +---vrf-import
```

```
|   |   |   |   +---vrf-target
|   |   |   |   +---default-route-tag <number>
|   |   |   |   +---shutdown
```

The associated locator must be configured to enable SRv6 on the VPRN service. In addition, the following rules apply.

- The function value can be statically configured, or it is dynamically allocated.
- Any Layer 3 function behavior can be configured, although VPN-IPv4 routes are advertised with **end-dt4** or **end-dt46** in that preference order (if they exist) and VPN-IPv6 is advertised with **end-dt6** or **end-dt46** in that preference order.
- The VPRN/label-mode is not relevant to SRv6 and setting it has no effect on the behavior of the SRv6 feature.
- The following is supported:
  - BGP-IPVPN and BGP-EVPN (EVPN-IFL) families are simultaneously supported in the same VPRN where SRv6 is enabled
  - up to two BGP instances per VPRN are supported
  - the two BGP instances can be associated to the same family or different families
  - a family cannot have two BGP instances of the same encapsulation, however, two bgp-ipvpn instances can be configured with SRv6 and MPLS encapsulations respectively
- VPRN feature interaction with SRv6:
  - Commands under the VPRN context that only operate on MPLS encapsulations:
    - **class-forwarding**
    - **entropy-label**
    - **hash-label**
    - **label-mode**
    - **ttl-propagate**
  - Commands under the VPRN context that are mutually exclusive with SRv6:
    - **carrier-carrier-vpn**
    - **network-interface**
    - **export-inactive-bgp**
  - VPRN features that work for MPLS and SRv6 encapsulations:
    - **vprn-type**
    - **allow-export-bgp-vpn**
    - **ecmp-unequal-cost**
    - **bgp-vpn-backup**
    - Unicast protocols on PE-CE interfaces
    - Commands such as ipsec/nat/subscriber-interfaces are supported (no interaction).
    - ECMP and edge PIC are supported for SRv6 and also across routes of the same family with different encapsulations; for example, the same prefix resolved to SRv6 and MPLS tunnels.
    - Route-target based leaking is supported for SRv6 routes in the VPRN.

3.9.3 SRv6 VPRN and BGP Path Attribute Propagation Between RTM BGP Owners

As is the case with existing MPLS encapsulation, BGP Path Attribute Propagation between BGP owners of the same VPRN RTM is supported, including routes with SRv6 encapsulation.

- BGP Path Attribute propagation for SRv6 routes does not require enabling a CLI knob
- In cases where multiple BGP owners coexist in the same VPRN RTM with a single instance, propagation is supported in the following cases, irrespective of the encapsulation of the route (MPLS or SRv6):
  - VPN-IPv4/6 ← → EVPN-IFL
  - VPN-IPv4/6 ← → VPN-IPv4/6 – when **allow-export-bgp-vpn** is enabled
  - EVPN-IFL← → EVPN-IFL – when **allow-export-bgp-vpn** is enabled
  - VPN-IPv4/6 ← → IPv4/v6
  - EVPN-IFL ← → IPv4/v6
  - VPN-IPv4/6 ← → EVPN-IFF (requires **iff-attribute-uniform-propagation**)
  - EVPN-IFL ← → EVPN-IFF (requires **iff-attribute-uniform-propagation**)
- In the case of multi-instance bgp VPRNs, VPN-IPv4/6 and EVPN-IFL routes received in one instance are readvertised into the other instance, including all the BGP Path Attributes of the original route.
- The following attributes are filtered out before the path attributes are propagated in all the previously described cases:
  - all type 0x06 extended communities
  - the BGP encapsulation extended community
  - the BGP encapsulation attribute
  - the BGP Prefix-SID attribute (which includes SRv6 Service SID TLVs)
  - all Route Target extended communities

3.9.4 Migration from MPLS to SRv6 in VPRN Services

To allow a seamless migration from MPLS tunnels to SRv6 in VPRN services, both MPLS auto-bind tunnels and SRv6 are supported in the same VPRN service.

Routes for the same prefix follow regular RTM selection in the VPRN. At the end of the selection process, if there are still SRv6 and MPLS routes, the SRv6 route is selected first and the MPLS route is removed from consideration.

3.9.5 SRv6 Service SIDs and BGP Routes in the Base Router

In the base router BGP instance, BGP routes may be originated, propagated or received with SRv6 TLVs in the prefix SID attribute. The processing rules are expected to use the following logic:

- A VPN-IPv4 or VPN-IPv6 route that is imported by the base router BGP instance from a VPRN (through the VRF export process) and that already has a prefix SID attribute with an SRv6 TLV at this stage, is advertised to all VPN-IPv4/VPN-IPv6 peers of the base router BGP instance, with **next-hop-self** as normal. There is no option to strip the SRv6 TLV/prefix SID attribute toward any of these peers, however, there is an option to drop the entire route toward selected peers or groups. (This is similar to the effect of a BGP export policy that drops the route, but route policies cannot match routes based on presence of an SRv6 TLV.)

- An IPv6 route that is imported by the base router BGP instance from another protocol's route that was added to base router RTM (static, ospf, isis, and so on) does not have a prefix-SID attribute carrying the SRv6 TLV added to it by default. This can only be achieved by configuring the **config**>**router**>**bgp**>**segment-routing-v6**>**family ipv6 add-srv6-tlvs** command, but this command also has the effect of adding a prefix SID attribute with SRv6 TLV to BGP IPv6 routes received from other peers without the SRv6 TLV and that are propagated to other family IPv6 peers with **next-hop-self** applied.
- Any BGP route received with a prefix SID attribute carrying SRv6 TLVs that is not an imported VPN-IPv4/VPN-IPv6/EVPN route or an IPv6 unlabeled unicast route is treated the same as existing routes, that is, by ignoring the prefix SID attribute contents, resolving the route based only on its BGP next hop, and propagating the prefix SID attribute to other peers unchanged.
- If an IPv6 unlabeled unicast route is received with a prefix SID attribute carrying an SRv6 TLV and the **config**>**router**>**bgp**>**segment-routing-v6**>**family ipv6 ignore-received-srv6-tlvs** command is configured, that route is treated the same as an existing route, that is, by ignoring the prefix SID attribute contents, resolving the route based only on its BGP next hop, and propagating the attribute to other peers unchanged, irrespective of the next-hop-self.
- If an IPv6 unlabeled unicast route is received with a prefix SID attribute carrying an SRv6 TLV and the **config**>**router**>**bgp**>**segment-routing-v6**>**family ipv6 ignore-received-srv6-tlvs** command is set to FALSE then that route should be considered resolved if and only if its BGP next hop is reachable AND the locator prefix is reachable. The datapath/FNH programming and IGP cost to reach the next hop (used by the BGP decision process) is based on the route to the locator prefix. The IPv6 route is propagated to other family IPv6 peers with the prefix SID attribute and its SRv6 TLV unchanged, irrespective of the next-hop-self.
- An IPv6 unlabeled unicast route that is received without a prefix SID attribute containing an SRv6 TLV does not have a prefix-SID attribute carrying the SRv6 TLV added to it by default, even if it is re-advertised with the next-hop-self applied. This can only be achieved by configuring the **config**>**router**>**bgp**>**segment-routing-v6**>**family ipv6 add-srv6-tlvs**. However, this command also has the effect of adding a prefix SID attribute with SRv6 TLV to imported/redistributed routes as noted in point 2.

```
configure
+--router
   +--segment-routing
|   |   +--- base-routing-instance
|   |   |   |   locator <locator-name>
|   |   |   |   +---function
|   |   |   |   |   +---end-dt6 <integer>
```

```
configure
+--router
   +--bgp
      +--segment-routing-v6
         +--source-address <ipv6-address>
         +--family*
            +--add-srv6-tlvs
            |  +--locator <locator-name>
            +--ignore-received-srv6-tlvs
```

Note:

- When an SRv6 TLV is added to an IPv6 unlabeled unicast route, the signaled behavior is End.DT6 and the SID Structure Sub-Sub-TLV contains a Transposition Offset and Transposition Length of 0.

- As in the case of VPRNs, a locator (that is associated to an FPE SRv6 function) is configured for the base router. If the End.DT6 function for that locator is not statically configured, a dynamically-allocated function is reserved for the End.DT6 behavior. This is internally mapped to a label, as in the case of VPRNs.

4 MPLS Forwarding Policy

The MPLS forwarding policy provides an interface for adding user-defined label entries into the label FIB of the router and user-defined tunnel entries into the tunnel table.

The endpoint policy allows the user to forward unlabeled packets over a set of user-defined direct or indirect next hops with the option to push a label stack on each next hop. Routes are bound to an endpoint policy when their next hop matches the endpoint address of the policy.

The user defines an endpoint policy by configuring a set of next-hop groups, each consisting of a primary and a backup next hops, and binding an endpoint to it.

The label-binding policy provides the same capability for labeled packets. In this case, labeled packets matching the ILM of the policy binding label are forwarded over the set of next hops of the policy.

The user defines a label-binding policy by configuring a set of next-hop groups, each consisting of a primary and a backup next hops, and binding a label to it.

This feature is targeted for router programmability in SDN environments.

4.1 Introduction to MPLS Forward Policy

This section provides information about configuring and operating a MPLS forwarding policy using CLI.

There are two types of MPLS forwarding policy:

- endpoint policy
- label-binding policy

The endpoint policy allows the user to forward unlabeled packets over a set of user-defined direct or indirect next hops, with the option to push a label stack on each next hop. Routes are bound to an endpoint policy when their next hop matches the endpoint address of the policy.

The label-binding policy provides the same capability for labeled packets. In this case, labeled packets matching the ILM of the policy binding label are forwarded over the set of next hops of the policy.

The data model of a forwarding policy represents each pair of {primary next hop, backup next hop} as a group and models the ECMP set as the set of Next-Hop Groups (NHGs). Flows of prefixes can be switched on a per NHG basis from the primary next hop, when it fails, to the backup next hop without disturbing the flows forwarded over the other NHGs of the policy. The same can be performed when reverting back from a backup next hop to the restored primary next hop of the same NHG.

4.2 Feature Validation and Operation Procedures

The MPLS forwarding policy follows a number of configuration and operation rules which are enforced for the lifetime of the policy.

There are two levels of validation:

- The first level validation is performed at provisioning time. The user can bring up a policy (**no shutdown** command) once these validation rules are met. Afterwards, the policy is stored in the forwarding policy database.
- The second level validation is performed when the database resolves the policy.

4.2.1 Policy Parameters and Validation Procedure Rules

The following policy parameters and validation rules apply to the MPLS forwarding policy and are enforced at configuration time:

- A policy must have either the **endpoint** or the **binding-label** command to be valid or the **no shutdown** will not be allowed. These commands are mutually exclusive per policy.
- The **endpoint** command specifies that this policy is used for resolving the next hop of IPv4 or IPv6 packets, of BGP prefixes in GRT, of static routes in GRT, of VPRN IPv4 or IPv6 prefixes, or of service packets of EVPN prefixes. It is also used to resolve the next hop of BGP-LU routes.

  The resolution of prefixes in these contexts matches the IPv4 or IPv6 next-hop address of the prefix against the address of the endpoint. The family of the primary and backup next hops of the NHGs within the policy are not relevant to the resolution of prefixes using the policy.

  See *Tunnel Table Handling of MPLS Forwarding Policy* for information about CLI commands for binding these contexts to an endpoint policy.
- The **binding-label** command allows the user to specify the label for binding to the policy such that labeled packets matching the ILM of the binding label can be forwarded over the NHG of the policy.

  The ILM entry is created only when a label is configured. Only a provisioned binding label from a reserved label block is supported. The name of the reserved label block using the **reserved-label-block** command must be configured.

  The payload of the packet forwarded using the ILM (payload underneath the swapped label) can be IPv4, IPv6, or MPLS. The family of the primary and backup next hops of the NHG within the policy are not relevant to the type of payload of the forwarded packets.
- Changes to the values of the **endpoint** and **binding-label** parameters require a **shutdown** of the specific forwarding policy context.
- A change to the name of the **reserved-label-block** requires a **shutdown** of the **forwarding-policies** context. The **shutdown** is not required if the user extends or shrinks the range of the **reserved-label-block**.
- The **preference** parameter allows the user to configure multiple endpoint forwarding policies with the same endpoint address value or multiple label-binding policies with the same binding label; providing the capability to achieve a 1:N backup strategy for the forwarding policy. Only the most preferred, lowest numerical preference value, policy is activated in data path as explained in *Policy Resolution and Operational Procedures*.
- Changes to the value of parameter **preference** requires a shutdown of the specific **forwarding-policy** context.

- A maximum of eight label-binding policies, with different preference values, are allowed for each unique value of the binding label.

  Label-binding policies with exactly the same value of the tuple {**binding label** | **preference**} are duplicate and their configuration is not allowed.

  The user cannot perform **no shutdown** on the duplicate policy.
- A maximum eight endpoint policies, with different preference values, are allowed for each unique value of the tuple {**endpoint**}.

  Endpoint policies with exactly the same value of the tuple {**endpoint**, **reference**} are duplicate and their configuration is not allowed.

  The user can not perform **no shutdown** on the duplicate policy.
- The **metric** parameter is supported with the endpoint policy only and is inherited by the routes which resolve their next hop to this policy.
- The **revert-timer** command configures the time to wait before switching back the resolution from the backup next hop to the restored primary next hop within a given NHG. By default, this timer is disabled meaning that the NHG will immediately revert to the primary next hop when it is restored.

  The revert timer is restarted each time the primary next hop flaps and comes back up again while the previous timer is still running. If the revert timer value is changed while the timer is running, it is restarted with the new value.
- The MPLS forwarding policy feature allows for a maximum of 32 NHGs consisting of, at most, one primary next hop and one backup next hop.
- The **next-hop** command allows the user to specify a direct next-hop address or an indirect next-hop address.
- A maximum of ten labels can be specified for a primary or backup direct next hop using the **pushed-labels** command. The label stack is programmed using a super-NHLFE directly on the outgoing interface of the direct primary or backup next hop.

  

  Note: This policy differs from the SR-TE LSP or SR policy implementation which can push a total of 11 labels due to the fact it uses a hierarchical NHLFE (super-NHLFE with maximum 10 labels pointing to the top SID NHLFE).

- The **resolution-type** {**direct**| **indirect**} command allows a limited validation at configuration time of the NHGs within a policy. The **no shutdown** command fails if any of these rules are not satisfied. The following are the rules of this validation:

  – NHGs within the same policy must be of the same resolution type.
  – A forwarding policy can have a single NHG of resolution type **indirect** with a primary next hop only or with both primary and backup next hops. An NHG with backup a next hop only is not allowed.
  – A forwarding policy will have one or more NHGs of resolution type **direct** with a primary next hop only or with both primary and backup next hops. An NHG with a backup next hop only is not allowed.
  – A check is performed to make sure the address value of the primary and backup next hop, within the same NHG, are not duplicates. No check is performed for duplicate primary or backup next-hop addresses across NHGs.
  – A maximum of 64,000 forwarding policies of any combination of label binding and endpoint types can be configured on the system.

- The IP address family of an endpoint policy is determined by the family of the **endpoint** parameter. It is populated in the TTMv4 or TTMv6 table accordingly. A label-binding policy does not have an IP address family associated with it and is programmed into the label (ILM) table.

  The following are the IP type combinations for the primary and backup next hops of the NHGs of a policy:

  – A primary or a backup indirect next hop with no pushed labels (label-binding policy) can be IPv4 or IPv6. A mix of both IP types is allowed within the same NHG.
  – A primary or backup direct next hop with no pushed labels (label-binding policy) can be IP types IPv4 or IPv6. A mix of both families is allowed within the same NHG.
  – A primary or a backup direct next hop with pushed labels (both endpoint and label binding policies) can be IP types IPv4 or IPv6. A mix of both families is allowed within the same NHG.

4.2.2 Policy Resolution and Operational Procedures

This section describes the validation of parameters performed at resolution time, as well as the details of the resolution and operational procedures.

- The following parameter validation is performed by the forwarding policy database at resolution time; meaning each time the policy is re-evaluated:

  – If the NHG primary or backup next hop resolves to a route whose type does not match the configured value in **resolution-type**, that next hop is made operationally "down".

    A DOWN reason code shows in the state of the next hop.
  – The primary and backup next hops of an NHG are looked up in the routing table. The lookups can match a direct next hop in the case of the direct resolution type and therefore the next hop can be part of the outgoing interface primary or secondary subnet. They can also match a static, IGP, or BGP route for an indirect resolution type, but only the set of IP next hops of the route are selected. Tunnel next hops are not selected and if they are the only next hops for the route, the NHG will be put in operationally "down" state.
  – The first 32, out of a maximum of 64, resolved IP next hops are selected for resolving the primary or backup next hop of a NHG of **resolution-type indirect**.
  – If the primary next hop is operationally "down", the NHG will use the backup next hop if it is UP. If both are operationally DOWN, the NHG is DOWN. See *Data Path Support* for details of the active path determination and the failover behavior.
  – If the binding label is not available, meaning it is either outside the range of the configured **reserved-label-block**, or is used by another MPLS forwarding policy or by another application, the label-binding policy is put operationally "down" and a retry mechanism will check the label availability in the background.

    A policy level DOWN reason code is added to alert users who may then choose to modify the binding label value.
  – No validation is performed for the pushed label stack of or a primary or backup next hop within a NHG or across NHGs. Users are responsible for validating their configuration.

- The forwarding policy database activates the best endpoint policy, among the named policies sharing the same value of the tuple {**endpoint**}, by selecting the lowest preference value policy. This policy is then programmed into the TTM and into the tunnel table in the data path.

  If this policy goes DOWN, the forwarding policy database performs a re-evaluation and activates the named policy with the next lowest preference value for the same tuple {**endpoint**}.

  If a more preferred policy comes back up, the forwarding policy database reverts to the more preferred policy and activates it.
- The forwarding policy database similarly activates the best label-binding policy, among the named policies sharing the same binding label, by selecting the lowest preference value policy. This policy is then programmed into the label FIB table in the data path as detailed in *Data Path Support*.

  If this policy goes DOWN, the forwarding policy database performs a re-evaluation and activates the named policy with the next lowest preference value for the same binding label value.

  If a more preferred policy comes back up, the forwarding policy database reverts to the more preferred policy and activates it.
- The active policy performs ECMP, weighted ECMP, or CBF over the active (primary or backup) next hops of the NHG entries.
- When used in the PCEP application, each LSP in a label-binding policy is reported separately by PCEP using the same binding label. The forwarding behavior on the node is the same whether the binding label of the policy is advertised in PCEP or not.
- A policy is considered UP when it is the best policy activated by the forwarding policy database and when at least one of its NHGs is operationally UP. A NHG of an active policy is considered UP when at least one of the primary or backup next hops is operationally UP.
- When the **config>router>mpls** or **config>router>mpls>forwarding-policies** context is set to **shutdown**, all forwarding policies are set to DOWN in the forwarding policy database and deprogrammed from IOM and data path.

  Prefixes which were being forwarded using the endpoint policies revert to the next preferred resolution type configured in the specific context (GRT, VPRN, or EVPN).
- When an NHG is set to **shutdown**, it is deprogrammed from the IOM and data path. Flows of prefixes which were being forwarded to this NHG are re-allocated to other NHGs based on the ECMP, Weighted ECMP, or CBF rules.
- When a policy is set to **shutdown**, it is deleted in the forwarding policy database and deprogrammed from the IOM and data path. Prefixes which were being forwarded using this policy will revert to the next preferred resolution type configured in the specific context (GRT, VPRN, or EVPN).
- The **no forwarding-policies** command deletes all policies from the forwarding policy database provided none of them are bound to any forwarding context (GRT, VPRN, or EVPN). Otherwise, the command fails.

4.3 Tunnel Table Handling of MPLS Forwarding Policy

An endpoint forwarding policy once validated as the most preferred policy for given endpoint address is added to the TTMv4 or TTMv6 according to the address family of the address of the **endpoint** parameter. A new owner of **mpls-fwd-policy** is used. A tunnel ID is allocated to each policy and is added into the TTM entry for the policy. For more information about the **mpls-fwd-policy** command, used to enable MPLS forwarding policy in different services, refer to the following guides:

- *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 2 Services and EVPN Guide: VLL, VPLS, PBB, and EVPN*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 3 Services Guide: IES and VPRN*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Router Configuration Guide*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Unicast Routing Protocols Guide*

The TTM preference value of a forwarding policy is configurable using the parameter **tunnel-table-pref**. The default value of this parameter is 255.

Each individual endpoint forwarding policy can also be assigned a preference value using the **preference** command with a default value of 255. When the forwarding policy database compares multiple forwarding policies with the same endpoint address, the policy with the lowest numerical preference value is activated and programmed into TTM. The TTM preference assigned to the policy is its own configured value in the **tunnel-table-pref** parameter.

If an active forwarding policy preference has the same value as another tunnel type for the same destination in TTM, then routes and services which are bound to both types of tunnels use the default TTM preference for the two tunnel types to select the tunnel to bind to as shown in *Table 21: Route Preferences* .

Table 21: Route Preferences

Route Preference	Value	Release Introduced
ROUTE_PREF_RIB_API	3	new in 16.0.R4 for RIB API IPv4 and IPv6 tunnel table entry
ROUTE_PREF_MPLS_FWD_POLICY	4	new in 16.0.R4 for MPLS forwarding policy of endpoint type
ROUTE_PREF_RSVP	7	—
ROUTE_PREF_SR_TE	8	new in 14.0
ROUTE_PREF_LDP	9	—
ROUTE_PREF_OSPF_TTM	10	new in 13.0.R1
ROUTE_PREF_ISIS_TTM	11	new in 13.0.R1
ROUTE_PREF_BGP_TTM	12	modified in 13.0.R1 (pref was 10 in R12)
ROUTE_PREF_UDP	254	introduced with 15.0 MPLS-over-UDP tunnels

Route Preference	Value	Release Introduced
ROUTE_PREF_GRE	255	—

An active endpoint forwarding policy populates the highest pushed label stack size among all its NHGs in the TTM. Each service and shortcut application on the router will use that value and perform a check of the resulting net label stack by counting all the additional labels required for forwarding the packet in that context.

This check is similar to the one performed for SR-TE LSP and SR policy features. If the check succeeds, the service is bound or the prefix is resolved to the forwarding policy. If the check fails, the service will not bind to this forwarding policy. Instead, it will bind to a tunnel of a different type if the user configured the use of other tunnel types. Otherwise, the service will go down. Similarly, the prefix will not get resolved to the forwarding policy and will either be resolved to another tunnel type or will become unresolved.

For more information about the **resolution-filter** CLI commands for resolving the next hop of prefixes in GRT, VPRN, and EVPN MPLS into an endpoint forwarding policy, refer to the following guides:

- *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 2 Services and EVPN Guide: VLL, VPLS, PBB, and EVPN*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Layer 3 Services Guide: IES and VPRN*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Router Configuration Guide*
- *7450 ESS, 7750 SR, 7950 XRS, and VSR Unicast Routing Protocols Guide*

BGP-LU routes can also have their next hop resolved to an endpoint forwarding policy.

4.4 Data Path Support

Note: The data path model for both the MPLS forwarding policy and the RIB API is the same. Unless explicitly stated, the selection of the active next hop within each NHG and the failover behavior within the same NHG or across NHGs is the same.

4.4.1 NHG of Resolution Type Indirect

Each NHG is modeled as a single NHLFE. The following are the specifics of the data path operation:

- Forwarding over the primary or backup next hop is modeled as a swap operation from the binding label to an implicit-null label over multiple outgoing interfaces (multiple NHLFEs) corresponding to the resolved next hops of the indirect route.
- Packets of flows are sprayed over the resolved next hops of an NHG with resolution of type indirect as a one-level ECMP spraying. See *Spraying of Packets in a MPLS Forwarding Policy* .
- An NHG of resolution type **indirect** uses a single NHLFE and does not support uniform failover. It will have CPM program only the active, the primary or backup, and the indirect next hop at any given point in time.
- Within a given NHG, the primary next hop is the preferred active path in the absence of any failure of the NHG of resolution type **indirect**.

- The forwarding database tracks the primary or backup next hop in the routing table. A **route delete** of the primary indirect next hop causes CPM to program the backup indirect next hop in the data path.

  A **route modify** of the indirect primary or backup next hop causes CPM to update the its resolved next hops and to update the data path if it is the active indirect next hop.
- When the primary indirect next hop is restored and is added back into the routing table, CPM waits for an amount of time equal to the user programmed revert-timer before updating the data path. However, if the backup indirect next hop fails while the timer is running, CPM updates the data path immediately.

4.4.2 NHG of Resolution Type Direct

The following rules are used for a NHG with a resolution type of **direct**:

- Each NHG is modeled as a pair of {primary, backup} NHLFEs. The following are the specifics of the label operation:
  - For a label-binding policy, forwarding over the primary or backup next hop is modeled as a swap operation from the binding label to the configured label stack or to an implicit-null label (if the **pushed-labels** command is not configured) over a single outgoing interface to the next hop.
  - For an endpoint policy, forwarding over the primary or backup next hop is modeled as a push operation from the binding label to the configured label stack or to an implicit-null label (if the **pushed-labels** command is not configured) over a single outgoing interface to the next hop.
  - The labels, configured by the **pushed-labels** command, are not validated.
- By default, packets of flows are sprayed over the set of NHGs with resolution of type **direct** as a one-level ECMP spraying. See *Spraying of Packets in a MPLS Forwarding Policy* .
- The user can enable weighted ECMP forwarding over the NHGs by configuring weight against all the NHGs of the policy. See *Spraying of Packets in a MPLS Forwarding Policy* .
- Within a given NHG, the primary next hop is the preferred active path in the absence of any failure of the NHG of resolution type direct.

  Note: The RIB API feature can change the active path away from the default. The gRPC client can issue a next-hop switch instruction to activate any of the primary or backup path at any time.

- The NHG supports uniform failover. The forwarding policy database assigns a Protect-Group ID (PG-ID) to each of the primary next hop and the backup next hop and programs both of them in the data path. A failure of the active path switches traffic to the other path following the uniform failover procedures as described in *Active Path Determination and Failover in a NHG of Resolution Type Direct*.
- The forwarding database tracks the primary or backup next hop in the routing table. A **route delete** of the primary or backup direct next hop causes CPM to send the corresponding PG-ID switch to the data path.

  A **route modify** of the direct primary or backup next hop causes CPM to update the MPLS forwarding database and to update the data path since both next hops are programmed.
- When the primary direct next hop is restored and is added back into the routing table, CPM waits for an amount of time equal to the user programmed **revert-timer** before activating it and updating the data path. However, if the backup direct next hop fails while the timer is running, CPM activates it and updates the data path immediately. The latter failover to the restored primary next hop is performed

using the uniform failover procedures as described in *Active Path Determination and Failover in a NHG of Resolution Type Direct*.

Note: RIB API does not support the revert timer. The gRPC client can issue a next-hop switch instruction to activate the restored primary next hop.

- CPM keeps track and updates the IOM for each NHG with the state of active or inactive of its primary and backup next hops following a failure event, a reversion to the primary next hop, or a successful next-hop switch request instruction (RIB API only).

4.4.2.1 Active Path Determination and Failover in a NHG of Resolution Type Direct

An NHG of resolution type **direct** supports uniform failover either within an NHG or across NHGs of the same policy. These uniform failover behaviors are mutually exclusive on a per-NHG basis depending on whether it has a single primary next hop or it has both a primary and backup next hops.

When an NHG has both a primary and a backup next hop, the forwarding policy database assigns a Protect-Group ID (PG-ID) to each and programs both in data path. The primary next hop is the preferred active path in the absence of any failure of the NHG.

During a failure affecting the active next hop, or the primary or backup next hop, CPM signals the corresponding PG-ID switch to the data path which then immediately begins using the NHLFE of the other next hop for flow packets mapped to NHGs of all forwarding polices which share the failed next hop.

An interface down event sent by CPM to the data path causes the data path to switch the PG-ID of all next hops associated with this interface and perform the uniform failover procedure for NHGs of all policies which share these PG-IDs.

Any subsequent network event causing a failure of the newly active next hop while the originally active next hop is still down, blackholes traffic of this NHG until CPM updates the policy to redirect the affected flows to the remaining NHGs of the forwarding policy.

When the NHG has only a primary next hop and it fails, CPM signals the corresponding PG-ID switch to the data path which then uses the uniform failover procedure to immediately re-assign the affected flows to the other NHGs of the policy.

A subsequent failure of the active next hop of a NHG the affected flow was re-assigned to in the first failure event, causes the data path to use the uniform failover procedure to immediately switch the flow to the other next hop within the same NHG.

Figure 41: NHG Failover Based on PG-ID Switch illustrates the failover behavior for the flow packets assigned to an NHG with both a primary and backup next hop and to an NHG with a single primary next hop.

The notation NHGi{Pi,Bi} refers to NHG "i" which consists of a primary next hop (Pi) and a backup next hop (Bi). When an NHG does not have a backup next hop, it is referred to as NHGi{Pi,Bi=null}.

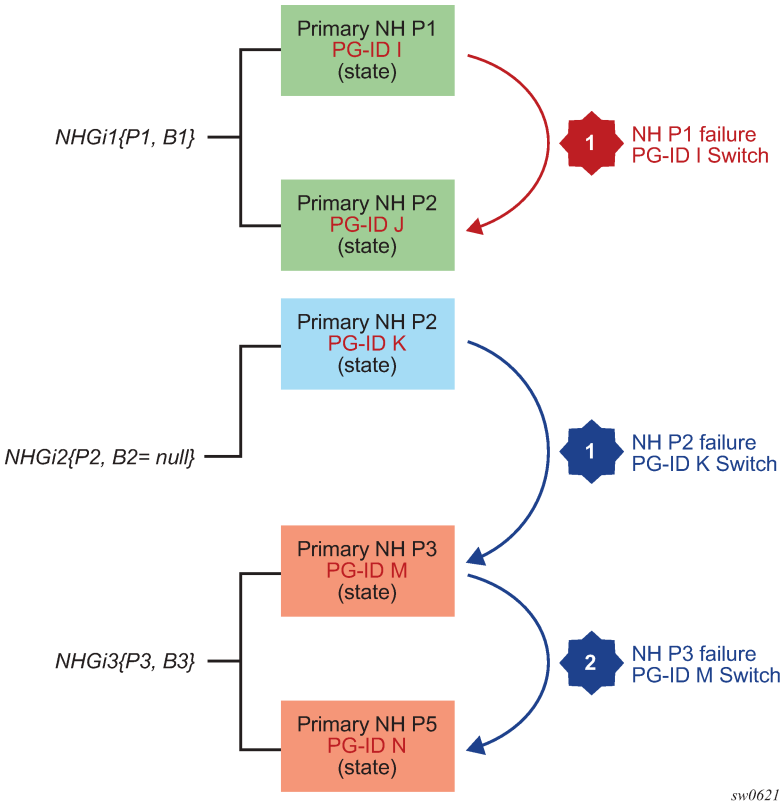

Figure 41: NHG Failover Based on PG-ID Switch

4.4.3 Spraying of Packets in a MPLS Forwarding Policy

When the node operates as an LER and forwards unlabeled packets over an endpoint policy, the spraying of packets over the multiple NHGs of type **direct** or over the resolved next hops of a single NHG of type **indirect** follows prior implementation. Refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR Interface Configuration Guide*.

When the node operates as an LSR, it forwards labeled packets matching the ILM of the binding label over the label-binding policy. An MPLS packet, including a MPLS-over-GRE packet, received over any network IP interface with a binding label in the label stack, is forwarded over the primary or backup next hop of either the single NHG of type **indirect** or of a selected NHG among multiple NHGs of type **direct**.

The router performs the following procedures when spraying labeled packets over the resolved next hops of a NHG of resolution type **indirect** or over multiple NHGs of type **direct**.

1. The router performs the GRE header processing as described in *MPLS-over-GRE termination* if the packet is MPLS-over-GRE encapsulated. Refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR Router Configuration Guide*.
2. The router then pops one or more labels and if there is a match with the ILM of a binding label, the router swaps the label to implicit-null label and forwards the packet to the outgoing interface. The

outgoing interface is selected from the set of primary or backup next hops of the active policy based on the LSR hash on the headers of the received MPLS packet.

a. The hash calculation follows the method in the user configuration of the command **lsr-load-balancing** {**lbl-only** | **lbl-ip** | **ip-only**} if the packet is MPLS-only encapsulated.
b. The hash calculation follows the method described in *LSR Hashing of MPLS-over-GRE Encapsulated Packet* if the packet is MPLS-over-GRE encapsulated. Refer to the *7450 ESS, 7750 SR, 7950 XRS, and VSR Interface Configuration Guide*.

4.4.4 Outgoing Packet Ethertype Setting and TTL Handling in Label Binding Policy

The following rules determine how the router sets the Ethertype field value of the outgoing packet:

- If the swapped label is not the Bottom-of-Stack label, the Ethertype is set to the MPLS value.
- If the swapped label is the Bottom-of-Stack label and the outgoing label is not implicit-null, the Ethertype is set to the MPLS value.
- If the swapped label is the Bottom-of-Stack label and the outgoing label is implicit-null, the Ethertype is set to the IPv4 or IPv6 value when the first nibble of the exposed IP packet is 4 or 6 respectively.

The router sets the TTL of the outgoing packet as follows:

- The TTL of a forwarded IP packet is set to MIN(MPLS_TTL-1, IP_TTL), where MPLS_TTL refers to the TTL in the outermost label in the popped stack and IP_TTL refers to the TTL in the exposed IP header.
- The TTL of a forwarded MPLS packet is set to MIN(MPLS_TTL-1, INNER_MPLS_TTL), where MPLS_TTL refers to the TTL in the outermost label in the popped stack and INNER_MPLS_TTL refers to the TTL in the exposed label.

4.4.5 Ethertype Setting and TTL Handling in Endpoint Policy

The router sets the Ethertype field value of the outgoing packet to the MPLS value.

The router checks and decrements the TTL field of the received IPv4 or IPv6 header and sets the TTL of all labels of the label stack specified in the **pushed-labels** command according to the following rules:

1. The router propagates the decremented TTL of the received IPv4 or IPv6 packet into all labels of the pushed label stack for a prefix in GRT.
2. The router then follows the configuration of the TTL propagation in the case of a IPv4 or IPv6 prefix forwarded in a VPRN context:

```
– config>router>ttl-propagate>vprn-local {none | vc-only | all}
– config>router>ttl-propagate>vprn-transit {none | vc-only | all}
– config>service>vprn>ttl-propagate>local {inherit | none | vc-only | all}
– config>service>vprn>ttl-propagate>transit {inherit | none | vc-only | all}
```

When a IPv6 packet in GRT is forwarded using an endpoint policy with an IPv4 endpoint, the IPv6 explicit null label is pushed first before the label stack specified in the **pushed-labels** command.

4.5 Weighted ECMP Enabling and Validation Rules

Weighted ECMP is supported within an endpoint or a label-binding policy when the NHGs are of resolution type **direct**. Weighted ECMP is not supported with an NHG of type **indirect**.

Weighted ECMP is performed on labeled or unlabeled packets forwarded over the set of NHGs in a forwarding policy when all NHG entries have a **load-balancing-weight** configured. If one or more NHGs have **no load-balancing-weight** configured, the spraying of packets over the set of NHGs reverts to plain ECMP.

Also, the **weighted-ecmp** command in GRT (**config>router>weighted-ecmp**) or in a VPRN instance (**config>service>vprn>weighted-ecmp**) are not required to enable the weighted ECMP forwarding in an MPLS forwarding policy. These commands are used when forwarding over multiple tunnels or LSPs. Weighted ECMP forwarding over the NHGs of a forwarding policy is strictly governed by the explicit configuration of a weight against each NHG.

The weighted ECMP normalized weight calculated for a NHG index causes the data path to program this index as many times as the normalized weight dictates for the purpose of spraying the packets.

4.6 Statistics

4.6.1 Ingress Statistics

The ingress statistics feature is associated with the binding label, that is the ILM of the forwarding policy, and provides aggregate packet and octet counters for packets matching the binding label.

The per-ILM statistic index for the MPLS forwarding policy features is assigned at the time the first instance of the policy is programmed in the data path. All instances of the same policy, for example, policies with the same binding-label, regardless of the **preference** parameter value, share the same statistic index.

The statistic index remains assigned as long as the policy exists and the **ingress-statistics** context is not shutdown. If the last instance of the policy is removed from the forwarding policy database, the CPM frees the statistic index and returns it to the pool.

If ingress statistics are not configured or are shutdown in a specific instance of the forwarding policy, identified by a unique value of pair {**binding-label**, **preference**} of the forwarding policy, an assigned statistic index is not incremented if that instance of the policy is activated

If a statistic index is not available at allocation time, the allocation fails and a retry mechanism will check the statistic index availability in the background.

4.6.2 Egress Statistics

Egress statistics are supported for both binding-label and endpoint MPLS forwarding policies; however, egress statistics are only supported in case where the next-hops configured within these policies are of resolution type **direct**. The counters are attached to the NHLFE of each next hop. Counters are effectively allocated by the system at the time the instance is programmed in the data-path. Counters are maintained

even if an instance is deprogrammed and values are not reset. If an instance is reprogrammed, traffic counting resumes at the point where it last stopped. Traffic counters are released and thus traffic statistics are lost when the instance is removed from the database when the egress statistic context is deleted, or when egress statistics are disabled (**egress-statistics shutdown**).

No retry mechanism is available for egress statistics. The system maintains a state per next-hop and per-instance regarding whether or not the allocation of statistic indices is successful. If the system is not able to allocate all the desired indices on a specified instance due to a lack of resources, the user should disable egress statistics on that instance, free the required number of statistics indices, and re-enable egress statistics on the desired entry. The selection of which other construct to release statistic indices from is beyond the scope of this document.

4.7 Configuring Static Label Routes using MPLS Forwarding Policy

4.7.1 Steering Flows to an Indirect Next-Hop

Figure 42: Traffic Steering to an Indirect Next-hop using a Static Label Route illustrates the traffic forwarding from a Virtual Network Function (VNF1) residing in a host in a Data Center (DC1) to VNF2 residing in a host in DC2 over the segment routing capable backbone network. DC1 and DC2 do not support segment routing and MPLS while the DC Edge routers do not support segment routing. Hence, MPLS packets of VNF1 flows are tunneled over a UDP/IP or GRE/IP tunnel and a static label route is configured on DC Edge1/2 to steer the decapsulated packets to the remote DC Edge3/4.

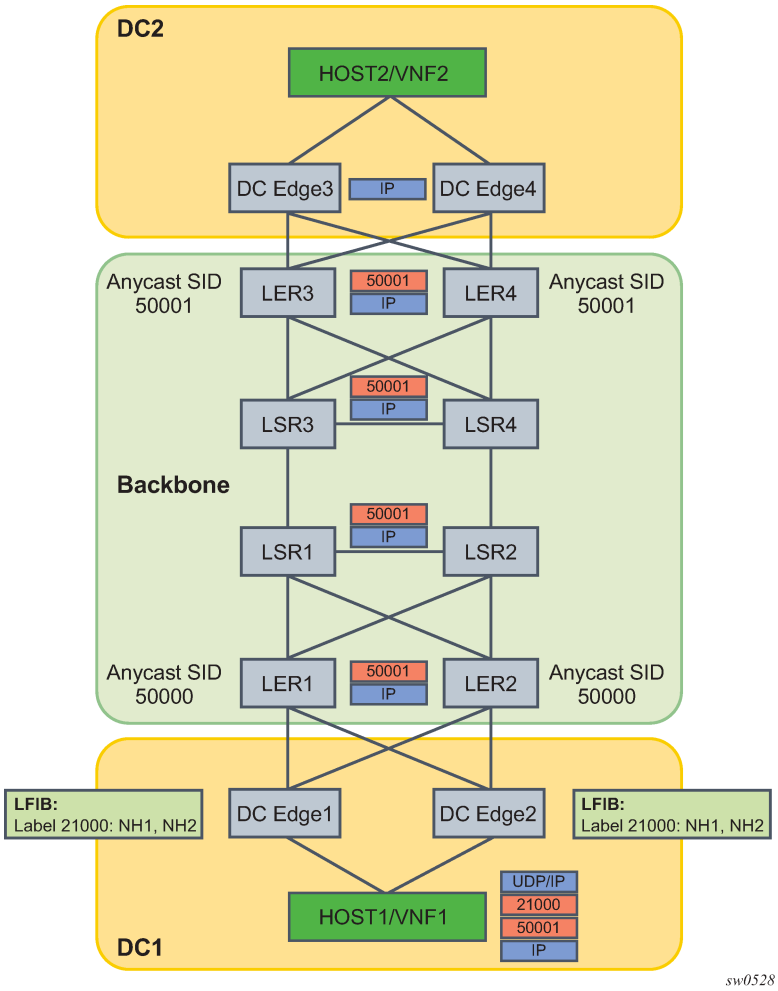

Figure 42: Traffic Steering to an Indirect Next-hop using a Static Label Route

The following are the data path manipulations of a packet across this network:

1. Host in DC1 pushes MPLS-over-UDP (or MPLS-over-GRE) header with outer IP destination address matching its local DC Edge1/2. It also pushes a static label 21000 which corresponds to the binding label of the MPLS forwarding policy configured in DC Edge1/2 to reach remote DC Edge3/4 (anycast address). The bottom of the label stack is the anycast SID for the remote LER3/4.
2. The label 21000 is configured on both DC Edge1 and DC Edge2 using a label-binding policy with an indirect next-hop pointing to the static route to the destination prefix of DC Edge3/4. The backup next-hop will point to the static route to reach some DC Edge5/6 in another remote DC (not shown).
3. There is EBGP peering between DC Edge1/2 and LER1/2 and between DC Edge3/4 and LER3/4.
4. DC Edge1/2 removes the UDP/IP header (or GRE/IP header) and swaps label 21000 to implicit-null and forwards (ECMP spraying) to all resolved next-hops of the static route of the primary or backup next-hop of the label-binding policy.
5. LER1/2 forwards based on the anycast SID to remote LER3/4.
6. LER3/4 removes the anycast SID label and forwards the inner IP packet to DC Edge3/4 which will then forward to Host2 in DC2.

The following CLI commands configure the static label route to achieve this use case. It creates a label-binding policy with a single NHG that is pointing to the first route as its primary indirect next-hop and the

second route as its backup indirect next-hop. The primary static route corresponds to a prefix of remote DC Edge3/4 router and the backup static route to the prefix of a pair of edge routers in a different remote DC. The policy is applied to routers DC Edge1/2 in DC1.

```
– config>router
    – static-route-entry fd84:a32e:1761:1888::1/128
        – next-hop 3ffe::e0e:e05
            – no shutdown
        – next-hop 3ffe::f0f:f01
            – no shutdown
    – static-route-entry fd22:9501:806c:2387::2/128
        – next-hop 3ffe::1010:1002
            – no shutdown
        – next-hop 3ffe::1010:1005
            – no shutdown
–
– config>router>mpls-labels
    – reserved-label-block static-label-route-lbl-block
        – start-label 20000 end-label 25000
–
– config>router>mpls
    – forwarding-policies
        – reserved-label-block static-label-route-lbl-block
        – forwarding-policy static-label-route-indirect
            – binding-label 21000
            – revert-timer 5
            – next-hop-group 1 resolution-type indirect
                – primary-next-hop
                    – next-hop fd84:a32e:1761:1888::1
                – backup-next-hop
                    – next-hop fd22:9501:806c:2387::2
```

4.7.2 Steering Flows to a Direct Next-Hop

Figure 43: Traffic Steering to a Direct Next-hop using a Static Label Route illustrates the traffic forwarding from a Virtual Network Function (VNF1) residing in a host in a Data Centre (DC1) to outside of the customer network via the remote peering Point Of Presence (POP1).

The traffic is forwarded over a segment routing capable backbone. DC1 and POP1 do not support segment routing and MPLS while the DC Edge routers do not support segment routing. Hence, MPLS packets of VNF1 flows are tunneled over a UDP/IP or GRE/IP tunnel and a static label route is configured on POP Edge3/4 to steer the decapsulated packets to the desired external BGP peer.

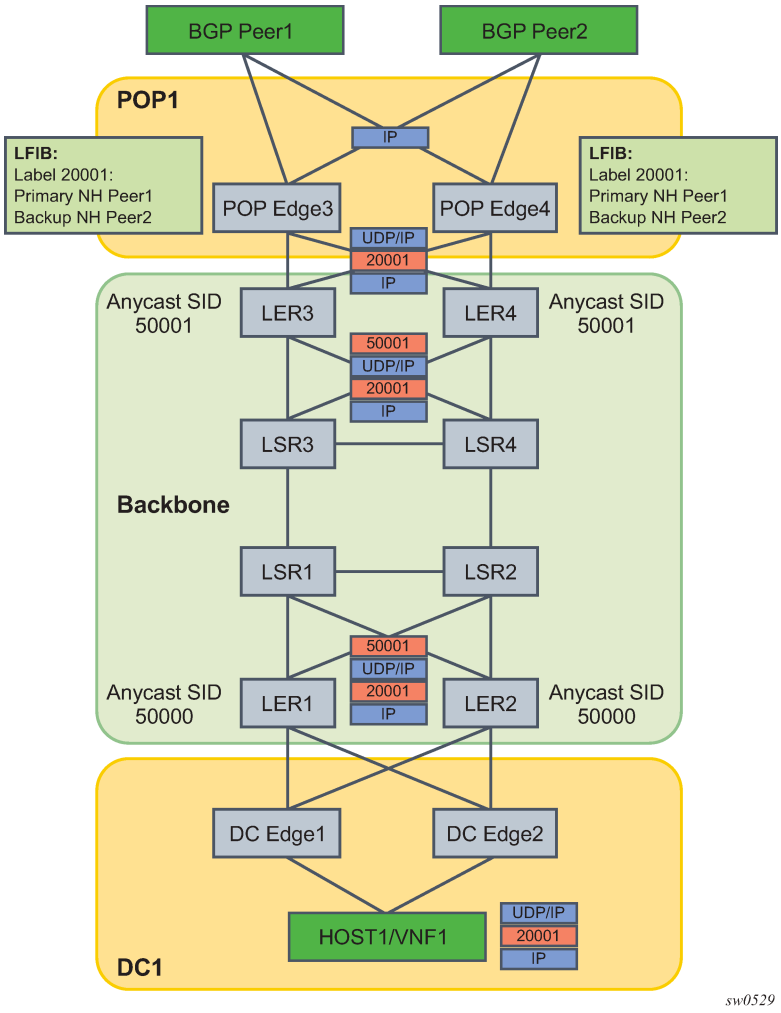

Figure 43: Traffic Steering to a Direct Next-hop using a Static Label Route

The intent is to override the BGP routing table at the peering routers (POP Edge3 and Edge4) and force packets of a flow originated in VNF1 to exit the network using a primary external BGP peer Peer1 and a backup external BGP peer Peer2, if Peer1 is down. This application is also referred to as Egress Peer Engineering (EPE).

The following are the data path manipulations of a packet across this network:

1. DC Edge1/2 receives a MPLS-over-UDP (or a MPLS-over-GRE) encapsulated packet from the host in the DC with the outer IP destination address set to the remote POP Edge3/4 routers in peering POP1 (anycast address). The host also pushes the static label 20001 for the remote external BGP Peer1 it wants to send to.
2. This label 20001 is configured on POP Edge3/4 using the MPLS forwarding policy feature with primary next-hop of Peer1 and backup next-hop of Peer2.
3. There is EBGP peering between DC Edge1/2 and LER1/2, and between POP Edge3/4 and LER3/4, and between POP Edge3/4 and Peer1/2.
4. LER1/LER2 pushes the anycast SID of remote LER3/4 as part of the BGP route resolution to a SR-ISIS tunnel or SR-TE policy.
5. LER3/4 removes the anycast SID and forwards the GRE packet to POP Edge3/4.

6. POP Edge3/4 removes UDP/IP (or GRE/IP) header and swaps the static label 20001 to implicit null and forwards to Peer1 (primary next-hop) or to Peer2 (backup next-hop).

The following CLI commands configure the static label route to achieve this use case. It creates a label-binding policy with a single NHG containing a primary and backup direct next-hops and is applied to peering routers POP Edge3/4.

