Implementing IS-IS

Integrated Intermediate System-to-Intermediate System (IS-IS), Internet Protocol Version 4 (IPv4), is a standards-based Interior Gateway Protocol (IGP). The Cisco software implements the IP routing capabilities described in International Organization for Standardization (ISO)/International Engineering Consortium (IEC) 10589 and RFC 1195, and adds the standard extensions for single topology and multitopology IS-IS for IP Version 6 (IPv6).

This module describes how to implement IS-IS (IPv4 and IPv6) on your Cisco IOS XR network.

Enable IS-IS and Configure Level 1 or Level 2 Routing

This task explains how to enable IS-IS and configure the routing level for an area.

Note Configuring the routing level in Step 4 is optional, but is highly recommended to establish the proper level of adjacencies.

Note Users can configure the **no max-metric** command only with levels 1 or 2, that is, **no max-metric level {1|2}** in order to view the result in the output of the **show configuration** command. Else, the maximum metric configuration is not displayed in the output. This behavior is observed before committing the configuration to the router.

Before you begin

Although you can configure IS-IS before you configure an IP address, no IS-IS routing occurs until at least one IP address is configured.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **net** *network-entity-title*
4. **is-type** { **level-1** | **level-1-2** | **level-2-only** }
5. Use the **commit** or **end** command.
6. **show isis** [**instance** *instance-id*] **protocol**

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- By default, all IS-IS instances are automatically Level 1 and Level 2. 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 **net** *network-entity-title*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# net 47.0004.004d.0001.0001.0c11.1110.00
```

Configures network entity titles (NETs) for the routing instance.

- Specify a NET for each routing instance if you are configuring multi-instance IS-IS.
- This example configures a router with area ID 47.0004.004d.0001 and system ID 0001.0c11.1110.00.
- To specify more than one area address, specify additional NETs. Although the area address portion of the NET differs, the systemID portion of the NET must match exactly for all of the configured items.

Step 4 **is-type** { **level-1** | **level-1-2** | **level-2-only** }

Example:

```
RP/0/RP0/CPU0:router(config-isis)# is-type level-2-only
```

(Optional) Configures the system type (area or backbone router).

- By default, every IS-IS instance acts as a **level-1-2** router.
- The **level-1** keyword configures the software to perform Level 1 (intra-area) routing only. Only Level 1 adjacencies are established. The software learns about destinations inside its area only. Any packets containing destinations outside the area are sent to the nearest **level-1-2** router in the area.
- The **level-2-only** keyword configures the software to perform Level 2 (backbone) routing only, and the router establishes only Level 2 adjacencies, either with other Level 2-only routers or with **level-1-2** routers.
- The **level-1-2** keyword configures the software to perform both Level 1 and Level 2 routing. Both Level 1 and Level 2 adjacencies are established. The router acts as a border router between the Level 2 backbone and its Level 1 area.

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.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

Step 6 **show isis** [**instance** *instance-id*] **protocol**

Example:

```
RP/0/RP0/CPU0:router# show isis protocol
```

(Optional) Displays summary information about the IS-IS instance.

Single-Topology IPv6

Single-topology IPv6 allows IS-IS for IPv6 to be configured on interfaces along with an IPv4 network protocol. All interfaces must be configured with the identical set of network protocols, and all routers in the IS-IS area (for Level 1 routing) or the domain (for Level 2 routing) must support the identical set of network layer protocols on all interfaces.

In single-topology mode, IPv6 topologies work with both narrow and wide metric styles in IPv4 unicast topology. During single-topology operation, one shortest path first (SPF) computation for each level is used to compute both IPv4 and IPv6 routes. Using a single SPF is possible because both IPv4 IS-IS and IPv6 IS-IS routing protocols share a common link topology.

Configure Single Topology for IS-IS

After an IS-IS instance is enabled, it must be configured to compute routes for a specific network topology.

This task explains how to configure the operation of the IS-IS protocol on an interface for an IPv4 or IPv6 topology.

Before you begin

Note To enable the router to run in single-topology mode, configure each of the IS-IS interfaces with all of the address families enabled and "single-topology" in the address-family IPv6 unicast in the IS-IS router stanza. You can use either the IPv6 address family or both IPv4 and IPv6 address families, but your configuration must represent the set of all active address families on the router. Additionally, explicitly enable single-topology operation by configuring it in the IPv6 router address family submode.

Two exceptions to these instructions exist:

1. If the address-family stanza in the IS-IS process contains the **adjacency-check disable** command, then an interface is not required to have the address family enabled.
2. The **single-topology** command is not valid in the ipv4 address-family submode.

The default metric style for single topology is narrow metrics. However, you can use either wide metrics or narrow metrics. How to configure them depends on how single topology is configured. If both IPv4 and IPv6 are enabled and single topology is configured, the metric style is configured in the **address-family ipv4** stanza. You may configure the metric style in the **address-family ipv6** stanza, but it is ignored in this case. If only IPv6 is enabled and single topology is configured, then the metric style is configured in the **address-family ipv6** stanza.

SUMMARY STEPS

1. **configure**
2. **interface** *type interface-path-id*
3. Do one of the following:
   - **ipv4 address** *address mask*
   - **ipv6 address** *ipv6-prefix* / *prefix-length* [**eui-64**]

- **ipv6 address** *ipv6-address* { */ prefix-length* | *link-local* }
- **ipv6 enable**

4. **exit**
5. **router isis** *instance-id*
6. **net** *network-entity-title*
7. **address-family ipv6** [**unicast**]
8. **single-topology**
9. exit
10. **interface** *type interface-path-id*
11. **circuit-type** { **level-1** | **level-1-2** | **level-2-only** }
12. **address-family** { **ipv4** | **ipv6** } [**unicast**]
13. Use the **commit** or **end** command.
14. **show isis** [**instance** *instance-id*] **interface** [*type interface-path-id*] [**detail**] [**level** { **1** | **2** }]
15. **show isis** [**instance** *instance-id*] **topology** [**systemid** *system-id*] [**level** { **1** | **2** }] [**summary**]

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config)# interface
```

Enters interface configuration mode.

Step 3 Do one of the following:

- **ipv4 address** *address mask*
- **ipv6 address** *ipv6-prefix / prefix-length* [**eui-64**]
- **ipv6 address** *ipv6-address* { */ prefix-length* | *link-local* }
- **ipv6 enable**

Example:

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.0.1.3 255.255.255.0
```

or

```
RP/0/RP0/CPU0:router(config-if)# ipv6 address 3ffe:1234:c18:1::/64 eui-64
RP/0/RP0/CPU0:router(config-if)# ipv6 address FE80::260:3EFF:FE11:6770 link-local
```

```
RP/0/RP0/CPU0:router(config-if)# ipv6 enable
```

or

Defines the IPv4 address for the interface. An IP address is required on all interfaces in an area enabled for IS-IS if any one interface is configured for IS-IS routing.

or

Specifies an IPv6 network assigned to the interface and enables IPv6 processing on the interface with the **eui-64** keyword.

or

Specifies an IPv6 address assigned to the interface and enables IPv6 processing on the interface with the **link-local** keyword.

or

Automatically configures an IPv6 link-local address on the interface while also enabling the interface for IPv6 processing.

- The link-local address can be used only to communicate with nodes on the same link.
- Specifying the **ipv6 address** *ipv6-prefix* / *prefix-length* interface configuration command without the **eui-64** keyword configures site-local and global IPv6 addresses.
- Specifying the **ipv6 address** *ipv6-prefix* / *prefix-length* command with the **eui-64** keyword configures site-local and global IPv6 addresses with an interface ID in the low-order 64 bits of the IPv6 address. Only the 64-bit network prefix for the address needs to be specified; the last 64 bits are automatically computed from the interface ID.
- Specifying the **ipv6 address** command with the **link-local** keyword configures a link-local address on the interface that is used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface.

Step 4 **exit**

Example:

```
RP/0/RP0/CPU0:router(config-if)# exit
```

Exits interface configuration mode, and returns the router to mode.

Step 5 **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.

- By default, all IS-IS instances are Level 1 and Level 2. You can change the level of routing to be performed by a particular routing instance by using the **is-type** command.

Step 6 **net** *network-entity-title*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# net 47.0004.004d.0001.0001.0c11.1110.00
```

Configures NETs for the routing instance.

- Specify a NET for each routing instance if you are configuring multi-instance IS-IS. You can specify a name for a NET and for an address.
- This example configures a router with area ID 47.0004.004d.0001 and system ID 0001.0c11.1110.00.
- To specify more than one area address, specify additional NETs. Although the area address portion of the NET differs, the system ID portion of the NET must match exactly for all of the configured items.

