Configure Segment Routing for IS-IS Protocol

Integrated Intermediate System-to-Intermediate System (IS-IS), Internet Protocol Version 4 (IPv4), is a standards-based Interior Gateway Protocol (IGP). The Cisco IOS XR software implements the IP routing capabilities described in International Organization for Standardization (ISO)/International Engineering Consortium (IEC) 10589 and RFC 1995, and adds the standard extensions for single topology and multitopology IS-IS for IP Version 6 (IPv6).

This module provides the configuration information used to enable segment routing for IS-IS.

Note For additional information on implementing IS-IS on your Cisco 8000 Series Router, see the *Implementing IS-IS* module in the *Routing Configuration Guide for Cisco 8000 Series Routers*.

- Enabling Segment Routing for IS-IS Protocol, on page 1
- Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface, on page 4
- Configuring an Adjacency SID, on page 6
- IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability, on page 12
- IS-IS Unreachable Prefix Announcement, on page 16
- Conditional Prefix Advertisement, on page 18

Enabling Segment Routing for IS-IS Protocol

Segment routing on the IS-IS control plane supports the following:

- IPv4 and IPv6 control plane
- Level 1, level 2, and multi-level routing
- Prefix SIDs for host prefixes on loopback interfaces
- Adjacency SIDs for adjacencies
- MPLS penultimate hop popping (PHP) and explicit-null signaling

This task explains how to enable segment routing for IS-IS.

Before you begin

Your network must support the MPLS Cisco IOS XR software feature before you enable segment routing for IS-IS on your router.

Note You must enter the commands in the following task list on every IS-IS router in the traffic-engineered portion of your network.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **address-family** { **ipv4** | **ipv6** } [**unicast**]
4. **metric-style wide** [**level** { **1** | **2** }]
5. **router-id loopback** *loopback interface used for prefix-sid*
6. **segment-routing mpls** [**sr-prefer**]
7. **exit**
8. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters mode.
Step 2	**router isis** *instance-id* **Example:** `RP/0/RP0/CPU0:router(config)# router isis isp`	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode. **Note** You can change the level of routing to be performed by a particular routing instance by using the **is-type** router configuration command.
Step 3	**address-family** { **ipv4**	**ipv6** } [**unicast**] **Example:** `RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
Step 4	**metric-style wide** [**level** { **1**	**2** }] **Example:**	Configures a router to generate and accept only wide link metrics in the Level 1 area.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-isis-af)# metric-style wide level 1`	
Step 5	**router-id loopback** *loopback interface used for prefix-sid* **Example:** `RP/0/(config-isis-af)#router-id loopback0`	Configures router ID for each address-family (ipv4/ipv6).
Step 6	**segment-routing mpls** [**sr-prefer**] **Example:** `RP/0/RP0/CPU0:router(config-isis-af)# segment-routing mpls`	Segment routing is enabled by the following actions: • MPLS forwarding is enabled on all interfaces where IS-IS is active. • All known prefix-SIDs in the forwarding plain are programmed, with the prefix-SIDs advertised by remote routers or learned through local or remote mapping server. • The prefix-SIDs locally configured are advertised. Use the **sr-prefer** keyword to set the preference of segment routing (SR) labels over label distribution protocol (LDP) labels.

Step 7	**exit**
	Example:
	`RP/0/RP0/CPU0:router(config-isis-af)# exit`
	`RP/0/RP0/CPU0:router(config-isis)# exit`	

Step 8	Use the **commit** or **end** command.	**commit** —Saves the configuration changes and remains within the configuration session. **end** —Prompts user to take one of these actions: • **Yes** — Saves configuration changes and exits the configuration session. • **No** —Exits the configuration session without committing the configuration changes. • **Cancel** —Remains in the configuration session, without committing the configuration changes.

What to do next

Configure the prefix SID.

Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface

A prefix segment identifier (SID) is associated with an IP prefix. The prefix SID is manually configured from the segment routing global block (SRGB) range of labels. A prefix SID is configured under the loopback interface with the loopback address of the node as the prefix. The prefix segment steers the traffic along the shortest path to its destination.