```
—
— config>router>mpls-labels
    — reserved-label-block static-label-route-lbl-block
        — start-label 20000 end-label 25000
—
— config>router>mpls
    — forwarding-policies
        — forwarding-policy static-label-route-direct
            — binding-label 20001
            — revert-timer 10
            — next-hop-group 1 resolution-type direct
                — primary-next-hop
                    — next-hop fd84:a32e:1761:1888::1
                — backup-next-hop
                    — next-hop fd22:9501:806c:2387::2
```

5 gRPC-based RIB API

5.1 RIB/FIB API Overview

Each router stores information about how to forward IP and MPLS packets in a set of RIB (routing information base) and FIB (forwarding information base) tables. These tables are conventionally populated by the management plane of the router (static entries) and by control plane protocols such as BGP, OSPF, ISIS, RSVP, LDP or segment routing.

In some SDN (Software Defined Networking) use cases, it may be useful to augment the RIB/FIB state held by a router to also include forwarding entries programmed by an external controller. SR OS supports a powerful and flexible gRPC-based RIB API service for this purpose. Using gRPC between a controller (gRPC client) and a router (gRPC server) has many benefits:

- gRPC is open source, with broad industry support and a rich ecosystem of tools and applications.
- gRPC is fast and efficient. The combination of HTTP/2 and protocol buffers ensures that a minimum number of bytes are sent across the wire as part of an RPC.
- gRPC is supported by many languages and platforms, including C, C++, C#, Go, Java, Node.js, Python, and Ruby.

To build a gRPC client that implements the RIB API service you must obtain, from Nokia, the protobuf definition file for the Nokia SR OS RIB API service. This protobuf file defines a RIB API service and its supported RPCs. The RIB API service supports one bidirectional streaming RPC called *Modify* and one unary RPC called GetVersion. The *Modify* RPC is used to add, delete or replace entries in any of the following RIB/FIB tables:

- IPv4 route table of the base router
- IPv6 route table of the base router
- IPv4 tunnel table
- IPv6 tunnel table
- MPLS label forwarding table

The *GetVersion* RPC allows the client to request the overall RIB API version and the individual RIB table versions supported by the router.

For maximum programming flexibility and speed, the entries added by the RIB-API service are not processed or stored as configuration data; they are provided directly to the control plane and modeled as though learned from a pseudo-protocol. RIB-API entries have the same persistence characteristics as protocol routes: if a router (gRPC server) detects that a gRPC client has disconnected or terminated its RPC, or if the router reboots, dependent RIB API entries are removed and must be re-programmed if persistence is required.

A gRPC client cannot delete entries it does not own, including routes from other protocols, but it can supersede routes from other sources through appropriate programming of preference values.

A gRPC client can read RIB/FIB entries programmed using the RIB API service (by any client) and obtain other state information that it needs using the gNMI management interface. gNMI is another gRPC-

based service supported by the router and it supports RPCs for configuration, one-time state retrieval and telemetry state subscriptions. The same client can have active gNMI and RIB API RPCs with the same target router and at the same time using the same TCP connection.

5.2 RIB/FIB API Fundamentals

The *Modify* RPC allows a gRPC client to add, modify or delete RIB/FIB entries. To accomplish this, the client sends a stream of *ModifyRequest* messages to the server (router) and receives, in return, a stream of *ModifyResponse* messages. Each *ModifyRequest* message can include multiple *Request* messages. Each *Request* message has a 64-bit ID (used to pair it with a *Response* message) and conveys one of the following instructions:

- A request to **add** an entry to one of the five supported RIB/FIB tables. The add operation requires the client to specify values for all parameters of the route, tunnel or label entry being programmed. If the add operation is successful, the RIB/FIB entry is considered owned by the client that carried out the transaction.
- A request to **replace** an entry in one of the five supported RIB/FIB tables. This operation completely replaces a RIB/FIB entry that was previously programmed by the same client. All the parameters of the new route, tunnel or label entry must be specified, even values that did not change from the previous entry.
- A request to **delete** an entry in one of the five supported RIB/FIB tables. This operation deletes a RIB/FIB entry that was previously programmed by the same client. The delete operation requires only the key values of the entry that should be deleted.
- An **end-of-rib** marker for one of the five supported RIB/FIB tables. This operation is used to accelerate the removal of stale entries associated with a RIB/FIB table, rather than waiting for purge timers to expire. Additional details are discussed in this chapter.
- A **next-hop-switch** instruction. The client sends this request in order to manually activate the primary or backup next hop associated with a specific next-hop-group of a specific tunnel or label entry. This might be done to facilitate a maintenance action or to manually revert traffic back to a primary next hop after it recovers from a failure that diverted traffic to a backup next hop.

The following general points should be noted:

- The router does not support multiple RIB API RPC sessions with the same client IP. If a client machine has multiple independent controllers, they need to interact with the router using different IP addresses.
- It is up to the client to choose a unique 64-bit identifier for each *Request* transaction, but Request IDs must increase throughout the lifetime of the RPC session.
- If a gRPC client omits any parameter that is considered mandatory by the server side, the router assumes that the intended value for the parameter is zero (0). This may cause an error if the zero value is invalid or unavailable.
- A status code of OK sent by the router to a client (in a *ModifyResponse* message) only indicates that the request was valid. In the case of an add/delete/modify operation, it does not mean that the FIB was actually modified and in the case of a next-hop-switchover it does not mean that the switchover has actually occurred.
- A RIB/FIB entry programmed by a gRPC client may be unusable because none of its next hops are resolvable or the requested label resources are not available. The entry is still accepted by the router and acknowledged with an OK status code response to the client. If at a later time the entry becomes usable, it is activated by the router automatically.

5.2.1 RIB/FIB API Entry Persistence

All states created by the RIB API service are ephemeral. In other words, when the router reboots, none of the API-programmed entries are preserved. The necessary entries must be reprogrammed by a gRPC client in the same way that BGP routes must be relearned from BGP peers after a reboot.

The persistence of a programmed RIB/FIB entry also depends on the liveness of the RPC session with the client that owns the entry, and this in turn depends on the liveness of the underlying TCP connection. If the TCP connection with a client goes down (due to link and/or router failures, client failure, or CPM switchover by the router) the router starts a purge timer for all affected clients and marks their owned RIB/FIB entries as stale. When a client's purge timer expires all of its stale entries are removed. While a purge timer is running, the associated stale entries remain valid and usable for forwarding but are less preferred than any non-stale entry. The purge timer gives an opportunity for the disconnected client or some other client to re-program the necessary RIB/FIB tables so that forwarding continues uninterrupted.

Detection of TCP connection failures by the router (gRPC server) can be assisted by enabling TCP **keepalive** on the gRPC TCP connections. When it is enabled, TCP keepalive messages are sent to all gRPC clients, regardless of the RPCs they support (gNMI or RIB API).

On the router, TCP **keepalive** is configured by specifying 3 parameters: **idle-time**, **interval**, and **retries**. These parameters are configured in the **config>system>grpc>tcp-keepalive** context. The sending of TCP keepalives starts when the connection has been idle (no TCP segments sent or received) for more than **idle-time** seconds. At that point the router sends a probe (TCP ACK with a sequence number = current sequence number - 1) and expects a TCP ACK. It repeats this probe every **interval** seconds for the configured number of **retries** and if no response is received to any of them the connection is immediately closed, starting the purge timer if the TCP connection is supporting a Modify RPC.

When a client is done programming all entries in a particular RIB/FIB table it can optionally send an **end-of-rib** request for that table in order to immediately remove all stale RIB entries associated with that table, regardless of the owner client IP.

5.3 RIB/FIB API Configuration Overview

Configuration related to the RIB/FIB API service on the router is spread across two general areas:

- system-level GRPC configuration (**config>system>grpc** or **config>system>security>profile>grpc**)
- routing instance configuration (**config>router** or **config>service>vprn**)

To enable the router to receive and process RIB API requests from a client perform the following steps.

Procedure

1. The RIB API service must be enabled at the gRPC system level: **config>system>grpc>rib-api>no shutdown**.
2. Optionally, a non-zero purge-timeout can be configured: **config>system>grpc>rib-api>purge-timeout**.
   The purge-timeout applies to all gRPC clients participating in the RIB API service.
3. Optionally, the sending of TCP keepalives can be enabled towards all gRPC clients by configuring values under the **config>system>grpc>tcp-keepalive** context.

4. One or more gRPC user accounts should be created, and these user accounts should be attached to a profile that authorizes the *GetVersion* and *Modify* RPCs associated with the RIB API service. Clients need to send a valid username and password when initiating any RPC.

5. Nokia recommends using TLS-based encryption between the client and server. This involves associating a **tls-server-profile** with the gPRC server. For more information, refer to the 7450 ESS, 7750 SR, 7950 XRS, and VSR System Management Guide.

6. If you want to use the RIB API service to program MPLS label entries then a **reserved-label-block** must be configured using the **config>router>rib-api>mpls>reserved-label-block** command and MPLS programming functionality must be enabled using the **config>router>rib-api>mpls>no shutdown** command.

To enable the router to use RIB API tunnel entries for resolving certain types of static and BGP routes, additional configuration is needed. For more information, refer to the 7450 ESS, 7750 SR, 7950 XRS, and VSR Unicast Routing Protocols Guide.

5.4 RIB/FIB API - IPv4 Route Table Programming

The RIB API service proto definition requires the client to provide values for all of the parameters listed in *Table 22: IPv4 Route Table Programming* when performing an **add** or **replace** of an IPv4 route. When performing a **delete** operation, only the bolded parameters (the lookup keys) are required. *Table 22: IPv4 Route Table Programming* describes the meaning of each parameter and its valid range.

Table 22: IPv4 Route Table Programming

Parameter	Type	Description
prefix	string	IPv4 prefix and prefix-length in CIDR format
preferences	uint32 (0-65535)	RIB API preference, used to compare one RIB API entry to another one; the lowest preference wins
rtm_preference	uint32 (0-255)	RTM preference, used to compare RIB API entry to other routes in RTM; the lowest preference wins
metric	uint32 (0-16777215)	Route cost/metric
tunnel_next_hop	string	A remote IPv4 address that must correspond to an API-programmed IPv4 tunnel

The router's RIB API database can hold up to eight different gPRC-programmed entries per IPv4 prefix. Typically N entries would be associated with N different gRPC clients although the same client can program multiple entries for the same prefix as long as the preference values are unique.

When an IPv4 route entry is successfully added or modified in the RIB API database, the router assesses whether the entry is valid or invalid and constantly re-evaluates this status. The entry is invalid if its next hop cannot be resolved to a gRPC-programmed IPv4 tunnel that is up.

If the entry is valid, the router compares it to all other valid API-programmed entries for the same IPv4 prefix. The router chooses any non-stale entry over a stale entry, then the entry with the lowest preference value, and then if there is a tie, the lowest metric, and then if there is a still a tie, the entry from the client with the lowest 128-bit IP address (an IPv4 address is encoded in the lower 32 bits).

If the entry is valid and the best relative to other RIB API entries then it is submitted to the route table manager. This software task compares the API route to all other non-API routes it has for the same IPv4 prefix. The router chooses the entry with the lowest RTM preference value, and then if there is a tie, the lowest metric, and then if there is still a tie, the entry submitted by the protocol with the lowest default preference.

If the route table manager selects the API route as the best route it is sent to the FIB manager for programming into the data path.

5.5 RIB/FIB API - IPv6 Route Table Programming

The RIB API service proto definition requires the client to provide values for all of the parameters listed in *Table 23: IPv6 Route Table Programming* when performing an **add** or **replace** of an IPv6 route. When performing a **delete** operation, only the bolded parameters (the lookup keys) are required. *Table 23: IPv6 Route Table Programming* describes the meaning of each parameter and its valid range.

Table 23: IPv6 Route Table Programming

Parameter	Type	Description
prefix	string	IPv6 prefix and prefix-length in CIDR format
preferences	uint32 (0-65535)	RIB API preference, used to compare one RIB API entry to another one; the lowest preference wins
rtm_preference	uint32 (0-255)	RTM preference, used to compare RIB API entry to other routes in RTM; the lowest preference wins
metric	uint32 (0-16777215)	Route cost/metric

Parameter	Type	Description
tunnel_next_hop	string	A remote IPv6 address that must correspond to an API-programmed IPv6 tunnel

The router's RIB API database can hold up to eight different gPRC-programmed entries per IPv6 prefix. Typically N entries would be associated with N different gRPC clients although the same client can program multiple entries for the same prefix as long as the preference values are unique.

When an IPv6 route entry is successfully added or modified in the RIB API database, the router assesses whether the entry is valid or invalid and constantly re-evaluates this status. The entry is invalid if its next hop cannot be resolved to a gRPC-programmed IPv6 tunnel that is up.

If the entry is valid, the router compares it to all other valid API-programmed entries for the same IPv6 prefix. The router chooses any non-stale entry over a stale entry, then the entry with the lowest preference value, and then if there is a tie, the lowest metric, and then if there is a still a tie, the entry from the client with the lowest 128-bit IP address (an IPv4 address is encoded in the lower 32 bits).

If the entry is valid and the best relative to other RIB API entries then it is submitted to the route table manager. This software task compares the API route to all other non-API routes it has for the same IPv6 prefix. The router chooses the entry with the lowest RTM preference value, and then if there is a tie, the lowest metric, and then if there is still a tie, the entry submitted by the protocol with the lowest default preference.

If the route table manager selects the API route as the best route it is sent to the FIB manager for programming into the data path.

5.6 RIB/FIB API - IPv4 Tunnel Table Programming

The RIB API service proto definition requires the client to provide values for all of the parameters listed in *Table 24: IPv4 Tunnel Table Programming* when performing an **add** or **replace** of an IPv4 MPLS tunnel. When performing a **delete** operation, only the bolded parameters (the lookup keys) are required. *Table 24: IPv4 Tunnel Table Programming* describes the meaning of each parameter and its valid range.

Table 24: IPv4 Tunnel Table Programming

Parameter	Type	Description
prefix	string	IPv4 host address
preferences	uint32 (0-65535)	RIB API preference, used to compare one RIB API entry to another one; the lowest preference wins

Parameter	Type	Description
ttm_preference	uint32 (0-255)	TTM preference, used in the programming of the tunnel in TTM
metric	uint32 (0-16777215)	Route cost/metric
next-hop-group[id]	list	A list of next-hop groups
id	uint32 (1-32)	Unique identifier of the next-hop group. Selected by the client
weight	uint32	Weight assigned to the next-hop-group when weighted ECMP is desired between next-hop-groups
primary	—	Mandatory
ip_address	string	IPv4 or IPv6 address on a local subnet; can be a secondary address
pushed_label_stack	list of uint32	A list of one or more MPLS labels, up to ten MPLS labels
backup	—	Optional
ip_address	string	IPv4 or IPv6 address on a local subnet; can be a secondary address
pushed_label_stack	list of uint32	A list of up to ten MPLS labels
egress-statistics	—	—
enable	boolean	Indicates whether statistics collection is enabled for this entry

The router's RIB API database can hold up to eight different gPRC-programmed entries per IPv4 tunnel endpoint. Typically N entries would be associated with N different gRPC clients although the same client can program multiple entries for the same tunnel endpoint as long as the preference values are unique

When an IPv4 tunnel endpoint entry is successfully added or modified in the RIB API database, the router assesses whether the entry is valid or invalid and constantly re-evaluates this status. The tunnel is invalid if none of its primary next hops can be resolved to an interface that is up or if MPLS programming using the RIB API is currently administratively disabled.

If the IPv4 tunnel entry is valid the router compares it to all other valid API-programmed entries for the same IPv4 endpoint address. The router chooses any non-stale entry over a stale entry, then the entry with the lowest preference value, and then if there is a tie, the lowest metric, and then if there is a still a tie, the entry from the client with the lowest 128-bit IP address (an IPv4 address is encoded in the lower 32 bits).

If the entry is valid and the best relative to other RIB API entries then it is programmed into the FIB and added to the base router IPv4 tunnel table. The tunnel entry is now active and can be used to resolve the next hops of other routes. For more information, refer to the 7450 ESS, 7750 SR, 7950 XRS, and VSR Unicast Routing Protocols Guide.

5.7 RIB/FIB API - IPv6 Tunnel Table Programming

The RIB API service proto definition requires the client to provide values for all of the parameters listed in *Table 25: IPv6 Tunnel Table Programming* when performing an **add** or **replace** of an IPv6 MPLS tunnel. When performing a **delete** operation, only the bolded parameters (the lookup keys) are required. *Table 25: IPv6 Tunnel Table Programming* describes the meaning of each parameter and its valid range.

Table 25: IPv6 Tunnel Table Programming

Parameter	Type	Description
prefix	string	IPv6 host address
preferences	uint32 (0-65535)	RIB API preference, used to compare one RIB API entry to another one; the lowest preference wins
ttm_preference	uint32 (0-255)	TTM preference, used in the programming of the tunnel in TTM
metric	uint32 (0-16777215)	Route cost/metric
next-hop-group[id]	list	A list of next-hop groups
id	uint32 (1-32)	Unique identifier of the next-hop group; selected by the client

Parameter	Type	Description
weight	uint32	Weight assigned to the next-hop-group when weighted ECMP is desired between next-hop-groups
primary	—	Mandatory
ip_address	string	IPv4 or IPv6 address on a local subnet; can be a secondary address.
pushed_label_stack	list of uint32	A list of one or more MPLS labels, up to ten MPLS labels
backup	—	Optional
ip_address	string	IPv4 or IPv6 address on a local subnet; can be a secondary address
pushed_label_stack	list of uint32	A list of up to ten MPLS labels
egress-statistics	—	—
enable	boolean	Indicates whether statistics collection is enabled for this entry

The router's RIB API database can hold up to eight different gPRC-programmed entries per IPv6 tunnel endpoint. Typically N entries would be associated with N different gRPC clients although the same client can program multiple entries for the same tunnel endpoint as long as the preference values are unique.

When an IPv6 tunnel endpoint entry is successfully added or modified in the RIB API database, the router assesses whether the entry is valid or invalid and constantly re-evaluates this status. The tunnel is invalid if none of its primary next hops can be resolved to an interface that is up or if MPLS programming using the RIB API is currently administratively disabled.

If the IPv6 tunnel entry is valid the router compares it to all other valid API-programmed entries for the same IPv6 endpoint address. The router chooses any non-stale entry over a stale entry, then the entry with the lowest preference value, and then if there is a tie, the lowest metric, and then if there is a still a tie, the entry from the client with the lowest 128-bit IP address (an IPv4 address is encoded in the lower 32 bits).

If the entry is valid and the best relative to other RIB API entries then it is programmed into the FIB and added to the base router IPv6 tunnel table. The tunnel entry is now active and can be used to resolve the next hops of other routes. For more information, refer to the 7450 ESS, 7750 SR, 7950 XRS, and VSR Unicast Routing Protocols Guide.

5.8 RIB/FIB API - MPLS LFIB Programming

The RIB API service proto definition requires the client to provide values for all of the parameters listed in *Table 26: MPLS LFIB Programming* when performing an **add** or **replace** of an MPLS LFIB entry. When performing a **delete** operation, only the bolded parameters (the lookup keys) are required. *Table 26: MPLS LFIB Programming* describes the meaning of each parameter and its valid range.

Table 26: MPLS LFIB Programming

Parameter	Type	Description
prefix	string	Incoming label value
preferences	uint32 (0-65535)	RIB API preference, used to compare one RIB API entry to another one; the lowest preference wins
next-hop-group[id]	list	A list of next-hop groups; required for a SWAP operation; omitted when the operation is a POP
id	uint32 (1-32)	Unique identifier of the next-hop group; selected by the client
weight	uint32	Weight assigned to the next-hop-group when weighted ECMP is desired between next-hop-groups
primary	—	Mandatory
ip_address	string	IPv4 or IPv6 address on a local subnet; can be a secondary address
pushed_label_stack	list of uint32	A list of zero or more MPLS labels, up to ten MPLS labels
backup	—	Optional
ip_address	string	IPv4 or IPv6 address on a local subnet; can be a secondary address

Parameter	Type	Description
pushed_label_stack	list of uint32	A list of up to ten MPLS labels
ingress-statistics	—	—
enable	bool	—
type	enum 0, 1, 2	INVALID = 0 POP = 1 SWAP = 2
egress-statistics	—	—
enable	boolean	Indicates whether statistics collection is to be enabled for this entry

The router's RIB API database can hold up to eight different gPRC-programmed entries per MPLS label value. Typically N entries would be associated with N different gRPC clients although the same client can program multiple entries for the same label value as long as the preference values are unique.

When an MPLS label entry is successfully added or modified in the RIB API database, the router assesses whether the entry is valid or invalid and constantly re-evaluates this status. The label entry is invalid if it is a SWAP operation and none of the primary next hops can be resolved to an interface that is up or if MPLS programming using the RIB API is administratively disabled or if the requested incoming label has already been allocated to another owner sharing the same reserved label block or if the requested incoming label is outside the reserved label block range.

If the label entry is valid the router compares it to all other valid API-programmed entries for the same label value. The router chooses any non-stale entry over a stale entry, then the entry with the lowest preference value, and then if there is a tie, the entry from the client with the lowest 128-bit IP address (an IPv4 address is encoded in the lower 32 bits).

If the label entry is valid and best relative to other RIB API entries then it is programmed into the forwarding plane.

5.9 RIB/FIB API - Using Next-Hop-Groups, Primary Next Hops, and Backup Next Hops

The RIB API service proto definition allows each MPLS tunnel and each MPLS label entry to have multiple next-hop-groups, each with a primary next hop and optionally one backup next hop. When a tunnel or label entry has more than one next-hop-group, this instructs the router to spray matching traffic across the next-hop-groups based on an ECMP or weighted-ECMP algorithm.

At any time, traffic hashed to a particular next-hop-group uses only the primary or backup next hop for forwarding. The selection of the active next hop within each next-hop-group is influenced by failures and by **next-hop-switch** *Request* messages made by the owner gRPC client. The specific rules are:

- If the primary next hop is resolved to an up interface when the next-hop-group is initially activated then it immediately becomes the active next hop.
- If the primary next hop is unresolved when the next-hop-group is initially activated then no next hop is immediately activated (even if the backup next hop is up) and a fixed wait-timer is started (three seconds). If the primary next hop comes up during that timer window then it is immediately activated. If the timer runs out and the primary has not yet come up the backup next hop is activated and stays active even if the primary comes up a short while later, after the timer expired.
- If the currently active next hop fails, the system automatically activates the other next hop.
- If the system receives a **next-hop-switch** *Request* targeting this specific entry and next-hop-group then the next hop indicated in the *Request* message is immediately activated, as long as it is up. If the requested next hop is down the message is ignored.

Note: The router returns a status of OK in response to a **next-hop-switch** *Request*as long as the key values identify a next-hop-group that exists for a tunnel or label entry owned by the gRPC client, even if the desired next hop is not activated.

5.10 RIB/FIB API - State and Telemetry

A gRPC client can use the gRPC gNMI service (Get RPC, Subscribe RPC) to retrieve state information from the router that can help it make better programming decisions. All states maintained by the router (and exposed to model-driven management interfaces) are available to the gRPC client.

RIB/FIB API also introduces additional YANG state models that are complementary to the programming actions. This new state is available through the following YANG paths:

- `state/router/route-fib`
- `state/router/tunnel-fib`
- `state/router/label-fib`
- `state/router/rib-api/route`
- `state/router/rib-api/tunnel`
- `state/router/rib-api/label`

The corresponding **show** commands are also provided for reference.

- **show router fib-telemetry route**
- **show router fib-telemetry tunnel**
- **show router fib-telemetry label**
- **show router rib-api route**
- **show router rib-api tunnel**
- **show router rib-api label**

The state information represented by the `state/router/route-fib` and `state/router/tunnel-fib` paths, and the **show router fib-telemetry route** and **show router fib-telemetry tunnel** show commands list are not collected by default, since it requires additional processing. In order for this state

to be collected you must configure the **configure router fib-telemetry** command. If this command is not configured then these states are not collected at all, and telemetry subscriptions are not supported for any of the following paths:

- `/state/router/route-fib`
- `/state/router/tunnel-fib`
- `/state/router/label-fib`

It is not possible for a single telemetry subscription to include any of these three paths in addition to other state paths outside of this tree. This is because of the potential volume of information in the tables described in this chapter.

For gNMI telemetry subscriptions, the following restrictions should be noted:

- If a route, tunnel or MPLS label entry is modified, and it covered by an ON-CHANGE subscription to a state path enabled by **config>router>fib-telemetry**, the update replays the current values of the entire entry (except for statistics), including values did not change from the last update. It is up to the client to compare the update to the previous one received if it needs to know the exact properties that changed.
- Subscriptions to list keys of state paths enabled by **fib-telemetry** are not supported.

5.11 Traffic Statistics

A gRPC client can make a request for traffic statistics to be collected. Both ingress and egress statistics are available but not all types of entries support both.

Traffic statistics are expressed in number of packets and in octets and are provided without forwarding-class or QoS profile distinction.

The system provides capabilities to display or show, clear, or monitor statistics.

5.11.1 Ingress statistics

Only RIB-API MPLS tunnel table entries support ingress statistics. The counters are attached to the ILM entry that is formed when the RIB-API entry is programmed. When different RIB-API entries use the same ILM or label, then the traffic statistics for these RIB-API entries are identical. Traffic counters are kept until the ILM entry is removed. Due to a lack of resources, the system may not be able to allocate counters (statistic indices) to an ILM. In this case, the system automatically retries until it succeeds.

5.11.2 Egress statistics

Egress statistics are supported for the three RIB-API tunnel tables (IPv4, IPv6, and MPLS). The counters are attached to the NHLFE of each next hop. Counters are effectively allocated by the system at the time that the instance is programmed in the data-path. Counters are maintained even if an instance is deprogrammed and values are not reset. This means that, if an instance is reprogrammed, traffic counting resumes at the point where it last stopped. Traffic counters are released and thus traffic statistics are lost when the instance or entry is removed from the database.

No retry mechanism is available for egress statistics. The system maintains a state per next-hop and per instance identifying whether or not allocation of statistic indices is successful. If the system is not able to

allocate all the desired indices on a specified instance due to a lack of resources, then the user should disable egress statistics on that instance. This action frees enough statistic indices and re-enables egress statistics on the desired entry. The selection of which other construct to release statistic indices from is beyond the scope of this document.

6 Path Computation Element Protocol (PCEP)

6.1 Introduction to the Path Computation Element Protocol (PCEP)

The Path Computation Element Protocol (PCEP) is one of several protocols used for communication between a Wide-Area Network (WAN) Software-Define Networking (SDN) controller and network elements.

The Nokia WAN SDN Controller is known as the Network Services Platform (NSP).

Figure 44: NSP Architecture illustrates the architecture of the NSP.

The NSP implements a few components that provide service provisioning, automation, optimization, and element management functions for both IP and optical networks. The following is an overview of the NSP components. More details can be found in the *NSP Planning Guide*:

1. NSP cluster

   The NSP cluster is the core component that hosts the common services (nspOS) as well as all the major NSP software applications. Among the applications hosted by the NSP cluster is the Model-driven Mediation (MDM) which provides mediation between model-driven NSP applications and Nokia or third-party network devices. The Workflow Manager (WFM) allows for the creation and execution of workflows. The NSP Baseline Analytics monitor network traffic to establish baselines and can flag anomalous traffic patterns.
2. IP resource control (IPRC)

   The IPRC provides service provisioning and activation as well as the Network Resource Controller for packet networks (NRC-P). The NRC-P hosts a path computation engine and implements a stateful Path Computation Element (PCE). The PCE instantiates and manages LSPs across IP network elements (NEs), and supports RSVP and segment routing LSP technologies. It also provides flow-based protocols such as OpenFlow and BGP FlowSpec to perform intelligent traffic steering and to automate policy based redirection at the flow or route level.
3. Cross domain resource control (CDRC)

   This component optimizes network resources across different layers and domains of IP/MPLS, and optical networks.
4. Simulation tool

   A traffic engineering tool that can be used by network engineers to design a new network, or optimize and simulate failures in an existing network that is imported into the tool.

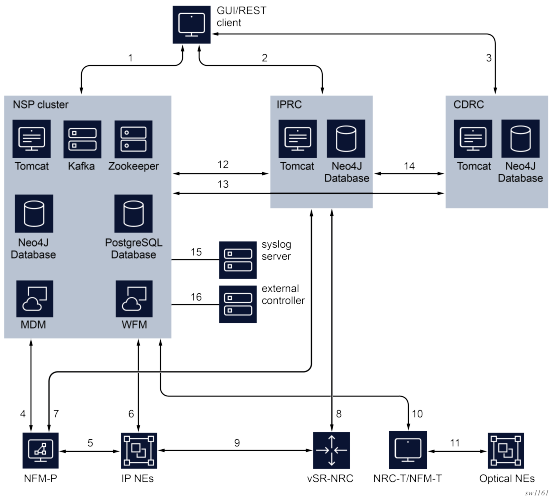

Figure 44: NSP Architecture

Figure 45: Packet Network Resource Controller (NRC-P) Architecture illustrates the NRC-P.

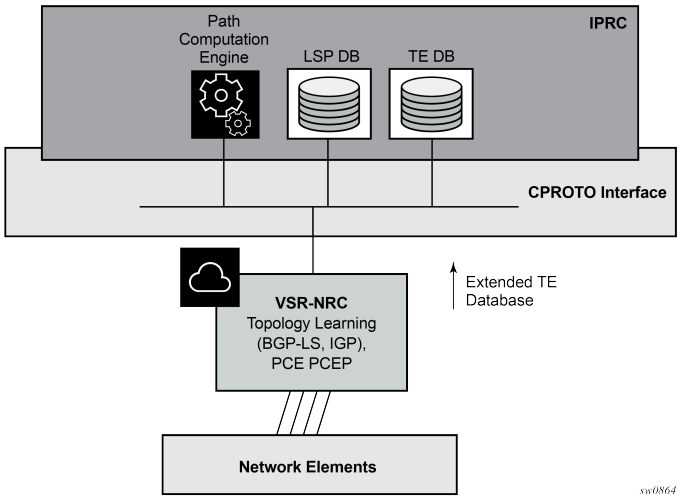

Figure 45: Packet Network Resource Controller (NRC-P) Architecture

The NRC-P has the following architecture:

- a single Virtual Machine (VM) handling the Java implementation of an MPLS path computation engine, a Traffic Engineering (TE) graph database (TE-DB), and an LSP database (LSP-DB). This is part of the IPRC as shown in *Figure 44: NSP Architecture*.
- a single VM running a SR OS image handles the functions of topology discovery of multiple IGP instances and areas via IGP or BGP-LS and the PCE PCEP functions. This is referred to as the VSR Network Resource Controller (VSR-NRC).
- a plug-in adapter using the Nokia CPROTO interface, providing reliable, TCP-based message delivery between VSR-NRC and the IPRC. The plug-in adapter implements a compact encoding/decoding (codec) function for the message content using Google ProtoBuf. Google ProtoBuf also provides for automatic C++ (VSR-NRC side) and Java (IPRC side) code generation to process the exchanged message content.

The VSR-NRC implements a PCEP PCE function, an OpenFlow controller, a BMP station database, a Route Origination module, and a TE database populated using IGP and BGP-LS.

The NRC-P module of the NSP and the VSR-NRC communicate using a reliable proprietary TCP-based channel called CPROTO. The NRC-P module acts as the server side, it is the module which always initiates the establishment of the CPROTO session toward the VSR-NRC.

The message data within the CPROTO channel is encoded and serialized using Google Protocol Buffers (PROTOBUF).

The VSR-NRC implements an NSP-Proxy module that manages all databases and channels used in the communications with the NRC-P. The NSP-Proxy opens a dedicated UDP port number 4199 for this communication and operates as the client side. This port is managed by the NSP-PROXY.

The NRC-P initiates a separate CPROTO session with a dedicated PROTOBUF channel service for the information exchanged for each type of capability supported:

1. PCEP
2. OPEN_FLOW
3. BGP_LS (covers topology discovered using BGP-LS and IGP)
4. BMP_STATION
5. ROM_SRTE (Route Origination Module for BGP families **sr-policy-ipv4** and **sr-policy-ipv6**)
6. ROM_IPV4/V6 (BGP families IPv4 and IPv6)
7. ROM_FLOSPEC (BGP families **flowspec-ipv4** and **flowspec-ipv6**)
8. ROM_LABELV4/V6 (BGP families **label-ipv4** and **label-ipv6**)
9. Global Health and Notification

The NRC-P, via the VSR-NRC, uses PCEP to communicate with its clients, referred to as PCE Clients (PCCs). Each router acting as a PCC initiates a PCEP session to the PCE in its domain.

When the user enables PCE control for one or more segment routing or RSVP-TE LSPs, the PCE owns the path updating and periodic re-optimization of the LSP. In this case, the PCE acts in an active stateful role. The PCE can also act in a stateful passive role for other LSPs on the router by discovering them and taking into account their resource consumption when computing the path for the LSPs it has control ownership of.

The following is a high-level description of the PCE and PCC capabilities:

- base PCEP implementation, per RFC 5440
- active and passive stateful PCE LSP update, as per RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*
- delegation of LSP control to PCE
- synchronization of the LSP database (LSP-DB) with network elements for PCE-controlled LSPs and network element-controlled LSPs
- support for the RSVP-TE P2P LSP type
- support for the SR-TE P2P LSP type, as per *draft-ietf-pce-segment-routing-08, PCEP Extensions for Segment Routing*
- support for PCC-initiated LSPs, as per RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*
- support for PCE-initiated LSPs, as per RFC 8281, *PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model*

- support for LSP path diversity across different LERs using extensions to the PCE path profile, as per *draft-alvarez-pce-path-profiles*
- support for LSP path bidirectionality constraints using extensions to the PCE path profile, as per *draft-alvarez-pce-path-profiles*

6.1.1 PCC and PCE Configuration

The following PCE parameters cannot be modified while the PCEP session is operational:

- **local-address**
- **keepalive**
- **dead-timer**

The **unknown-message-rate** PCE parameter can be modified while the PCEP session is operational.

The following PCC parameters cannot be modified while the PCEP session is operational:

- **local-address**
- **keepalive**
- **dead-timer**
- **peer** (regardless of **shutdown** state)

The following PCC parameters can be modified while the PCEP session is operational:

- **report-path-constraints**
- **unknown-message-rate**

6.1.2 Base Implementation of Path Computation Elements (PCE)

The base implementation of PCE uses the PCEP extensions defined in RFC 5440.

The main functions of the PCEP are:

- PCEP session establishment, maintenance, and closing
- path computation requests using the PCReq message
- path computation replies using the PCRep message
- notification messages (PCNtf) by which the PCEP speaker can inform its peer about events, such as path request cancellation by PCC or path computation cancellation by PCE
- error messages (PCErr) by which the PCEP speaker can inform its peer about errors related to processing requests, message objects, or TLVs

Table 27: Base PCEP Message Objects and TLVs lists the base PCEP messages and objects.

Table 27: Base PCEP Message Objects and TLVs

TLV, Object, or Message	Contained in Object	Contained in Message
OPEN Object	—	OPEN, PCErr
Request Parameter (RP) Object	—	PCReq, PCRep, PCErr, PCNtf
NO-PATH Object	—	PCRep
END-POINTS Object	—	PCReq
BANDWIDTH Object	—	PCReq, PCRep, PCRpt, PCInitiate
METRIC Object	—	PCReq, PCRep, PCRpt, PCInitiate
Explicit Route Object (ERO)	—	PCRep
Reported Route Object (RRO)	—	PCReq
LSPA Object	—	PCReq, PCRep, PCRpt, PCInitiate
Include Route Object (IRO)	—	PCReq, PCRep
SVEC Object	—	PCReq
NOTIFICATION Object	—	PCNtf
PCEP-ERROR Object	—	PCErr
LOAD-BALANCING Object	—	PCReq
CLOSE Object	—	CLOSE

The behavior and limitations of the implementation of the objects in *Table 27: Base PCEP Message Objects and TLVs* are as follows:

- PCE treats all supported objects received in a PCReq message as mandatory, regardless of whether the P-flag in the object's common header is set (mandatory object) or not (optional object).
- The PCC implementation will always set the B-flag (B=1) in the METRIC object containing the hop metric value, which means that a bound value must be included in PCReq. PCE returns the computed value in PCRep with flags set identically to PCReq.
- The PCC implementation will always set flags B=0 and C=1 in the METRIC object for the IGP or TE metric values in the PCReq message. This means that the request is to optimize (minimize) the metric without providing a bound. PCE returns the computed value in PCRep with flags set identically to PCReq.
- The IRO and LOAD-BALANCING objects are not in the NSP PCE feature. If the PCE receives a PCReq message with one or more of these objects, it will ignore them regardless of the setting of the P-flag, and will process the path computations normally.
- LSP path setup and hold priorities will be configurable during SR-TE LSP configuration on the router, and PCC will pass the configurations on in an LSPA object. However, PCE does not implement LSP pre-emption.
- The LSPA, METRIC, and BANDWIDTH objects are also included in the PCRpt message.

The following features are not supported in the SR OS:

- PCE discovery using IS-IS, per RFC 5089, and OSPF, per RFC 5088, along with corresponding extensions for discovering stateful PCE, per *draft-sivabalan-pce-disco-stateful*
- security of the PCEP session using MD5 or TLS between PCEP peers
- PCEP synchronization optimization as per *draft-ietf-pce-stateful-sync-optimizations*
- support of end-to-end secondary backup paths for an LSP. PCE standards do not currently support an LSP container with multiple paths, and treats each request as a path with a unique PLSP ID. It is up to the router to tie the two paths together to create 1:1 protection, and to request path or SRLG diversity among them when it makes the request to PCE. This is not specific to PCE controlling an SR-TE LSP, but also to controlling an RSVP LSP.
- jitter, latency, and packet loss metrics support as per RFC 7471 and *draft-ietf-isis-te-metric-extensions*, and their use in the PCE METRIC object as per *draft-ietf-pce-pcep-service-aware*

6.1.3 PCEP Session Establishment and Maintenance

The PCEP protocol operates over TCP using destination TCP port 4189. The PCC always initiates the connection. Once the user configures the PCEP local address and the peer address on the PCC, the PCC initiates a TCP connection to the PCE. Once a connection is established, the PCC and PCE exchange OPEN messages, which initializes the PCEP session and exchanges the session parameters to be negotiated.

The PCC always checks first if the remote PCE address is reachable out-of-band via the management port. If not, it will try to reach the remote PCE address in-band. When the session comes up out-of-band, the management IP address is always used; the local address configured by the user is ignored and is only used for an in-band session.

A keep-alive mechanism is used as an acknowledgment of the acceptance of the session within the negotiated parameters. It is also used as a maintenance function to detect whether or not the PCEP peer is still alive.

The negotiated parameters include the Keepalive timer and the DeadTimer, and one or more PCEP capabilities such as support of Stateful PCE and the SR-TE LSP Path type.

The PCEP session initialization steps are illustrated in *Figure 46: PCEP Session Initialization*.

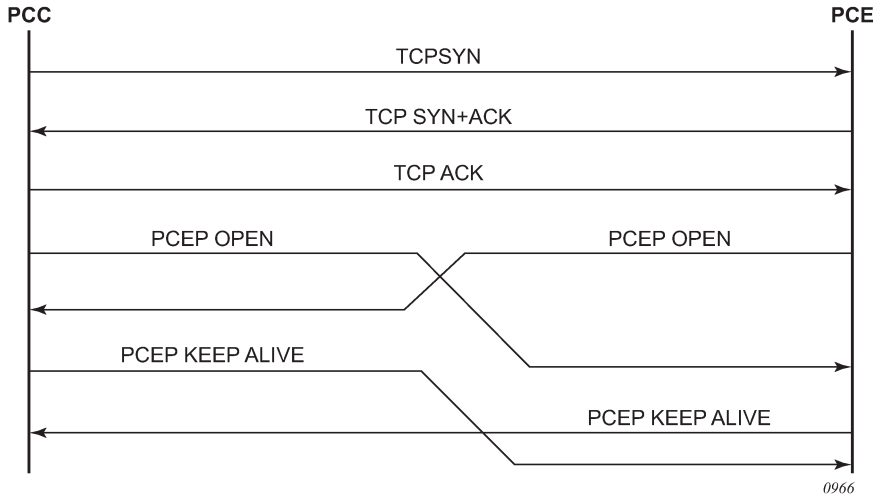

Figure 46: PCEP Session Initialization

If the session to the PCE times out, the router acting as a PCC keeps the last successfully-programmed path provided by the PCE until the session to the PCE is re-established. Any subsequent change to the state of an LSP is synchronized at the time the session is re-established.

When a PCEP session to a peer times out or closes, the rate at which the PCEP speaker attempts the establishment of the session is subject to an exponential back-off mechanism.

6.1.4 PCEP Parameters

The following PCEP parameters are user-configurable on both the PCC and PCE. On the PCE, the configured parameter values are used on sessions to all PCCs.

- Keep-alive timer — A PCEP speaker (PCC or PCE) must send a keep-alive message if no other PCEP message is sent to the peer at the expiry of this timer. This timer is restarted every time a PCEP message is sent or the keep-alive message is sent.

  The keep-alive mechanism is asymmetric, meaning that each peer can use a different keep-alive timer value.

  The range of this parameter is 1 to 255 seconds, and the default value is 30 seconds. The no version returns to the default value.
- Dead timer — This timer tracks the amount of time a PCEP speaker (PCC or PCE) waits after the receipt of the last PCEP message before declaring its peer down.

  The dead timer mechanism is asymmetric, meaning that each PCEP speaker can propose a different dead timer value to its peer to use to detect session timeouts.

  The range of this parameter is 1 to 255 seconds, and the default value is 120 seconds. The no version returns to the default value.

- Maximum rate of unknown messages — When the rate of received unrecognized or unknown messages reaches this limit, the PCEP speaker closes the session to the peer.
- Session re-delegation and state timeout — If the PCEP session to the PCE goes down, all delegated PCC-initiated LSPs have their state maintained in the PCC and are not timed out. The PCC will continue to try re-establishing the PCEP session. When the PCEP session is re-established, the LSP database is synchronized with the PCE, and any LSP which went down since the last time the PCEP session was up will have its path updated by the PCE.

6.1.4.1 Stateful PCE

The main function introduced by stateful PCE over the base PCE implementation is the ability to synchronize the LSP state between the PCC and the PCE. This allows the PCE to have all the required LSP information to perform re-optimization and updating of the LSP paths.

Table 28: PCEP Stateful PCE Extension Objects and TLVs describes the messages and objects supported by stateful PCE in the SR OS.

Table 28: PCEP Stateful PCE Extension Objects and TLVs

TLV, Object, or Message	Contained in Object	Contained in Message
Path Computation State Report (PCRpt)	—	New message
Path Computation Update Request (PCUpd)	—	New message
Stateful PCE Capability TLV	OPEN	OPEN
Stateful Request Parameter (SRP) Object	—	PCRpt, PCErr, PCInitiate
LSP Object	ERO	PCRpt, PCReq, PCRep, PCInitiate
LSP Identifiers TLV	LSP	PCRpt
Symbolic Path Name TLV	LSP, SRP	PCRpt, PCInitiate
LSP Error Code TLV	LSP	PCRpt
RSVP Error Spec TLV	LSP	PCRpt

The behavior and limitations of the implementation of the objects in *Table 28: PCEP Stateful PCE Extension Objects and TLVs* are as follows:

- PCC and PCE support all PCEP capability TLVs defined in this document and will always advertise them. If the OPEN object received from a PCEP speaker does not contain one or more of the capabilities, the PCE or PCC will not use them during that specific PCEP session.
- The PCC always includes the LSP object in the PCReq message to make sure that the PCE can correlate the PLSP-ID for this LSP when a subsequent PCRpt message arrives with delegation bit set. The PCE will, however, still honor a PCReq message without the LSP Object.
- PCE path computation will only consider the bandwidth used by LSPs in its LSP-DB. As a result, there are two situations where PCE path computation will not accurately take into account the bandwidth used in the network:
  - When there are LSPs which are signaled by the routers but are not synchronized up with the PCE. The user can enable the reporting of the LSP to the PCE LSP database for each LSP.
  - When the stateful PCE is peering with a third party stateless PCC, implementing only the original RFC 5440. While PCE will be able to bring the PCEP session up, the LSP database will not be updated, since stateless PCC does not support the PCRpt message. As such, PCE path computation will not accurately take into account the bandwidth used by these LSPs in the network.
- PCE ignores the R-flag (re-optimize flag) in the PCReq message when acting in stateful-passive mode for a given LSP, and will always return the new computed path, regardless if it is link-by-link identical or has the same metric as the current path. The decision whether or not to initiate the new path in the network belongs to the PCC.
- The SVEC object is not supported in the SR OS and the NSP. If the PCE receives a PCReq message with the SVEC object, it will ignore the SVEC object and treat each path computation request in the PCReq message as independent, regardless of the setting of the P-flag in the SVEC object common header.
- When an LSP is delegated to the PCE, there can be no prior state in the NRC-P LSP database for the LSP. This could be due to the PCE not having received a PCReq message for the same PLSP-ID. In order for the PCE to become aware of the original constraints of the LSP, the following additional procedures are performed.
  - PCC appends a duplicate of each of the LSPA, METRIC, and BANDWIDTH objects in the PCRpt message. The only difference between the two objects of the same type is that the P-flag is set in the common header of the duplicate object to indicate a mandatory object for processing by the PCE.
  - The value of the metric or bandwidth in the duplicate object contains the original constraint value, while the first object contains the operational value. This is applicable to hop metrics in the METRIC object and BANDWIDTH object only. SR OS PCC does not support putting a bound on the IGP or TE metric in the path computation.
  - The path computation on the PCE uses the first set of objects when updating a path if the PCRpt contains a single set. If the PCRpt contains a duplicate set, PCE path computation must use the constraints in the duplicate set.
  - For interoperability, implementations compliant to PCEP standards should be able to accept the first metric object and ignore the second object without additional error handling. Since there are also BANDWIDTH and LSPA objects, the [**no**] **report-path-constraints** command is provided in the PCC on a per-PCEP session basis to disable the inclusion of the duplicate objects. Duplicate objects are included by default.

Stateful PCE uses the additional messages, TLVs, and objects described in *Table 29: PCEP Stateful PCE Extension Objects and TLVs Locations* for PCE initiation of LSPs.

Table 29: PCEP Stateful PCE Extension Objects and TLVs Locations

TLV, Object, or Message	Contained in Object	Contained in Message
PCE LSP Initiate Message (PCInitiate)	—	New message
PCC LSP Create Flag (C-Flag)	LSP	PCRpt
PATH_PROFILE_ID TLV	Path Profile	N/A

6.1.4.2 PCEP Extensions in Support of SR-TE LSPs

In order for the PCE and PCC to manage the path of an SR-TE LSP, they both implement the following extensions to PCEP in support of segment routing.

