

Segment Routing Traffic Engineering Commands

This chapter describes the commands used to configure and use Segment Routing Traffic Engineering (SR-TE).

To use commands of this module, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using any command, contact your AAA administrator for assistance.

- accounting interfaces segment-routing mpls, on page 3
- affinity-map, on page 4
- autoroute include ipv6 all, on page 5
- bgp prefix-path-label ignore, on page 6
- binding-sid, on page 7
- clear segment-routing srv6 sid, on page 9
- distribute link-state (SRTE), on page 10
- distribute link-state, on page 11
- effective-metric admin-distance, on page 12
- hw-module profile cef sropt enable, on page 14
- kshortest-paths, on page 16
- nexthop validation use-locator, on page 17
- neighbor sr-policy name targeted, on page 18
- on-demand constraints, on page 19
- on-demand dynamic affinity sid-algorithm, on page 20
- on-demand dynamic affinity disjoint-path, on page 21
- on-demand maximum-sid-depth, on page 23
- on-demand source-address, on page 24
- on-demand steering, on page 25
- path-invalidation drop, on page 26
- policy binding-sid, on page 28
- policy candidate-paths, on page 29
- policy candidate-paths constraints disjoint-path, on page 30
- policy candidate-paths constraints resources, on page 32
- policy color, on page 33
- policy source-address, on page 34
- policy steering, on page 35
- resource-list, on page 36
- segment-list, on page 37

- separate-next-hop, on page 38
- show segment-routing traffic-eng database explicit-bsid srv6, on page 39
- steering labeled-services, on page 41
- te-latency, on page 42

accounting interfaces segment-routing mpls

To enable per-application accounting for segment routing MPLS IPv4 or IPv6 traffic and obtain traffic statistics on the interfaces, use the **accounting interfaces segment-routing mpls** command in the accounting interface configuration mode.

accounting interfaces segment-routing mpls { **ipv4** | **ipv6** }

Syntax Description

ipv4	Enables segment routing MPLS IPv4 accounting.
ipv6	Enables segment routing MPLS IPv6 accounting.

Command Default

No default behavior or values

Command Modes

Accounting interface configuration mode

Command History

Usage Guidelines

No specific guidelines impact the use of this command.

Examples

This example shows how to enable per-application accounting to obtain statistics for segment routing MPLS IPv4 traffic on the interfaces.

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# accounting
RP/0/RP0/CPU0:router(config-acct)# interfaces
RP/0/RP0/CPU0:router(config-acct-if)# segment-routing mpls ipv4
```

affinity-map

To define an affinity map, use the **affinity-map name** *name* **bit-position** *bit-position* command in SR-TE sub-mode.

affinity-map **name** *name* **bit-position** *bit-position*

Syntax Description

name *name*	Specify the name of the affinity-map.
bit-position *bit-position*	Specify the bit position in the Extended Admin Group bitmask.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 7.3.1	This command was introduced.

Usage Guidelines

Configure affinity maps on the following routers:

- Routers with interfaces that have an associated admin group attribute.
- Routers that act as SR-TE head-ends for SR policies that include affinity constraints.

Example

This example shows how to define an affinity map:

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# affinity-map
Router(config-sr-te-affinity-map)# name RED bit-position 23
```

autoroute include ipv6 all

To enable IPv6 autoroute support for SR-TE policies with IPv4 endpoints, use the **autoroute include ipv6 all** command in the SR-TE policy and PCC profile modes. To disable this feature, use the **no** form of this command.

autoroute include ipv6 all
no autoroute include ipv6 all

Syntax Description

This command has no keywords or arguments.

Command Default

IPv6 autoroute support is disabled.

Command Modes

SR-TE policy

PCC profile

Command History

Release	Modification
Release 7.5.4	This command was introduced.

Usage Guidelines

The **include ipv6 all** command form enables autoroute support for IPv6 prefixes, for a specified SR-TE policy. This command can be used in the SR-TE policy and PCC profile modes.