A prefix SID can be a node SID or an Anycast SID. A node SID is a type of prefix SID that identifies a specific node. An Anycast SID is a type of prefix SID that identifies a set of nodes, and is configured with n-flag clear. The set of nodes (Anycast group) is configured to advertise a shared prefix address and prefix SID. Anycast routing enables the steering of traffic toward multiple advertising nodes. Packets addressed to an Anycast address are forwarded to the topologically nearest nodes.

Strict-SPF SIDs are used to forward traffic strictly along the SPF path. IS-IS advertises the SR Algorithm sub Type Length Value (TLV) (in the SR Router Capability SubTLV) to include both algorithm 0 (SPF) and algorithm 1 (Strict-SPF). Strict-SPF SIDs are also used to program the backup paths for prefixes, node SIDs, and adjacency SIDs.

The prefix SID is globally unique within the segment routing domain.

This task explains how to configure prefix segment identifier (SID) index or absolute value on the IS-IS enabled Loopback interface.

Before you begin

Ensure that segment routing is enabled on the corresponding address family.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **interface Loopback** *instance*
4. **address-family** { **ipv4** | **ipv6** } [**unicast**]
5. **prefix-sid** [**algorithm** *algorithm-number*] {**index** *SID-index* | **absolute** *SID-value*} [**n-flag-clear**] [**explicit-null**]
6. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters XR Config mode.

	Command or Action	Purpose
Step 2	**router isis** *instance-id* **Example:** `RP/0/RP0/CPU0:router(config)# router isis 1`	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode. • You can change the level of routing to be performed by a particular routing instance by using the **is-type** router configuration command.
Step 3	**interface Loopback** *instance* **Example:** `RP/0/RP0/CPU0:router(config-isis)# interface Loopback0`	Specifies the loopback interface and instance.
Step 4	**address-family** { **ipv4** | **ipv6** } [**unicast**] **Example:** The following is an example for ipv4 address family: `RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.

Step 5	**prefix-sid** [**algorithm** *algorithm-number*] {**index** *SID-index* | **absolute** *SID-value*} [**n-flag-clear**] [**explicit-null**]
	Example:
	`RP/0/RP0/CPU0:router(config-isis-if-af)# prefix-sid index 1001`
	`RP/0/RP0/CPU0:router(config-isis-if-af)# prefix-sid absolute 17001`
		Configures the prefix-SID index or absolute value for the interface.
		Specify **algorithm** *algorithm-number* to configure SR Flexible Algorithm. See Enabling Segment Routing Flexible Algorithm.
		Specify **index** *SID-index* for each node to create a prefix SID based on the lower boundary of the SRGB + the index.
		Specify **absolute** *SID-value* for each node to create a specific prefix SID within the SRGB.
		By default, the n-flag is set on the prefix-SID, indicating that it is a node SID. For specific prefix-SID (for example, Anycast prefix-SID), enter the **n-flag-clear** keyword. IS-IS does not set the N flag in the prefix-SID sub Type Length Value (TLV).
		To disable penultimate-hop-popping (PHP) and add explicit-Null label, enter **explicit-null** keyword. IS-IS sets the E flag in the prefix-SID sub TLV. Any upstream neighbor of the Prefix-SID originator replaces the Prefix-SID with a Prefix-SID having an Explicit NULL value.

Step 6	Use the **commit** or **end** command.	**commit** —Saves the configuration changes and remains within the configuration session. **end** —Prompts user to take one of these actions: • **Yes** — Saves configuration changes and exits the configuration session.

	Command or Action	Purpose
		• **No** —Exits the configuration session without committing the configuration changes. • **Cancel** —Remains in the configuration session, without committing the configuration changes.

Verify the prefix-SID configuration:

```
RP/0/RP0/CPU0:router# show isis database verbose

IS-IS 1 (Level-2) Link State Database
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
router.00-00        * 0x0000039b   0xfc27        1079            0/0/0
  Area Address: 49.0001
  NLPID:        0xcc
  NLPID:        0x8e
  MT:           Standard (IPv4 Unicast)
  MT:           IPv6 Unicast                                     0/0/0
  Hostname:     router
  IP Address:   10.0.0.1
  IPv6 Address: 2001:0db8:1234::0a00:0001
  Router Cap:   10.0.0.1, D:0, S:0
    Segment Routing: I:1 V:1, SRGB Base: 16000 Range: 8000
    SR Algorithm:
      Algorithm: 0
      Algorithm: 1
<...>
  Metric: 0          IP-Extended 10.0.0.1/32
    Prefix-SID Index: 1001, Algorithm:0, R:0 N:1 P:0 E:0 V:0 L:0
    Prefix-SID Index: 101, Algorithm:1, R:0 N:1 P:0 E:0 V:0 L:0
<...>
```