- A new Segment Routing capability TLV in the OPEN object to indicate support of segment routing tunnels by the PCE and the PCC during PCEP session initialization. This TLV is referred to as the SR-PCE-CAPABILITY TLV.
- The PCC and PCE support all PCEP capability TLVs defined in this chapter, and will always advertise them. If the OPEN object received from a PCEP speaker does not contain one or more of the capabilities, the PCE or the PCC will not use them during that specific PCEP session.
- A new Path Setup Type TLV for SR-TE LSPs to be included in the Stateful PCE Request Parameters (SRP) Object during path report (PCRpt) messages by the PCC.

  A Path Setup Type TLV with a value of 1 identifies an SR-TE LSP.
- A new Segment Routing ERO and RRO with sub-objects, referred to as SR-ERO and SR-RRO sub-objects, which encode the SID information in PCRpt messages.
- The PCE implementation supports the Segment-ID (SID) Depth value in the METRIC object. This is always signaled by the PCC in the PCEP Open object as part of the as SR-PCE-CAPABILITY TLV. It is referred to as the Maximum Stack Depth (MSD). In addition, the per-LSP value for the **max-sr-labels** option, if configured, is signaled by the PCC to the PCE in the Segment-ID (SID) Depth value in a METRIC object for both a PCE-computed LSP and a PCE-controlled LSP. PCE will compute and provide the full explicit path with TE-links specified. If there is no path with the number of hops lower than the MSD value, or the Segment-ID (SID) Depth value if signaled, a reply with no path will be returned to the PCC.For a PCC controlled LSP, if the label stack returned by the TE-DB's hop-to-label translation exceeds the per LSP maximum SR label stack size, the LSP is brought down.
- If the Path Setup Type (PST) TLV is not included in the PCReq message, the PCE or PCC must assume it is for an RSVP-TE LSP.

Table 30: PCEP Segment Routing Extension Objects and TLVs describes the segment routing extension objects and TLVs supported in the SR OS.

Table 30: PCEP Segment Routing Extension Objects and TLVs

TLV, Object, or Message	Contained in Object	Contained in Message
SR PCE CAPABILITY TLV	OPEN	OPEN
Path Setup Type (PST) TLV	SRP	PCReq, PCRep, PCRpt
SR-ERO Sub-object	ERO	PCRep, PCRpt
SR-RRO Sub-object	RRO	PCReq, PCRpt
Segment-ID (SID) Depth Value in METRIC Object	METRIC	PCReq, PCRpt

6.1.5 LSP Initiation

An LSP that is configured on the router is referred to as a PCC-initiated LSP. An LSP that is not configured on the router, but is instead created by the PCE at the request of an application or a service instantiation, is referred to as a PCE-initiated LSP.

The SR OS support three different modes of operations for PCC-initiated LSPs which are configurable on a per-LSP basis.

1. When the path of the LSP is computed and updated by the router acting as a PCE Client (PCC), the LSP is referred to as PCC-initiated and PCC-controlled.

   A PCC-initiated and PCC-controlled LSP has the following characteristics:

   - The LSP can contain strict or loose hops, or a combination of both.
   - CSPF is supported for RSVP-TE LSPs. Local path computation takes the form of hop-to-label translation for SR-TE LSPs.
   - LSPs can be reported to synchronize the LSP database of a stateful PCE server using the **pce-report** option. In this case, the PCE acts in passive stateful mode for this LSP. The LSP path cannot be updated by the PCE. In other words, the control of the LSP is maintained by the PCC.
2. When the path of the LSP is computed by the PCE at the request of the PCC, it is referred to as PCC-initiated and PCE-computed.

   A PCC-initiated and PCE-computed LSP has the following characteristics:

   - The user must enable the **path-computation-method pce** option for the LSP so that the PCE can perform path computation at the request of the PCC only. PCC retains control.
   - LSPs can be reported to synchronize the LSP database of a stateful PCE server using the **pce-report** option. In this case, the PCE acts in passive stateful mode for this LSP.

3. When the path of the LSP is updated by the PCE following a delegation from the PCC, it is referred to as PCC-initiated and PCE-controlled.

   A PCC-initiated and PCE-controlled LSP has the following characteristics:

   - The user must enable the **pce-control** option for the LSP so that the PCE can perform path updates following a network event without an explicit request from the PCC. PCC delegates full control.
   - The user must enable the **pce-report** option for LSPs that cannot be delegated to the PCE. The PCE acts in active stateful mode for this LSP.

The SR OS also supports PCE-initiated LSPs. PCE-initiated LSP is a feature that allows a WAN SDN Controller, for example, the Nokia NSP, to automatically instantiate an LSP based on a service or application request. Only SR-TE PCE-initiated LSPs are supported.

The instantiated LSP does not have a configuration on the network routers and is therefore treated the same way as an auto-LSP. The parameters of the LSP are provided using policy lookup in the NSP and are passed to the PCC using PCEP as per RFC 8281. Missing LSP parameters are added using a default or specified LSP template on the PCC.

PCE-initiated LSPs have the following characteristics:

- The user must enable **pce-initiated-lsp sr-te** to enable the PCC to accept and process PCInitiate messages from the PCE.
- The user must configure one or more LSP templates of type **pce-init-p2p-srte** for SR-TE LSPs. A default template is supported that is used for LSPs for which no ID or an ID of 0 is included in the PCInitiate message. The user must configure at least one default PCE-initiated LSP template.

PCE-initiated LSPs are a form of SR-TE Auto-LSP and are available to the same forwarding contexts. See *Forwarding Contexts Supported with SR-TE Auto-LSP*. Similar to other auto-LSPs, they are installed in the TTM and are therefore available to advanced policy-based services using auto-bind such as VPRN and E-VPN. However, they cannot be used with provisioned SDPs.

6.1.5.1 PCC-Initiated and PCE-Computed/Controlled LSPs

The following is the procedure for configuring and programming a PCC-initiated SR-TE LSP when control is delegated to the PCE.

Procedure

1. The LSP configuration is created on the PE router via CLI or via the OSS/NSP NFM-P.

   The configuration dictates which PCE control mode is desired: active (**pce-control** and **pce-report** options enabled) or passive (**path-computation-method pce** enabled and **pce-control** disabled).

2. PCC assigns a unique PLSP-ID to the LSP.

   The PLSP-ID uniquely identifies the LSP on a PCEP session and must remain constant during its lifetime. PCC on the router must keep track of the association of the PLSP-ID to the Tunnel-ID and Path-ID, and use the latter to communicate with MPLS about a specific path of the LSP. PCC also uses the SRP-ID to correlate PCRpt messages for each new path of the LSP.

3. The PE router does not validate the entered path.

   Note however that in the SR OS, the PCE supports the computation of a path for an LSP with empty-hops in its path definition. While PCC will include the IRO objects in the PCReq message to PCE, the PCE will ignore them and compute the path with the other constraints except the IRO.

4. The PE router sends a PCReq message to the PCE to request a path for the LSP.

   The PCReq message includes the LSP parameters in the METRIC object, the LSPA object, and the BANDWIDTH object. The PE router also includes the LSP object with the assigned PLSP-ID. At this point, the PCC does not delegate the control of the LSP to the PCE.

5. The PCE computes a new path, reserves the bandwidth, and returns the path in a PCRep message with the computed ERO in the ERO object. It also includes the LSP object with the unique PLSP-ID, the METRIC object with any computed metric value, and the BANDWIDTH object.

   Note: For the PCE to be able to use the SRLG path diversity and admin-group constraints in the path computation, the user must configure the SRLG and admin-group membership against the MPLS interface and make sure that the **traffic-engineering** option is enabled in IGP. This causes IGP to flood the link SRLG and admin-group membership in its participating area, and for PCE to learn it in its TE database.

6. The PE router updates the CPM and the data path with the new path.

   Up to this point, the PCC and PCE are using passive stateful PCE procedures. The next steps will synchronize the LSP database of the PCC and PCE for both PCE-computed and PCE-controlled LSPs. They will also initiate the active PCE stateful procedures for the PCE-controlled LSP only.

7. The PE router sends a PCRpt message to update the PCE with an UP state, and also sends the RRO as confirmation.

   It now includes the LSP object with the unique PLSP-ID. For a PCE-controlled LSP, the PE router also sets the delegation control flag to delegate control to the PCE. The state of the LSP is now synchronized between the router and the PCE.

8. Following a network event or a re-optimization, the PCE computes a new path for a PCE-controlled LSP and returns it in a PCUpd message with the new ERO.

   It will include the LSP object with the same unique PLSP-ID assigned by the PCC, as well as the Stateful Request Parameter (SRP) object with a unique SRP-ID-number to track error and state messages specific to this new path.

9. The PE router updates the CPM and the data path with the new path.

10. The PE router sends a PCRpt message to inform the PCE that the older path is deleted.

    It includes the unique PLSP-ID value in the LSP object and the R (Remove) bit set.

11. The PE router sends a new PCRpt message to update PCE with an UP state, and also sends the RRO to confirm the new path.

    The state of the LSP is now synchronized between the router and the PCE.

12. If PCE owns the delegation of the LSP and is making a path update, MPLS will initiate the LSP and update the operational value of the changed parameters while the configured administrative values will not change.

    Both the administrative and operational values are shown in the details of the LSP path in MPLS.

13. If the user makes any configuration change to the PCE-computed or PCE-controlled LSP, MPLS requests that the PCC first revoke delegation in a PCRpt message (PCE-controlled only), and then MPLS and PCC follow the above steps to convey the changed constraint to PCE which will result in the programming of a new path into the data path, the synchronization of the PCC and PCE LSP databases, and the return of delegation to PCE.

The above procedure is followed when the user performs a **no shutdown** command on a PCE-controlled or PCE-computed LSP. The starting point is an LSP which is administratively down with no active path. For an LSP with an active path, the following items can apply:

1. If the user enabled the **path-computation-method pce** option on a PCC-controlled LSP with an active path, no action is performed until the next time the router needs a path for the LSP following a network event of a LSP parameter change. At that point, the prior procedure is followed.
2. If the user enabled the **pce-control** option on a PCC-controlled or PCE-computed LSP with an active path, the PCC will issue a PCRpt message to the PCE with an UP state, as well as the RRO of the active path. It will set the delegation control flag to delegate control to the PCE. The PCE will keep the active path of the LSP and make no updates to it until the next network event or re-optimization. At that point, the prior procedure is followed.

6.1.5.2 PCE-Initiated LSPs

The following is the procedure for configuring and programming a PCE-initiated SR-TE LSP.

Procedure

1. The user must enable **pce-initiated-lsp sr-te** using the CLI or using the OSS.

   The user can also optionally configure a limit to the number of PCE-Initiated LSPs that the PCE can instantiate on a node using the **max-srte-pce-init-lsps** command in the CLI or using the OSS.

2. The user must configure at least one LSP template of type **pce-init-p2p-srte** to select the value of the LSP parameters that remain under the control of the PCC.

   At a minimum, a default template should be configured (type **pce-init-p2p-srte default**). In addition, LSP templates with a defined template ID can be configured. The template ID can be included in the path profile of the PCEInitiate message to indicate which non-default template to use for a particular LSP. If the PCInitiate message does not include the PCE path profile, MPLS uses the default PCE-initiated LSP template. *Table 31: LSP Template Parameters* lists the applicable LSP template parameters. These are grouped into:

   - parameters that are controlled by the PCE and that the PCC cannot change (invalid, implicit, and signaled in PCEP)
   - parameters that are controlled by the PCC and are used for signaling the LSP in the control plane

- parameters that are controlled by the PCC and are related to the usability of the LSP by MPLS and other applications such as routing protocols, services, and OAM

The user can configure these parameters in the template.

Table 31: LSP Template Parameters

Controlled by PCE			Controlled by PCC	
Invalid	Implicit	Signaled in PCEP	LSP Signaling Options	LSP Usability Options
auto-bandwidth	pce-report	bandwidth	—	—
retry-limit	—	exclude	—	bgp-shortcut
retry-timer	pce-control	from	—	bgp-transport-tunnel
shutdown	pce-report	hop-limit	default-path (mandatory, must be empty)	—
least-fill	path-computation-method pce	include	—	—
metric-type		—	—	entropy-label
—	—	setup-priority	—	igp-shortcut
—	—	hold-priority	—	—
—	—	—	—	load-balancing-weight
—	—	—	—	max-sr-labels
—	—	—	—	additional-frr-labels
—	—	—	—	metric

Controlled by PCE			Controlled by PCC	
Invalid	Implicit	Signaled in PCEP	LSP Signaling Options	LSP Usability Options
—	—	—	—	vprn-auto-bind
—	—	—	—	admin-tag

All PCE-initiated LSPs using a particular LSP template are deleted if the user deletes the template. The default template can be created or deleted if the **pce-initiated-lsp>sr-te** context does not exist. However, the **pce-init-p2p-sr-te default lsp-template** cannot be deleted if the **pce-initiated-lsp>sr-te** context exists and is not shutdown. This context must be shutdown to delete the **pce-init-p2p-sr-te default** LSP template, which brings down all PCE Initiated LSPs. The **pce-initiated-lsp>sr-te** context cannot be administratively enabled if the **pce-init-p2p-sr-te default lsp-template** is not configured.

A shutdown of an LSP template does not bring down any already established LSPs. Parameters can only be changed once in the shutdown state and the changes do not take effect until a **no shutdown** is performed. This means that PCE updates use older parameters if the template is still shut down.

MPLS copies the lsp-template parameters into the lsp-entry when a PCE initiated LSP is created. MPLS handles lsp-updates based on the last copied parameters.

After the lsp-template parameter changes, when the lsp-template is **no shutdown**.

- MPLS copies the related TTM parameters (listed below) into the LSP entry, and updates TTM

- If there is a change in **max-sr-labels**, MPLS re-evaluates the related LSPs, and brings paths down if applicable (for example, if current hopCount is greater than the applicable **max-sr-labels** value).

  The TTM LSP-related parameters include:

  – Metric
  – VprnAutoBind
  – LoadBalWeight
  – MaxSrLabels
  – AdditionalFrrLabels
  – MetricOffset
  – IgpShortCut
  – IgpShortcutLfaOnly
  – IgpShortcutLfaProtect
  – LspBgpShortCut
  – LspBgpTransTunnel

A PCE-initiated LSP "update" request will be accepted regardless of the LSP template administrative state, as follows:

- If the LSP template is adminUp, the system copies the LSP template parameters to the LSP/path.
- If the LSP template is adminDown, the system uses the previously copied LSP template parameters and responds to the update with an LSP operUp report.

3. The user can set the redelegation and state timers on the PCC.

   Redelegation timeout and state timeout timers are started when the PCEP session goes down or the PCE signals overload. The redelegation timer applies to both PCC-initiated and PCE-initiated LSPs, while the state timer applies only to PCE-initiated LSPs. The redelegation and state timers are configured in the CLI or through management, as follows:

   config>router>pcep>pcc>

   [no] redelegation-timer *seconds*

   [no] state-timer *seconds* [**action** {**remove** | **none**}]

   If the delegated PCE-initiated LSPs cannot be redelegated by the time these timers expire, a configurable action is performed by the PCC. The supported actions are **remove** or **none**, with a default of **remove**.

4. The PCE can then initiate and remove LSPs on the PCC.

   These procedures are described in *LSP Instantiation Using PCEP*, *LSP Deletion Using PCEP*, and *Dynamic State Handling for PCE Initiated LSPs*.

6.1.5.2.1 LSP Instantiation Using PCEP

The following procedures are followed in the instantiation of a PCE-initiated LSP by both the NSP and SR OS router. Further protocol details can be found in RFC 8281.

NSP Generation of PCInitiate

Procedure

1. When the PCEP session is established from the PCC to PCE, the PCC and PCE exchange the Open object and both set the new "I flag, LSP-INSTANTIATION CAPABILITY" flag, in the STATEFUL-PCE-CAPABILITY TLV flag field.
2. The operator, using the north-bound REST interface, the NSD or another interface, makes a request to the NSP to initiate an LSP.

   The following parameters are specified:

   a. source address
   b. destination address
   c. LSP type (SR-TE)
   d. bandwidth value
   e. include/exclude admin-group constraints
   f. optional PCE path profile ID for the path computation at the PCE
   g. optional PCE-initiated LSP template ID for use by the PCC to complete the instantiation of the LSP
3. The NSP crafts the PCInitiate message and sends it to the PCC using PCEP.

   The message contains the LSP object with PLSP-ID=0, the SRP object, the ENDPOINTS object, the computed SR-ERO (SR-TE) object, and the list of LSP attributes (bandwidth object, one or more metric objects, and the LSPA object). The LSP path name is inserted into the Symbolic Path Name TLV in the LSP object.
4. The PCE-initiated LSP template ID to be used at the PCC, if any, is included in the PATH-PROFILE-ID TLV of the Path Profile object.

   The Profile ID matches the PCE-initiated LSP template ID at the PCC and is not the same asThe Path Profile ID is used on the PCE to compute the path of this PCE-initiated LSP.
5. The Path Profile ID is used on the PCE to compute the path of this PCE-initiated LSP.

SR OS Router Procedures on Receiving a PCInitiate Message

Procedure

1. If a PCInitiate message includes a name that is a duplicate of an existing LSP on the router, the system generates an error.
2. The router assigns a PLSP-ID and looks up the specified PCE-initiated LSP template ID, if any, or the default PCE-initiated LSP template, to retrieve the local parameters, and instantiates the SR-TE LSP.
3. The instantiated LSP is added to the TTM and is used by all applications that look up a tunnel in the TTM.
4. The router crafts a PCRpt message with the Tunnel-ID, LSP-ID, and the RRO and passes it along with the PLSP-ID set to the assigned value and the delegation bit set in the LSP object to the PCE.

NSP Procedures on Receiving a PCRpt Message for a PCE

Procedure

1. The NSP confirms the bandwidth reservation and updates its LSP database. The PCC and PCE are synchronized at this point.
2. The NSP reports the PLSP-ID/Tunnel-ID to the application, for example NSD, or to the operator that uses it in the specific application that originated the request.
3. The PCE can perform updates to the path during the lifetime of the LSP by using the PCUpd message in the same way as with a delegated PCC-initiated LSP.

6.1.5.2.2 LSP Deletion Using PCEP

The following procedures apply in the deletion of a PCE-initiated LSP. More protocol level details are provided in RFC 8281. These procedures are applicable when the user manually deletes the PCE-initiated LSP or the NSP application, or when NSD requests the deletion of the PCE-initiated LSP. The procedures that apply when a network event occurs are described in *SR OS Router Procedures*.

Procedure

The NSP crafts a PCInitiate message for the corresponding PLSP-ID and sets the R-bit in the SRP object flags to indicate to the PCC that it must delete the LSP. The NSP sends the message to the PCC using PCEP.

SR OS Router Procedures on Receipt of a PCInitiate with the R-bit Set

Procedure

1. The router deletes the state of the LSP.
2. The router crafts a PCRpt message with the R-bit set in the LSP object flags.

NSP Procedures Upon Issuance of pce-init delete Command

Procedure

The NSP deletes the LSP from its LSP database.

6.1.5.2.3 Dynamic State Handling for PCE Initiated LSPs

NSP Procedures

Procedure

1. The NRC-P controls the creation and the deletion of the PCE-initiated LSP.

2. All LSP creation retries are performed by the NSP. If the PCC rejects an instantiation, the NSP can issue a new request for instantiation or give up and delete the LSP state locally after a configurable maximum number of retries.

3. The NSP can reject an instantiation request if it does not receive a PCRpt from the PCC message within a configured timeframe.

4. When the PCEP session comes up and the LSP DB synchronization from the PCC to PCE is complete, the NSP reinitiates the PCE-initiated LSPs that are missing from the PCC reports.

5. If a PCEP session goes down, the NSP stops sending any new or updated PCE-initiated LSP paths to that PCC; therefore, the LSP DB on the NSP and PCC can go out of synchronization during that time.

6. If the PCEP session to a PCC goes down, the NSP marks all PCE-initiated and PCC-initiated LSPs for that PCC as stale but keeps their reservation for an amount of time equal to the **state-timeout** timer. The **state-timeout** timer applies to both PCE-initiated and PCC-initiated LSPs on the PCE and is set to a fixed value of 10 minutes.

   Note: The **state-timeout** timer must be considerably larger than the maximum state timer on the PCC to give the PCC time to clean up PCE-initiated LSPs and prevent PCInit requests for duplicate LSPs.

   a. If the PCEP session was re-established within that time, the NRC-P reinitiates all PCE-initiated LSPs toward the PCC from which a PCRpt remove with the special error code LSP_ERR_SYNC_DELETE was received during the LSP DB synchronization with the PCC.
   b. If the **state-timeout** timer expires, the NRC-P releases the resources but does not delete the LSPs from the LSP DB. If the PCEP session comes up subsequently, the NSP recomputes the path of each LSP from which a PCRpt remove with the special error code LSP_ERR_SYNC_DELETE was received during the LSP DB synchronization with the PCC and sends the PCC a PCInitiate message for each LSP.

7. If the NSP is informed by the VSR-NRC of a PCRpt with the remove flag in the LSP object and SRP object set for each of them, it follows the same procedures for these LSPs as when the PCEP session goes down.

SR OS Router Procedures

Table 32: Impact of PCC Operational Events summarizes the impact of various PCC operational events on the status of PCE-initiated LSPs.

Table 32: Impact of PCC Operational Events

Event	Impact on PCE-initiated LSPs	
	Oper-down	Deleted
MPLS shutdown	✓ *1*	

Event	Impact on PCE-initiated LSPs	
	Oper-down	Deleted
no mpls		✓[2]
no pce-initiated-lsp		✓ (all) [2]
no sr-te		✓ (sr-te)[2]
sr-te shutdown	✓ (sr-te) [1]	
pcc shutdown		✓ (all) [3]
pcc peer shutdown		✓[3]
Delete LSP template ID		✓ (LSPs using template)[2]
Delete default LSP template		✓ (all) [2]

Note:

1. Also results in a PCRpt to the PCE with LSP error admin down.
2. Also results in a PCRpt to the PCE with LSP deleted.
3. A PCRpt with delete and a special error code, for example, LSP_ERR_SYNC_DELETE, is sent during the PCC rejoin synchronization that occurs when the PCC or PCC peer comes back up.

The following list describes in more detail the actions that the PCC takes on PCE-initiated LSPs as a result of PCC operational events.

Procedure

1. If any event causes PCE-initiated LSPs to be deleted by the PCC, the PCC sends a PCRpt with remove the flag in both the SRP object and the LSP object set for each impacted LSP. If the event is a failure of the PCEP session to the PCE, or a shutdown of the PCC or PCC peer, the PCRpt is sent, with the special error code LSP_ERR_SYNC_DELETE, only after the PCEP session comes back up during the PCC resynchronization with the PCE.

2. If any event causes PCE-initiated LSPs to go operationally down, the PCC router sends a PCRpt with the operational bits in the LSP object set to DOWN for each impacted LSP.

3. If the user shuts down the PCC process on the router, all PCE-initiated LSPs are deleted. When the user performs a **no shutdown** of the PCC process, the PCC reports to the PCE so that the NSP is aware.

4. If a PCEP peer is shut down, the PCEP session goes down but the PCC keeps the state of all PCE-initiated LSPs, subject to the following rules regarding redelegation and the cleanup of state. See section 5.7.5 of RFC 8231 and section 6 of RFC 8281. These rules apply to all LSPs delegated to the PCE.

   Redelegation timeout and state timeout timers are started when the PCEP session goes down or the PCE signals overload. Configuration of these timers is described in *PCE-Initiated LSPs*. The system enforces that the **state-timer** be greater than the **redelegation-timer**, as specified in RFC 8231.

   The objectives of redelegation are described in Section 5.7.5 of RFC 8231. The redelegation process is as follows for both PCE-initiated and PCC-initiated LSPs.

   The existing LSP delegation state is maintained while the LSP redelegation timer is running. This gives the PCE time to recover. At the expiry of the redelegation timer, the PCC attempts to redelegate the LSPs to the PCE, as follows:

   - if the PCEP session to the existing PCE is still down or the PCE is still in overload, return delegation state to the PCC for all the delegated LSPs
   - wait until the PCEP session comes up and then attempt to redelegate the remaining LSPs back to the PCE. For each LSP, set a redelegation attempted flag once redelegation is attempted. If redelegation is accepted for all PCE-initiated LSPs delegated to the PCC before the state timeout timer expires, the system is behaving as expected.
   - if the state timeout timer expires, wait until all LSPs have been processed. The LSPs that are not redelegated but have the redelegation attempted flag set have the configured action applied to them. If this is **delete**, LSPs are deleted; otherwise, wait until the PCEP session comes up and then attempt to redelegate the remaining LSPs back to the PCE.

6.1.5.2.4 PCEP Support for RSVP-TE LSP

This section describes PCEP support of PCE Client-initiated (PCC-initiated) RSVP-TE LSP. The PCEP support of an RSVP-TE LSP provides the following modes of operation:

- PCC-initiated and PCC-controlled
- PCC-initiated and PCE-computed
- PCC-initiated and PCE-controlled

Each primary and secondary path is assigned its own unique Path LSP-ID (PLSP-ID). PCC indicates to PCE the state of each path (both UP and DOWN) and which path is currently active and carrying traffic (ACTIVE state).

The PCEP support of an RSVP-TE LSP differs from that of an SR-TE LSP in that PCE initiated RSVP-TE LSPs are not supported.

Feature Configuration

The following MPLS-level and LSP-level CLI commands, used to configure RSVP-TE LSP in a router acting as a PCEP Client (PCC).

- **config>router>mpls>pce-report rsvp-te** {**enable** | **disable**}
- **config>router>mpls>lsp>path-profile** *profile-id range* [**path-group** *group-id range*]
- **config>router>mpls>lsp>pce-report** {**enable** | **disable** | **inherit**}
- **config>router>mpls>lsp>path-computation-method pce**
- **config>router>mpls>lsp>pce-control**

Note: The PCE function implemented in the Nokia Network Services Platform (NSP) and referred to as the Network Resource Controller for Packet (NRC-P), supports only Shared Explicit (SE) style bandwidth management for TE LSPs. The PCEP protocol does not have means for the PCC to convey this value to the PCE, so, regardless of whether the LSP configuration option **rsvp-resv-style** is set to **se** or **ff**, the PCE will always use the SE style in the CSPF computation of the path for a PCE-computed or PCE-controlled RSVP-TE LSP.

A **one-hop-p2p** or a **mesh-p2p** RSVP-TE **auto-lsp** only supports the **pce-report** command in the LSP template:

config>router>mpls>lsp-template>pce-report {**enable** | **disable** | **inherit**}

The user must first shut down the LSP template before changing the value of the **pce-report option**.

A manual bypass LSP does not support any of the PCE-related commands. Reporting a bypass LSP to PCE is not required because it does not book bandwidth.

All other MPLS, LSP, and path-level commands are supported, with the exception of **backup-class-type**, **class-type**, **least-fill**, **main-ct-retry-limit**, **mbb-prefer-current-hops**, and **srlg** (on secondary standby path), which, if enabled, will result in a no operation.

The same instantiation modes are supported for RSVP-TE PCC-initiated LSPs as the SR-TE PCC-initiated LSPs. See *LSP Initiation* for more information.

Behavior of the LSP Path Update

When the **pce-control** option is enabled, the PCC delegates the control of the RSVP-TE LSP to the PCE.

The NRC-P sends a path update using the PCUpd message in the following cases:

- a failure event that impacts a link or a node in the path of a PCE-controlled LSP

  The operation is performed by the PCC as an MBB if the LSP remained in the UP state due to protection provided by FRR or a secondary path. If the LSP went down, then the update brings it into the UP state. A PCRpt message is sent by the PCC for each change to the state of the LSP during this process.
- a topology change that impacts a link in the path of a PCE-controlled LSP

  This topology change can be a change to the IGP metric, the TE metric, admin-group, or SRLG membership of an interface. This update is performed as an MBB by the PCC.

- the user performed a manual resignal of PCE-controlled RSVP-TE LSP path from the NRC-P

  This update is performed as an MBB by the PCC.
- the user performed a Global Concurrent Optimization (GCO) on a set of PCE-controlled RSVP-TE LSPs from the NRC-P

  This update is performed as an MBB by the PCC.

The procedures for the path update are the same as those for an SR-TE LSP. See *LSP Initiation* for more information. However, the PCUpd message from the PCE does not contain the label for each hop in the computed ERO. PCC then signals the path using the ERO returned by the PCE and, if successful, programs the data path, then sends the PCRpt message with the resulting RRO and hop labels provided by RSVP-TE signaling.

If the signaling of the ERO fails, then the ingress LER returns a PCErr message to PCE with the LSP Error code field of the LSP-ERROR-CODE TLV set to a value of 8 (RSVP signaling error).

If an RSVP-TE LSP has the **no adaptive** option set, the ingress LER cannot perform an MBB for such an LSP. A PCUpd message received from the PCE is then failed by the ingress LER, which returns a PCErr message to PCE with the LSP Error code field of the LSP-ERROR-CODE TLV set to a value of 8 (RSVP signaling error).

When the NRC-P reoptimizes the path of a PCE-controlled RSVP-TE LSP, it is possible that a path that satisfies the constraints of the LSP no longer exists. In this case, the NRC-P sends a PCUpd message with an empty ERO, which forces the PCC to bring down the path of the RSVP-TE LSP.

NRC-P sends a PCUpd message with an empty ERO if the following cases are true.

- The requested bandwidth is the same as current bandwidth, which avoids bringing down the path on a resignal during a MBB transition.
- Local protection is not currently in use, which avoids bringing down a path that activated an FRR backup path. The LSP can remain on the FRR backup path until a new primary path can be found by NRC-P.
- The links of the current path are all operationally up, which allows NRC-P to make sure that the RSVP control plane will report the path down when a link is down and not prematurely bring the path down with an empty ERO.

Behavior of LSP MBB

In addition to the Make-Before-Break (MBB) support when the PCC receives a path update, as described in *Behavior of the LSP Path Update*, an RSVP-TE LSP supports the MBB procedure for any parameter configuration change, including the PCEP-related commands when they result in a change to the path of the LSP.

If the user adds or modifies the **path-profile** command for an RSVP-TE LSP, a Config Change MBB is only performed if the **path-computation-method pce**, **pce-report**, or **pce-control** options are enabled on the LSP. Otherwise, no action occurs. When **path-computation-method pce**, **pce-report**, or **pce-control** are enabled on the LSP, the Path Update MBB (**tools perform router mpls update-path**) will be failed, resulting in a no operation.

MBB is also supported for the Manual Resignal and Auto-Bandwidth MBB types.

When the LSP goes into a MBB state at the ingress LER, the behavior is dependent on the LSP's operating mode.

PCE-Controlled LSP

The LSP MBB procedures for a PCE-controlled LSP (**pce-control** enabled) are as follows.

Items *1* through *5* of the following procedures apply to the Config Change, Manual Resignal, and Auto-Bandwidth MBB types. The Delayed Retry MBB type used with the SRLG on secondary standby LSP feature is not supported with a PCE controlled LSP. See *Behavior of Secondary LSP Paths* for information about the SRLG on secondary standby LSP feature.

Procedure

1. PCC temporarily removes delegation by sending a PCRpt message for the corresponding PLSP-ID with the delegation D-bit clear.
2. For an LSP with **path-computation-method** disabled, MPLS submits a path request to the local CSPF including the updated path constraints.
3. For an LSP with **path-computation-method pce** enabled, PCC issues a PCReq for the same PLSP-ID and includes the updated constraints in the metric, LSPA, or bandwidth objects.

   The bandwidth object contains the current operational bandwidth of the LSP in the case of the auto-bandwidth MBB.

   - If the PCE successfully finds a path, it replies with a PCRep message with the ERO.
   - If the PCE does not find a path, it replies with a PCRep message containing the No-Path object.

4. If the local CSPF or the PCE return a path, the PCC performs the following actions.
   a. PCC signals the LSP with RSVP control plane and moves traffic to the new MBB path. It then sends a PCRpt message with the delegation D-bit set to return delegation and containing the RRO and LSP object, with the LSP identifiers TLV containing the LSP-ID of the new MBB path. The message includes the metric, LSPA, and bandwidth objects where the P-flag is clear, which indicates the operational values of these parameters. Unless the user disabled the **report-path-constraints** option under the **pcc** context, the PCC also includes a second set of metric, LSPA, or bandwidth objects with the P-flag set to convey to PCE the constraints of the path.
   b. PCC sends a PathTear message to delete the state of the older path in the network. PCC then sends a PCRpt message to PCE with the older path PLSP-ID and the remove R-bit set to also have PCE remove the state of that LSP from its database.
5. If the local CSPF or the PCE returns no path or the RSVP-TE signaling of the returned path fails, the router makes no further requests. That is, there is no retry for the MBB.
   a. The PCC sends a PCErr message to PCE with the LSP Error code field of the LSP-ERROR-CODE TLV set to a value of 8 (RSVP signaling error) if the MBB failed due to a RSVP-TE signaling error.
   b. The PCC sends a PCRpt message with the delegation D-bit set to return delegation and containing the RRO and LSP objects with the LSP identifiers TLV containing the LSP-ID of the currently active path. The message includes the metric, LSPA, and bandwidth objects with the P-flag is clear to indicate the operational values of these parameters. Unless the user disabled the **report-path-constraints** option under the **pcc** context, the PCC also includes a second set of metric, LSPA, and bandwidth objects with the P-flag set to convey to PCE the constraints of the path.

6. The ingress LER takes no action in the case of a network event triggered MBB, such as FRR Global Revertive, TE Graceful Shutdown, or Soft Pre-Emption.
   a. The ingress PE keeps the information as required and sets the state of MBB to one of the FRR global Revertive, TE Graceful Shutdown, or Soft Pre-emption MBB values but does not perform the MBB action.
   b. The NRC-P computes a new path in the case of Global Revertive MBB due to a failure event. This computation uses the PCUpd message to update the path using the MBB procedure described in *Behavior of the LSP Path Update*. The activation of a bypass LSP by a PLR in the network causes the PCC to issue an updated PCRpt message with the new RRO reflecting the PLR and RRO hops. PCE should release the bandwidth on the links that are no longer used by the LSP path.
   c. The NRC-P computes a new path in the case of the TE graceful MBB if the RSVP-TE is using the TE metric, because the TE metric of the link in TE graceful shutdown is set to infinity. This computation uses the PCUpd message to update the path using the MBB procedure described in *Behavior of the LSP Path Update*.
   d. The NRC-P does not act on the TE graceful MBB if the RSVP-TE is using the IGP metric or is on the soft pre-emption MBB; however, the user can perform a manual resignal of the LSP path from the NRC-P to force a new path computation, which accounts for the newly available bandwidth on the link that caused the MBB event. This computation uses the PCUpd message to update the path using the MBB procedure described in *Behavior of the LSP Path Update*.
   e. The user can perform a manual resignal of the LSP path from the ingress LER, which forces an MBB for the path as per the remove-delegation/MBB/return-delegation procedures described in this section.
   f. If the user performs **no pce-control** while the LSP still has the state for any of the network event triggered MBBs, the MBB is performed immediately by the PCC as described in the procedures in *PCE-Computed LSP* for a PCE-computed LSP and as described in the procedures in *PCC-Controlled LSP* for a PCC-controlled LSP.

7. The timer-based resignal MBB behaves like the TE graceful or soft pre-emption MBB. The user can perform a manual resignal of the LSP path from the ingress LER or from PCE.

8. The Path Update MBB (**tools perform router mpls update-path**) is failed and will result in a no operation. This is true in all cases when the RSVP-TE LSP enables the **pce-report** option.

PCE-Computed LSP

All MBB types are supported for PCE-computed LSP. The LSP MBB procedures for a PCE-computed LSP (**path-computation-method pce** enabled and **pce-control** disabled) are as follows.

Procedure

1. PCC issues a PCReq for the same PLSP-ID and includes the updated constraints in the metric, LSPA, and bandwidth objects.

   - If PCE successfully finds a path, it replies with a PCRep message with the ERO.
   - If PCE does not find a path, it replies with a PCRep message containing the No-Path object.

2. If the PCE returns a path, the PCC signals the LSP with RSVP control plane and moves traffic to the new MBB path.

   If **pce-report** is enabled for this LSP, the PCC sends a PCRpt message with the delegation D-bit clear to retain control and containing the RRO and LSP object with the LSP identifiers TLVs containing the LSP-ID of the new MBB path. The message includes the metric, LSPA, and bandwidth objects where the P-flag is clear, which indicates the operational values of these parameters. Unless the user disables

the **report-path-constraints** option under the **pcc** context, PCC also includes a second set of metric, LSPA, and bandwidth objects with the P-flag set to convey to PCE the constraints of the path.

3. If the PCE returns no path or the RSVP-TE signaling of the returned path failed, MPLS puts the LSP into retry mode and sends a request to PCE every *retry-timer* seconds and up to the value of *retry-count*.

4. When the **pce-report** is enabled for the LSP and the FRR Global Revertive MBB is triggered following a bypass LSP activation by a PLR in the network, PCC issues an updated PCRpt message with the new RRO reflecting the PLR and RRO hops. PCE releases the bandwidth on the links that are no longer used by the LSP path.

5. If the user changes the RSVP-TE LSP configuration from **path-computation-method pce** to **no path-computation-method**, then MBB procedures are not supported. In this case, the LSP path is torn down and is put into retry mode to compute a new path from the local CSPF on the router to signal it.

PCC-Controlled LSP

All MBB types are supported for PCC-controlled LSP. The LSP MBB procedures for a PCC-controlled LSP (**path-computation-method pce** and **pce-control** disabled) are as follows.

Procedure

1. MPLS submits a path request, including the updated path constraints, to the local CSPF.

2. If the local CSPF returns a path, PCC signals the LSP with RSVP control plane and moves traffic to the new MBB path.

   If **pce-report** is enabled for this LSP, the PCC sends a PCRpt message with the delegation bit clear to retain control and containing the RRO and LSP object with the LSP identifiers TLV containing the LSP-ID of the new MBB path. It includes the metric, LSPA, and bandwidth objects where the P-flag is clear, which indicates the operational values of these parameters. Unless the user disables the **report-path-constraints** option under the **pcc** context, PCC also includes a second set of metric, LSPA, and bandwidth objects with the P-flag set to convey to PCE the constraints of the path.

3. If the CSPF returns no path or the RSVP-TE signaling of the returned path fails, MPLS puts the LSP into retry mode and sends a request to the local CSPF every *retry-timer* seconds and up to the value of *retry-count*.

4. When **pce-report** is enabled for the LSP and the FRR Global Revertive MBB is triggered following a bypass LSP activation by a PLR in the network, PCC issues an updated PCRpt message with the new RRO reflecting the PLR and RRO hops. PCE releases the bandwidth on the links that are no longer used by the LSP path.

6.1.5.2.5 Behavior of Secondary LSP Paths

Each of the primary, secondary standby, and secondary non-standby paths of the same LSP must use a separate PLSP-ID. In the PCE function of the NSP, the NRC-P, checks the LSP-IDENTIFIERS TLVs in the LSP object and can identify which PLSP-IDs are associated with the same LSP or the same RSVP session. The parameters are the IPv4 Tunnel Sender Address, the Tunnel ID, the Extended Tunnel ID, and the IPv4 Tunnel Endpoint Address. This approach allows the use of all the PCEP procedures for all three types of LSP paths.

PCC indicates to PCE the following states for the path in the LSP object: down, up (signaled but is not carrying traffic), or active (signaled and carrying traffic).

PCE tracks active paths and displays them in the NSP GUI. It also provides only the tunnel ID of an active PLSP-ID to a specific destination prefix when a request is made by a service or a steering application.

PCE recomputes the paths of all PLSP-IDs that are affected by a network event. The user can select each path separately on the NSP GUI and trigger a manual resignal of one or more paths of the LSP.

Note: Enabling the **srlg** option on a secondary standby path results in a **no** operation. The NRC-P supports link and SRLG disjointness using the PCE path profile, and the user can apply to the primary and secondary paths of the same LSP. See *PCE Path Profile Support* for more information.

6.1.5.2.6 PCE Path Profile Support

The PCE path profile ID and path group ID are configured at the LSP level.

The NRC-P can enforce path disjointness and bidirectionality among a pair of forward and a pair of reverse LSP paths. Both pairs of LSP paths must use a unique path group ID along with the same Path Profile ID, which is configured on the NRC-P to enforce path disjointness or path bidirectionality constraints.

When the user wants to apply path disjointness and path bidirectionality constraints to LSP paths, it is important to follow the following guidelines. The user can configure the following sets of LSP paths.

- Configure a set consisting of a pair of forward LSPs and a pair of reverse LSPs each with a single path, primary or secondary. The pair of forward LSPs can originate and terminate on different routers. The pair of reverse LSPs must mirror the forward pair. In this case, the path profile ID and the path group ID configured for each LSP must match. Because each LSP has a single path, the bidirectionality constraint applies automatically to the forward and reverse LSPs, which share the same originating node and the same terminating routers.
- Configure a pair consisting of a forward LSP and a reverse LSP, each with a primary path and a single secondary path, or each with a couple of secondary paths. Because the two paths of each LSP inherit the same LSP level path profile ID and path group ID configuration, the NRC-P path computation algorithm cannot guarantee that the primary paths in both directions meet the bidirectionality constraint. That is, it is possible that the primary path for the forward LSP shares the same links as the secondary path of the reverse LSP and vice-versa.

6.1.6 LSP Path Diversity and Bidirectionality Constraints

The PCE path profile defined in *draft-alvarez-pce-path-profiles* is used to request path diversity or a disjoint for two or more LSPs originating on the same or different PE routers. It is also used to request that paths of two unidirectional LSPs between the same two routers use the same TE links. This is referred to as the bidirectionality constraint.

Path profiles are defined by the user directly on the NRC-P Policy Manager with a number of LSP path constraints, which are metrics with upper bounds specified, and with an objective, which are metrics optimized with no bound specified. The NRC-P Policy Manager allows the following PCE constraints to be configured within each PCE Path Profile:

- path diversity, node-disjoint, link-disjoint

- path bidirectionality, symmetric reverse route preferred, symmetric reverse route required
- maximum path IGP metric (cost)
- maximum path TE metric
- maximum hop count

The user can also specify which PCE objective to use to optimize the path of the LSP in the PCE Path Profile:

- IGP metric (cost)
- TE metric
- hops (span)

The CSPF algorithm will optimize this objective. If a constraint is provided for the same metric, then the CSPF algorithm makes sure that the selected path achieves a lower or equal value to the bound specified in the constraint.

For hop-count metrics, if a constraint is sent in a METRIC object, and is also specified in a PCE profile referenced by the LSP, the constraint in the METRIC object is used.

For IGP and TE metrics, if an objective is sent in a METRIC object, and is also specified in a PCE profile referenced by the LSP, the objective in the Path Profile is used.

The constraints in the Bandwidth object and the LSPA object, specifically the include/exclude admin-group constraints and setup and hold priorities, are not supported in the PCE profile.

In order to indicate the path diversity and bidirectionality constraints to the PCE, the user must configure the profile ID and path group ID of the PCE path that the LSP belongs to. The CLI for this is described in the *Configuring and Operating SR-TE* section. The path group ID does not need to be defined in the PCE as part of the path profile configuration, and identifies implicitly the set of paths which must have the path diversity constraint applied.

The user can only associate a single path group ID with a specific PCE path profile ID for a given LSP. However, the same path group ID can be associated with multiple PCE profile IDs for the same LSP.

The path profiles are inferred using the path ID in the path request by the PCC. When the PE router acting as a PCC wants to request path diversity from a set of other LSPs belonging to a path group ID value, it adds a new path profile object into the PCReq message. The object contains the path profile ID and the path group ID as an extended ID field. In other words, the diversity metric is carried in an opaque way from PCC to PCE.

The bidirectionality constraint operates the same way as the diversity constraint. The user can configure a PCE profile with both the path diversity and bidirectionality constraints. PCE will check if there is an LSP in the reverse direction which belongs to the same path group ID as an originating LSP it is computing the path for, and will enforce the constraint.

In order for the PCE to be aware of the path diversity and bidirectionality constraints for an LSP that is delegated but for which there is no prior state in the NRC-P LSP database, the path profile object is included in the PCRpt message with the P-flag set in the common header to indicate that the object must be processed.

Table 33: PCEP Path Profile Extension Objects and TLVs describes the new objects introduced in the PCE path profile.

Table 33: PCEP Path Profile Extension Objects and TLVs

TLV, Object, or Message	Contained in Object	Contained in Message
PATH-PROFILE-CAPABILITY TLV	OPEN	OPEN
PATH-PROFILE Object	—	PCReq, PCRpt, PCInitiate

A path profile object can contain multiple TLVs containing each profile-id and extend-id, and should be processed properly. If multiple path profile objects are received, the first object is interpreted and the others are ignored. The PCC and the PCE support all PCEP capability TLVs defined in this chapter and will always advertise them. If the OPEN object received from a PCEP speaker does not contain one or more of the capabilities, the PCE or PCC will not use them during that PCEP session.

6.1.7 Path Computation Fallback for PCC-Initiated LSPs

For PCC-initiated RSVP-TE and SR-TE LSPs, the router supports fallback to a local path computation method in the case where the configured PCEP sessions are down or the PCE is unreachable, or when all configured PCEs are signaling overload and the redelegation timer expires while all configured LSPs are signaling overload so that the LSP cannot be redelegated. The fallback method can be configured to be the local CSPF or none. In the latter case, MPLS uses the hop-to-label translation (SR-TE LSPs) or the explicit IGP path (RSVP-TE LSPs).

This capability is supported by both active and passive stateful LSPs. Active stateful LSPs are fully delegated to the PCE by being both PCE computed (**path-computation-method pce**) and PCE controlled. Passive stateful LSPs are PCE computed.

Note: For the passive stateful case, it is important that the **retry-timer** and **retry-limit** values exceed the **redelegation-timer** value, otherwise, the LSP may go operationally down before the fallback path computation has occurred.

A fallback path computation method is configured as follows:

```
configure>router>
   mpls
      lsp <xyz>
         pce-control
         path-computation-method {pce | local-cspf}
         fallback-path-computation-method {none | local-cspf}
```

If **none** is configured, MPLS uses the default method based on the configured path, which is hop-to-label path computation for SR-TE LSPs and IGP-based path computation for RSVP-TE LSPs.

The **fallback-path-computation-method** command is only valid for **path-computation-method pce**, irrespective of whether **pce-control** is configured. It is mutually exclusive with the **path-computation-method local-cspf** and **no path-computation-method** commands.

The fallback mechanism is only triggered if PCC informs MPLS that PCEP is down. It is not triggered while the PCC is administratively down or not yet configured.

Note: On the first local path computation following a fallback, MPLS is not aware of the list of SRLGs or administrative groups that are used by the original path computed by PCE. As a result, MPLS can only provide a list of hops or links to avoid on the first computation.

PCE reports are sent, where applicable, with the delegation bit cleared.

6.2 TE-DB and LSP-DB Partial Synchronization

VSR-NRC to NSP partial synchronization of TE database (TE-DB) and LSP database (LSP-DB) allows the VSR-NRC to send incremental TE-DB and LSP-DB records to the NRC-P when the CPROTO session flaps. Without this feature, a full synchronization of all records is performed each time the CPROTO session flaps which increases the convergence time of the NRC-P PCE.

With partial synchronization, VSR-NRC keeps track of the last record acknowledged by the NRC-P before the CPROTO session went down. When the CPROTO session is re-established, VSR-NRC sends only the records received from the network after the last acknowledged record.

Figure 47: TE and LSP Database Partial Synchronization illustrates the behavior of the partial synchronization of the TE-DB and LSP-DB.

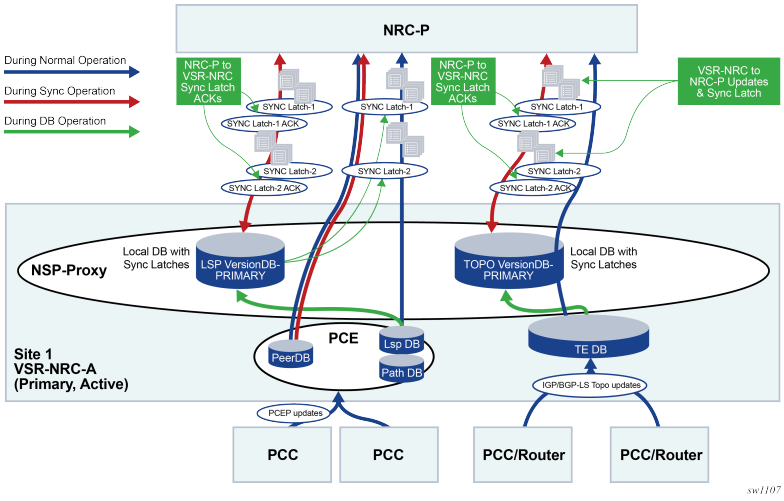

Figure 47: TE and LSP Database Partial Synchronization

DB refers to the local TE-DB or LSP-DB maintained by the NSP-PROXY on the VSR-NRC.

The phrase Version-DB refers to the new copy of the same DB, TE-DB or LSP-DB, augmented with synchronization latches, and is used during the synchronization process to play back the records and latches to the NRC-P.

The main processes of the partial synchronization feature are as follows.

1. Enhancements to DB maintenance in active VSR-NRC:

   a. The active VSR-NRC, VSR-NRC-A or VAR-NRC-B (see *VSR-NRC 1+1 Redundancy*), maintains a local copy of the database, referred to as Version-DB, for each record NSP-PROXY sends to NRC-P.
   b. Every 10 seconds, the NSP-PROXY also sends to NRC-P a message which contains a Latch Reference ID (LRID) only.
   c. The active VSR-NRC maintains a latch in Version-DB for each LRID sent to NRC-P. This latch is used to identify the start point for SYNC_START message processing based on the LRID received from NRC-P.

2. Enhancements to active VSR-NRC and NRC-P DB synchronization:

   a. After opening a PCEP or a BGP_LS service channel, the NRC-P in the primary site sends a SYNC_START with an LRID (empty for full sync) to the active VSR-NRC (VSR-NRC-A or VSR-NRC-B).
   b. The NSP-PROXY on the active VSR-NRC begins sync with the NRC-P of the Version-DB records, which include saved interleaved synchronization latches, from LRID specified in SYNC_START. An empty SYNC_START means the full content of the Version-DB is played back to NRC-P.
   c. After processing all the records up to the specified LRID, NRC-P sends an acknowledgement of that LRID to NSP-PROXY using SYNC_ACK.
   d. After initial sync, the NSP-PROXY on the active VSR-NRC resumes sending records from the main DB and inserts every 10 seconds an LRID message between the records it sends to NRC-P. The same stream of records interleaved with LRID messages is saved in the local Version-DB.
   e. As in the initial synchronization phase, NRC-P acknowledges back an LRID to NSP-PROXY using a SYNC_ACK after the complete processing of all the records up to that LRID.

Figure 48: TE-DB Partial Synchronization Message Sequence shows the enhancements to the CPROTO protobuf messages exchanged between VSR-NRC and NRC-P to provide full or partial synchronization of the TE-DB.

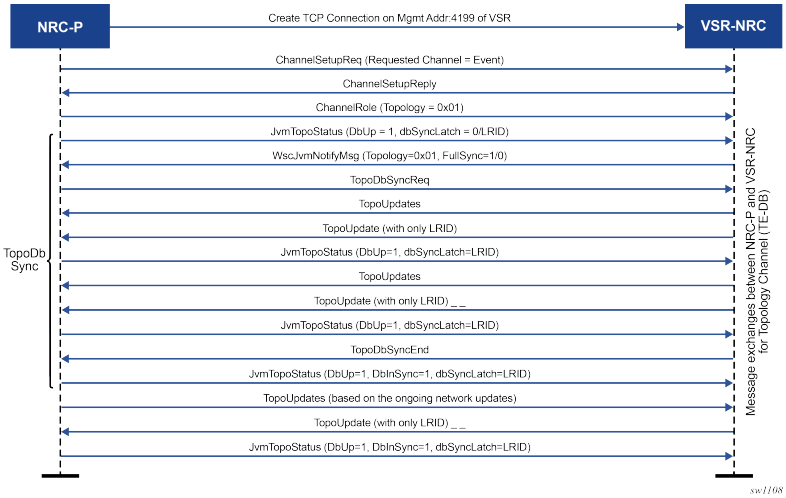

Figure 48: TE-DB Partial Synchronization Message Sequence

Figure 49: LSP-DB Partial Synchronization Message Sequence shows the enhancements to the CPROTO protobuf messages exchanged between VSR-NRC and NRC-P to provide full or partial synchronization of the LSP-DB.

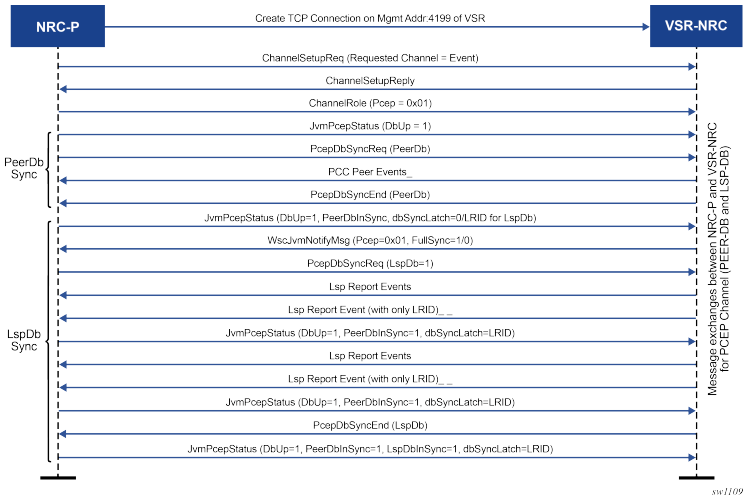

Figure 49: LSP-DB Partial Synchronization Message Sequence

6.3 NSP and VSR-NRC PCE Redundancy

This feature introduces resilience support to the PCE and PCC capabilities.

6.3.1 Overview of NSP Ecosystem Redundancy

The NSP ecosystem resilience consists of local, or single site, and remote, or dual site, redundancy mechanisms.

6.3.1.1 Redundancy in a Single Site Deployment

Figure 50: NSP Ecosystem Redundancy in Single-Site Deployment illustrates the NSP ecosystem and provisioning of redundancy within a single-site deployment.

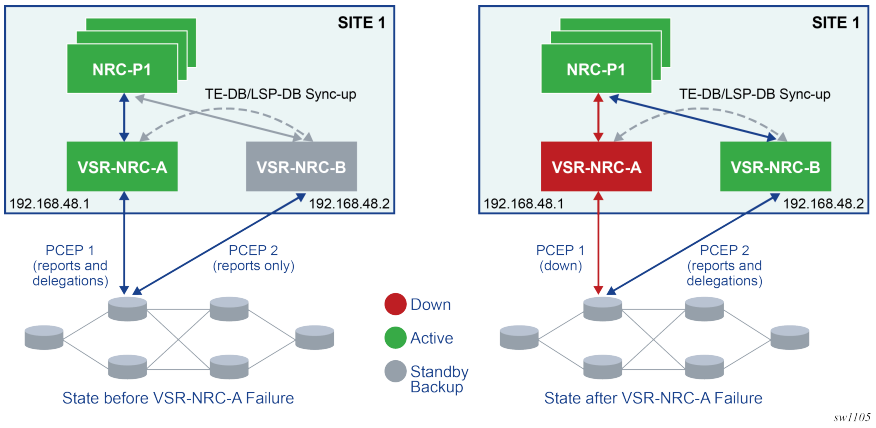

Figure 50: NSP Ecosystem Redundancy in Single-Site Deployment

The NSP, where the NRC-P component resides, is protected by a cluster of 3 Virtual Machines (VMs). This local redundancy scheme elects one VM as the active and the other two VMs become standby backups. NSP must always be deployed in a cluster of 3 VMs.

The VSR-NRC module runs the integrated SR OS virtual SIM model (1 CPM+IOM VM) and can be deployed standalone.

The VSR-NRC can be protected with a VM cluster implementing the 1+1 redundancy scheme. The local redundancy provides for continuous full and partial synchronization of the TE-DB and LS-DB between the active VSR-NRC and the backup standby VSR-NRC, as well as with the local NRC-P.

The details of the VSR-NRC 1+1 single-site redundancy mechanism are provided in *VSR-NRC 1+1 Single-Site Redundancy*.

6.3.1.2 Redundancy in a Dual Site Deployment

Figure 51: NSP ecosystem Redundancy in Dual-Site Deployment illustrates the NSP ecosystem and provisioning of redundancy within a dual-site deployment.

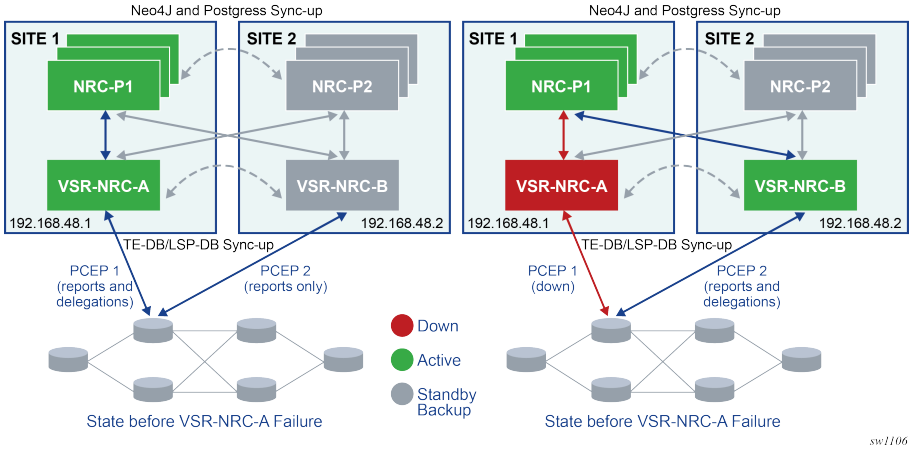

Figure 51: NSP ecosystem Redundancy in Dual-Site Deployment

Both local and remote redundancy are deployed. The remote redundancy, sometimes referred to as Disaster Recovery (DR) or geo redundancy, consists of a primary site and a secondary backup site each with an NSP VM cluster and a single VSR-NRC VM.

A heartbeat protocol runs between the NSP clusters in the primary site and the standby backup sites.

The VSR-NRC connects to both the NRC-P within its own site and to the NRC-P in the remote site. A failover to the remote VSR-NRC occurs when the primary site fails entirely, when the primary NRC-P fails, and when the VSR-NRC only fails.

TE-DB and LSP-DB full and partial synchronization among the cluster of two VSR-NRCs improves the coverage of remote redundancy. In addition, the VSR-NRC 1+1 redundancy scheme is extended to the remote site. The details of the VSR-NRC 1+1 dual site redundancy are provided in *VSR-NRC Dual-Site Redundancy*.

6.3.2 PCC and PCE Configuration

The following CLI command enables the configuration on the PCC of a second PCEP session to the secondary backup PCE peer. A **preference** parameter value is used to indicate the primary or the secondary backup PCE peer role:

configure router pcep pcc peer *ip-address* [**preference** *preference*]

A maximum of two PCE peers are supported. The PCE peer that is not in overload is always selected by the PCC as the active PCE. However, if neither of the PCEs are signaling the overload state, the PCE with the higher numerical preference value is selected. In case of a tie, the PCE with the lower IP address is selected.

To change the value of the **preference** parameter, the peer must be deleted and recreated.

CPROTO channels are established through the management port and, by default, use the local system address and open TCP port 4199 on both the primary and secondary VSR-NRCs.

In addition, the NRC-P always provides the active VSR-NRC acting as a CPROTO server with the system address of the mate VSR-NRC which to initiate the CPROTO channel to. The address is provided using the new Global Health and Notification CPROTO channel.

NRC-P provides a configuration for the primary VSR-NRC. This is the preferred active VSR-NRC. The other VSR-NRC is secondary. It is recommended to set the VSR-NRC co-located with the NRC-P as the primary VSR-NRC to take advantage of lowest latency and more reliable CPROTO channel.

In *Figure 51: NSP ecosystem Redundancy in Dual-Site Deployment*, the primary VSR-NRC in the local site is VSR-NRC-A and the secondary VSR-NRC in the remote site is VSR-NRC-B. The reverse configuration is performed in the remote site. With single-site VSR-NRC redundancy, both VSR-NRCs are local and either can be configured as the primary VSR-NRC.

6.3.3 NSP Cluster Redundancy

The following describes NSP cluster redundancy rules:

- At each site, a master is elected among the cluster of three VMs. In a DR deployment, the cluster in one site is designated as the primary, meaning it is the preferred active cluster. The site is referred to as the primary site. The second cluster and site are referred to as secondary and therefore act as the standby backup cluster or site.
- The application processes at the standby site are shut down, but the neo4j and other databases are synchronized with the primary active site.
- Switching to the secondary standby site can be initiated manually or by using an automated approach stemming from the loss of heartbeat between the primary and standby sites.
- When the NSP cluster at the primary active site is down (two out of three servers must be inactive, shut down, or failed), the heartbeat mechanism between the primary and standby NSP clusters fails after three timeouts. This initiates the active role at the secondary standby site.
- When the NSP cluster at the primary site is back up, the heartbeat mechanism between the primary/standby and secondary/active NSP clusters is restored. The primary site can be restored to the active role manually. Automatic reversion to the primary NSP cluster is not supported.

6.3.4 VSR-NRC 1+1 Redundancy

This feature implements support for the single site or local 1+1 redundancy of the VSR-NRC .

6.3.4.1 VSR-NRC 1+1 Single-Site Redundancy

Single-site or local 1+1 redundancy of the VSR-NRC relies on extending the communication of the NSP-PROXY to synchronize the contents of the TE-DB and LSP-DB between the primary VSR-NRC (VSR-NRC-A) and the secondary VSR-NRC (VSR-NRC-B). This is supported by the CPROTO sync channel running between the two VSR-NRCs.

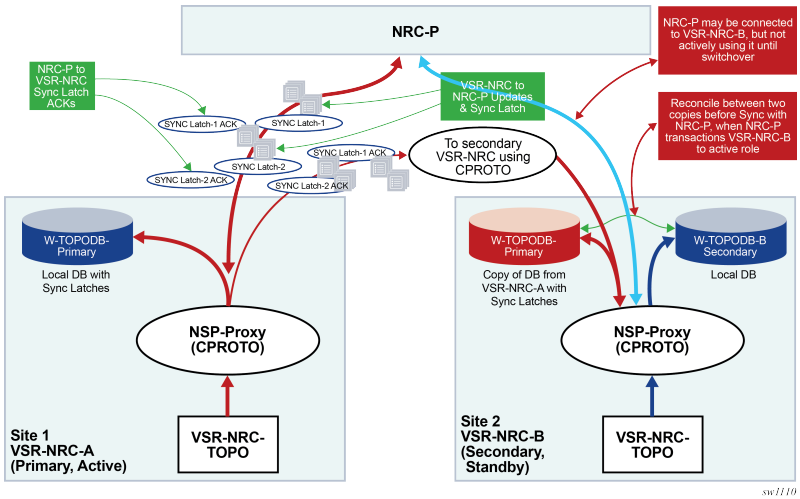

Figure 52: VSR-NRC 1+1 Single-Site Redundancy

6.3.4.1.1 Initial Establishment of Active/ Standby VSR-NRC Roles

The initial establishment of active/standby VSR-NRC roles is as follows.

1. NRC-P performs up to 3 attempts at 10-second intervals to establish the global CPROTO channel to the primary VSR-NRC (for example, VSR-NRC-A in the primary site). If not successful, it performs another 3 attempts to the secondary VSR-NRC (VSR-NRC-B in primary site). This process is continued cycling among the two VSR-NRCs until the global channel is established to either VSR-NRC-A or VSR-NRC-B which then becomes the target active VSR-NRC. The CPROTO channel establishment attempts continue to the other VSR-NRC which becomes the target standby VSR-NRC.

   This CPROTO channel establishment process is also followed when the global CPROTO channel to the active VSR-NRC goes down beginning always with an attempt to the primary VSR-NRC (VSR-NRC-A) and then onto the secondary VSR-NRC (VSR-NRC-B) and continuing to cycling between them until a channel is established.
2. At successful global CPROTO channel establishment, the NRC-P sends a notify message (wscIsActive=TRUE, mateAddr, matePort) requesting a transition to active role from the VSR-NRC (VSR-NRC-A or VSR-NRC-B).
3. When NRC-P also established the TOPO and PCEP CPROTO channels, the active VSR-NRC begins the partial database synchronization procedures to NRC-P, as mentioned in *TE-DB and LSP-DB Partial Synchronization*.

4. When the reconcile process is complete, the active VSR-NRC acknowledges the NRC-P by sending a notify reply message (notifyReply=TRUE).
5. The active VSR-NRC attempts to establish a CPROTO sync channel to the mate VSR-NRC. After it is successfully established, it begins the full or partial database synchronization procedures to the mate VSR-NRC following the procedures, as mentioned in *TE-DB and LSP-DB Partial Synchronization*.
6. After the global CPROTO channel to the target standby VSR-NRC is established, NRC-P sends a notify message (wscIsActive=FALSE, mateAddr, matePort) requesting transition to a standby role. The target standby VSR-NRC acknowledges by sending a notify reply message (notifyReply=TRUE). No database synchronization occurs between the standby VSR-NRC and the NRC-P.
7. The standby VSR-NRC maintains a copy of the mate active VSR-NRC database and independently builds its own database using BGP-LS and PCEP peerings with the network.
8. When the active and standby roles have been assigned by the NRC-P, the primary and secondary VSR-NRCs keep that role until further notice from NRC-P. The roles are not affected by the state of the CPROTO sync channel.

6.3.4.1.2 Failover to Backup VSR-NRC

The failover to backup VSR-NRC process is as follows.

1. If the active VSR-NRC fails, the NRC-P detects it when the global CPROTO channel goes down.

   The NRC-P uses a keep-alive timer of 60 seconds and a multiplier of 2.2 for a total keep-alive timeout of 132 seconds.

   At keep-alive timeout, NRC-P determines that the channel is idle and closes it. NRC-P also closes the TOPO and PCEP CPROTO channels, if they are not already down.
2. Next, the NRC-P begins its CPROTO establishment cycle as detailed in *Initial Establishment of Active/ Standby VSR-NRC Roles*. If the failed VSR-NRC is the primary VSR-NRC (VSR-NRC-A in the primary site), three attempts are performed to bring the global CPROTO channel back up. If the global CPROTO channel is successfully restored, the primary VSR-NRC remains the target active VSR-NRC. If not, three attempts are performed to the secondary VSR-NRC (VSR-NRC-B) and so on.

   Note: It can take up to 162 seconds for the NRC-P to switch to the secondary standby VSR-NRC. This includes the keep-alive idle time of 132 seconds plus up to 3 attempts of 10-second intervals to establish the global CPROTO channel, 3 unsuccessful attempts to primary VSR-NRC and one successful attempt to the secondary VSR-NRC. If the user reboots the primary VSR-NRC (VSR-NRC-A) while it is active, it may come back up faster and therefore it will remain the target active VSR-NRC and NRC-P does not switch to secondary VSR-NRC (VSR-NRC-B).
3. When the global CPROTO channel is up to the target active VSR-NRC, either the primary or secondary VSR-NRC, NRC-P sends a notify message requesting active role and providing the IP address for the mate VSR-NRC (wscIsActive=TRUE, mateAddr, matePort).
4. The target active VSR-NRC begins the transition to the active role by flapping the CPROTO sync session to the mate VSR-NRC, if not already down. Then it begins reconciling the local copy of the

mate databases with its own network-learned databases. The common records only have a difference in the setting of the Delegate bit in the PCEP Report messages.

The following reconcile process is followed:

a. Common records of the local database are carried over with setting of the Delegate bit in the PCEP Report messages and with the LRID information from the mate database.
b. Records in the mate database which are not reconciled with the local database are deleted.
c. Records in the local database which are not reconciled with the mate database are always carried over.

Common records of the local database are preferred over those of the mate copy database except for the LRID. After the reconcile process is complete, the now newly active VSR-NRC destroys the mate copy database and acknowledges the NRC-P by sending a notify reply message (notifyReply=TRUE).

5. The newly active VSR-NRC begins the partial database synchronization procedures to NRC-P as described in *TE-DB and LSP-DB Partial Synchronization*. The newly active VSR-NRC stops accepting new records from its own TE-DB and LSP-DB until after it completes the reconcile between the mate copy databases and its local databases and completes the partial sync with NRC-P.
6. The newly active VSR-NRC notifies the local PCE process to set the PCEP overload to OFF and to start an overload timer, hard-coded to 10 minutes, for each PCEP session. At the receipt of the first PCEP redelegation from a PCC, VSR-NRC stops the timer for that PCC and sends a PCC ready message to NRC-P which can then begin sending update messages to that PCC. If the overload timer expires before receiving a PCEP redelegation message, the newly active VSR-NRC clears all delegations of the corresponding PCC toward NRC-P.
7. The newly active VSR-NRC attempts to establish a CPROTO channel to the mate VSR-NRC. After successfully establishing a channel, it begins the full or partial database synchronization procedures to the mate VSR-NRC following similar procedures as mentioned in *TE-DB and LSP-DB Partial Synchronization*.

6.3.4.1.3 Recovery of the Failed VSR-NRC

The recovery of the failed VSR-NRC process is as follows.

1. The recovered VSR-NRC assumes the role of None, meaning it is neither active nor standby, until the global channel is successfully established from the NRC-P. In this state, it does not accept a CPROTO SYNC channel from its mate VSR-NRC.
2. After successfully opening a global CPROTO channel, NRC-P sends a notify message (wscIsActive=FALSE, mateAddr, matePort) to the recovered VSR-NRC to request transition to standby role.
3. The recovered VSR-NRC then accepts the CPROTO SYNC channel from its mate and prepares the mate copy DBs to accept the updates from its mate VSR-NRC.
4. VSR-NRC sends to NRC-P a notify reply message (notifyReply=TRUE) to accept the standby role assigned by NSP.
5. The newly standby VSR-NRC notifies the local PCE process to set the PCEP overload to ON.
6. NRC-P keeps this global CPROTO channel alive by sending KAs at regular intervals. No other service CPROTO channel is created while this VSR-NRC is in standby role.
7. NRC-P does not automatically revert the active role to the recovered VSR-NRC. A manual reversion procedure is supported. See *Manual Switchover to the Mate VSR-NRC* for the reversion procedure.

6.3.4.1.4 Manual Switchover to the Mate VSR-NRC

The NRC-P provides an API to perform a manual switchover to the mate VSR-NRC. This could be used, for example, to revert the active role back to a recovered primary VSR-NRC.

1. NRC-P sends a notify message (wscIsActive=FALSE, mateAddr, matePort) over the global channel to currently active VSR-NRC to request transition to standby role.
2. The currently active VSR-NRC shuts down the CPROTO SYNC channel to its mate and prepares the mate copy DBs to accept the updates from its mate VSR-NRC.
3. The currently active VSR-NRC sends to NRC-P a notify reply message (notifyReply=TRUE) to accept the standby role requested.
4. NRC-P then follows the steps in *Initial Establishment of Active/ Standby VSR-NRC Roles* to transition the currently standby VSR-NRC to active role.

6.3.4.2 VSR-NRC Dual-Site Redundancy

The behavior of dual-site redundancy follows the single site redundancy procedures because only the active NRC-P can establish CPROTO channels to the pair of primary and secondary VSR-NRCs.

When the NRC-P in the secondary site becomes active, it attempts to establish a CPROTO global channel to both primary and secondary VSR-NRCs. Nokia recommends configuring the local VSR-NRC (VSR-NRC-B) as the primary VSR-NRC in the secondary site.

6.3.4.3 Global Health and Notification CPROTO Channel

The VSR-NRC 1+1 redundancy features introduces a global channel between each VSR-NRC and the NRC-P for exchanging channel health and notification messages that are not application-specific.

Figure 53: Global Health and Notification Channel Message Sequence illustrates the sequence of messages to establish the global health and notification channel as well as the sequence of messages used for establishing the VSR-NRC role of active or standby.

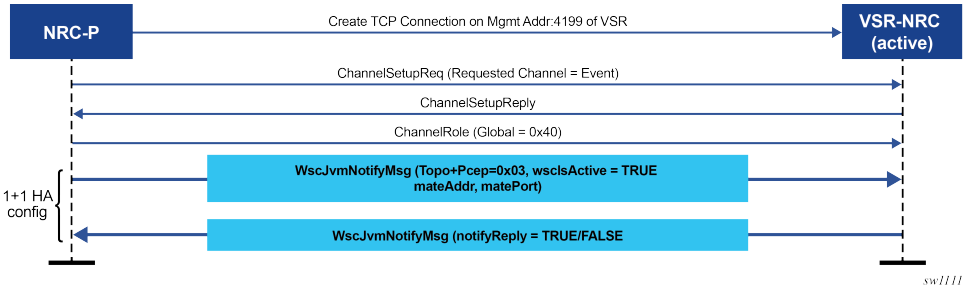

Figure 53: Global Health and Notification Channel Message Sequence

6.3.5 PCE Southbound and PCC Behavior

6.3.5.1 PCE Southbound Behavior

The following describes VSR-NRC PCE redundancy rules:

- steady state behavior
  - The PCC establishes a PCEP session to each of the primary active VSR-NRC and secondary standby VSR-NRC. The secondary standby VSR-NRC is either in the primary site with single site redundancy or at the secondary site with dual site redundancy, however, the secondary standby VSR-NRC sets PCEP sessions with the PCCs in the overload state. The VSR-NRC enters this PCEP overload state when its upstream CPROTO session to the NSP cluster is down, or is being instructed by the NRC-P to enter the standby state as described in *VSR-NRC 1+1 Redundancy*.
  - The VSR-NRC acting as a PCE signals the overload state to the PCCs in a PCEP notification message. While in the overload state, the VSR-NRC PCE accepts reports (PCRpt) without delegation but rejects requests (PCReq) and reject reports (PCRpt) with delegation. The VSR-NRC PCE also does not originate initiate messages (PCInitiate) and update messages (PCUpd).
  - The secondary standby VSR-NRC maintains its BGP and IGP peerings with the network and updates its TE database as a result of any network topology changes.
- primary active NSP cluster failure

  When the NSP cluster at the primary active site is down (two out of three servers must be inactive, shut down, or failed), the heartbeat mechanism between the primary active and secondary standby NSP clusters fails. This initiates the NSP cluster activity at the secondary standby site.

  The following are the procedures on the VSR-NRC:

  - The primary VSR-NRC detects CPROTO global channel failure and puts all its PCEP sessions to the PCCs into the overload state.
  - The NRC-P in the NSP cluster at the secondary site follows the procedures in *VSR-NRC 1+1 Redundancy* to transition the secondary VSR-NRC into active state.
  - The VSR-NRC at the primary site must also return the delegation of all LSPs back to the PCCs by sending an empty LSP Update Request that has the Delegate flag set to 0 as per RFC 8231. To accommodate third party PCE implementations which may not return delegations, each PCC will concurrently revoke the delegation of its LSPs from the primary VSR-NRC PCE. This allows the PCCs to delegate all eligible LSPs, including PCE-initiated LSPs, to the PCE function in the VSR-NRC at the secondary site. If the entire primary active site fails, the PCE side procedure in this step does not apply.
- VSR-NRC complex failure at the primary site (NSP server is still up)

  A VSR-NRC complex failure at the primary active NSP triggers the failover to backup VSR-NRC procedures in *VSR-NRC 1+1 Redundancy*.

6.3.5.2 PCC Behavior

The following describes PCC rules with PCE redundancy:

- PCCs can establish upstream PCEP sessions with at most two VSR-NRC PCEs.
- Each upstream session has a preference that takes effect when both upstream PCEP sessions are successfully established. The PCE peer that is not in overload is always selected by the PCC as the active PCE. However, if neither of the PCEs are signaling the overload state, the PCE with the higher numerical preference value is selected, and in case of a tie, the PCE with the lower IP address is selected.

- In the steady state, because one upstream VSR-NRC PCE is in overload, only one PCEP session is active. The PCCs send request messages (PCReq) active VSR-NRC PCE only. Similarly, the PCCs delegate an LSP using a report message (PCRpt) with the Delegate flag set to the active VSR-NRC PCE only. PCRpt messages are sent with the Delegate flag clear to the secondary standby VSR-NRC PCE in overload.
- If the current active PCEP session signals overload state, the PCC will select the other PCE as the active PCE as long as the corresponding PCEP session is not in overload. Any new path request message (PCReq) or path report message (PCRpt) with the Delegate flag set, is sent to the new PCE.

  The PCE in overload returns the delegation of all existing LSPs back to this PCC by sending an empty LSP Update message that has the Delegate flag set as per RFC 8231. To accommodate third party PCE implementations which may not return delegations, each PCC will concurrently revoke the delegation of its LSPs from the current PCE. The PCC will then delegate these LSPs to the new active PCE by sending a path report (PCRpt) with the Delegate flag set.
- If the current active PCEP session goes operationally down, the PCC starts the redelegation timer (default 90 seconds) and state timeout timer (default 180 seconds).

  – If the PCEP session is restored before the redelegation timer expires, no delegation change is performed and the LSP state is maintained.
  – Upon expiration of the redelegation timer, the PCC looks for the other PCEP session and, if not in overload, it immediately delegates the LSPs to the newly active PCE. If the new PCE accepts the delegation, the LSP state is maintained.
  – If the PCEP session does not recover before the redelegation timer expires and the PCC fails to find another active PCEP session, then by default the PCC clears the LSP state of PCE-initiated LSPs after state timeout expiry; the PCC deletes the PCE-initiated LSPs and releases all their resources. A configuration option of the redelegation timer CLI command allows the user to keep the state of the pce-initiated LSPs instead. The PCC does not clear the state of PCC-initiated LSPs; however, the user can do this by deleting the configuration.

6.4 Configuring and Operating RSVP-TE LSP with PCEP

This section provides information about configuring and operating RSVP-TE LSP with PCEP using CLI.

The following describes the detailed configuration of an inter-area RSVP-TE LSP with both a primary path and a secondary path. The network uses IS-IS with the backbone area in Level 2 and the leaf areas in Level 1. Topology discovery is learned by NRC-P using BGP-LS.

The LSP uses an admin-group constraint to keep the paths of the secondary and primary link disjoint in the backbone area. The LSP is PCE-controlled but also has **path-computation-method pce** enabled so the initial path, and any MBB path, is also computed by PCE.

The NSP and SR OS load versions used to produce this example are:

- NSP: NSP-2.0.3-rel.108
- PCE SR OS: TiMOS-B-0.0.W129
- PCC: TiMOS-B-0.0.I4902

Figure 54: Multi-level IS-IS Topology in the NSP GUI shows a multi-level IS-IS topology in the NSP GUI:

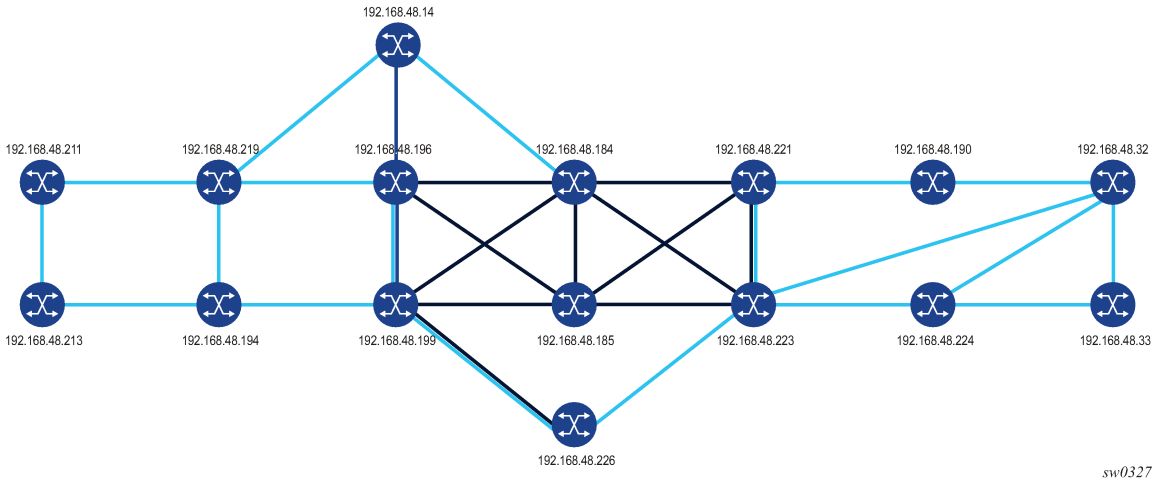

Figure 54: Multi-level IS-IS Topology in the NSP GUI

The following example shows the configuration and **show** command output of the PCEP on the PCE node and the PCC node.