Example

The following example shows how to configure the IPv6 autoroute function for an SR-TE policy with an IPv4 endpoint:

```
Router# configure
Router(config)# segment-routing traffic-eng policy pol12
Router(config-sr-te-policy)# autoroute include ipv6 all
Router(config-sr-te-policy)# commit
```

The following example shows how to configure the IPv6 autoroute function for a PCE-instantiated SR-TE policy with an IPv4 endpoint:

```
Router# configure
Router(config)# segment-routing traffic-eng pcc profile 10
Router(config-pcc-prof)# autoroute include ipv6 all
Router(config-pcc-prof)# commit
```

bgp prefix-path-label ignore

To indicate BGP to ignore the programming of the service route's prefix label when recursing onto the BSID of an SR-TE policy, use the **bgp prefix-path-label ignore** command in SR-TE policy steering config mode.

bgp prefix-path-label ignore

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

SR-TE policy steering

Command History

Release	Modification
Release 7.9.1	This command was introduced.

Usage Guidelines

This command can be configured for manual SR policies.

Example

The following example shows how to configure BGP to ignore the programming of the service route's prefix label when recursing onto the BSID of an SR-TE policy:

```
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy POLICY1
Router(config-sr-te-policy)# steering
Router(config-sr-te-policy-steering)# bgp prefix-path-label ignore
```

binding-sid

To specify the binding SID (BSID) allocation behavior, use the **binding-sid** command in SR-TE sub-mode.

binding-sid { **dynamic disable** | **explicit** { **enforce-srlb** | **fallback-dynamic** | **enforce** } }

Syntax Description

dynamic disable	Disables dynamic binding SID allocation. Candidate paths without an explicit BSID will be considered invalid.
explicit enforce-srlb	Specifies strict SRLB enforcement. If the BSID is not within the SRLB, the policy stays down.
explicit fallback-dynamic	Specifies that, if the BSID is not available, the BSID is allocated dynamically and the policy comes up.
explicit enforce	Enforces explicit binding SID allocation.

Command Default

Binding SIDs are dynamically allocated

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 26.1.1	The **enforce** keyword was introduced.
Release 7.3.1	This command was introduced.

Usage Guidelines

Explicit BSIDs are allocated from the segment routing local block (SRLB) or the dynamic range of labels. A best-effort is made to request and obtain the BSID for the SR-TE policy. If requested BSID is not available (if it does not fall within the available SRLB or is already used by another application or SR-TE policy), the policy stays down.

This command specifies how the BSID allocation behaves if the BSID value is not available:

- Fallback to dynamic allocation – If the BSID is not available, the BSID is allocated dynamically and the policy comes up.
- Strict SRLB enforcement – If the BSID is not within the SRLB, the policy stays down.

Example

This example shows how to configure an SR policy to use an explicit BSID of 1000. If the BSID is not available, the BSID is allocated dynamically and the policy comes up.

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# binding-sid explicit fallback-dynamic
Router(config-sr-te)# policy goo
Router(config-sr-te-policy)# binding-sid mpls 1000
```

The following example enforces the SR-TE policy to explicitly use the binding SID allocation:

```
Router(config)# segment-routing
Router(config-sr)# traffic-engineering
Router(config-sr-te)# policy MY_SRV6_POLICY
Router(config-sr-te-policy)# srv6
Router(config-sr-te-policy-srv6)# binding-sid explicit enforce
Router(config-sr-te-policy-srv6)# commit
```

clear segment-routing srv6 sid

To resolve SID ownership conflicts that prevent an explicit BSID from being successfully allocated to an SR-TE policy, use the **clear segment-routing srv6 sid** command.

clear segment-routing srv6 sid *address*

Syntax Description

<*address*>	SRv6 SID in IPv6 address format.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 26.1.1	This command was introduced.

Usage Guidelines

Use this command for explicit (static) SRv6 SIDs. You cannot clear dynamically allocated SIDs with this command.

Task ID

Task ID	Operation
rib	EXECUTE
cisco-support	EXECUTE

This example shows how the **clear segment-routing srv6 sid** <*address*> command manually clears an explicit SRv6 SID in case of route conflicts.