Configuring an Adjacency SID

An adjacency SID (Adj-SID) is associated with an adjacency to a neighboring node. The adjacency SID steers the traffic to a specific adjacency. Adjacency SIDs have local significance and are only valid on the node that allocates them.

An adjacency SID can be allocated dynamically from the dynamic label range or configured manually from the segment routing local block (SRLB) range of labels.

Adjacency SIDs that are dynamically allocated do not require any special configuration, however there are some limitations:

- A dynamically allocated Adj-SID value is not known until it has been allocated, and a controller will not know the Adj-SID value until the information is flooded by the IGP.
- Dynamically allocated Adj-SIDs are not persistent and can be reallocated after a reload or a process restart.
- Each link is allocated a unique Adj-SID, so the same Adj-SID cannot be shared by multiple links.

Manually allocated Adj-SIDs are persistent over reloads and restarts. They can be provisioned for multiple adjacencies to the same neighbor or to different neighbors. You can specify that the Adj-SID is protected. If the Adj-SID is protected on the primary interface and a backup path is available, a backup path is installed. By default, manual Adj-SIDs are not protected.

Adjacency SIDs are advertised using the existing IS-IS Adj-SID sub-TLV. The S and P flags are defined for manually allocated Adj-SIDs.

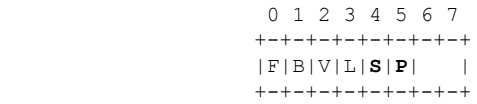

Table 1: Adjacency Segment Identifier (Adj-SID) Flags Sub-TLV Fields

Field	Description
S (Set)	This flag is set if the same Adj-SID value has been provisioned on multiple interfaces.
P (Persistent)	This flag is set if the Adj-SID is persistent (manually allocated).

Manually allocated Adj-SIDs are supported on point-to-point (P2P) interfaces.

This task explains how to configure an Adj-SID on an interface.

Before you begin

Ensure that segment routing is enabled on the corresponding address family.

Use the **show mpls label table detail** command to verify the SRLB range.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **interface** *type interface-path-id*
4. **point-to-point**
5. **address-family** { **ipv4** | **ipv6** } [**unicast**]
6. **adjacency-sid** {**index** *adj-SID-index* | **absolute** *adj-SID-value* } [**protected**]
7. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters mode.

	Command or Action	Purpose
Step 2	**router isis** *instance-id* **Example:** `RP/0/RP0/CPU0:router(config)# router isis 1`	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode. • You can change the level of routing to be performed by a particular routing instance by using the **is-type** router configuration command.
Step 3	**interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet0/0/0/7`	Specifies the interface and enters interface configuration mode.
Step 4	**point-to-point** **Example:** `RP/0/RP0/CPU0:router(config-isis-if)# point-to-point`	Specifies the interface is a point-to-point interface.
Step 5	**address-family** { **ipv4** | **ipv6** } [**unicast**] **Example:** The following is an example for ipv4 address family: `RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.

Step 6	**adjacency-sid** {**index** *adj-SID-index* | **absolute** *adj-SID-value* } [**protected**]
	Example:
	`RP/0/RP0/CPU0:router(config-isis-if-af)# adjacency-sid index 10`
	`RP/0/RP0/CPU0:router(config-isis-if-af)# adjacency-sid absolute 15010`
		Configures the Adj-SID index or absolute value for the interface.
		Specify **index** *adj-SID-index* for each link to create an Ajd-SID based on the lower boundary of the SRLB + the index.
		Specify **absolute** *adj-SID-value* for each link to create a specific Ajd-SID within the SRLB.
		Specify if the Adj-SID is **protected**. For each primary path, if the Adj-SID is protected on the primary interface and a backup path is available, a backup path is installed. By default, manual Adj-SIDs are not protected.