Step 7 **address-family ipv6** [**unicast**]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# address-family ipv6 unicast
```

Specifies the IPv6 address family and enters router address family configuration mode.

- This example specifies the unicast IPv6 address family.

Step 8 **single-topology**

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# single-topology
```

(Optional) Configures the link topology for IPv4 when IPv6 is configured.

- The **single-topology** command is valid only in IPv6 submode. The command instructs IPv6 to use the single topology rather than the default configuration of a separate topology in the multitopology mode.

Step 9 exit

Example:

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

Exits router address family configuration mode, and returns the router to router configuration mode.

Step 10 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE 0/1/0/3
```

Enters interface configuration mode.

Step 11 **circuit-type** { **level-1** | **level-1-2** | **level-2-only** }

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# circuit-type level-1-2
```

(Optional) Configures the type of adjacency.

- The default circuit type is the configured system type (configured through the **is-type** command).
- Typically, the circuit type must be configured when the router is configured as only **level-1-2** and you want to constrain an interface to form only **level-1** or **level-2-only** adjacencies.

Step 12 **address-family** { **ipv4** | **ipv6** } [**unicast**]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
```

Specifies the IPv4 or IPv6 address family, and enters interface address family configuration mode.

- This example specifies the unicast IPv4 address family on the interface.

Step 13 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 14 **show isis** [**instance** *instance-id*] **interface** [*type interface-path-id*] [**detail**] [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router# show isis interface
```

(Optional) Displays information about the IS-IS interface.

Step 15 **show isis** [**instance** *instance-id*] **topology** [**systemid** *system-id*] [**level** { **1** | **2** }] [**summary**]

Example:

```
RP/0/RP0/CPU0:router# show isis topology
```

(Optional) Displays a list of connected routers in all areas.

Configuring Single-Topology IS-IS for IPv6: Example

The following example shows single-topology mode being enabled. An IS-IS instance is created, the NET is defined, IPv6 is configured along with IPv4 on an interface, and IPv4 link topology is used for IPv6. This configuration allows POS interface 0/3/0/0 to form adjacencies for both IPv4 and IPv6 addresses.

```
router isis isp
 net 49.0000.0000.0001.00
 address-family ipv6 unicast
  single-topology
 interface tenGigE 0/11/0/0
  address-family ipv4 unicast
  !
  address-family ipv6 unicast
  !
  exit
!
interface tenGigE 0/11/0/0
 ipv4 address 10.0.1.3 255.255.255.0
```

```
ipv6 address 2001::1/64
```

Set SPF Interval for a Single-Topology Configuration

This task explains how to make adjustments to the SPF calculation to tune router performance. This task is optional.

Because the SPF calculation computes routes for a particular topology, the tuning attributes are located in the router address family configuration submode. SPF calculation computes routes for Level 1 and Level 2 separately.

When IPv4 and IPv6 address families are used in a single-topology mode, only a single SPF for the IPv4 topology exists. The IPv6 topology "borrows" the IPv4 topology; therefore, no SPF calculation is required for IPv6. To tune the SPF calculation parameters for single-topology mode, configure the **address-family ipv4 unicast** command.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **address-family** { **ipv4** | **ipv6** } [**unicast**]
4. **spf-interval** {[**initial-wait** *initial* | **secondary-wait** *secondary* | **maximum-wait** *maximum*] ...} [**level** { **1** | **2** }]
5. Use the **commit** or **end** command.
6. **show isis** [**instance** *instance-id*] [[**ipv4** | **ipv6** | **afi-all**] [**unicast** | **safi-all**]] **spf-log** [**level** { **1** | **2** }] [**fspf** | **prc** | **nhc**] [**detail** | **verbose**] [**last** *number* | **first** *number*]

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **router isis** *instance-id*

Example:

```
Router(config)# router isis isp
```

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 **address-family** { **ipv4** | **ipv6** } [**unicast**]

Example:

```
Router(config-isis)#address-family ipv4 unicast
```

Specifies the IPv4or IPv6 address family, and enters router address family configuration mode.

Step 4 **spf-interval** {[**initial-wait** *initial* | **secondary-wait** *secondary* | **maximum-wait** *maximum*] ...} [**level** { **1** | **2** }]

Example:

```
Router(config-isis-af)# spf-interval maximum-wait 5000 initial-wait 50 secondary-wait 200
```

(Optional) Controls the minimum time between successive SPF calculations.