```
Router# clear segment-routing srv6 sid fc20:253f:11:fe05::
Tue Feb 17 05:04:09.407 +00
Router#
```

distribute link-state (SRTE)

To enable reporting of SRTE policies, use the **distribute link-state** command in the SR-TE configuration mode.

distribute link-state [**report-candidate-path-inactive**]

Table 1: Syntax Description:

Syntax	Description
report-candidate-path-inactive	Enables reporting of SRTE policies using BGP-LS.

Command Default

The reporting of policies to BGP-LS is disabled by default.

Command Modes

SR-TE configuration (config-sr-te)

Command History

Release	Modification
Release 24.1.1	Supports reporting of SR-TE policies using BGP- Link State for SRv6.
Release 7.10.1	This command was introduced and supports reporting of SR-TE policies using BGP- Link State for SR-MPLS.

Task ID

Task ID	Operation
distribute link-state	write/read

Example

This example shows how to enable BGP-LS reporting and syncing of SRTE Policies:

```
Router# config
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# distribute link-state
Router(config-sr-te-distribute-ls)# report-candidate-path-inactive
Router(config-sr-te-distribute-ls)# exit
```

distribute link-state

To enable reporting of SRTE policies, use the **distribute link-state** command in the SR-TE configuration mode.

distribute link-state [**report-candidate-path-inactive**]

Table 2: Syntax Description:

Syntax	Description
report-candidate-path-inactive	Enables reporting of SRTE policies using BGP-LS.

Command Default

The reporting of policies to BGP-LS is disabled by default.

Command Modes

SR-TE configuration (config-sr-te)

Command History

Release	Modification
Release 24.1.1	Supports reporting of SR-TE policies using BGP- Link State for SRv6.
Release 7.10.1	This command was introduced and supports reporting of SR-TE policies using BGP- Link State for SR-MPLS.

Task ID

Task ID	Operation
distribute link-state	write/read

Example

This example shows how to enable BGP-LS reporting and syncing of SRTE Policies:

```
Router# config
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# distribute link-state
Router(config-sr-te-distribute-ls)# report-candidate-path-inactive
Router(config-sr-te-distribute-ls)# exit
```

effective-metric admin-distance

To configure administrative distance for different metric types, use the **effective-metric admin-distance** command in the SR-TE configuration mode.

effective-metric admin-distance [**metric-type** { **igp** | **te** | **latency** | **hopcount** | **unknown** } | **flex-algo-metric-type** { *4-127* | **bandwidth** | **generic** | **igp** | **latency** | **te** }] **admin-distance** *distance*

Syntax Description

`metric-type { igp | te | latency | hopcount | unknown }`	Specify the metric type advertised to other protocols. • igp: IGP metric type • te: TE metric type • latency: LATENCY metric type • hopcount: HOPCOUNT metric type • unknown: Unknown metric type
`flex-algo-metric-type { 4-127 | bandwidth | generic | igp | latency | te }`	Specify the flex-algo metric type advertised to other protocols. • <4-127>: IANA defined metric types • bandwidth: BANDWIDTH metric type • generic: USER-DEFINED metric types • igp: IGP metric type • latency: LATENCY metric type • te: TE metric type
admin-distance *distance*	Administrative distance (1-255) advertised for the specified metric type.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 7.5.2	This command was introduced.
Release 24.4.1	The keyword **flex-algo-metric-type** was introduced.

Usage Guidelines

None.

Task ID

Task ID	Operation
config-services	read, write

Example

In this example, the administrative distance for SR-TE generic user-defined metric is changed to 120 using the **effective-metric admin-distance** command.