Step 7	Use the **commit** or **end** command.	**commit** —Saves the configuration changes and remains within the configuration session. **end** —Prompts user to take one of these actions: • **Yes** — Saves configuration changes and exits the configuration session. • **No** —Exits the configuration session without committing the configuration changes.

	Command or Action	Purpose
		• **Cancel** —Remains in the configuration session, without committing the configuration changes.

Verify the Adj-SID configuration:

```
RP/0/RP0/CPU0:router# show isis segment-routing label adjacency persistent
Mon Jun 12 02:44:07.085 PDT

IS-IS 1 Manual Adjacency SID Table

15010 AF IPv4
      GigabitEthernet0/0/0/3: IPv4, Protected 1/65/N, Active
      GigabitEthernet0/0/0/7: IPv4, Protected 2/66/N, Active

15100 AF IPv6
      GigabitEthernet0/0/0/3: IPv6, Not protected 255/255/N, Active
```

Verify the labels are added to the MPLS Forwarding Information Base (LFIB):

```
RP/0/RP0/CPU0:router# show mpls forwarding labels 15010
Mon Jun 12 02:50:12.172 PDT
Local  Outgoing    Prefix             Outgoing     Next Hop        Bytes
Label  Label       or ID              Interface                    Switched
------ ----------- ------------------ ------------ --------------- ------------
15010  Pop         SRLB (idx 10)      Gi0/0/0/3    10.0.3.3        0
       Pop         SRLB (idx 10)      Gi0/0/0/7    10.1.0.5        0
       16004       SRLB (idx 10)      Gi0/0/0/7    10.1.0.5        0           (!)
       16004       SRLB (idx 10)      Gi0/0/0/3    10.0.3.3        0           (!)
```

Manually Configure a Layer 2 Adjacency SID

Typically, an adjacency SID (Adj-SID) is associated with a Layer 3 adjacency to a neighboring node, to steer the traffic to a specific adjacency. If you have Layer 3 bundle interfaces, where multiple physical interfaces form a bundle interface, the individual Layer 2 bundle members are not visible to IGP; only the bundle interface is visible.

You can configure a Layer 2 Adj-SID for the individual Layer 2 bundle interfaces. This configuration allows you to track the availability of individual bundle member links and to verify the segment routing forwarding over the individual bundle member links, for Operational Administration and Maintenance (OAM) purposes.

A Layer 2 Adj-SID can be allocated dynamically or configured manually.

- IGP dynamically allocates Layer 2 Adj-SIDs from the dynamic label range for each Layer 2 bundle member. A dynamic Layer 2 Adj-SID is not persistent and can be reallocated as the Layer 3 bundle link goes up and down.
- Manually configured Layer 2 Adj-SIDs are persistent if the Layer 3 bundle link goes up and down. Layer 2 Adj-SIDs are allocated from the Segment Routing Local Block (SRLB) range of labels. However, if the configured value of Layer 2 Adj-SID does not fall within the available SRLB, a Layer 2 Adj-SID will not be programmed into forwarding information base (FIB).

Restrictions

- Adj-SID forwarding requires a next-hop, which can be either an IPv4 address or an IPv6 address, but not both. Therefore, manually configured Layer 2 Adj-SIDs are configured per address-family.
- Manually configured Layer 2 Adj-SID can be associated with only one Layer 2 bundle member link.
- A SID value used for Layer 2 Adj-SID cannot be shared with Layer 3 Adj-SID.
- SR-TE using Layer 2 Adj-SID is not supported.

This task explains how to configure a Layer 2 Adj-SID on an interface.

Before you begin

Ensure that segment routing is enabled on the corresponding address family.

Use the **show mpls label table detail** command to verify the SRLB range.

SUMMARY STEPS

1. **configure**
2. **segment-routing**
3. **adjacency-sid**
4. **interface** *type interface-path-id*
5. **address-family** { **ipv4** | **ipv6** } [**unicast**]
6. **l2-adjacency sid** {**index** *adj-SID-index* | **absolute** *adj-SID-value* } [**next-hop** {*ipv4_address* | *ipv6_address* }]
7. Use the **commit** or **end** command.
8. **end**
9. **router isis** *instance-id*
10. **address-family** { **ipv4** | **ipv6** } [**unicast**]
11. **segment-routing bundle-member-adj-sid**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters mode.
Step 2	**segment-routing** **Example:** `RP/0/RP0/CPU0:Router(config)# segment-routing`	Enters segment routing configuration mode.

