



Nokia Service Router Linux

SEGMENT ROUTING GUIDE
RELEASE 21.11

3HE 17920 AAAA

Issue 1

December 2021

© 2021 Nokia.

Use subject to Terms available at: www.nokia.com/terms/.

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.

This document includes Nokia proprietary and confidential information, which may not be distributed or disclosed to any third parties without the prior written consent of Nokia.

This document is intended for use by Nokia's customers ("You"/"Your") in connection with a product purchased or licensed from any company within Nokia Group of Companies. Use this document as agreed. You agree to notify Nokia of any errors you may find in this document; however, should you elect to use this document for any purpose(s) for which it is not intended, You understand and warrant that any determinations You may make or actions You may take will be based upon Your independent judgment and analysis of the content of this document.

Nokia reserves the right to make changes to this document without notice. At all times, the controlling version is the one available on Nokia's site.

No part of this document may be modified.

NO WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY OF AVAILABILITY, ACCURACY, RELIABILITY, TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, IS MADE IN RELATION TO THE CONTENT OF THIS DOCUMENT. IN NO EVENT WILL NOKIA BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL OR ANY LOSSES, SUCH AS BUT NOT LIMITED TO LOSS OF PROFIT, REVENUE, BUSINESS INTERRUPTION, BUSINESS OPPORTUNITY OR DATA THAT MAY ARISE FROM THE USE OF THIS DOCUMENT OR THE INFORMATION IN IT, EVEN IN THE CASE OF ERRORS IN OR OMISSIONS FROM THIS DOCUMENT OR ITS CONTENT.

Copyright and trademark: Nokia is a registered trademark of Nokia Corporation. Other product names mentioned in this document may be trademarks of their respective owners.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

© 2021 Nokia.

Table of contents

1 About this guide.....	4
1.1 What's new.....	4
1.2 Precautionary and information messages.....	4
1.3 Conventions.....	4
2 About segment routing.....	6
2.1 IS-IS extensions for segment routing.....	7
2.2 Supported functionality.....	9
3 SR-MPLS configuration on the default network-instance.....	11
3.1 Defining the SRGB and enabling SR-MPLS.....	11
4 SR-MPLS configuration on the IS-IS instance.....	12
4.1 Configuring the node SID.....	12
4.2 Defining the dynamic SRLB.....	12
4.3 Enabling SR-MPLS on the IS-IS instance.....	13
5 BGP shortcuts configuration over segment routing tunnels.....	15
5.1 Configuring BGP shortcuts.....	15
6 Segment routing display commands.....	16
6.1 Displaying the SID database.....	16
6.2 Displaying label block information.....	17
6.3 Displaying tunnel table entries.....	17

1 About this guide

This document describes configuration details for the Segment Routing feature set used with the Nokia Service Router Linux (SR Linux).

This document is intended for network technicians, administrators, operators, service providers, and others who need to understand how the router is configured.

**Note:**

This manual covers the current release and may also contain some content that will be released in later maintenance loads. See the *SR Linux Release Notes* for information on features supported in each load.

1.1 What's new

This is the first release of this document.

1.2 Precautionary and information messages

The following are information symbols used in the documentation.



DANGER: Danger warns that the described activity or situation may result in serious personal injury or death. An electric shock hazard could exist. Before you begin work on this equipment, be aware of hazards involving electrical circuitry, be familiar with networking environments, and implement accident prevention procedures.



WARNING: Warning indicates that the described activity or situation may, or will, cause equipment damage, serious performance problems, or loss of data.



Caution: Caution indicates that the described activity or situation may reduce your component or system performance.



Note: Note provides additional operational information.



Tip: Tip provides suggestions for use or best practices.

1.3 Conventions

Nokia SR Linux documentation uses the following command conventions.

- **Bold** type indicates a command that the user must enter.
- Input and output examples are displayed in `Courier` text.

-
- An open right-angle bracket indicates a progression of menu choices or simple command sequence (often selected from a user interface). Example: **start > connect to**.
 - Angle brackets (< >) indicate an item that is not used verbatim. For example, for the command **show ethernet <name>**, *name* should be replaced with the name of the interface.
 - A vertical bar (|) indicates a mutually exclusive argument.
 - Square brackets ([]) indicate optional elements.
 - Braces ({ }) indicate a required choice. When braces are contained within square brackets, they indicate a required choice within an optional element.
 - *Italic* type indicates a variable.