```
Router(config)#segment-routing
Router(config-sr)#traffic-eng
Router(config-sr-te)#effective-metric admin-distance flex-algo-metric-type generic 130
admin-distance 120
Router(config-sr-te)#commit
```

hw-module profile cef sropt enable

To enable Segment Routing Encap object optimization, use the **hw-module profile cef sropt enable** command in XR Config mode.

hw-module profile cef sropt enable

Syntax Description

This command has no keywords or arguments.

Command Default

Segment Routing Encap object optimization is disabled.

Command Modes

XR Configuration

Command History

Release	Modification
Release 7.5.4	This command was introduced.

Usage Guidelines

It does not support any overlay protocols, which includes BGP LU, L3VPN, and SRTE. Furthermore, RSVP-TE is not supported in the underlay, which may constrain certain deployment scenarios where TE tunnels are expected.

After you enter this command, you must reload the router.

Segment Routing Encap object optimization minimizes the Encap resource consumption of the forwarding ASIC. With this feature, instead of consuming an Encap entry for each outgoing path, the forwarding chain of a labeled prefix with ECMP consumes only a single global Encap entry.

SR Encap object optimization is triggered only when all ECMP paths of a labeled prefix (primary and backup) perform the same egress action (either all pop or all swap); and have the same outgoing label for the swap egress action. If this condition is not met, then the prefix is programmed with a dedicated Encap object per outgoing path.

SR Encap object optimization is supported for both labeled IPv4 /32 (SR-MPLSv4) and labeled IPv6 /128 (SR-MPLSv6).

All paths associated with the prefix (primary and backup) must have the same outgoing label value for SR Encap object optimization to be triggered. For example:

- For prefixes with LFA backup paths, the SR Encap object optimization is triggered because these backup paths do not require an extra label to be pushed.
- For prefixes with TI-LFA backup paths requiring extra labels to be pushed, the SR Encap object optimization is not triggered because all the paths associated with the prefix do not have the same outgoing label value.

Per-label per-interface egress counters are not supported when SR Encap object optimization is enabled. Instead, per-label aggregate egress counters are supported.

SR MicroLoop Avoidance is not supported when SR Encap object optimization is enabled.

Example

This example shows how to enable Segment Routing Encap object optimization:

```
Router(config)# hw-module profile cef sropt enable

In order to activate/deactivate SROPT feature, you must manually reload the chassis/all
line cards

Router(config)# commit
Router(config)# end

Router# reload location all
Proceed with reload? [confirm] y

Router# show hw-module profile cef

-----------------------------------------------------------------
Knob                          Status          Applied   Action
-----------------------------------------------------------------
CBF Enable                    Unconfigured    N/A       None
CBF forward-class-list        Unconfigured    N/A       None
BGPLU                         Unconfigured    N/A       None
LPTS ACL                      Unconfigured    N/A       None
Dark Bandwidth                Unconfigured    N/A       None
SR-OPT Enable                 Configured      Yes       None
IP Redirect Punt              Unconfigured    N/A       None
IPv6 Hop-limit Punt           Unconfigured    N/A       None
MPLS Per Path Stats           Unconfigured    N/A       None
Tunnel TTL Decrement          Unconfigured    N/A       None
High-Scale No-LDP-Over-TE     Unconfigured    N/A       None
Label over TE counters        Unconfigured    N/A       None
Highscale LDPoTE No SRoTE     Unconfigured    N/A       None
LPTS Pifib Entry Counters     Unconfigured    N/A       None
```

kshortest-paths

To set the maximum number of attempts for SRTE to compute paths that satisfy cumulative metric bounds criteria, use the **kshortest-paths** command in SR-TE configuration mode. To revert to the default number of attempts (100), use the **no** form of the command.

kshortest-paths *max-attempts*

no kshortest-paths

Syntax Description

max-attempts	Maximum number of attempts. Choose a value between 1 and 200.

Command Default

100 attempts are made to compute paths that satisfy the cumulative metric bounds criteria.

Command Modes

SR-TE configuration (config-sr-te)

Command History

Release	Modification
Release 7.3.1	This command was introduced.

Usage Guidelines

By default, a maximum of 100 attempts are made. To update the value, you can use this command.

You can use the **show segment-routing traffic-eng policy color** command (**Number of K-shortest-paths** field) to see the K-shortest path algorithm computation result. For example, if the **Number of K-shortest-paths** field displays 4, it means that the K-shortest path algorithm took 4 computations to find the right path. The 4 shortest paths that are computed using K-shortest path algorithm did not respect the cumulative bounds, and the fifth shortest path was valid against the bounds.

Example

This example shows how to set the maximum number of attempts for computing paths that satisfy the cumulative metric bounds criteria:

```
Router# configure terminal
Router(config)# segment-routing traffic-eng
Router(config-sr-te)# kshortest-paths 120
Router(config-sr-te)# commit
```

nexthop validation use-locator

To enable next-hop validation to use the locator for SRv6, use the **nexthop validation use-locator** command in BGP configuration mode.

nexthop validation use-locator

Syntax Description

This command has no keywords or arguments.

Command Default

BGP NH validation is disabled.

Command Modes

BGP configuration

Command History

Release	Modification
Release 26.1.1	This command was introduced.

Usage Guidelines

- You must manually configure BGP SID locator tracking because it is disabled by default.

  If SID structure sub-TLVs are missing, BGP cannot extract locator information and will use traditional next-hop tracking (NHT) behavior.

- You must enable this feature on all devices in the network to avoid the risk of a routing loop.

Example

The following example shows how to enable next-hop validation to use the locator for SRv6:

```
Router(config)#router bgp 200
Router(config-bgp)#nexthop validation use-locator
```

neighbor sr-policy name targeted

To configure the SR policy name under LDP, use the **neighbor sr-policy***name* **targeted** command in SR-TE configuration mode.

neighbor sr-policy *name* **targeted**

Table 3: Syntax Description

Syntax	Description
name	Use the command to configure the SR policy name under LDP *name* is the auto-generated SR policy name assigned by the router when creating an LDP targeted adjacency over an SR policy. **Note** You can use the **show segment-routing traffic-eng policy** command to display the auto generated SR policy name. Auto-generated SR policy name uses the following naming convention: **srte_c_***color_val***_ep_***endpoint-address*. For example, srte_c_1000_ep_10.1.1.2.

Command Default

None

Command Modes

SR-TE configuration mode

Command History

Release	Modification
Release 7.10.1	This command was introduced.

Example

The following example shows how to configure the SR policy name under LDP:

```
Router(config)# mpls ldp
Router(config-ldp)# address-family ipv4
Router(config-ldp-af)# neighbor sr-policy srte_c_1000_ep_10.1.1.2 targeted
Router(config-ldp-af)#commit
```

on-demand constraints

Note From Cisco IOS XR Release 7.9.1, you must reconfigure all SR-ODN configurations with Flexible Algorithm constraints that use the on-demand dynamic sid-algorithm with this command.

To configure the SR Flexible Algorithm constraints, use the **constraints segments sid-algorithm** command in SR-TE sub-mode.

on-demand color *color* **constraints** { **segments sid-algorithm** *algo* | **resources** { **exclude resource-list** *name* | **exclude-group** *group_name* | **apply-group** *group_name* } }

Syntax Description

segments	Specify constraints for segments of a path in a network.
sid-algorithm *algo*	Specify the SR Flexible Algorithm value. The *algo* range is from 128 to 255.
resources	Specify resource constraints for path computation.
exclude	Exclude resources from path computation.
resource-list *name*	Specify the name of the resource-list to exclude from the path computation.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 24.1.1	The resources option was introduced.
Release 7.9.1	You must reconfigure all SR-ODN configurations with Flexible Algorithm constraints that use the on-demand dynamic sid-algorithm with this command.
Release 7.4.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Example

The following example shows how to add an SR Flexible Algorithm constraint:

```
Router(config-sr-te-color)# constraints segments sid-algorithm 128
```

The following example shows how to associate the excluded IPv4 addresses for ODN SR-TE policies:

```
Router(config)#segment-routing
Router(config-sr)#traffic-eng
Router(config-sr-te)#on-demand color 7001
Routerconfig-sr-te-color)#constraints resources exclude resource-list node_resc_list
```

on-demand dynamic affinity sid-algorithm

Note You must reconfigure all SR-ODN configurations with Flexible Algorithm constraints that use this command with the **constraints segments sid-algorithm** *algo* command.

To configure the SR Flexible Algorithm constraints, use the **on-demand dynamic sid-algorithm** command in SR-TE sub-mode.

on-demand color *color* **dynamic sid-algorithm** *algo*

Syntax Description

sid-algorithm `algo`	Specify the SR Flexible Algorithm value . The *algo* range is from 128 to 255.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 6.3.1	This command was introduced.
Release 7.4.1	This command was replaced by the **constraints segments sid-algorithm** *algo* command.
Release 7.9.1	You must reconfigure all SR-ODN configurations with Flexible Algorithm constraints that use this command with the **constraints segments sid-algorithm** *algo* command. .

Usage Guidelines

This command was replaced by the **constraints segments sid-algorithm** *algo* command.

Example