	Command or Action	Purpose
Step 3	**adjacency-sid** **Example:** `RP/0/RP0/CPU0:Router(config-sr)# adjacency-sid`	Enters adjacency SID configuration mode.
Step 4	**interface** *type interface-path-id* **Example:** `RP/0/RP0/CPU0:Router(config-sr-adj)# interface GigabitEthernet0/0/0/3`	Specifies the interface and enters interface configuration mode.
Step 5	**address-family** { **ipv4** | **ipv6** } [**unicast**] **Example:** `RP/0/RP0/CPU0:Router(config-sr-adj-intf)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
Step 6	**l2-adjacency sid** {**index** *adj-SID-index* | **absolute** *adj-SID-value* } [**next-hop** { *ipv4_address* | *ipv6_address* }] **Example:** `RP/0/RP0/CPU0:Router(config-sr-adj-intf-af)# l2-adjacency sid absolute 15015 next-hop 10.1.1.4`	Configures the Adj-SID index or absolute value for the interface. Specify **index** *adj-SID-index* for each link to create an Ajd-SID based on the lower boundary of the SRLB + the index. Specify **absolute** *adj-SID-value* for each link to create a specific Ajd-SID within the SRLB. For point-to-point interfaces, you are not required to specify a next-hop. However, if you do specify the next-hop, the Layer 2 Adj-SID will be used only if the specified next-hop matches the neighbor address. For LAN interfaces, you must configure the next-hop IPv4 or IPv6 address. If you do not configure the next-hop, the Layer 2 Adj-SID will not be used for LAN interface.
Step 7	Use the **commit** or **end** command.	**commit** —Saves the configuration changes and remains within the configuration session. **end** —Prompts user to take one of these actions: • **Yes** — Saves configuration changes and exits the configuration session. • **No** —Exits the configuration session without committing the configuration changes. • **Cancel** —Remains in the configuration session, without committing the configuration changes.
Step 8	**end**	
Step 9	**router isis** *instance-id* **Example:**	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:Router(config)# router isis isp`	
Step 10	**address-family** { **ipv4** | **ipv6** } [**unicast**] **Example:** `RP/0/RP0/CPU0:Router(config-isis)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family, and enters router address family configuration mode.
Step 11	**segment-routing bundle-member-adj-sid** **Example:** `RP/0/RP0/CPU0:Router(config-isis-af)# segment-routing bundle-member-adj-sid`	Programs the dynamic Layer 2 Adj-SIDs, and advertises both manual and dynamic Layer 2 Adj-SIDs. **Note** This command is not required to program manual L2 Adj-SID, but is required to program the dynamic Layer 2 Adj-SIDs and to advertise both manual and dynamic Layer 2 Adj-SIDs.

Verify the configuration:

```
Router# show mpls forwarding detail | i "Pop|Outgoing Interface|Physical Interface"
Tue Jun 20 06:53:51.876 PDT
. . .
15001  Pop         SRLB (idx 1)       BE1          10.1.1.4        0
     Outgoing Interface: Bundle-Ether1 (ifhandle 0x000000b0)
     Physical Interface: GigabitEthernet0/0/0/3 (ifhandle 0x000000b0)


Router# show running-config segment-routing
Tue Jun 20 07:14:25.815 PDT
segment-routing
 adjacency-sid
  interface GigabitEthernet0/0/0/3
   address-family ipv4 unicast
    l2-adjacency-sid absolute 15001
   !
  !
 !
!
```

Associated Commands

- l2-adjacency sid
- segment-routing bundle-member-adj-sid

IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability

The following sub-TLVs support the advertisement of IPv4 and IPv6 prefix attribute flags and the source router ID of the router that originated a prefix advertisement, as described in RFC 7794.

- Prefix Attribute Flags
- IPv4 and IPv6 Source Router ID

Prefix Attribute Flags

The Prefix Attribute Flag sub-TLV supports the advertisement of attribute flags associated with prefix advertisements. Knowing if an advertised prefix is directly connected to the advertising router helps to determine how labels that are associated with an incoming packet should be processed.