Generic IP addresses are used in examples. Replace these with the appropriate IP addresses used in the system.

2 About segment routing

Segment routing can perform shortest path routing or source routing using the concept of abstract segment. A segment can represent the local prefix of a node, a specific adjacency of the node (interface or next hop), a service context, or a specific explicit path over the network. Each segment is identified by a Segment ID (SID).

With segment routing, the source router can define the end-to-end path for packets to reach a destination using an ordered list of segments represented by their SIDs. Each SID represents actions that subsequent nodes in the network execute, such as forwarding a packet to a specific destination or interface.

Unlike a typical Multiprotocol Label Switching (MPLS) architecture, a segment routing network does not require Label Distribution Protocol (LDP) or Resource Reservation Protocol - Traffic Engineering (RSVP-TE) to set up tunnels. Instead, segment routing extensions to the IGP (such as IS-IS) provide support for allocation and signaling of SID information.

SR-MPLS

When segment routing is instantiated over the MPLS data plane (referred to as SR-MPLS), a SID is represented by a standard MPLS label or an index that maps to an MPLS label. To route a packet to a destination, the source router pushes onto the packet one or more MPLS labels representing the required SIDs. Each subsequent node forwards the packet based on the outer label, removes the outer label, and forwards the packet to the next segment on the path.

The MPLS data plane requires no modifications to support SR-MPLS.

Node SID

A prefix SID is a segment type that represents the ECMP-aware shortest path to reach a particular IP prefix from any IGP topology location. A node SID is a type of prefix SID that identifies a specific node in the IGP topology using a loopback address as the prefix. When a node SID is included in an MPLS label stack, it instructs the receiving node to forward the packet to the destination along the ECMP-aware shortest path determined by the IGP.

You allocate the node SIDs to use in the network in a similar fashion as the loopback IP addresses. Like all prefix SIDs, node SIDs must be unique within the domain, and are allocated in the form of an MPLS label (or index). The range of labels available for prefix and node SIDs is defined in the SRGB.

SRGB

The Segment Routing Global Block (SRGB) defines the range of MPLS labels available for global segments, such as prefix and node SIDs. The SRGB is defined as one contiguous block of MPLS labels. Nokia strongly recommends using identical SRGBs on all nodes within the SR domain. The use of identical SRGBs simplifies troubleshooting because the same MPLS label represents the same prefix or node SID on each node.

The SRGB configuration is supported on the default network-instance only.

SRLB

While the SRGB defines a range of MPLS labels for global segments, the Segment Routing Label Block (SRLB) defines a range of MPLS labels for local segments, such as adjacency SIDs. An adjacency SID is a segment type that represents an instruction to forward packets over a specific (one-hop) adjacency between two nodes in the IGP.

The SRLB is configurable per IGP protocol instance (on the default network-instance only). It is defined as one contiguous block of MPLS labels per IGP instance. SR Linux currently supports only one dynamic label block for the SRLB.



Note: Adjacency SID functionality is not supported in the current release. However, to enable segment routing on the IS-IS instance, you must configure a dynamic label block SRLB to reserve labels for future allocation of dynamic adjacency SIDs. Allocation of dynamic adjacency SIDs from this block is targeted for a future release.

For more information about MPLS and MPLS labels, see the *SR Linux MPLS Guide*.

2.1 IS-IS extensions for segment routing

IS-IS provides extensions to advertise segment routing specific TLVs. These TLVs allow IS-IS to advertise the SIDs in the domain for each segment.

Segment routing with IS-IS is also known as SR-ISIS. SR-ISIS refers to the segment routing extensions of the IS-IS IGP protocol and the forwarding entries created by those extensions.

With segment routing enabled on the IS-IS instance, IS-IS extensions can support TLVs to advertise IPv4/IPv6 prefix SIDs. The following new sub-TLVs are defined in RFC 8667 and are supported in the implementation of SR-ISIS:

- prefix SID sub-TLV
- SR-Capabilities sub-TLV
- SR-Algorithm sub-TLV

In the current release, SR Linux advertises each configured prefix SID as a node SID.

The following sections describes the behaviors and limitations of the SR-ISIS TLV and sub-TLVs.