```
*A:PCE Server 226>config>router>pcep>pce# info
----------------------------------------------
            local-address 192.168.48.226
            no shutdown
----------------------------------------------
*A:Reno 194>config>router>pcep>pcc# info
----------------------------------------------
            peer 192.168.48.226
                no shutdown
            exit
            no shutdown
----------------------------------------------

*A:PCE Server 226>config>router>pcep>pce# show router pcep pce status
===============================================================================
Path Computation Element Protocol (PCEP) Path Computation Element (PCE) Info
===============================================================================
Admin Status           : Up              Oper Status          : Up
Unknown Msg Limit      : 10 msg/min
Keepalive Interval     : 30 seconds      DeadTimer Interval   : 120 seconds
Capabilities List      : stateful-delegate stateful-pce segment-rt-path
Local Address          : 192.168.48.226
PCE Overloaded         : false
-------------------------------------------------------------------------------
PCEP Path Computation Element (PCE) Peer Info
-------------------------------------------------------------------------------
Peer                       Sync State             Oper Keepalive/Oper DeadTimer
-------------------------------------------------------------------------------
192.168.48.190:4189        done                   30/120
192.168.48.194:4189        done                   30/120
192.168.48.198:4189        done                   30/120
192.168.48.199:4189        done                   30/120
192.168.48.219:4189        done                   30/120
192.168.48.221:4189        done                   30/120
192.168.48.224:4189        done                   30/120
-------------------------------------------------------------------------------
===============================================================================

*A:Reno 194# show router pcep pcc status
===============================================================================
```

```
Path Computation Element Protocol (PCEP) Path Computation Client (PCC) Info
===============================================================================
Admin Status           : Up              Oper Status          : Up
Unknown Msg Limit      : 10 msg/min
Keepalive Interval     : 30 seconds      DeadTimer Interval   : 120 seconds
Capabilities List      : stateful-delegate stateful-pce segment-rt-path
Address                : 192.168.48.194
Report Path Constraints: True
-------------------------------------------------------------------------------
PCEP Path Computation Client (PCC) Peer Info
-------------------------------------------------------------------------------
Peer                       Admin State/Oper State Oper Keepalive/Oper DeadTimer
-------------------------------------------------------------------------------
192.168.48.226             Up/Up                  30/120
-------------------------------------------------------------------------------
===============================================================================

*A:Reno 194# show router pcep pcc lsp-db
===============================================================================
PCEP Path Computation Client (PCC) LSP Update Info
===============================================================================
PCEP-specific LSP ID: 11
LSP ID               : 14378             LSP Type            : rsvp-p2p
Tunnel ID            : 1                 Extended Tunnel Id  : 192.168.48.194
LSP Name             : From Reno to Atlanta RSVP-TE::primary_empty
Source Address       : 192.168.48.194    Destination Address : 192.168.48.224
LSP Delegated        : True              Delegate PCE Address: 192.168.48.226
Oper Status          : active
-------------------------------------------------------------------------------
PCEP-specific LSP ID: 12
LSP ID               : 14380             LSP Type            : rsvp-p2p
Tunnel ID            : 1                 Extended Tunnel Id  : 192.168.48.194
LSP Name             : From Reno to Atlanta RSVP-TE::secondary_empty
Source Address       : 192.168.48.194    Destination Address : 192.168.48.224
LSP Delegated        : True              Delegate PCE Address: 192.168.48.226
Oper Status          : up
===============================================================================
```

The following examples shows the configuration and **show** command output of BGP on the PCE node and the ABR node-to-learn topology using the BGP-LS NLRI family.

```
*A:PCE Server 226>config>router>bgp# info
----------------------------------------------
            family bgp-ls
            min-route-advertisement 1
            link-state-export-enable
            group "IBGP_L2"
                family bgp-ls
                peer-as 65000
                neighbor 192.168.48.198
                exit
                neighbor 192.168.48.199
                exit
                neighbor 192.168.48.221
                exit
            exit
            no shutdown
----------------------------------------------

*A:Chicago 221>config>router>bgp# info
----------------------------------------------
            min-route-advertisement 1
            advertise-inactive
            link-state-import-enable
```

```
            group "IBGP_L2"
                family bgp-ls
                peer-as 65000
                neighbor 192.168.48.226
                exit
            exit
            no shutdown
-------------------------------------------

*A:PCE Server 226# show router bgp summary
===============================================================================
 BGP Router ID:192.168.48.226    AS:65000       Local AS:65000
===============================================================================
BGP Admin State         : Up          BGP Oper State              : Up
Total Peer Groups       : 1           Total Peers                 : 3
Total BGP Paths         : 182         Total Path Memory           : 44896
Total IPv4 Remote Rts   : 0           Total IPv4 Rem. Active Rts  : 0
Total McIPv4 Remote Rts : 0           Total McIPv4 Rem. Active Rts: 0
Total McIPv6 Remote Rts : 0           Total McIPv6 Rem. Active Rts: 0
Total IPv6 Remote Rts   : 0           Total IPv6 Rem. Active Rts  : 0
Total IPv4 Backup Rts   : 0           Total IPv6 Backup Rts       : 0
Total Supressed Rts     : 0           Total Hist. Rts             : 0
Total Decay Rts         : 0
Total VPN Peer Groups   : 0           Total VPN Peers             : 0
Total VPN Local Rts     : 0
Total VPN-IPv4 Rem. Rts : 0           Total VPN-IPv4 Rem. Act. Rts: 0
Total VPN-IPv6 Rem. Rts : 0           Total VPN-IPv6 Rem. Act. Rts: 0
Total VPN-IPv4 Bkup Rts : 0           Total VPN-IPv6 Bkup Rts     : 0
Total VPN Supp. Rts     : 0           Total VPN Hist. Rts         : 0
Total VPN Decay Rts     : 0
Total L2-VPN Rem. Rts   : 0           Total L2VPN Rem. Act. Rts   : 0
Total MVPN-IPv4 Rem Rts : 0           Total MVPN-IPv4 Rem Act Rts : 0
Total MDT-SAFI Rem Rts  : 0           Total MDT-SAFI Rem Act Rts  : 0
Total MSPW Rem Rts      : 0           Total MSPW Rem Act Rts      : 0
Total RouteTgt Rem Rts  : 0           Total RouteTgt Rem Act Rts  : 0
Total McVpnIPv4 Rem Rts : 0           Total McVpnIPv4 Rem Act Rts : 0
Total McVpnIPv6 Rem Rts : 0           Total McVpnIPv6 Rem Act Rts : 0
Total MVPN-IPv6 Rem Rts : 0           Total MVPN-IPv6 Rem Act Rts : 0
Total EVPN Rem Rts      : 0           Total EVPN Rem Act Rts      : 0
Total FlowIpv4 Rem Rts  : 0           Total FlowIpv4 Rem Act Rts  : 0
Total FlowIpv6 Rem Rts  : 0           Total FlowIpv6 Rem Act Rts  : 0
Total LblIpv4 Rem Rts   : 0           Total LblIpv4 Rem. Act Rts  : 0
Total LblIpv6 Rem Rts   : 0           Total LblIpv6 Rem. Act Rts  : 0
Total LblIpv4 Bkp Rts   : 0           Total LblIpv6 Bkp Rts       : 0
Total Link State Rem Rts: 271         Total Link State Rem. Act Rts: 0
===============================================================================
BGP Summary
===============================================================================
Legend : D - Dynamic Neighbor
===============================================================================
Neighbor
Description
                  AS PktRcvd InQ  Up/Down   State|Rcv/Act/Sent (Addr Family)
                     PktSent OutQ
-------------------------------------------------------------------------------
192.168.48.198
               65000        0    0 02h42m56s Active
                            0    0
192.168.48.199
               65000      503    0 02h42m56s 76/0/0 (LinkState)
                          328    0
192.168.48.221
               65000      519    0 02h42m56s 195/0/0 (LinkState)
                          328    0
-------------------------------------------------------------------------------
```

```
*A:PCE Server 226# show router bgp routes bgp-ls hunt link
===============================================================================
 BGP Router ID:192.168.48.226    AS:65000       Local AS:65000
===============================================================================
 Legend -
 Status codes  : u - used, s - suppressed, h - history, d - decayed, * - valid
                 l - leaked, x - stale, > - best, b - backup, p - purge
 Origin codes  : i - IGP, e - EGP, ? - incomplete
===============================================================================
BGP-LS Link NLRIs
===============================================================================
-------------------------------------------------------------------------------
RIB In Entries
-------------------------------------------------------------------------------
Network:
 Type          : LINK-NLRI
 Protocol      : ISIS Level-2           Identifier     : 0xa
 Local Node descriptor:
  Autonomous System  : 0.0.253.232
  Link State Id      : 10
  IGP Router Id      : 0x38120048184
 Remote Node descriptor:
  Autonomous System  : 0.0.253.232
  Link State Id      : 10
  IGP Router Id      : 0x38120048223
 Link descriptor:
  IPV4 Interface Addr: 10.0.14.184
  IPV4 Neighbor Addr : 10.0.14.223
Nexthop        : 192.168.48.199
From           : 192.168.48.199
Res. Nexthop   : 0.0.0.0
Local Pref.    : 100                    Interface Name : NotAvailable
Aggregator AS  : None                   Aggregator     : None
Atomic Aggr.   : Not Atomic             MED            : None
AIGP Metric    : None
Connector      : None
Community      : No Community Members
Cluster        : No Cluster Members
Originator Id  : None                   Peer Router Id : 192.168.48.199
Flags          : Valid  Best  IGP
Route Source   : Internal
AS-Path        : No As-Path
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                      Dest Class     : 0
Add Paths Send : Default
Last Modified  : 02h27m50s
-------------------------------------------------------------------------------
Link State Attribute TLVs :
 Administrative group (color) : 0x0
 Maximum link bandwidth : 100000 Kbps
 Max. reservable link bandwidth : 100000 Kbps
 Unreserved bandwidth0 : 100000 Kbps
 Unreserved bandwidth1 : 100000 Kbps
 Unreserved bandwidth2 : 100000 Kbps
 Unreserved bandwidth3 : 100000 Kbps
 Unreserved bandwidth4 : 100000 Kbps
 Unreserved bandwidth5 : 100000 Kbps
 Unreserved bandwidth6 : 100000 Kbps
 Unreserved bandwidth7 : 100000 Kbps
 TE Default Metric : 100
 IGP Metric : 100
 Adjacency Segment Identifier (Adj-SID) :       flags 0x30 weight 0 sid 262136
-------------------------------------------------------------------------------
Network:
```

```
 Type          : LINK-NLRI
 Protocol      : ISIS Level-2          Identifier     : 0xa
 Local Node descriptor:
  Autonomous System  : 0.0.253.232
  Link State Id      : 10
  IGP Router Id      : 0x38120048184
 Remote Node descriptor:
  Autonomous System  : 0.0.253.232
  Link State Id      : 10
  IGP Router Id      : 0x38120048223
 Link descriptor:
  IPV4 Interface Addr: 10.0.14.184
  IPV4 Neighbor Addr : 10.0.14.223
Nexthop        : 192.168.48.221
From           : 192.168.48.221
Res. Nexthop   : 0.0.0.0
Local Pref.    : 100                   Interface Name : NotAvailable
Aggregator AS  : None                  Aggregator     : None
Atomic Aggr.   : Not Atomic            MED            : None
AIGP Metric    : None
Connector      : None
Community      : No Community Members
Cluster        : No Cluster Members
Originator Id  : None                  Peer Router Id : 192.168.48.221
Flags          : Valid  IGP
TieBreakReason : OriginatorID
Route Source   : Internal
AS-Path        : No As-Path
Route Tag      : 0
Neighbor-AS    : N/A
Orig Validation: N/A
Source Class   : 0                     Dest Class     : 0
Add Paths Send : Default
Last Modified  : 02h27m54s
-------------------------------------------------------------------------------
Link State Attribute TLVs :
 Administrative group (color) : 0x0
 Maximum link bandwidth : 100000 Kbps
 Max. reservable link bandwidth : 100000 Kbps
 Unreserved bandwidth0 : 100000 Kbps
 Unreserved bandwidth1 : 100000 Kbps
 Unreserved bandwidth2 : 100000 Kbps
 Unreserved bandwidth3 : 100000 Kbps
 Unreserved bandwidth4 : 100000 Kbps
 Unreserved bandwidth5 : 100000 Kbps
 Unreserved bandwidth6 : 100000 Kbps
 Unreserved bandwidth7 : 100000 Kbps
 TE Default Metric : 100
 IGP Metric : 100
 Adjacency Segment Identifier (Adj-SID) :       flags 0x30 weight 0 sid 262136
-------------------------------------------------------------------------------
```

Figure 55: Primary and Secondary RSVP-TE LSP Paths in the NSP GUI shows primary and secondary RSVP-TE LSP paths in the NSP GUI.

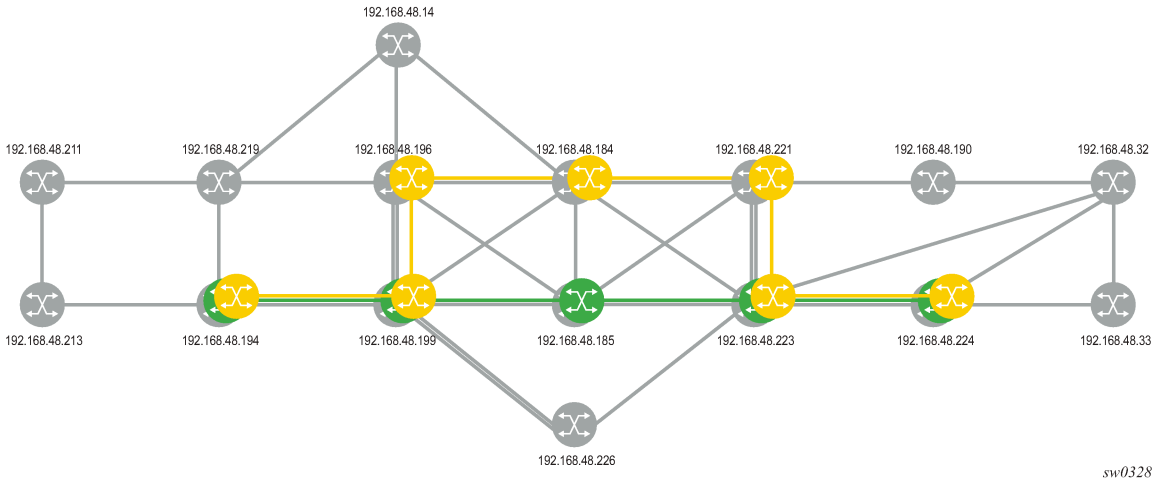

Figure 55: Primary and Secondary RSVP-TE LSP Paths in the NSP GUI

The following example shows the configuration and **show** command output of the MPLS on the PCC node.