```
Router(config-sr-te-color-dyn)# sid-algorithm 128
```

on-demand dynamic affinity disjoint-path

To configure the disjoint-path constraints, use the **on-demand dynamic disjoint-path** command in SR-TE sub-mode.

on-demand color *color* **dynamic disjoint-path group-id** *id* **type** { **link** | **node** | **srlg** | **srlg-node** } [{ **sub-id** *sub_id* | **fallback disable** }]

Syntax Description

group-id *id*	Specify the group ID of the disjoint path. Valid values are from 1 to 65535.
type {**link** | **node** | **srlg** | **srlg-node** }	Specify the type of disjointness.
sub-id *id*	Specify the sub-group ID of the disjoint path. Valid values are from 1 to 65535.
fallback disable	Disable all fallback behavior in case the requested disjointness cannot be achieved.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 24.1.1	The **fallback disable** keyword was introduced.
Release 6.3.1	This command was introduced.

Usage Guidelines

Configures the disjoint group ID and defines the preferred level of disjointness (the type of resources that should not be shared by the two paths):

- link—Specifies that links are not shared on the computed paths.
- node—Specifies that nodes are not shared on the computed paths.
- srlg—Specifies that links with the same SRLG value are not shared on the computed paths
- srlg-node—Specifies that SRLG and nodes are not shared on the computed paths.

If a pair of paths that meet the requested disjointness level cannot be found, then the paths will automatically fallback to a lower level:

- If the requested disjointness level is SRLG or node, then link-disjoint paths will be computed.
- If the requested disjointness level was link, or if the first fallback from SRLG or node disjointness failed, then the lists of segments encoding two shortest paths, without any disjointness constraint, will be computed.

Example

```
Router(config-sr-te-color-dyn)# disjoint-path group-id 775 type link
```

The following example indicates how to configure strict disjointness for an ODN SR-TE policy:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#on-demand color 4
Router(config-sr-te-color)#dynamic
Router(config-sr-te-color-dyn)#disjoint-path group-id 1 type node fallback disable
Router(config-sr-te-color-dyn)#commit
```

on-demand maximum-sid-depth

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

on-demand source-address

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

on-demand steering

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

path-invalidation drop

To enable the dropping of traffic when an SR Policy becomes invalid, use the **path-invalidation drop** command.

policy *policy* **steering path-invalidation drop**

on-demand color *color* **steering path-invalidation drop**

pcc profile *profile* **steering path-invalidation drop**

Syntax Description

This command has no keywords or arguments.

Command Default

Disabled

Command Modes

SR-TE Policy

SR-TE ODN

SR-TE PCC

Command History

Release	Modification
Release 7.11.1	This command was introduced.

Usage Guidelines

By default, if an SR policy becomes invalid, traffic would fall back to the native SR forwarding path. In some scenarios, a network operator may require that certain traffic be only carried over the path associated with an SR policy and never allow the native SR LSP to be used. This command is introduced to meet this requirement.

With the **path-invalidation drop** command is enabled, an SR policy that would become invalid (for example, no valid candidate path available) is programmed to drop traffic. At the same time, the SR policy remains operationally UP to prevent prefixes steered over this SR policy from falling back to the native SR path.

Forwarding over the SR policy path resumes without droppic traffic, when the SR policy path becomes valid.

Example

The following example shows how to enable the dropping of traffic when an On-Demand SR Policy becomes invalid.

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# on-demand color 10
Router(config-sr-te-color)# steering
Router(config-sr-te-on-demand-color-steering)# path-invalidation drop
```

The following example shows how to enable the dropping of traffic when an SR Policy becomes invalid.

```
Router# configure
Router(config)# segment-routing
```

```
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy FOO
Router(config-sr-te-policy)# steering
Router(config-sr-te-policy-steering)# path-invalidation drop
```

The following example shows how to enable the dropping of traffic when a PCE-initiated SR Policy becomes invalid.