Router capability advertisement

In SR Linux, advertisement of the router capability TLV (TLV 242) with the segment routing related sub-TLVs is automatically enabled when segment routing is configured on the IS-IS instance. There is no explicit configuration to enable or disable router capability advertisement. The scope of segment routing capabilities is the entire IS-IS domain.

Origination of SR-Algorithm sub-TLVs

When SR Linux originates the SR-Algorithm capability sub-TLV, the algorithm field is set to 0, indicating shortest path first (SPF) algorithm based on link metric.

Origination of IPv4 and IPv6 prefix SID sub-TLVs

SR Linux originates prefix SID sub-TLVs with the following processing rules and encoding of flags.

- Both IPv4 and IPv6 prefix sub-TLVs originate within MT=0.
- SR Linux originates a single prefix SID sub-TLV per IS-IS IP reachability TLV.
- SR Linux encodes the 32-bit index in the prefix SID sub-TLV. The 24-bit label is not supported.

Table 1: Flag encoding for prefix SID sub-TLV

Flag	Encoding
R-flag	Set if the prefix SID sub-TLV and its corresponding IP reachability TLV are propagated between levels; see below for more details about prefix propagation
N-flag	Always set because SR Linux supports a prefix SID type that is node SID only
P-flag (no-PHP flag)	Always set, meaning the label for the prefix SID is pushed by the PHP router when forwarding to this router. When the SR Linux PHP router (with P-flag set to 1), processes a received prefix SID with the P-flag set to zero, it uses implicit-null for the outgoing label toward the router that advertised it.
E-flag (Explicit-Null flag)	Always set to 0. However, the SR Linux PHP router processes a received prefix SID with the E-flag set to 1, and if the P-flag is also set to 1, it pushes explicit-null for the outgoing label toward the router which advertised it. The system ignores the value of the E-flag if the P-flag is not set.
V-flag	Always set to 0 to indicate an index value for the SID
L-flag	Always set to 0 to indicate that the SID index value is not locally significant
algorithm field	Always set to 0 to indicate shortest path first (SPF) algorithm based on link metric and is not checked on a received prefix SID sub-TLV

Receiving IPv4 and IPv6 prefix SID sub-TLVs

SR Linux processes a prefix SID sub-TLV using the following rules.

- Decodes prefix SID sub-TLVs that are received in TLVs 135, 235, 236 and 237 for representation in the YANG state model of the LSDB
- Ignores prefix SID sub-TLVs received in TLVs 235 and 237 for further processing. They do not appear in the state representation of the local SID database (only in the LSDB).
- If the local flag is set or the algorithm is not 0, deems a prefix SID sub-TLV in a received TLV 135 or TLV 236 invalid and makes it inactive.
 - If the MPLS label value implied by the received prefix SID sub-TLV falls outside the range of the SRGB block, deems the node SID invalid and makes it inactive (reason = sid-index-out-of-range). No forwarding state is created for invalid entries.
- Resolves a prefix SID sub-TLV received without the N-flag set but with the prefix length equal to 32 (for IPv4) or 128 (for IPv6).
- Does not resolve a prefix SID sub-TLV received with the N-flag set and a prefix length different than 32 (for IPv4) or 128 (for IPv6).
- Resolves a prefix SID received within an IP reachability TLV based on the following route preference:
 - SID received via L1 in a prefix SID sub-TLV part of the IP reachability TLV
 - SID received via L2 in a prefix SID sub-TLV part of the IP reachability TLV
- Processes the first prefix SID sub-TLV only if multiple are received within the same IS-IS IP reachability TLV.
- An L1L2 router propagates a prefix received in an IP reachability TLV, along with the prefix SID sub-TLV, from L1 to L2. A router in L2 sets up an SR tunnel to the L1 router via the L1L2 router, which acts as an LSR.

- By default, an L1L2 router does not propagate from L2 to L1 a prefix received in an IP reachability TLV, along with the prefix SID sub-TLV.
- If an ABR summarizes a prefix, the prefix SID sub-TLV is not propagated with the summarized route between levels. To propagate the node SID for a /32 prefix, you must disable route summarization.

Origination of IPv4 and IPv6 adjacency SID sub-TLVs

The current release does not support origination of adjacency SID sub-TLVs in TLVs 22 and 222.

Receiving IPv4 and IPv6 adjacency SID sub-TLVs

- Adjacency SID sub-TLVs received in TLVs 22 and 222 are decoded for representation in the YANG state model of the LSDB.
- The current release does not validate received adjacency SID sub-TLVs or use them to program any datapath objects.