```
*A:Reno 194>config>router>mpls>lsp# info
----------------------------------------------
               to 192.168.48.224
               egress-statistics
                   shutdown
               exit
               fast-reroute facility
                   no node-protect
               exit
               path-computation-method pce
               pce-report enable
               pce-control
               revert-timer 1
               primary "primary_empty"
                   exclude "top"
                   bandwidth 10
               exit
               secondary "secondary_empty"
                   standby
                   exclude "bottom"
                   bandwidth 5
               exit
               no shutdown
----------------------------------------------

*A:Reno 194# show router mpls lsp "From Reno to Atlanta RSVP-TE" path detail
===============================================================================
MPLS LSP From Reno to Atlanta RSVP-TE Path  (Detail)
===============================================================================
Legend :
    @ - Detour Available             # - Detour In Use
    b - Bandwidth Protected          n - Node Protected
    s - Soft Preemption
    S - Strict                       L - Loose
    A - ABR
===============================================================================
-------------------------------------------------------------------------------
LSP From Reno to Atlanta RSVP-TE Path primary_empty
-------------------------------------------------------------------------------
LSP Name         : From Reno to Atlanta RSVP-TE
Path LSP ID      : 14382
```

```
From             : 192.168.48.194       To                   : 192.168.48.224
Admin State      : Up                   Oper State           : Up
Path Name        : primary_empty        Path Type            : Primary
Path Admin       : Up                   Path Oper            : Up
Out Interface    : 1/1/1                Out Label            : 262094
Path Up Time     : 0d 00:00:22          Path Down Time       : 0d 00:00:00
Retry Limit      : 0                    Retry Timer          : 30 sec
Retry Attempt    : 0                    Next Retry In        : 0 sec
BFD Template     : None                 BFD Ping Interval    : 60
BFD Enable       : False
Adspec           : Disabled             Oper Adspec          : Disabled
CSPF             : Enabled              Oper CSPF            : Enabled
Least Fill       : Disabled             Oper LeastFill       : Disabled
FRR              : Enabled              Oper FRR             : Enabled
FRR NodeProtect  : Disabled             Oper FRR NP          : Disabled
FR Hop Limit     : 16                   Oper FRHopLimit      : 16
FR Prop Admin Gr*: Disabled             Oper FRPropAdmGrp    : Disabled
Propogate Adm Grp: Disabled             Oper Prop Adm Grp    : Disabled
Inter-area       : False
PCE Updt ID      : 0
PCE Report       : Enabled              Oper PCE Report      : Enabled
PCE Control      : Enabled              Oper PCE Control     : Enabled
PCE Compute      : Enabled
Neg MTU          : 1496                 Oper MTU             : 1496
Bandwidth        : 10 Mbps              Oper Bandwidth       : 10 Mbps
Hop Limit        : 255                  Oper HopLimit        : 255
Record Route     : Record               Oper Record Route    : Record
Record Label     : Record               Oper Record Label    : Record
Setup Priority   : 7                    Oper Setup Priority  : 7
Hold Priority    : 0                    Oper Hold Priority   : 0
Class Type       : 0                    Oper CT              : 0
Backup CT        : None
MainCT Retry     : n/a
    Rem          :
MainCT Retry     : 0
    Limit        :
Include Groups   :                      Oper Include Groups  :
None                                           None
Exclude Groups   :                      Oper Exclude Groups  :
top                                            top
Adaptive         : Enabled              Oper Metric          : 40
Preference       : n/a
Path Trans       : 7                    CSPF Queries         : 7172
Failure Code     : noError
Failure Node     : n/a
Explicit Hops    :
    No Hops Specified
Actual Hops      :
    10.202.5.194 (192.168.48.194) @              Record Label        : N/A
 -> 10.202.5.199 (192.168.48.199) @              Record Label        : 262094
 -> 192.168.48.185 (192.168.48.185)              Record Label        : 262111
 -> 10.0.5.185                                   Record Label        : 262111
 -> 192.168.48.223 (192.168.48.223)              Record Label        : 262121
 -> 10.0.7.223                                   Record Label        : 262121
 -> 192.168.48.224 (192.168.48.224)              Record Label        : 262116
 -> 10.101.4.224                                 Record Label        : 262116
Computed Hops    :
    10.202.5.199(S)
 -> 10.0.5.185(S)
 -> 10.0.7.223(S)
 -> 10.101.4.224(S)
Resignal Eligible: False
Last Resignal    : n/a                  CSPF Metric          : 40
-------------------------------------------------------------------------------
LSP From Reno to Atlanta RSVP-TE Path secondary_empty
-------------------------------------------------------------------------------
```