This section describes the behavior of each flag when a prefix advertisement is learned from one level to another.

Note Prefix attributes are only added when wide metric is used.

Prefix Attribute Flags Sub-TLV Format

```
 0 1 2 3 4 5 6 7 ...
+-+-+-+-+-+-+-+-+...
|X|R|N|           ...
+-+-+-+-+-+-+-+-+...
```

Prefix Attribute Flags Sub-TLV Fields

Field	Description
X (External Prefix Flag)	This flag is set if the prefix has been redistributed from another protocol. The value of the flag is preserved when the prefix is propagated to another level.
R (Re-advertisement Flag)	This flag is set to 1 by the Level 1-2 router when the prefix is propagated between IS-IS levels (from Level 1 to Level 2, or from Level 2 to Level 1). This flag is set to 0 when the prefix is connected locally to an IS-IS-enabled interface (regardless of the level configured on the interface).

Field	Description
N (Node Flag)	For prefixes that are propagated from another level: **1.** Copy the N-flag from the prefix attribute sub-TLV, if present in the source level. **2.** Copy the N-flag from the prefix-SID sub-TLV, if present in the source level. **3.** Otherwise, set to 0. For connected prefixes: **1.** Set to 0 if **prefix-attributes n-flag-clear** is configured (see Configuring Prefix Attribute N-flag-clear, on page 15). **2.** Set to 0 if **prefix-sid** {**index***SID-index* | **absolute** *SID-value*} {**n-flag-clear**] is configured . **3.** Otherwise, set to 1 when the prefix is a host prefix (/32 for IPV4, /128 for IPv6) that is associated with a loopback address. **Note** If the flag is set and the prefix length is not a host prefix, then the flag must be ignored.

IPv4 and IPv6 Source Router ID

The Source Router ID sub-TLV identifies the source of the prefix advertisement. The IPv4 and IPv6 source router ID is displayed in the output of the **show isis database verbose** command.

The Source Router ID sub-TLV is added when the following conditions are met:

1. The prefix is locally connected.
2. The N-flag is set to 1 (when it's a host prefix and the **n-flag-clear** configuration is not used).
3. The router ID is configured in the corresponding address family.

The source router ID is propagated between levels.

Table 2: Source Router Sub-TLV Format

IPv4 Source Router ID	Type: 11 Length: 4 Value: IPv4 Router ID of the source of the prefix advertisement
IPv6 Source Router ID	Type: 12 Length: 16 Value: IPv6 Router ID of the source of the prefix advertisement

Configuring Prefix Attribute N-flag-clear

The N-flag is set to 1 when the prefix is a host prefix (/32 for IPV4, /128 for IPv6) that is associated with a loopback address. The advertising router can be configured to not set this flag. This task explains how to clear the N-flag.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **interface Loopback** *instance*
4. **prefix-attributes n-flag-clear** [**Level-1** | **Level-2**]
5. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `RP/0/RP0/CPU0:router# configure`	Enters mode.
Step 2	**router isis** *instance-id* **Example:** `RP/0/RP0/CPU0:router(config)# router isis 1`	
Step 3	**interface Loopback** *instance* **Example:** `RP/0/RP0/CPU0:router(config)# interface Loopback0`	Specifies the loopback interface.
Step 4	**prefix-attributes n-flag-clear** [**Level-1**	**Level-2**] **Example:** `RP/0/RP0/CPU0:router(config-if)# isis prefix-attributes n-flag-clear`	Clears the prefix attribute N-flag explicitly.
Step 5	Use the **commit** or **end** command.	**commit** —Saves the configuration changes and remains within the configuration session. **end** —Prompts user to take one of these actions: • **Yes** — Saves configuration changes and exits the configuration session. • **No** —Exits the configuration session without committing the configuration changes.

	Command or Action	Purpose
		• **Cancel** —Remains in the configuration session, without committing the configuration changes.