2.2 Supported functionality

SR-MPLS-IPV4 dataplane on 7250 IXR-6/10

- Maximum of one label pushed per packet (corresponding to a remote node SID)
- Maximum of one label popped per packet (corresponding to a local node SID)
- Transit and egress LSR ECMP hashing
- Ingress LSR ECMP hashing
- Prefix/Node SID (IPv4/IPv6)

MPLS label management

- Configurable SRGB for the default network-instance:
 - SRGB is a reference to a static shared label range
 - SRGB must be one contiguous block
- Configurable SRLB for the IS-IS instance of the default network-instance:
 - SRLB is a reference to a dedicated, dynamic label range
 - Each SRLB must be one contiguous block

IS-IS segment routing extensions

- IPv4/IPv6 node/prefix SID
- SID collision handling (no event generated when collision detected)

BGP shortcuts over segment routing tunnels

SR-ISIS-IPv4 tunnels in TTM can resolve BGP IPv4 routes.

SR-MPLS OAM

- ICMP tunneling
- RFC 4950 extensions

ICMP tunneling

SR Linux supports ICMP extensions for MPLS to support debugging and tracing in MPLS networks, including SR-MPLS networks. With ICMP tunneling enabled, ICMP messages can be tunneled to the endpoint of the tunnel and then returned through IP routing. For more information, see the *SR Linux MPLS Guide*.

3 SR-MPLS configuration on the default network-instance

To configure segment routing on the MPLS data plane, you must define the SRGB and enable SR-MPLS on the default network-instance. SR-MPLS is supported on the default network-instance only.

3.1 Defining the SRGB and enabling SR-MPLS

The SRGB references a static shared MPLS label range that must be one contiguous block of labels. As part of the SRGB configuration, you must add the **segment-routing mpls** context on the default network-instance, which enables segment routing based on the MPLS dataplane.

To configure the SRGB, you must first define a static MPLS label range in the **system** context. You must then assign that MPLS label range to be available for use by the SRGB.



Note: Nokia strongly recommends to use identical SRGBs on all nodes within the SR domain.

Example: Define the MPLS label range for the SRGB

The following example defines static MPLS label range `srgb-range-1` for the SRGB.

```
--{ * candidate shared default }--[ ]--
# info system mpls label-ranges static srgb-range-1
  system {
    mpls {
      label-ranges {
        static srgb-range-1 {
          shared true
          start-label 16001
          end-label 16999
        }
      }
    }
  }
}
```

Example: Assign the MPLS label range to the SRGB

The following example assigns static label range `srgb-range-1` to the SRGB.

```
--{ * candidate shared default }--[ ]--
# info network-instance default segment-routing mpls global-block
  network-instance default {
    segment-routing {
      mpls {
        global-block {
          label-range srgb-range-1
        }
      }
    }
  }
}
```

4 SR-MPLS configuration on the IS-IS instance

To configure SR-MPLS on an IS-IS instance, you must specify the node SID, define the Segment Routing Label Block (SRLB), and enable SR-MPLS on the IS-IS instance. SR-MPLS is configurable on one IS-IS instance only (on the default network-instance).

4.1 Configuring the node SID

Prerequisites

- Configure the primary address of a loopback (loN.n) or system0.0 subinterface with the required node SID prefix, and set the subinterface as an IS-IS interface.

To define the node SID in the IS-IS interface configuration, you must specify an index value that is relative to the SRGB. In this case, the MPLS label value is calculated as follows:

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

For example, if the SRGB range starts at 16000, and the index value is 1, the resulting MPLS label value for the SID is 16001.

To configure the node SID corresponding to an IPv4 or IPv6 prefix, assign an index value to the loopback (or system) subinterface in the IS-IS instance.

Example: Configure node SID

The following example adds IPv4 node SID index 1 to loopback subinterface lo0.1.

```
--{ * candidate shared default }--[ ]--
# info network-instance default protocols isis instance sr-isis-1 interface lo0.1
network-instance default {
  protocols {
    isis {
      instance sr-isis-1 {
        interface lo0.1 {
          segment-routing {
            mpls {
              ipv4-node-sid {
                index 1
              }
            }
          }
        }
      }
    }
  }
}
```

4.2 Defining the dynamic SRLB

To enable MPLS segment routing in the single IS-IS instance of the default network-instance, you must configure the IS-IS instance with an SRLB. The SRLB references a dynamic, dedicated (non-shared) label range that must be one contiguous block of labels.