```
LSP Name         : From Reno to Atlanta RSVP-TE
Path LSP ID      : 14384
From             : 192.168.48.194       To                  : 192.168.48.224
Admin State      : Up                   Oper State          : Up
Path Name        : secondary_empty      Path Type           : Standby
Path Admin       : Up                   Path Oper           : Up
Out Interface    : 1/1/1                Out Label           : 262091
Path Up Time     : 0d 00:00:25          Path Down Time      : 0d 00:00:00
Retry Limit      : 0                    Retry Timer         : 30 sec
Retry Attempt    : 0                    Next Retry In       : 0 sec
BFD Template     : None                 BFD Ping Interval   : 60
BFD Enable       : False
Adspec           : Disabled             Oper Adspec         : Disabled
CSPF             : Enabled              Oper CSPF           : Enabled
Least Fill       : Disabled             Oper LeastFill      : Disabled
Propogate Adm Grp: Disabled             Oper Prop Adm Grp   : Disabled
Inter-area       : False
PCE Updt ID      : 0
PCE Report       : Enabled              Oper PCE Report     : Enabled
PCE Control      : Enabled              Oper PCE Control    : Enabled
PCE Compute      : Enabled
Neg MTU          : 1496                 Oper MTU            : 1496
Bandwidth        : 5 Mbps               Oper Bandwidth      : 5 Mbps
Hop Limit        : 255                  Oper HopLimit       : 255
Record Route     : Record               Oper Record Route   : Record
Record Label     : Record               Oper Record Label   : Record
Setup Priority   : 7                    Oper Setup Priority : 7
Hold Priority    : 0                    Oper Hold Priority  : 0
Class Type       : 0                    Oper CT             : 0
Include Groups   :                      Oper Include Groups :
None                                      None
Exclude Groups   :                      Oper Exclude Groups :
bottom                                    bottom
Adaptive         : Enabled              Oper Metric         : 60
Preference       : 255
Path Trans       : 28                   CSPF Queries        : 10
Failure Code     : noError
Failure Node     : n/a
Explicit Hops    :
    No Hops Specified
Actual Hops      :
    10.202.5.194 (192.168.48.194)               Record Label        : N/A
 -> 10.202.5.199 (192.168.48.199)               Record Label        : 262091
 -> 10.0.9.198 (192.168.48.198)                 Record Label        : 262096
 -> 192.168.48.184 (192.168.48.184)             Record Label        : 262102
 -> 10.0.2.184                                  Record Label        : 262102
 -> 192.168.48.221 (192.168.48.221)             Record Label        : 262119
 -> 10.0.4.221                                  Record Label        : 262119
 -> 192.168.48.223 (192.168.48.223)             Record Label        : 262088
 -> 10.0.10.223                                 Record Label        : 262088
 -> 192.168.48.224 (192.168.48.224)             Record Label        : 262115
 -> 10.101.4.224                                Record Label        : 262115
Computed Hops    :
    10.202.5.199(S)
 -> 10.0.9.198(S)
 -> 10.0.2.184(S)
 -> 10.0.4.221(S)
 -> 10.0.10.223(S)
 -> 10.101.4.224(S)
Srlg             : Disabled
Srlg Disjoint    : False
Resignal Eligible: False
Last Resignal    : n/a                  CSPF Metric         : 60
===============================================================================
```

7 Standards and Protocol Support

Note:

The information provided in this chapter is subject to change without notice and may not apply to all platforms.

Nokia assumes no responsibility for inaccuracies.

7.1 Access Node Control Protocol (ANCP)

draft-ietf-ancp-protocol-02, *Protocol for Access Node Control Mechanism in Broadband Networks*

RFC 5851, *Framework and Requirements for an Access Node Control Mechanism in Broadband Multi-Service Networks*

7.2 Application Assurance (AA)

3GPP Release 12, *ADC rules over Gx interfaces*

RFC 3507, *Internet Content Adaptation Protocol (ICAP)*

7.3 Asynchronous Transfer Mode (ATM)

AF-ILMI-0065.000 Version 4.0, *Integrated Local Management Interface (ILMI)*

AF-PHY-0086.001 Version 1.1, *Inverse Multiplexing for ATM (IMA) Specification*

AF-TM-0121.000 Version 4.1, *Traffic Management Specification*

GR-1113-CORE Issue 1, *Asynchronous Transfer Mode (ATM) and ATM Adaptation Layer (AAL) Protocols Generic Requirements*

GR-1248-CORE Issue 3, *Generic Requirements for Operations of ATM Network Elements (NEs)*

RFC 1626, *Default IP MTU for use over ATM AAL5*

RFC 2684, *Multiprotocol Encapsulation over ATM Adaptation Layer 5*

7.4 Bidirectional Forwarding Detection (BFD)

draft-ietf-idr-bgp-ls-sbfd-extensions-01, *BGP Link-State Extensions for Seamless BFD*

RFC 5880, *Bidirectional Forwarding Detection (BFD)*

RFC 5881, *Bidirectional Forwarding Detection (BFD) IPv4 and IPv6 (Single Hop)*

RFC 5882, *Generic Application of Bidirectional Forwarding Detection (BFD)*

RFC 5883, *Bidirectional Forwarding Detection (BFD) for Multihop Paths*

RFC 7130, *Bidirectional Forwarding Detection (BFD) on Link Aggregation Group (LAG) Interfaces*

RFC 7880, *Seamless Bidirectional Forwarding Detection (S-BFD)*

RFC 7881, *Seamless Bidirectional Forwarding Detection (S-BFD) for IPv4, IPv6, and MPLS*

RFC 7883, *Advertising Seamless Bidirectional Forwarding Detection (S-BFD) Discriminators in IS-IS*

RFC 7884, *OSPF Extensions to Advertise Seamless Bidirectional Forwarding Detection (S-BFD) Target Discriminators*

7.5 Border Gateway Protocol (BGP)

draft-hares-idr-update-attrib-low-bits-fix-01, *Update Attribute Flag Low Bits Clarification*

draft-ietf-idr-add-paths-guidelines-08, *Best Practices for Advertisement of Multiple Paths in IBGP*

draft-ietf-idr-best-external-03, *Advertisement of the best external route in BGP*

draft-ietf-idr-bgp-flowspec-oid-03, *Revised Validation Procedure for BGP Flow Specifications*

draft-ietf-idr-bgp-gr-notification-01, *Notification Message support for BGP Graceful Restart*

draft-ietf-idr-bgp-ls-app-specific-attr-01, *Application Specific Attributes Advertisement with BGP Link-State*

draft-ietf-idr-bgp-optimal-route-reflection-10, *BGP Optimal Route Reflection (BGP-ORR)*

draft-ietf-idr-error-handling-03, *Revised Error Handling for BGP UPDATE Messages*

draft-ietf-idr-flowspec-interfaceset-03, *Applying BGP flowspec rules on a specific interface set*

draft-ietf-idr-flowspec-path-redirect-05, *Flowspec Indirection-id Redirect* - localised ID

draft-ietf-idr-flowspec-redirect-ip-02, *BGP Flow-Spec Redirect to IP Action*

draft-ietf-idr-link-bandwidth-03, *BGP Link Bandwidth Extended Community*

draft-ietf-idr-long-lived-gr-00, *Support for Long-lived BGP Graceful Restart*

draft-ietf-sidr-origin-validation-signaling-04, *BGP Prefix Origin Validation State Extended Community*

RFC 1772, *Application of the Border Gateway Protocol in the Internet*

RFC 1997, *BGP Communities Attribute*

RFC 2385, *Protection of BGP Sessions via the TCP MD5 Signature Option*

RFC 2439, *BGP Route Flap Damping*

RFC 2545, *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*

RFC 2858, *Multiprotocol Extensions for BGP-4*

RFC 2918, *Route Refresh Capability for BGP-4*

RFC 3107, *Carrying Label Information in BGP-4*

RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*

RFC 4360, *BGP Extended Communities Attribute*

RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4456, *BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)*

RFC 4486, *Subcodes for BGP Cease Notification Message*

RFC 4659, *BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*

RFC 4684, *Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)*

RFC 4724, *Graceful Restart Mechanism for BGP* - helper mode

RFC 4760, *Multiprotocol Extensions for BGP-4*

RFC 4798, *Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)*

RFC 5004, *Avoid BGP Best Path Transitions from One External to Another*

RFC 5065, *Autonomous System Confederations for BGP*

RFC 5291, *Outbound Route Filtering Capability for BGP-4*

RFC 5396, *Textual Representation of Autonomous System (AS) Numbers* - asplain

RFC 5492, *Capabilities Advertisement with BGP-4*

RFC 5549, *Advertising IPv4 Network Layer Reachability Information with an IPv6 Next Hop*

RFC 5575, *Dissemination of Flow Specification Rules*

RFC 5668, *4-Octet AS Specific BGP Extended Community*

RFC 6286, *Autonomous-System-Wide Unique BGP Identifier for BGP-4*

RFC 6793, *BGP Support for Four-Octet Autonomous System (AS) Number Space*

RFC 6810, *The Resource Public Key Infrastructure (RPKI) to Router Protocol*

RFC 6811, *Prefix Origin Validation*

RFC 6996, *Autonomous System (AS) Reservation for Private Use*

RFC 7311, *The Accumulated IGP Metric Attribute for BGP*

RFC 7607, *Codification of AS 0 Processing*

RFC 7674, *Clarification of the Flowspec Redirect Extended Community*

RFC 7752, *North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP*

RFC 7854, *BGP Monitoring Protocol (BMP)*

RFC 7911, *Advertisement of Multiple Paths in BGP*

RFC 7999, *BLACKHOLE Community*

RFC 8092, *BGP Large Communities Attribute*

RFC 8212, *Default External BGP (EBGP) Route Propagation Behavior without Policies*

RFC 8571, *BGP - Link State (BGP-LS) Advertisement of IGP Traffic Engineering Performance Metric Extensions*

7.6 Broadband Network Gateway (BNG) - Control and User Plane Separation (CUPS)

3GPP 23.007, *Restoration procedures*

3GPP 29.244, *Interface between the Control Plane and the User Plane nodes*

3GPP 29.281, *General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U)*

BBF TR-459, *Control and User Plane Separation for a Disaggregated BNG*

RFC 8300, *Network Service Header (NSH)*

7.7 Circuit Emulation

RFC 4553, *Structure-Agnostic Time Division Multiplexing (TDM) over Packet (SAToP)*

RFC 5086, *Structure-Aware Time Division Multiplexed (TDM) Circuit Emulation Service over Packet Switched Network (CESoPSN)*

RFC 5287, *Control Protocol Extensions for the Setup of Time-Division Multiplexing (TDM) Pseudowires in MPLS Networks*

7.8 Ethernet

IEEE 802.1AB, *Station and Media Access Control Connectivity Discovery*

IEEE 802.1ad, *Provider Bridges*

IEEE 802.1ag, *Connectivity Fault Management*

IEEE 802.1ah, *Provider Backbone Bridges*

IEEE 802.1ak, *Multiple Registration Protocol*

IEEE 802.1aq, *Shortest Path Bridging*

IEEE 802.1ax, *Link Aggregation*

IEEE 802.1D, *MAC Bridges*

IEEE 802.1p, *Traffic Class Expediting*

IEEE 802.1Q, *Virtual LANs*

IEEE 802.1s, *Multiple Spanning Trees*

IEEE 802.1w, *Rapid Reconfiguration of Spanning Tree*

IEEE 802.1X, *Port Based Network Access Control*

IEEE 802.3ac, *VLAN Tag*

IEEE 802.3ad, *Link Aggregation*

IEEE 802.3ah, *Ethernet in the First Mile*

IEEE 802.3x, *Ethernet Flow Control*

ITU-T G.8031/Y.1342, *Ethernet Linear Protection Switching*

ITU-T G.8032/Y.1344, *Ethernet Ring Protection Switching*

ITU-T Y.1731, *OAM functions and mechanisms for Ethernet based networks*

7.9 Ethernet VPN (EVPN)

draft-ietf-bess-evpn-igmp-mld-proxy-05, *IGMP and MLD Proxy for EVPN*

draft-ietf-bess-evpn-irb-mcast-04, *EVPN Optimized Inter-Subnet Multicast (OISM) Forwarding* - ingress replication

draft-ietf-bess-evpn-pref-df-06, *Preference-based EVPN DF Election*

draft-ietf-bess-evpn-prefix-advertisement-11, *IP Prefix Advertisement in EVPN*

draft-ietf-bess-evpn-proxy-arp-nd-08, *Operational Aspects of Proxy-ARP/ND in EVPN Networks*

draft-ietf-bess-evpn-virtual-eth-segment-06, *EVPN Virtual Ethernet Segment*

draft-ietf-bess-pbb-evpn-isid-cmacflush-00, *PBB-EVPN ISID-based CMAC-Flush*

RFC 7432, *BGP MPLS-Based Ethernet VPN*

RFC 7623, *Provider Backbone Bridging Combined with Ethernet VPN (PBB-EVPN)*

RFC 8214, *Virtual Private Wire Service Support in Ethernet VPN*

RFC 8317, *Ethernet-Tree (E-Tree) Support in Ethernet VPN (EVPN) an Provider Backbone Bridging EVPN (PBB-EVPN)*

RFC 8365, *A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)*

RFC 8560, *Seamless Integration of Ethernet VPN (EVPN) with Virtual Private LAN Service (VPLS) and Their Provider Backbone Bridge (PBB) Equivalents*

RFC 8584, *DF Election and AC-influenced DF Election*

7.10 Frame Relay

ANSI T1.617 Annex D, *DSS1 - Signalling Specification For Frame Relay Bearer Service*

FRF.1.2, *PVC User-to-Network Interface (UNI) Implementation Agreement*

FRF.12, *Frame Relay Fragmentation Implementation Agreement*

FRF.16.1, *Multilink Frame Relay UNI/NNI Implementation Agreement*

FRF.5, *Frame Relay/ATM PVC Network Interworking Implementation*

FRF2.2, *PVC Network-to-Network Interface (NNI) Implementation Agreement*

ITU-T Q.933 Annex A, *Additional procedures for Permanent Virtual Connection (PVC) status management*

7.11 Generalized Multiprotocol Label Switching (GMPLS)

draft-ietf-ccamp-rsvp-te-srlg-collect-04, *RSVP-TE Extensions for Collecting SRLG Information*

RFC 3471, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description*

RFC 3473, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions*

RFC 4204, *Link Management Protocol (LMP)*

RFC 4208, *Generalized Multiprotocol Label Switching (GMPLS) User-Network Interface (UNI): Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Support for the Overlay Model*

RFC 4872, *RSVP-TE Extensions in Support of End-to-End Generalized Multi-Protocol Label Switching (GMPLS) Recovery*

RFC 5063, *Extensions to GMPLS Resource Reservation Protocol (RSVP) Graceful Restart* - helper mode

7.12 gRPC Remote Procedure Calls (gRPC)

cert.proto version 0.1.0, *gRPC Network Operations Interface (gNOI) Certificate Management Service*

file.proto version 0.1.0, *gRPC Network Operations Interface (gNOI) File Service*

gnmi.proto version 0.7.0, *gRPC Network Management Interface (gNMI) Service Specification*

PROTOCOL-HTTP2, *gRPC over HTTP2*

system.proto Version 1.0.0, *gRPC Network Operations Interface (gNOI) System Service*

7.13 Intermediate System to Intermediate System (IS-IS)

draft-ietf-isis-mi-02, *IS-IS Multi-Instance*

draft-kaplan-isis-ext-eth-02, *Extended Ethernet Frame Size Support*

ISO/IEC 10589:2002 Second Edition, *Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)*

RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*

RFC 2973, *IS-IS Mesh Groups*

RFC 3359, *Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System*

RFC 3719, *Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)*

RFC 3787, *Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)*

RFC 4971, *Intermediate System to Intermediate System (IS-IS) Extensions for Advertising Router Information*

RFC 5120, *M-ISIS: Multi Topology (MT) Routing in IS-IS*

RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*

RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*

RFC 5302, *Domain-wide Prefix Distribution with Two-Level IS-IS*

RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*

RFC 5304, *IS-IS Cryptographic Authentication*

RFC 5305, *IS-IS Extensions for Traffic Engineering TE*

RFC 5306, *Restart Signaling for IS-IS* - helper mode

RFC 5307, *IS-IS Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*

RFC 5308, *Routing IPv6 with IS-IS*

RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*

RFC 5310, *IS-IS Generic Cryptographic Authentication*

RFC 6119, *IPv6 Traffic Engineering in IS-IS*

RFC 6213, *IS-IS BFD-Enabled TLV*

RFC 6232, *Purge Originator Identification TLV for IS-IS*

RFC 6233, *IS-IS Registry Extension for Purges*

RFC 6329, *IS-IS Extensions Supporting IEEE 802.1aq Shortest Path Bridging*

RFC 7775, *IS-IS Route Preference for Extended IP and IPv6 Reachability*

RFC 7794, *IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability*

RFC 7987, *IS-IS Minimum Remaining Lifetime*

RFC 8202, *IS-IS Multi-Instance* - single topology

RFC 8570, *IS-IS Traffic Engineering (TE) Metric Extensions* - delay metric

RFC 8919, *IS-IS Application-Specific Link Attributes*

7.14 Internet Protocol (IP) — Fast Reroute

draft-ietf-rtgwg-lfa-manageability-08, *Operational management of Loop Free Alternates*

RFC 5286, *Basic Specification for IP Fast Reroute: Loop-Free Alternates*

RFC 7431, *Multicast-Only Fast Reroute*

RFC 7490, *Remote Loop-Free Alternate (LFA) Fast Reroute (FRR)*

7.15 Internet Protocol (IP) — General

draft-grant-tacacs-02, *The TACACS+ Protocol*

RFC 768, *User Datagram Protocol*

RFC 793, *Transmission Control Protocol*

RFC 854, *Telnet Protocol Specifications*

RFC 1350, *The TFTP Protocol (revision 2)*

RFC 2347, *TFTP Option Extension*

RFC 2348, *TFTP Blocksize Option*

RFC 2349, *TFTP Timeout Interval and Transfer Size Options*

RFC 2428, *FTP Extensions for IPv6 and NATs*

RFC 2784, *Generic Routing Encapsulation (GRE)*

RFC 2818, *HTTP Over TLS*

RFC 2890, *Key and Sequence Number Extensions to GRE*

RFC 3164, *The BSD syslog Protocol*

RFC 4250, *The Secure Shell (SSH) Protocol Assigned Numbers*

RFC 4251, *The Secure Shell (SSH) Protocol Architecture*

RFC 4252, *The Secure Shell (SSH) Authentication Protocol* - publickey, password

RFC 4253, *The Secure Shell (SSH) Transport Layer Protocol*

RFC 4254, *The Secure Shell (SSH) Connection Protocol*

RFC 4511, *Lightweight Directory Access Protocol (LDAP): The Protocol*

RFC 4513, *Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms* - TLS

RFC 4632, *Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan*

RFC 5082, *The Generalized TTL Security Mechanism (GTSM)*

RFC 5246, *The Transport Layer Security (TLS) Protocol Version 1.2* - TLS client, RSA public key

RFC 5656, *Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer* - ECDSA

RFC 5925, *The TCP Authentication Option*

RFC 5926, *Cryptographic Algorithms for the TCP Authentication Option (TCP-AO)*

RFC 6398, *IP Router Alert Considerations and Usage* - MLD

RFC 6528, *Defending against Sequence Number Attacks*

RFC 7011, *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*

RFC 7012, *Information Model for IP Flow Information Export*

RFC 7230, *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*

RFC 7231, *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content*

RFC 7232, *Hypertext Transfer Protocol (HTTP/1.1): Conditional Requests*

RFC 7301, *Transport Layer Security (TLS) Application Layer Protocol Negotiation Extension*

7.16 Internet Protocol (IP) — Multicast

cisco-ipmulticast/pim-autorp-spec01, *Auto-RP: Automatic discovery of Group-to-RP mappings for IP multicast* - version 1

draft-ietf-bier-pim-signaling-08, *PIM Signaling Through BIER Core*

draft-ietf-idmr-traceroute-ipm-07, *A "traceroute" facility for IP Multicast*

draft-ietf-l2vpn-vpls-pim-snooping-07, *Protocol Independent Multicast (PIM) over Virtual Private LAN Service (VPLS)*

RFC 1112, *Host Extensions for IP Multicasting*

RFC 2236, *Internet Group Management Protocol, Version 2*

RFC 2365, *Administratively Scoped IP Multicast*

RFC 2375, *IPv6 Multicast Address Assignments*

RFC 2710, *Multicast Listener Discovery (MLD) for IPv6*

RFC 3306, *Unicast-Prefix-based IPv6 Multicast Addresses*

RFC 3376, *Internet Group Management Protocol, Version 3*

RFC 3446, *Anycast Rendevous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)*

RFC 3590, *Source Address Selection for the Multicast Listener Discovery (MLD) Protocol*

RFC 3618, *Multicast Source Discovery Protocol (MSDP)*

RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*

RFC 3956, *Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address*

RFC 3973, *Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised)* - auto-RP groups

RFC 4541, *Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches*

RFC 4604, *Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast*

RFC 4607, *Source-Specific Multicast for IP*

RFC 4608, *Source-Specific Protocol Independent Multicast in 232/8*

RFC 4610, *Anycast-RP Using Protocol Independent Multicast (PIM)*

RFC 4611, *Multicast Source Discovery Protocol (MSDP) Deployment Scenarios*

RFC 5059, *Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)*

RFC 5186, *Internet Group Management Protocol Version 3 (IGMPv3) / Multicast Listener Discovery Version 2 (MLDv2) and Multicast Routing Protocol Interaction*

RFC 5384, *The Protocol Independent Multicast (PIM) Join Attribute Format*

RFC 5496, *The Reverse Path Forwarding (RPF) Vector TLV*

RFC 6037, *Cisco Systems' Solution for Multicast in MPLS/BGP IP VPNs*

RFC 6512, *Using Multipoint LDP When the Backbone Has No Route to the Root*

RFC 6513, *Multicast in MPLS/BGP IP VPNs*

RFC 6514, *BGP Encodings and Procedures for Multicast in MPLS/IP VPNs*

RFC 6515, *IPv4 and IPv6 Infrastructure Addresses in BGP Updates for Multicast VPNs*

RFC 6516, *IPv6 Multicast VPN (MVPN) Support Using PIM Control Plane and Selective Provider Multicast Service Interface (S-PMSI) Join Messages*

RFC 6625, *Wildcards in Multicast VPN Auto-Discover Routes*

RFC 6826, *Multipoint LDP In-Band Signaling for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Path*

RFC 7246, *Multipoint Label Distribution Protocol In-Band Signaling in a Virtual Routing and Forwarding (VRF) Table Context*

RFC 7385, *IANA Registry for P-Multicast Service Interface (PMSI) Tunnel Type Code Points*

RFC 7716, *Global Table Multicast with BGP Multicast VPN (BGP-MVPN) Procedures*

RFC 7761, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*

RFC 8279, *Multicast Using Bit Index Explicit Replication (BIER)*

RFC 8296, *Encapsulation for Bit Index Explicit Replication (BIER) in MPLS and Non-MPLS Networks* - MPLS encapsulation

RFC 8401, *Bit Index Explicit Replication (BIER) Support via IS-IS*

RFC 8444, *OSPFv2 Extensions for Bit Index Explicit Replication (BIER)*

RFC 8487, *Mtrace Version 2: Traceroute Facility for IP Multicast*

RFC 8534, *Explicit Tracking with Wildcard Routes in Multicast VPN* - (C-*,C-*) wildcard

RFC 8556, *Multicast VPN Using Bit Index Explicit Replication (BIER)*

7.17 Internet Protocol (IP) — Version 4

RFC 791, *Internet Protocol*

RFC 792, *Internet Control Message Protocol*

RFC 826, *An Ethernet Address Resolution Protocol*

RFC 951, *Bootstrap Protocol (BOOTP)* - relay

RFC 1034, *Domain Names - Concepts and Facilities*

RFC 1035, *Domain Names - Implementation and Specification*

RFC 1191, *Path MTU Discovery* - router specification

RFC 1519, *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*

RFC 1534, *Interoperation between DHCP and BOOTP*

RFC 1542, *Clarifications and Extensions for the Bootstrap Protocol*

RFC 1812, *Requirements for IPv4 Routers*

RFC 1918, *Address Allocation for Private Internets*

RFC 2003, *IP Encapsulation within IP*

RFC 2131, *Dynamic Host Configuration Protocol*

RFC 2132, *DHCP Options and BOOTP Vendor Extensions*

RFC 2401, *Security Architecture for Internet Protocol*

RFC 3021, *Using 31-Bit Prefixes on IPv4 Point-to-Point Links*

RFC 3046, *DHCP Relay Agent Information Option (Option 82)*

RFC 3768, *Virtual Router Redundancy Protocol (VRRP)*

RFC 4884, *Extended ICMP to Support Multi-Part Messages* - ICMPv4 and ICMPv6 Time Exceeded

7.18 Internet Protocol (IP) — Version 6

RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*

RFC 2529, *Transmission of IPv6 over IPv4 Domains without Explicit Tunnels*

RFC 3122, *Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification*

RFC 3315, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*

RFC 3587, *IPv6 Global Unicast Address Format*

RFC 3596, *DNS Extensions to Support IP version 6*

RFC 3633, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*

RFC 3646, *DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*

RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*

RFC 3971, *SEcure Neighbor Discovery (SEND)*

RFC 3972, *Cryptographically Generated Addresses (CGA)*

RFC 4007, *IPv6 Scoped Address Architecture*

RFC 4193, *Unique Local IPv6 Unicast Addresses*

RFC 4291, *Internet Protocol Version 6 (IPv6) Addressing Architecture*

RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*

RFC 4861, *Neighbor Discovery for IP version 6 (IPv6)*

RFC 4862, *IPv6 Stateless Address Autoconfiguration* - router functions

RFC 4890, *Recommendations for Filtering ICMPv6 Messages in Firewalls*

RFC 4941, *Privacy Extensions for Stateless Address Autoconfiguration in IPv6*

RFC 5007, *DHCPv6 Leasequery*

RFC 5095, *Deprecation of Type 0 Routing Headers in IPv6*

RFC 5722, *Handling of Overlapping IPv6 Fragments*

RFC 5798, *Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6* - IPv6

RFC 5952, *A Recommendation for IPv6 Address Text Representation*

RFC 6092, *Recommended Simple Security Capabilities in Customer Premises Equipment (CPE) for Providing Residential IPv6 Internet Service* - Internet Control and Management, Upper-Layer Transport Protocols, UDP Filters, IPsec and Internet Key Exchange (IKE), TCP Filters

RFC 6106, *IPv6 Router Advertisement Options for DNS Configuration*

RFC 6164, *Using 127-Bit IPv6 Prefixes on Inter-Router Links*

RFC 6437, *IPv6 Flow Label Specification*

RFC 8021, *Generation of IPv6 Atomic Fragments Considered Harmful*

RFC 8200, *Internet Protocol, Version 6 (IPv6) Specification*

RFC 8201, *Path MTU Discovery for IP version 6*

7.19 Internet Protocol Security (IPsec)

draft-ietf-ipsec-isakmp-mode-cfg-05, *The ISAKMP Configuration Method*

draft-ietf-ipsec-isakmp-xauth-06, *Extended Authentication within ISAKMP/Oakley (XAUTH)*

RFC 2401, *Security Architecture for the Internet Protocol*

RFC 2403, *The Use of HMAC-MD5-96 within ESP and AH*

RFC 2404, *The Use of HMAC-SHA-1-96 within ESP and AH*

RFC 2405, *The ESP DES-CBC Cipher Algorithm With Explicit IV*

RFC 2406, *IP Encapsulating Security Payload (ESP)*

RFC 2407, *IPsec Domain of Interpretation for ISAKMP (IPsec DoI)*

RFC 2408, *Internet Security Association and Key Management Protocol (ISAKMP)*

RFC 2409, *The Internet Key Exchange (IKE)*

RFC 2410, *The NULL Encryption Algorithm and Its Use With IPsec*

RFC 3526, *More Modular Exponential (MODP) Diffie-Hellman group for Internet Key Exchange (IKE)*

RFC 3566, *The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec*

RFC 3602, *The AES-CBC Cipher Algorithm and Its Use with IPsec*

RFC 3706, *A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers*

RFC 3947, *Negotiation of NAT-Traversal in the IKE*

RFC 3948, *UDP Encapsulation of IPsec ESP Packets*

RFC 4106, *The Use of Galois/Counter Mode (GCM) in IPsec ESP*

RFC 4210, *Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)*

RFC 4211, *Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)*

RFC 4301, *Security Architecture for the Internet Protocol*

RFC 4303, *IP Encapsulating Security Payload*

RFC 4307, *Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2)*

RFC 4308, *Cryptographic Suites for IPsec*

RFC 4434, *The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol (IKE)*

RFC 4543, *The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH*

RFC 4868, *Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPSec*

RFC 4945, *The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2 and PKIX*

RFC 5019, *The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments*

RFC 5280, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

RFC 5282, *Using Authenticated Encryption Algorithms with the Encrypted Payload of the IKEv2 Protocol*

RFC 5903, *ECP Groups for IKE and IKEv2*

RFC 5998, *An Extension for EAP-Only Authentication in IKEv2*

RFC 6379, *Suite B Cryptographic Suites for IPsec*

RFC 6380, *Suite B Profile for Internet Protocol Security (IPsec)*

RFC 6712, *Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)*

RFC 6960, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*

RFC 7296, *Internet Key Exchange Protocol Version 2 (IKEv2)*

RFC 7321, *Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)*

RFC 7383, *Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation*

RFC 7427, *Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)*

RFC 7468, *Textual Encodings of PKIX, PKCS, and CMS Structures*

7.20 Label Distribution Protocol (LDP)

draft-pdutta-mpls-ldp-adj-capability-00, *LDP Adjacency Capabilities*

draft-pdutta-mpls-ldp-v2-00, *LDP Version 2*

draft-pdutta-mpls-mldp-up-redundancy-00, *Upstream LSR Redundancy for Multi-point LDP Tunnels*

draft-pdutta-mpls-multi-ldp-instance-00, *Multiple LDP Instances*

draft-pdutta-mpls-tldp-hello-reduce-04, *Targeted LDP Hello Reduction*

RFC 3037, *LDP Applicability*

RFC 3478, *Graceful Restart Mechanism for Label Distribution Protocol* - helper mode

RFC 5036, *LDP Specification*

RFC 5283, *LDP Extension for Inter-Area Label Switched Paths (LSPs)*

RFC 5443, *LDP IGP Synchronization*

RFC 5561, *LDP Capabilities*

RFC 5919, *Signaling LDP Label Advertisement Completion*

RFC 6388, *Label Distribution Protocol Extensions for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths*

RFC 6512, *Using Multipoint LDP When the Backbone Has No Route to the Root*

RFC 6826, *Multipoint LDP in-band signaling for Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths*

RFC 7032, *LDP Downstream-on-Demand in Seamless MPLS*

RFC 7473, *Controlling State Advertisements of Non-negotiated LDP Applications*

RFC 7552, *Updates to LDP for IPv6*

7.21 Layer Two Tunneling Protocol (L2TP) Network Server (LNS)

draft-mammoliti-l2tp-accessline-avp-04, *Layer 2 Tunneling Protocol (L2TP) Access Line Information Attribute Value Pair (AVP) Extensions*

RFC 2661, *Layer Two Tunneling Protocol "L2TP"*

RFC 2809, *Implementation of L2TP Compulsory Tunneling via RADIUS*

RFC 3438, *Layer Two Tunneling Protocol (L2TP) Internet Assigned Numbers: Internet Assigned Numbers Authority (IANA) Considerations Update*

RFC 3931, *Layer Two Tunneling Protocol - Version 3 (L2TPv3)*

RFC 4719, *Transport of Ethernet Frames over Layer 2 Tunneling Protocol Version 3 (L2TPv3)*

RFC 4951, *Fail Over Extensions for Layer 2 Tunneling Protocol (L2TP) "failover"*

7.22 Multiprotocol Label Switching (MPLS)

draft-ietf-mpls-lsp-ping-ospfv3-codepoint-02, *OSPFv3 CodePoint for MPLS LSP Ping*

RFC 3031, *Multiprotocol Label Switching Architecture*

RFC 3032, *MPLS Label Stack Encoding*

RFC 3270, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services* - E-LSP

RFC 3443, *Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks*

RFC 4023, *Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)*

RFC 4182, *Removing a Restriction on the use of MPLS Explicit NULL*

RFC 5332, *MPLS Multicast Encapsulations*

RFC 5884, *Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)*

RFC 6374, *Packet Loss and Delay Measurement for MPLS Networks* - Delay Measurement, Channel Type 0x000C

RFC 6424, *Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels*

RFC 6425, *Detecting Data Plane Failures in Point-to-Multipoint Multiprotocol Label Switching (MPLS) - Extensions to LSP Ping*

RFC 6790, *The Use of Entropy Labels in MPLS Forwarding*

RFC 7510, *Encapsulating MPLS in UDP*

RFC 7746, *Label Switched Path (LSP) Self-Ping*

RFC 7876, *UDP Return Path for Packet Loss and Delay Measurement for MPLS Networks* - Delay Measurement

RFC 8029, *Detecting Multiprotocol Label Switched (MPLS) Data-Plane Failures*

7.23 Multiprotocol Label Switching — Transport Profile (MPLS-TP)

RFC 5586, *MPLS Generic Associated Channel*

RFC 5921, *A Framework for MPLS in Transport Networks*

RFC 5960, *MPLS Transport Profile Data Plane Architecture*

RFC 6370, *MPLS Transport Profile (MPLS-TP) Identifiers*

RFC 6378, *MPLS Transport Profile (MPLS-TP) Linear Protection*

RFC 6426, *MPLS On-Demand Connectivity and Route Tracing*

RFC 6427, *MPLS Fault Management Operations, Administration, and Maintenance (OAM)*

RFC 6428, *Proactive Connectivity Verification, Continuity Check and Remote Defect indication for MPLS Transport Profile*

RFC 6478, *Pseudowire Status for Static Pseudowires*

RFC 7213, *MPLS Transport Profile (MPLS-TP) Next-Hop Ethernet Addressing*

7.24 Network Address Translation (NAT)

draft-ietf-behave-address-format-10, *IPv6 Addressing of IPv4/IPv6 Translators*

draft-ietf-behave-v6v4-xlate-23, *IP/ICMP Translation Algorithm*

draft-miles-behave-l2nat-00, *Layer2-Aware NAT*

draft-nishitani-cgn-02, *Common Functions of Large Scale NAT (LSN)*

RFC 4787, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*

RFC 5382, *NAT Behavioral Requirements for TCP*

RFC 5508, *NAT Behavioral Requirements for ICMP*

RFC 6146, *Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers*

RFC 6333, *Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion*

RFC 6334, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Option for Dual-Stack Lite*

RFC 6887, *Port Control Protocol (PCP)*

RFC 6888, *Common Requirements For Carrier-Grade NATs (CGNs)*

RFC 7753, *Port Control Protocol (PCP) Extension for Port-Set Allocation*

RFC 7915, *IP/ICMP Translation Algorithm*

7.25 Network Configuration Protocol (NETCONF)

RFC 5277, *NETCONF Event Notifications*

RFC 6020, *YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)*

RFC 6022, *YANG Module for NETCONF Monitoring*

RFC 6241, *Network Configuration Protocol (NETCONF)*

RFC 6242, *Using the NETCONF Protocol over Secure Shell (SSH)*

RFC 6243, *With-defaults Capability for NETCONF*

RFC 8342, *Network Management Datastore Architecture (NMDA)* - Startup, Candidate, Running and Intended datastores

RFC 8525, *YANG Library*

RFC 8526, *NETCONF Extensions to Support the Network Management Datastore Architecture* - <get-data> operation

7.26 Open Shortest Path First (OSPF)

RFC 1586, *Guidelines for Running OSPF Over Frame Relay Networks*

RFC 1765, *OSPF Database Overflow*

RFC 2328, *OSPF Version 2*

RFC 3101, *The OSPF Not-So-Stubby Area (NSSA) Option*

RFC 3509, *Alternative Implementations of OSPF Area Border Routers*

RFC 3623, *Graceful OSPF Restart Graceful OSPF Restart* - helper mode

RFC 3630, *Traffic Engineering (TE) Extensions to OSPF Version 2*

RFC 4203, *OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS)*

RFC 4222, *Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance*

RFC 4552, *Authentication/Confidentiality for OSPFv3*

RFC 4576, *Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4577, *OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 5185, *OSPF Multi-Area Adjacency*

RFC 5187, *OSPFv3 Graceful Restart* - helper mode

RFC 5243, *OSPF Database Exchange Summary List Optimization*

RFC 5250, *The OSPF Opaque LSA Option*

RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*

RFC 5340, *OSPF for IPv6*

RFC 5642, *Dynamic Hostname Exchange Mechanism for OSPF*

RFC 5709, *OSPFv2 HMAC-SHA Cryptographic Authentication*

RFC 5838, *Support of Address Families in OSPFv3*

RFC 6549, *OSPFv2 Multi-Instance Extensions*

RFC 6987, *OSPF Stub Router Advertisement*

RFC 7684, *OSPFv2 Prefix/Link Attribute Advertisement*

RFC 7770, *Extensions to OSPF for Advertising Optional Router Capabilities*

RFC 8362, *OSPFv3 Link State Advertisement (LSA) Extensibility*

RFC 8920, *OSPF Application-Specific Link Attributes*

7.27 OpenFlow

TS-007 Version 1.3.1, *OpenFlow Switch Specification* - OpenFlow-hybrid switches

7.28 Path Computation Element Protocol (PCEP)

draft-alvarez-pce-path-profiles-04, *PCE Path Profiles*

draft-dhs-spring-pce-sr-p2mp-policy-00, *PCEP extensions for p2mp sr policy*

draft-ietf-pce-segment-routing-08, *PCEP Extensions for Segment Routing*

RFC 5440, *Path Computation Element (PCE) Communication Protocol (PCEP)*

RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*

RFC 8281, *PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model*

7.29 Point-to-Point Protocol (PPP)

RFC 1332, *The PPP Internet Protocol Control Protocol (IPCP)*

RFC 1377, *The PPP OSI Network Layer Control Protocol (OSINLCP)*

RFC 1661, *The Point-to-Point Protocol (PPP)*

RFC 1662, *PPP in HDLC-like Framing*

RFC 1877, *PPP Internet Protocol Control Protocol Extensions for Name Server Addresses*

RFC 1989, *PPP Link Quality Monitoring*

RFC 1990, *The PPP Multilink Protocol (MP)*

RFC 1994, *PPP Challenge Handshake Authentication Protocol (CHAP)*

RFC 2153, *PPP Vendor Extensions*

RFC 2516, *A Method for Transmitting PPP Over Ethernet (PPPoE)*

RFC 2615, *PPP over SONET/SDH*

RFC 2686, *The Multi-Class Extension to Multi-Link PPP*

RFC 2878, *PPP Bridging Control Protocol (BCP)*

RFC 4638, *Accommodating a Maximum Transit Unit/Maximum Receive Unit (MTU/MRU) Greater Than 1492 in the Point-to-Point Protocol over Ethernet (PPPoE)*

RFC 5072, *IP Version 6 over PPP*

7.30 Policy Management and Credit Control

3GPP TS 29.212 Release 11, *Policy and Charging Control (PCC); Reference points* - Gx support as it applies to wireline environment (BNG)

RFC 4006, *Diameter Credit-Control Application*

RFC 6733, *Diameter Base Protocol*

7.31 Pseudowire

draft-ietf-l2vpn-vpws-iw-oam-04, *OAM Procedures for VPWS Interworking*

MFA Forum 9.0.0, *The Use of Virtual trunks for ATM/MPLS Control Plane Interworking*

MFA Forum 12.0.0, *Multiservice Interworking - Ethernet over MPLS*

MFA Forum 13.0.0, *Fault Management for Multiservice Interworking v1.0*

MFA Forum 16.0.0, *Multiservice Interworking - IP over MPLS*

RFC 3916, *Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)*

RFC 3985, *Pseudo Wire Emulation Edge-to-Edge (PWE3)*

RFC 4385, *Pseudo Wire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN*

RFC 4446, *IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)*

RFC 4447, *Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*

RFC 4448, *Encapsulation Methods for Transport of Ethernet over MPLS Networks*

RFC 4619, *Encapsulation Methods for Transport of Frame Relay over Multiprotocol Label Switching (MPLS) Networks*

RFC 4717, *Encapsulation Methods for Transport Asynchronous Transfer Mode (ATM) over MPLS Networks*

RFC 4816, *Pseudowire Emulation Edge-to-Edge (PWE3) Asynchronous Transfer Mode (ATM) Transparent Cell Transport Service*

RFC 5085, *Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires*

RFC 5659, *An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge*

RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*

RFC 6073, *Segmented Pseudowire*

RFC 6310, *Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping*

RFC 6391, *Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network*

RFC 6575, *Address Resolution Protocol (ARP) Mediation for IP Interworking of Layer 2 VPNs*

RFC 6718, *Pseudowire Redundancy*

RFC 6829, *Label Switched Path (LSP) Ping for Pseudowire Forwarding Equivalence Classes (FECs) Advertised over IPv6*

RFC 6870, *Pseudowire Preferential Forwarding Status bit*

RFC 7023, *MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking*

RFC 7267, *Dynamic Placement of Multi-Segment Pseudowires*

RFC 7392, *Explicit Path Routing for Dynamic Multi-Segment Pseudowires* - ER-TLV and ER-HOP IPv4 Prefix

7.32 Quality of Service (QoS)

RFC 2430, *A Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)*

RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*

RFC 2597, *Assured Forwarding PHB Group*

RFC 3140, *Per Hop Behavior Identification Codes*

RFC 3246, *An Expedited Forwarding PHB (Per-Hop Behavior)*

7.33 Remote Authentication Dial In User Service (RADIUS)

RFC 2865, *Remote Authentication Dial In User Service (RADIUS)*

RFC 2866, *RADIUS Accounting*

RFC 2867, *RADIUS Accounting Modifications for Tunnel Protocol Support*

RFC 2868, *RADIUS Attributes for Tunnel Protocol Support*

RFC 2869, *RADIUS Extensions*

RFC 3162, *RADIUS and IPv6*

RFC 4818, *RADIUS Delegated-IPv6-Prefix Attribute*

RFC 5176, *Dynamic Authorization Extensions to RADIUS*

RFC 6911, *RADIUS attributes for IPv6 Access Networks*

RFC 6929, *Remote Authentication Dial-In User Service (RADIUS) Protocol Extensions*

7.34 Resource Reservation Protocol — Traffic Engineering (RSVP-TE)

draft-newton-mpls-te-dynamic-overbooking-00, *A Diffserv-TE Implementation Model to dynamically change booking factors during failure events*

RFC 2702, *Requirements for Traffic Engineering over MPLS*

RFC 2747, *RSVP Cryptographic Authentication*

RFC 2961, *RSVP Refresh Overhead Reduction Extensions*

RFC 3097, *RSVP Cryptographic Authentication -- Updated Message Type Value*

RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*

RFC 3473, *Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions* - IF_ID RSVP_HOP object with unnumbered interfaces and RSVP-TE graceful restart helper procedures

RFC 3477, *Signalling Unnumbered Links in Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)*

RFC 3564, *Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering*

RFC 3906, *Calculating Interior Gateway Protocol (IGP) Routes Over Traffic Engineering Tunnels*

RFC 4090, *Fast Reroute Extensions to RSVP-TE for LSP Tunnels*

RFC 4124, *Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*

RFC 4125, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*

RFC 4127, *Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*

RFC 4561, *Definition of a Record Route Object (RRO) Node-Id Sub-Object*

RFC 4875, *Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)*

RFC 4950, *ICMP Extensions for Multiprotocol Label Switching*

RFC 5151, *Inter-Domain MPLS and GMPLS Traffic Engineering -- Resource Reservation Protocol-Traffic Engineering (RSVP-TE) Extensions*

RFC 5712, *MPLS Traffic Engineering Soft Preemption*

RFC 5817, *Graceful Shutdown in MPLS and Generalized MPLS Traffic Engineering Networks*

7.35 Routing Information Protocol (RIP)

RFC 1058, *Routing Information Protocol*

RFC 2080, *RIPng for IPv6*

RFC 2082, *RIP-2 MD5 Authentication*

RFC 2453, *RIP Version 2*

7.36 Segment Routing (SR)

draft-bashandy-rtgwg-segment-routing-uloop-06, *Loop avoidance using Segment Routing*

draft-filsfils-spring-srv6-net-pgm-insertion-04, *SRv6 NET-PGM extension: Insertion*

draft-ietf-6man-spring-srv6-oam-10, *Operations, Administration, and Maintenance (OAM) in Segment Routing Networks with IPv6 Data plane (SRv6)*

draft-ietf-bess-srv6-services-07, *SRv6 BGP based Overlay Services*

draft-ietf-idr-bgp-ls-segment-routing-ext-16, *BGP Link-State extensions for Segment Routing*

draft-ietf-idr-bgp-ls-segment-routing-msd-09, *Signaling MSD (Maximum SID Depth) using Border Gateway Protocol Link-State*

draft-ietf-idr-segment-routing-te-policy-09, *Advertising Segment Routing Policies in BGP*

draft-ietf-isis-mpls-elc-10, *Signaling Entropy Label Capability and Entropy Readable Label Depth Using IS-IS* - advertising ELC

draft-ietf-lsr-flex-algo-08, *IGP Flexible Algorithm*

draft-ietf-lsr-isis-srv6-extensions-14, *IS-IS Extension to Support Segment Routing over IPv6 Dataplane*

draft-ietf-ospf-mpls-elc-12, *Signaling Entropy Label Capability and Entropy Readable Label-stack Depth Using OSPF* - advertising ELC

draft-ietf-rtgwg-segment-routing-ti-lfa-01, *Topology Independent Fast Reroute using Segment Routing*

draft-ietf-spring-conflict-resolution-05, *Segment Routing MPLS Conflict Resolution*

draft-ietf-spring-segment-routing-policy-08, *Segment Routing Policy Architecture*

draft-ietf-teas-sr-rsvp-coexistence-rec-02, *Recommendations for RSVP-TE and Segment Routing LSP co-existence*

draft-voyer-6man-extension-header-insertion-10, *Deployments With Insertion of IPv6 Segment Routing Headers*

draft-voyer-pim-sr-p2mp-policy-02, *Segment Routing Point-to-Multipoint Policy*

draft-voyer-spring-sr-p2mp-policy-03, *SR Replication Policy for P2MP Service Delivery*

RFC 8287, *Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes*

RFC 8476, *Signaling Maximum SID Depth (MSD) Using OSPF* - node MSD

RFC 8491, *Signaling Maximum SID Depth (MSD) Using IS-IS* - node MSD

RFC 8660, *Segment Routing with the MPLS Data Plane*

RFC 8661, *Segment Routing MPLS Interworking with LDP*

RFC 8663, *MPLS Segment Routing over IP* - BGP SR with SR-MPLS-over-UDP/IP

RFC 8665, *OSPF Extensions for Segment Routing*

RFC 8666, *OSPFv3 Extensions for Segment Routing*

RFC 8667, *IS-IS Extensions for Segment Routing*

RFC 8669, *Segment Routing Prefix Segment Identifier Extensions for BGP*

RFC 8754, *IPv6 Segment Routing Header (SRH)*

RFC 8986, *Segment Routing over IPv6 (SRv6) Network Programming*

7.37 Simple Network Management Protocol (SNMP)

RFC 1157, *A Simple Network Management Protocol (SNMP)*

RFC 1215, *A Convention for Defining Traps for use with the SNMP*

RFC 1901, *Introduction to Community-based SNMPv2*

RFC 3410, *Introduction and Applicability Statements for Internet Standard Management Framework*

RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*

RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*

RFC 3413, *Simple Network Management Protocol (SNMP) Applications*

RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*

RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*

RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*

RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)* - SNMP over UDP over IPv4

RFC 3584, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*

RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model*

7.38 Simple Network Management Protocol (SNMP) - Management Information Base (MIB)

draft-ietf-snmpv3-update-mib-05, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

draft-ietf-isis-wg-mib-06, *Management Information Base for Intermediate System to Intermediate System (IS-IS)*

draft-ietf-mboned-msdp-mib-01, *Multicast Source Discovery protocol MIB*

draft-ietf-mpls-ldp-mib-07, *Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)*

draft-ietf-mpls-lsr-mib-06, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base Using SMIv2*

draft-ietf-mpls-te-mib-04, *Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base*

draft-ietf-ospf-mib-update-08, *OSPF Version 2 Management Information Base*

draft-ietf-vrrp-unified-mib-06, *Definitions of Managed Objects for the VRRP over IPv4 and IPv6* - IPv6

ianaaddressfamilynumbers-mib, *IANA-ADDRESS-FAMILY-NUMBERS-MIB*

ianagmplstc-mib, *IANA-GMPLS-TC-MIB*

ianaiftype-mib, *IANAifType-MIB*

ianaiprouteprotocol-mib, *IANA-RTPROTO-MIB*

IEEE8021-CFM-MIB, *IEEE P802.1ag(TM) CFM MIB*

IEEE8021-PAE-MIB, *IEEE 802.1X MIB*

IEEE8023-LAG-MIB, *IEEE 802.3ad MIB*

LLDP-MIB, *IEEE P802.1AB(TM) LLDP MIB*

RFC 1212, *Concise MIB Definitions*

RFC 1213, *Management Information Base for Network Management of TCP/IP-based Internets: MIB-II*

RFC 1724, *RIP Version 2 MIB Extension*

RFC 2021, *Remote Network Monitoring Management Information Base Version 2 using SMIv2*

RFC 2115, *Management Information Base for Frame Relay DTEs Using SMIv2*

RFC 2206, *RSVP Management Information Base using SMIv2*

RFC 2213, *Integrated Services Management Information Base using SMIv2*

RFC 2494, *Definitions of Managed Objects for the DS0 and DS0 Bundle Interface Type*

RFC 2514, *Definitions of Textual Conventions and OBJECT-IDENTITIES for ATM Management*

RFC 2515, *Definitions of Managed Objects for ATM Management*

RFC 2578, *Structure of Management Information Version 2 (SMIv2)*

RFC 2579, *Textual Conventions for SMIv2*

RFC 2580, *Conformance Statements for SMIv2*

RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*

RFC 2819, *Remote Network Monitoring Management Information Base*

RFC 2856, *Textual Conventions for Additional High Capacity Data Types*

RFC 2863, *The Interfaces Group MIB*

RFC 2864, *The Inverted Stack Table Extension to the Interfaces Group MIB*

RFC 2933, *Internet Group Management Protocol MIB*

RFC 3014, *Notification Log MIB*

RFC 3165, *Definitions of Managed Objects for the Delegation of Management Scripts*

RFC 3231, *Definitions of Managed Objects for Scheduling Management Operations*

RFC 3273, *Remote Network Monitoring Management Information Base for High Capacity Networks*

RFC 3419, *Textual Conventions for Transport Addresses*

RFC 3498, *Definitions of Managed Objects for Synchronous Optical Network (SONET) Linear Automatic Protection Switching (APS) Architectures*

RFC 3592, *Definitions of Managed Objects for the Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) Interface Type*

RFC 3593, *Textual Conventions for MIB Modules Using Performance History Based on 15 Minute Intervals*

RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types*

RFC 3637, *Definitions of Managed Objects for the Ethernet WAN Interface Sublayer*

RFC 3877, *Alarm Management Information Base (MIB)*

RFC 3895, *Definitions of Managed Objects for the DS1, E1, DS2, and E2 Interface Types*

RFC 3896, *Definitions of Managed Objects for the DS3/E3 Interface Type*

RFC 4001, *Textual Conventions for Internet Network Addresses*

RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*

RFC 4113, *Management Information Base for the User Datagram Protocol (UDP)*

RFC 4220, *Traffic Engineering Link Management Information Base*

RFC 4273, *Definitions of Managed Objects for BGP-4*

RFC 4292, *IP Forwarding Table MIB*

RFC 4293, *Management Information Base for the Internet Protocol (IP)*

RFC 4631, *Link Management Protocol (LMP) Management Information Base (MIB)*

RFC 4878, *Definitions and Managed Objects for Operations, Administration, and Maintenance (OAM) Functions on Ethernet-Like Interfaces*

RFC 7420, *Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module*

SFLOW-MIB Version 1.3 (Draft 5), *sFlow MIB*

7.39 Timing

GR-1244-CORE Issue 3, *Clocks for the Synchronized Network: Common Generic Criteria*

GR-253-CORE Issue 3, *SONET Transport Systems: Common Generic Criteria*

IEEE 1588-2008, *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*

ITU-T G.781, *Synchronization layer functions*

ITU-T G.813, *Timing characteristics of SDH equipment slave clocks (SEC)*

ITU-T G.8261, *Timing and synchronization aspects in packet networks*

ITU-T G.8262, *Timing characteristics of synchronous Ethernet equipment slave clock (EEC)*

ITU-T G.8264, *Distribution of timing information through packet networks*

ITU-T G.8265.1, *Precision time protocol telecom profile for frequency synchronization*

ITU-T G.8275.1, *Precision time protocol telecom profile for phase/time synchronization with full timing support from the network*

RFC 3339, *Date and Time on the Internet: Timestamps*

RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*

7.40 Two-Way Active Measurement Protocol (TWAMP)

RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)* - server, unauthenticated mode

RFC 5938, *Individual Session Control Feature for the Two-Way Active Measurement Protocol (TWAMP)*

RFC 6038, *Two-Way Active Measurement Protocol (TWAMP) Reflect Octets and Symmetrical Size Features*

RFC 8545, *Well-Known Port Assignments for the One-Way Active Measurement Protocol (OWAMP) and the Two-Way Active Measurement Protocol (TWAMP)* - TWAMP

RFC 8762, *Simple Two-Way Active Measurement Protocol* - unauthenticated

7.41 Virtual Private LAN Service (VPLS)

RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*

RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*

RFC 5501, *Requirements for Multicast Support in Virtual Private LAN Services*

RFC 6074, *Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)*

RFC 7041, *Extensions to the Virtual Private LAN Service (VPLS) Provider Edge (PE) Model for Provider Backbone Bridging*

RFC 7117, *Multicast in Virtual Private LAN Service (VPLS)*

7.42 Voice and Video

DVB BlueBook A86, *Transport of MPEG-2 TS Based DVB Services over IP Based Networks*

ETSI TS 101 329-5 Annex E, *QoS Measurement for VoIP - Method for determining an Equipment Impairment Factor using Passive Monitoring*

ITU-T G.1020 Appendix I, *Performance Parameter Definitions for Quality of Speech and other Voiceband Applications Utilizing IP Networks - Mean Absolute Packet Delay Variation & Markov Models*

ITU-T G.107, *The E Model - A computational model for use in planning*

ITU-T P.564, *Conformance testing for voice over IP transmission quality assessment models*

RFC 3550, *RTP: A Transport Protocol for Real-Time Applications* - Appendix A.8

RFC 4585, *Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)*

RFC 4588, *RTP Retransmission Payload Format*

7.43 Wireless Local Area Network (WLAN) Gateway

3GPP TS 23.402, *Architecture enhancements for non-3GPP accesses* - S2a roaming based on GPRS

7.44 Yet Another Next Generation (YANG)

RFC 6991, *Common YANG Data Types*

RFC 7950, *The YANG 1.1 Data Modeling Language*

RFC 7951, *JSON Encoding of Data Modeled with YANG*

7.45 Yet Another Next Generation (YANG) - OpenConfig Modules

openconfig-aaa.yang version 0.4.0, *OpenConfig AAA Module*

openconfig-aaa-radius.yang version 0.3.0, *OpenConfig AAA RADIUS Module*

openconfig-aaa-tacacs.yang version 0.3.0, *OpenConfig AAA TACACS+ Module*

openconfig-acl.yang version 1.0.0, *OpenConfig ACL Module*

openconfig-bfd.yang version 0.1.0, *OpenConfig BFD Module*

openconfig-bgp.yang version 3.0.1, *OpenConfig BGP Module*

openconfig-bgp-common.yang version 3.0.1, *OpenConfig BGP Common Module*

openconfig-bgp-common-multiprotocol.yang version 3.0.1, *OpenConfig BGP Common Multiprotocol Module*

openconfig-bgp-common-structure.yang version 3.0.1, *OpenConfig BGP Common Structure Module*

openconfig-bgp-global.yang version 3.0.1, *OpenConfig BGP Global Module*

openconfig-bgp-neighbor.yang version 3.0.1, *OpenConfig BGP Neighbor Module*

openconfig-bgp-peer-group.yang version 3.0.1, *OpenConfig BGP Peer Group Module*

openconfig-bgp-policy.yang version 4.0.1, *OpenConfig BGP Policy Module*

openconfig-if-aggregate.yang version 2.0.0, *OpenConfig Interfaces Aggregated Module*

openconfig-if-ethernet.yang version 2.0.0, *OpenConfig Interfaces Ethernet Module*

openconfig-if-ip.yang version 2.0.0, *OpenConfig Interfaces IP Module*

openconfig-if-ip-ext.yang version 2.0.0, *OpenConfig Interfaces IP Extensions Module*

openconfig-interfaces.yang version 2.0.0, *OpenConfig Interfaces Module*

openconfig-isis.yang version 0.3.0, *OpenConfig IS-IS Module*

openconfig-isis-policy.yang version 0.3.0, *OpenConfig IS-IS Policy Module*

openconfig-isis-routing.yang version 0.3.0, *OpenConfig IS-IS Routing Module*

openconfig-lacp.yang version 1.1.0, *OpenConfig LACP Module*

openconfig-lldp.yang version 0.1.0, *OpenConfig LLDP Module*

openconfig-local-routing.yang version 1.0.1, *OpenConfig Local Routing Module*

openconfig-mpls.yang version 2.3.0, *OpenConfig MPLS Module*

openconfig-mpls-ldp.yang version 3.0.2, *OpenConfig MPLS LDP Module*

openconfig-mpls-rsvp.yang version 2.3.0, *OpenConfig MPLS RSVP Module*

openconfig-mpls-te.yang version 2.3.0, *OpenConfig MPLS TE Module*

openconfig-network-instance.yang version 0.8.0, *OpenConfig Network Instance Module*

openconfig-packet-match.yang version 1.0.0, *OpenConfig Packet Match Module*

openconfig-platform.yang version 0.12.2, *OpenConfig Platform Module*

openconfig-platform-fan.yang version 0.1.1, *OpenConfig Platform Fan Module*

openconfig-platform-linecard.yang version 0.1.2, *OpenConfig Platform Linecard Module*

openconfig-relay-agent.yang version 0.1.0, *OpenConfig Relay Agent Module*

openconfig-routing-policy.yang version 3.0.0, *OpenConfig Routing Policy Module*

openconfig-rsvp-sr-ext.yang version 0.1.0, *OpenConfig RSVP-TE and SR Extensions Module*

openconfig-system-logging.yang version 0.3.1, *OpenConfig System Logging Module*

openconfig-system-terminal.yang version 0.3.0, *OpenConfig System Terminal Module*

openconfig-telemetry.yang version 0.5.0, *OpenConfig Telemetry Module*

openconfig-vlan.yang version 2.0.0, *OpenConfig VLAN Module*

Customer Document and Product Support

Customer Documentation

Customer Documentation Welcome Page

Technical Support

Product Support Portal

Documentation Feedback

Customer Documentation Feedback