Verify the prefix attribute configuration:

```
RP/0/RP0/CPU0:router# show isis database verbose

IS-IS 1 (Level-2) Link State Database
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
router.00-00        * 0x0000039b   0xfc27        1079            0/0/0
  Area Address: 49.0001
  NLPID:        0xcc
  NLPID:        0x8e
  MT:           Standard (IPv4 Unicast)
  MT:           IPv6 Unicast                                     0/0/0
  Hostname:     router
  IP Address:   10.0.0.1
  IPv6 Address: 2001:0db8:1234::0a00:0001
  Router Cap:   10.0.0.1, D:0, S:0
    Segment Routing: I:1 V:1, SRGB Base: 16000 Range: 8000
    SR Algorithm:
      Algorithm: 0
      Algorithm: 1
<...>
  Metric: 0          IP-Extended 10.0.0.1/32
    Prefix-SID Index: 1001, Algorithm:0, R:1 N:0 P:1 E:0 V:0 L:0
    Prefix Attribute Flags: X:0 R:1 N:0
  Metric: 10         IP-Extended 10.0.0.2/32
    Prefix-SID Index: 1002, Algorithm:0, R:0 N:1 P:0 E:0 V:0 L:0
    Prefix Attribute Flags: X:0 R:0 N:1
    Source Router ID: 10.0.0.2
<...>
```

IS-IS Unreachable Prefix Announcement

Table 3: Feature History Table

Feature Name	Release	Description
IS-IS Unreachable Prefix Announcement	Release 7.8.1	The Unreachable Prefix Announcement (UPA) notifies the loss of prefix reachability between areas or domains, for prefixes that are covered by the summary address range during inter-area or inter-domain summarization. This feature helps in identifying the routers that are facing prefix unreachability issues faster and fix it. The new commands introduced for this feature are: • summary-prefix • prefix-unreachable

The organization of networks into levels or areas and/or IGP domains helps to limit the scope of link-state information within certain boundaries. However, the state that is related to prefix reachability often requires propagation across these areas (Level1/Level2) or domains (Autonomous System Boundary Router (ASBR)). An Autonomous System Boundary Router (ASBR) is a router that is running multiple protocols and serves as a gateway to routers outside the Open Shortest Path First (OSPF) domain and those operating with different protocols.

Route summarization, also known as route aggregation, is a method to minimize the number of routing tables in an IP network. It consolidates selected multiple routes into a single route advertisement.

Techniques such as summarization address the scale challenges associated with the advertizement of the individual prefix state outside of local area/domain. MPLS architecture did not allow for the effective use of the summarization due to its end-to-end Label Switched Path (LSP) requirement. With the introduction of the SRv6, which does not have such requirement, the use of summarization has become important again.

Summarization results in suppression of the individual prefix state that is useful for triggering fast-convergence mechanisms outside of the Interior Gateway Routing Protocols (IGPs (for example - Border Gateway Protocol - Prefix Independent Convergence (BGP PIC) Edge).

This feature enables the notification of the individual prefixes becoming unreachable in its area/domain, when the summarization is used between areas/domains to advertise the reachability for these prefixes.

There are existing SRv6 deployments that use summarization and require fast detection of the egress Provider Edge (PE) going down. To address these deployments in timely manner, we use the existing Protocol Data Units (PDUs) and Tag-Length-Values (TLVs), which is based on the Prefix Unreachability Advertisement (UPA).

Configuration Steps

The configuration steps that are required to set up the Unreachable Prefix Announcement (UPA) feature are as follows:

- **UPA Advertisement**

  An existing IS-IS address-family submode **summary-prefix** command was extended for UPA advertisement.

```
Router(config)#router isis 1
Router(config-isis)#address-family ipv6 unicast
Router(config-isis-af)#summary-prefix beef:10::/32 level 2 adv-unreachable
Router(config-isis-af)#summary-prefix beef:11::/32 level 2 algorithm 128 adv-unreachable
 unreachable-component-tag 777
Router(config-isis-af)#commit
```

- **Prefix Unreachable**

  The new **prefix-unreachable** command includes new commands that control the UPA advertisement such as, lifetime, metric, limit the maximum number if UPAs and UPA processing. For more details see, prefix-unreachable

```
Router(config)#router isis 1
Router(config-isis)#address-family ipv6
Router(config-isis-af)#prefix-unreachable
Router(config-isis-prefix-unreachable)#adv-lifetime 500
Router(config-isis-prefix-unreachable)#adv-metric 4261412866
Router(config-isis-prefix-unreachable)#adv-maximum 77
Router(config-isis-prefix-unreachable)#rx-process-enable
Router(config-isis-prefix-unreachable)#commit
```