```
Router# configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# pcc profile 7
Router(config-pcc-prof)# steering
Router(config-pcc-prof-steering)# path-invalidation drop
```

policy binding-sid

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

policy candidate-paths

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

policy candidate-paths constraints disjoint-path

To configure the disjoint-path constraints, use the **on-demand dynamic disjoint-path** command in SR-TE sub-mode.

policy *policy* **candidate-paths preference** *preference* **constraints disjoint-path group-id** *id* **type** { **link** | **node** | **srlg** | **srlg-node** } [{ **sub-id** *sub_id* | **shortest-path** | **fallback disable** }]

Syntax Description

group-id *id*	Specify the group ID of the disjoint path. Valid values are from 1 to 65535.
type {**link** | **node** | **srlg** | **srlg-node** }	Specify the type of disjointness.
sub-id *id*	Specify the sub-group ID of the disjoint path. Valid values are from 1 to 65535.
shortest-path	Enable shortest path computation for the selected candidate path.
fallback disable	Disable all fallback behavior in case the requested disjointness cannot be achieved.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 24.1.1	The **shortest-path** and **fallback disable** keywords were introduced.
Release 6.3.1	This command was introduced.

Usage Guidelines

Configures the disjoint group ID and defines the preferred level of disjointness (the type of resources that should not be shared by the two paths):

- link—Specifies that links are not shared on the computed paths.
- node—Specifies that nodes are not shared on the computed paths.
- srlg—Specifies that links with the same SRLG value are not shared on the computed paths
- srlg-node—Specifies that SRLG and nodes are not shared on the computed paths.

If a pair of paths that meet the requested disjointness level cannot be found, then the paths will automatically fallback to a lower level:

- If the requested disjointness level is SRLG or node, then link-disjoint paths will be computed.
- If the requested disjointness level was link, or if the first fallback from SRLG or node disjointness failed, then the lists of segments encoding two shortest paths, without any disjointness constraint, will be computed.

Example

```
Router(config-sr-te)# policy FOO
Router(config-sr-te-policy)# candidate-paths preference 100
Router(config-sr-te-poliilojkl,.cy-path-pref)# constraints disjoint-path group-id 775 type
 link
```

The following example indicates how to configure the shortest path preference for a disjoint path:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#policy dynamic_pcep_policy_disjoint
Router(config-sr-te-policy)#candidate-paths
Router(config-sr-te-policy-path)#preference 100
Router(config-sr-te-policy-path-pref)#constraints disjoint-path group-id 1 type link
shortest-path
```

The following example indicates how to configure strict disjointness for a SR-TE policy:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#policy foo
Router(config-sr-te-policy)#color 1 end-point ipv4 10.10.10.1
Router(config-sr-te-policy)#candidate-paths preference 100
Router(config-sr-te-policy-path-pref)#constraints disjoint-path group-id 1 type node fallback
 disable
Router(config-sr-te-policy-path-pref)#commit
```

policy candidate-paths constraints resources

To exclude IP addresses from the path computation for SR-TE policies, use the **policy candidate-paths constraints resources** command in the SR-TE configuration mode.

policy *policy* **candidate-paths preference** *preference* **constraints resources** { **exclude resource-list** *name* | **exclude-group** *group_name* | **apply-group** *group_name* }

Syntax Description

resources {**exclude-group** | **exclude** | **apply-group**}	Specify the resource constraints for path computation: • exclude. Excludes resources from the path computation. • exclude-group. Excludes the apply-group configuration from the group. • apply-group. Applies configuration from a group.
resource-list *name*	Specify the name of the resource-list to exclude from the path computation.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 24.1.1	This command was introduced.

Usage Guidelines

None.

Example

The following example shows how to exclude a list of IPv4 addresses from the network resource list:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#resource-list node_resc_list
Router(config-sr-te-rl)#index 1 ipv4 10.10.10.1
Router(config-sr-te-rl)#index 2 ipv4 10.10.10.8
```