This label block sets aside labels for future use. SR Linux does not actually allocate labels from this block in the current release.



Note: Adjacency SID functionality is not currently supported. However, to enable segment routing on the IS-IS instance, you must configure an SRLB to reserve labels for future use.

To configure the SRLB, you must first define a dynamic MPLS label range. You must then assign that MPLS label range to be available for use by the SRLB.

Example: Define the MPLS label range for the SRLB

The following example defines a dynamic MPLS label range (`srlb-dynamic-1`) for use by the SRLB.

```
--{ * candidate shared default }--[ ]--
# info system mpls label-ranges dynamic srlb-dynamic-1
  system {
    mpls {
      label-ranges {
        dynamic srlb-dynamic-1 {
          start-label 15001
          end-label 15999
        }
      }
    }
  }
}
```

Example: Assign the MPLS label range to the SRLB

The following example assigns the defined MPLS label range (`srlb-dynamic-1`) to the SRLB.

```
--{ * candidate shared default }--[ ]--
# info network-instance default protocols isis dynamic-label-block
  network-instance default {
    protocols {
      isis {
        dynamic-label-block srlb-dynamic-1
      }
    }
  }
}
```

4.3 Enabling SR-MPLS on the IS-IS instance

Prerequisites

- Configure IS-IS as the IGP on the segment routing nodes in your network. See the *SR Linux Configuration Basics Guide*.
- Define the dynamic SRLB.

To enable the IS-IS extensions for advertisement of segment routing capabilities, you must enable SR-MPLS on the IS-IS instance.

Example: Enable SR-MPLS

The following example shows the SR-MPLS configuration on IS-IS instance `sr-isis-1`.

```
--{ * candidate shared default }--[ ]--  
# info network-instance default protocols isis instance sr-isis-1 segment-routing mpls  
  network-instance default {  
    protocols {  
      isis {  
        instance sr-isis-1 {  
          segment-routing {  
            mpls {  
            }  
          }  
        }  
      }  
    }  
  }  
}
```

5 BGP shortcuts configuration over segment routing tunnels

When you have segment routing enabled in your network, you can specify a segment routing tunnel as the next hop for a BGP route. This capability is referred to as BGP shortcuts over segment routing.

To configure BGP shortcuts, you must configure the BGP protocol with the allowed tunnel types and the required tunnel resolution mode.

5.1 Configuring BGP shortcuts

To configure BGP shortcuts, you must configure the BGP protocol in the default network-instance with the allowed tunnel types for next-hop resolution. You must also define the tunnel resolution mode, which determines the order of preference and the fallback when using tunnels in the tunnel table instead of routes in the FIB. Available options are as follows:

- require: requires tunnel table lookup instead of FIB lookup
- prefer: prefers tunnel table lookup over FIB lookup
- disabled (default): performs FIB lookup only

Example: Configure BGP next-hop resolution to allow segment routing tunnels

The following example shows the BGP next-hop resolution configuration to allow SR-ISIS tunnels, with the tunnel mode set to prefer.

```
--{ * candidate shared default }--[ ]--
# info network-instance default protocols bgp ipv4-unicast next-hop-resolution ipv4-next-hops
  tunnel-resolution
    network-instance default {
      protocols {
        bgp {
          ipv4-unicast {
            next-hop-resolution {
              ipv4-next-hops {
                tunnel-resolution {
                  mode prefer
                  allowed-tunnel-types [
                    sr-isis
                  ]
                }
              }
            }
          }
        }
      }
    }
  }
```

6 Segment routing display commands

SR Linux supports display commands to provide operational information about segment routing, including the following:

- SID database
- label block information
- tunnel-table entries

6.1 Displaying the SID database

Use the **info from state** command to display information about the SID database.