Running Configuration

Execute the following show commands to review the L1/L2 (area) or ASBR (domain) running configuration:

Run the **show run router isis 1 address-family ipv6 unicast** command to view the summary prefix under as well as UPA parameters under it.

```
Router#sh run router isis 1 address-family ipv6 unicast
router isis 1
 address-family ipv6 unicast
  advertise application lfa link-attributes srlg
  advertise link attributes
  prefix-unreachable
   adv-lifetime 300
  !
  summary-prefix 10::/64
  summary-prefix beef:10::/32 adv-unreachable
  summary-prefix beef:11::/32 algorithm 128 adv-unreachable
  summary-prefix ceef:10::/32 adv-unreachable
  propagate level 2 into level 1 route-policy L2_TO_L1
  segment-routing srv6
   locator USID_ALG0
   !
   locator USID_ALG128
   !
  !
 !
!
```

Conditional Prefix Advertisement

In some situations, it's beneficial to make the IS-IS prefix advertisement conditional. For example, an Area Border Router (ABR) or Autonomous System Boundary Router (ASBR) that has lost its connection to one of the areas or autonomous systems (AS) might keep advertising a prefix. If an ABR or ASBR advertises the Segment Routing (SR) SID with this prefix, the label stack of the traffic routed toward the disconnected area or AS might use this SID, which would result in dropped traffic at the ABR or ASBR.

ABRs or ASBRs are often deployed in pairs for redundancy and advertise a shared Anycast prefix SID. Conditional Prefix Advertisement allows an ABR or an ASBR to advertise its Anycast SID only when connected to a specific area or domain. If an ABR or ASBR becomes disconnected from the particular area or AS, it stops advertising the address for a specified interface (for example, Loopback).

Configure the conditional prefix advertisement under a specific interface. The prefix advertisement on this interface is associated with the route-policy that tracks the presence of a set of prefixes (prefix-set) in the Routing Information Base (RIB).

For faster convergence, the route-policy used for conditional prefix advertisement uses the new event-based **rib-has-route async** condition to notify IS-IS of the following situations:

- When the last prefix from the prefix-set is removed from the RIB.
- When the first prefix from the prefix-set is added to the RIB.

Configuration

To use the conditional prefix advertisement in IS-IS, create a prefix-set to be tracked. Then create a route policy that uses the prefix-set.

```
Router(config)# prefix-set prefix-set-name
Router(config-pfx)# prefix-address-1/length[, prefix-address-2/length,,,
prefix-address-16/length]
Router(config-pfx)# end-set

Router(config)# route-policy rpl-name
Router(config-rpl)# if rib-has-route async prefix-set-name then
Router(config-rpl-if)# pass
Router(config-rpl-if)# endif
Router(config-rpl)# end-policy
```

To advertise the loopback address in IS-IS conditionally, use the **advertise prefix route-policy** command under IS-IS interface address-family configuration sub-mode.

```
Router(config)# router isis 1
Router(config-isis)# interface Loopback0
Router(config-isis-if)# address-family ipv4 unicast
Router(config-isis-if-af)# advertise prefix route-policy rpl-name
Router(config-isis-if-af)# commit
```

Example

```
Router(config)# prefix-set domain_2
Router(config-pfx)# 2.3.3.3/32, 2.4.4.4/32
Router(config-pfx)# end-set
Router(config)# route-policy track_domain_2
Router(config-rpl)# if rib-has-route async domain_2 then
Router(config-rpl-if)# pass
Router(config-rpl-if)# endif
Router(config-rpl)# end-policy
Router(config)# router isis 1
Router(config-isis)# interface Loopback0
Router(config-isis-if)# address-family ipv4 unicast
Router(config-isis-if-af)# advertise prefix route-policy track_domain-2
Router(config-isis-if-af)# commit
```

Running Configuration

```
prefix-set domain_2
  2.3.3.3/32,
  2.4.4.4/32
end-set
!
route-policy track_domain_2
  if rib-has-route async domain_2 then
    pass
  endif
end-policy
!
router isis 1
 interface Loopback0
  address-family ipv4 unicast
   advertise prefix route-policy track_domain_2
  !
 !
```

!