The following example shows how to associate the excluded IPv4 addresses to one or more candidate paths for SR-TE policies:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#policy dynamic_pcep_policy
Router(config-sr-te-policy)#candidate-paths
Router(config-sr-te-policy-path)#preference 100
Router(config-sr-te-policy-path-pref)#constraints resources exclude resource-list
node_resc_list
```

policy color

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

policy source-address

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

policy steering

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

resource-list

To configure a list of IPv4 addresses that you want to exclude from the network resource list for a candidate path, use the **resource-list** command in SR-TE configuration mode.

resource-list *name* **index** *"1-65535"* **ipv4** *ipv4-addr*

Syntax Description

resource-list *name*	Specify the resource-list name to exclude from the path computation.
index *1-65535*	Specify the index entry. Ranges from 1–65535.
ipv4 *ipv4-addr*	Specify the IPv4 address that you want to exclude from the network resource list.

Command Default

None

Command Modes

SR-TE configuration mode

Command History

Release	Modification
Release 24.1.1	This command was introduced.

Usage Guidelines

None.

Example

The following example shows how to configure a list of IPv4 addresses that you want to exclude from the network resource list:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#resource-list node_resc_list
Router(config-sr-te-rl)#index 1 ipv4 10.10.10.1
Router(config-sr-te-rl)#index 2 ipv4 10.10.10.8
```

segment-list

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

separate-next-hop

To enable SR-TE with next-hop independent scaling optimization, use the **separate-next-hop** command in ST-TE configuration mode.

segment-routing traffic-eng separate-next-hop

This command has no keywords or arguments.

Command Default

None

Command Modes

SR-TE configuration

Command History

Release	Modification
Release 7.3.1	This command was introduced.

Usage Guidelines

Example

show segment-routing traffic-eng database explicit-bsid srv6

To verify how the SR-TE process perceives and manages explicit SIDs from the SID Manager, use the **show segment-routing traffic-eng database explicit-bsid srv6** command.

show segment-routing traffic-eng database explicit-bsid srv6 [X:X::X | detail | private]

Syntax Description

detail	Displays operational status and programming details for explicit BSIDs.
private	Displays detailed explicit BSID information to assist with troubleshooting.
X:X::X	Displays information for a specific explicit BSID in IPv6 format.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 26.1.1	This command was introduced.

Usage Guidelines

None.

Task ID

Task ID	Operation
config-services	READ
cisco-support	READ

These examples show how to verify the summary or detailed information about explicit BSIDs.

```
Router# show segment-routing traffic-eng database explicit-bsid srv6 fc20:253f:11:fe05::
Tue Feb 17 05:02:24.088 +00

               Binding SID                     Behavior      Policy (Color,Endpoint)
-------------------------- ------------------------- ----------------------------
      fc20:253f:11:fe05::    uB6 (PSP/USD Insert.Red)                    50,22::22

Router# show segment-routing traffic-eng database explicit-bsid srv6 private
Tue Feb 17 05:02:33.031 +00

SRv6-TE Redistributed BSID database:
------------------------------------

Binding SID: fc20:253f:11:fe05::/64 (Last update: Feb 17 05:00:57.924)
  Behavior: uB6 (PSP/USD Insert.Red) (99)
  Policy identifier:
    Color: 50 Endpoint: 22::22
  Locator ID: 1
  Conflict: No
  Stale: No
  Configured: Yes
  Event history (oldest first):
```

```
    Time                    Event
    Feb 17 05:00:57.924     Added (SID Mgr configured: true)

Router# show segment-routing traffic-eng database explicit-bsid srv6 detail
Tue Feb 17 05:02:29.593 +00

SRv6-TE Redistributed BSID database:
------------------------------------

Binding SID: fc20:253f:11:fe05::/64 (Last update: Feb 17 05:00:57.924)
  Behavior: uB6 (PSP/USD Insert.Red) (99)
  Policy identifier:
    Color: 50 Endpoint: 22::22
  Locator ID: 1
  Conflict: No
```

steering labeled-services

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example

te-latency

Syntax Description

Command Default

Command Modes

Command History

Release	Modification

Usage Guidelines

Task ID

Task ID	Operation

Example