Example:

```
--{ candidate shared default }--[ ]--
# info from state network-instance default segment-routing mpls sid-database
network-instance default {
  segment-routing {
    mpls {
      sid-database {
        prefix-sid 10.20.1.1/32 sid-label-value 3010 protocol isis protocol-instance 0
protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 10.20.1.2/32 sid-label-value 3020 protocol isis protocol-instance 0
protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 10.20.1.3/32 sid-label-value 3030 protocol isis protocol-instance 0
protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 10.20.1.4/32 sid-label-value 3040 protocol isis protocol-instance 0
protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 10.20.1.5/32 sid-label-value 3050 protocol isis protocol-instance 0
protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 10.20.1.6/32 sid-label-value 3060 protocol isis protocol-instance 0
protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 2001:10:20:1::1/128 sid-label-value 3610 protocol isis protocol-
instance 0 protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 2001:10:20:1::2/128 sid-label-value 3620 protocol isis protocol-
instance 0 protocol-multi-topology 0 algorithm 0 {
          active true
        }
        prefix-sid 2001:10:20:1::3/128 sid-label-value 3630 protocol isis protocol-
instance 0 protocol-multi-topology 0 algorithm 0 {
```



```

        active true
    }
    prefix-sid 2001:10:20:1::4/128 sid-label-value 3640 protocol isis protocol-
instance 0 protocol-multi-topology 0 algorithm 0 {
    active true
    }
    prefix-sid 2001:10:20:1::5/128 sid-label-value 3650 protocol isis protocol-
instance 0 protocol-multi-topology 0 algorithm 0 {
    active true
    }
    prefix-sid 2001:10:20:1::6/128 sid-label-value 3660 protocol isis protocol-
instance 0 protocol-multi-topology 0 algorithm 0 {
    active true
    }
    }
    }
}

```

6.2 Displaying label block information

Use the **info from state** command to display information about the configured label blocks.

Example:

```

--{ * candidate shared default }--[ ]--
# info from state system mpls label-ranges
system {
    mpls {
        label-ranges {
            static {
                name srgb-range-1
                start-label 16001
                end-label 16999
                allocated-labels 10
                free-labels 989
                status ready
            }
        }
    }
}

```

6.3 Displaying tunnel table entries

Use the **show network-instance default tunnel-table** command to display information about segment routing tunnel table entries. You can adjust the output of the report to filter by address type, encapsulation type, tunnel type, and destination prefix.

Example: Show all tunnels

```

--{ running }--[ ]--
# show network-instance default tunnel-table all
-----
IPv4 tunnel table of network-instance "default"
-----
+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

```

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|IPv4 Prefix|Encaps|Tunnel|Tunnel|FIB|Metric|Preference|Last Update|Next-hop|Next-hop|
|Type|Type|ID| | | | | | (Type) | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|10.20.1.1/32|mpls|sr-isis|13010|Y|10|11|2021-11-10T|10.10.1.1|ethernet-1/33.1|
|14:17:32.202Z|(mpls)|
|10.20.1.3/32|mpls|sr-isis|13030|Y|10|11|2021-11-10T|10.10.3.3|ethernet-1/34.1|
|14:18:02.299Z|(mpls)|
|10.20.1.4/32|mpls|sr-isis|13040|Y|10|11|2021-11-10T|10.10.4.4|ethernet-1/8.1|
|14:18:26.729Z|(mpls)|
|10.20.1.5/32|mpls|sr-isis|13050|Y|10|11|2021-11-10T|10.10.23.5|ethernet-1/9.1|
|14:19:02.343Z|(mpls)|
|10.20.1.6/32|mpls|sr-isis|13060|Y|10|11|2021-11-10T|10.10.14.6|ethernet-1/11.1|
|14:19:38.140Z|(mpls)|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 SR-ISIS tunnels, 5 active, 0 inactive
-----
IPv6 tunnel table of network-instance "default"
-----
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|IPv6 Prefix|Encaps|Tunnel|Tunnel|FIB|Metric|Prefe|Last Update|Next-hop|Next-hop|
|Type|Type|ID| | | | | | (Type) | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|2001:10:20:|mpls|sr-isis|13610|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/33.1|
|1::1/128|14:17:32.204Z|1ff:feff:20(mpls)|
|2001:10:20:|mpls|sr-isis|13630|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/34.1|
|1::3/128|14:18:03.316Z|3ff:feff:21(mpls)|
|2001:10:20:|mpls|sr-isis|13640|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/8.1|
|1::4/128|14:18:31.638Z|4ff:feff:20(mpls)|
|2001:10:20:|mpls|sr-isis|13650|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/9.1|
|1::5/128|14:19:04.882Z|5ff:feff:20(mpls)|
|2001:10:20:|mpls|sr-isis|13660|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/11.1|
|1::6/128|14:19:43.545Z|6ff:feff:4(mpls)|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 SR-ISIS tunnels, 5 active, 0 inactive
--{ running }--[ ]--

```

Example: Show IPv4 tunnels

```

--{ running }--[ ]--
# show network-instance default tunnel-table ipv4
-----
IPv4 tunnel table of network-instance "default"
-----
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|IPv4 Prefix|Encaps|Tunnel|Tunnel|FIB|Metric|Preference|Last Update|Next-hop|Next-hop|
|Type|Type|ID| | | | | | (Type) | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|10.20.1.1/32|mpls|sr-isis|13010|Y|10|11|2021-11-10T|10.10.1.1|ethernet-1/33.1|
|14:17:32.202Z|(mpls)|
|10.20.1.3/32|mpls|sr-isis|13030|Y|10|11|2021-11-10T|10.10.3.3|ethernet-1/34.1|
|14:18:02.299Z|(mpls)|
|10.20.1.4/32|mpls|sr-isis|13040|Y|10|11|2021-11-10T|10.10.4.4|ethernet-1/8.1|
|14:18:26.729Z|(mpls)|
|10.20.1.5/32|mpls|sr-isis|13050|Y|10|11|2021-11-10T|10.10.23.5|ethernet-1/9.1|
|14:19:02.343Z|(mpls)|
|10.20.1.6/32|mpls|sr-isis|13060|Y|10|11|2021-11-10T|10.10.14.6|ethernet-1/11.1|
|14:19:38.140Z|(mpls)|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 SR-ISIS tunnels, 5 active, 0 inactive

```

Example: Show IPv6 tunnels

```
--{ running }--[ ]--
# show network-instance default tunnel-table ipv6
-----
IPv6 tunnel table of network-instance "default"
-----
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|IPv6 Prefix|Encaps|Tunnel|Tunnel|FIB|Metric|Prefe|Last Update|Next-hop|Next-hop|
| |Type|Type|ID| | |rence| | (Type) | |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|2001:10:20:|mpls|sr-isis|13610|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/33.1|
|1::1/128| | | | | | |14:17:32.202Z|1ff:feff:20(mpls)| |
|2001:10:20:|mpls|sr-isis|13630|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/34.1|
|1::3/128| | | | | | |14:18:02.299Z|3ff:feff:21(mpls)| |
|2001:10:20:|mpls|sr-isis|13640|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/8.1|
|1::4/128| | | | | | | |4ff:feff:20(mpls)| |
|2001:10:20:|mpls|sr-isis|13650|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/9.1|
|1::5/128| | | | | | | |5ff:feff:20(mpls)| |
|2001:10:20:|mpls|sr-isis|13660|Y|10|11|2021-11-10T|fe80::201:|ethernet-1/11.1|
|1::6/128| | | | | | | |6ff:feff:4(mpls)| |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
5 SR-ISIS tunnels, 5 active, 0 inactive
-----
```

Example: Show IPv4 tunnel by destination prefix

```
--{ running }--[ ]--
# show network-instance default tunnel-table ipv4 10.20.1.6/32 type sr-isis detail
-----
Show report for network instance "default" tunnel table
-----
=====
Destination      : 10.20.1.6/32
Encapsulation    : mpls
Tunnel Type      : sr-isis
Metric           : 10
Preference       : 11
Last Update      : 2021-11-10T14:19:38.140Z
FIB Status       : active
Next-hops
  10.10.14.6 (mpls) via [ethernet-1/11.1]
  pushed MPLS labels : [53060]
=====
```

Example: Show IPv6 tunnel by destination prefix

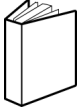
```
# show network-instance default tunnel-table ipv6 2001:10:20:1::6/128 detail
-----
Show report for network instance "default" tunnel table
-----
=====
Destination      : 2001:10:20:1::6/128
Encapsulation    : mpls
Tunnel Type      : sr-isis
Metric           : 10
Preference       : 11
Last Update      : 2021-11-10T14:19:43.545Z
FIB Status       : active
=====
```

Next-hops

fe80::201:6ff:feff:4 (mpls) via [ethernet-1/11.1]
pushed MPLS labels : [53660]

=====

Customer document and product support



Customer documentation

[Customer documentation welcome page](#)



Technical support

[Product support portal](#)



Documentation feedback

[Customer documentation feedback](#)