- This value imposes a delay in the SPF computation after an event trigger and enforces a minimum elapsed time between SPF runs.
- If this value is configured too low, the router can lose too many CPU resources when the network is unstable.
- Configuring the value too high delays changes in the network topology that result in lost packets.

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.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

Step 6 **show isis** [**instance** *instance-id*] [[**ipv4** | **ipv6** | **afi-all**] [**unicast** | **safi-all**]] **spf-log** [**level** { **1** | **2** }] [**fspf** | **prc** | **nhc**] [**detail** | **verbose**] [**last** *number* | **first** *number*]

Example:

```
Router# show isis instance 1 ipv4 spf-log
```

(Optional) Displays how often and why the router has run a full SPF calculation.

Customize Routes for IS-IS

This task explains how to perform route functions that include injecting default routes into your IS-IS routing domain and redistributing routes learned in another IS-IS instance. This task is optional.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*

3. **set-overload-bit** [**on-startup** { *delay* | **wait-for-bgp** }] [**level** { **1** | **2** }]
4. **address-family** { **ipv4** | **ipv6** } [**unicast**]
5. **default-information originate** [**route-policy** *route-policy-name*]
6. **distribute-list** { {**prefix-list** *prefix-list-name* | **route-policy** *route-policy-name*} } **in**
7. **redistribute isis** *instance* [**level-1** | **level-2** | **level-1-2**] [**metric** *metric*] [**metric-type** { **internal** | **external** }] [**policy** *policy-name*]
8. Do one of the following:
   - **summary-prefix** *address* / *prefix-length* [**level** { **1** | **2** }]
   - **summary-prefix** *ipv6-prefix* / *prefix-length* [**level** { **1** | **2** }]
9. **maximum-paths** *route-number*
10. **distance** *weight* [*address / prefix-length* [*route-list-name*]]
11. **set-attached-bit**
12. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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 process, and places the router in router configuration mode.

- By default, all IS-IS instances are automatically Level 1 and Level 2. You can change the level of routing to be performed by a particular routing instance by using the **is-type** command.

Step 3 **set-overload-bit** [**on-startup** { *delay* | **wait-for-bgp** }] [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# set-overload-bit
```

(Optional) Sets the overload bit.

Note
The configured overload bit behavior does not apply to NSF restarts because the NSF restart does not set the overload bit during restart.

Step 4 **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 5 **default-information originate** [**route-policy** *route-policy-name*]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# default-information originate
```

(Optional) Injects a default IPv4 or IPv6 route into an IS-IS routing domain.

- The **route-policy** keyword and *route-policy-name* argument specify the conditions under which the IPv4 or IPv6 default route is advertised.
- If the **route-policy** keyword is omitted, then the IPv4 or IPv6 default route is unconditionally advertised at Level 2.

Step 6 **distribute-list** { {**prefix-list** *prefix-list-name* | **route-policy** *route-policy-name*} } **in**

Example:

```
RP/0/RP0/CPU0:router(config-isis)# distribute-list { {prefix-list | prefix-list-name} |
{route-policy | route-policy-name} } in
```

(Optional) Filters the routes that Intermediate System-to-Intermediate System (IS-IS) installs in the Routing Information Base (RIB).

Warning
When **distribute-list in** command is configured, some routes that IS-IS computes are not installed in the forwarding plane of the local router, but other IS-IS routers will not be aware of this. This introduces a difference between the forwarding state computed by other IS-IS routers and the actual forwarding state on this router. In some cases, this could lead to traffic being dropped or looped. Hence, be careful about when to use this command.

Step 7 **redistribute isis** *instance* [**level-1** | **level-2** | **level-1-2**] [**metric** *metric*] [**metric-type** { **internal** | **external** }] [**policy** *policy-name*]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# redistribute isis 2 level-1
```

(Optional) Redistributes routes from one IS-IS instance into another instance.

- In this example, an IS-IS instance redistributes Level 1 routes from another IS-IS instance.

Step 8 Do one of the following:

- **summary-prefix** *address* / *prefix-length* [**level** { **1** | **2** }]
- **summary-prefix** *ipv6-prefix* / *prefix-length* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# summary-prefix 10.1.0.0/16 level 1
```

or

```
RP/0/RP0/CPU0:router(config-isis-af)# summary-prefix 3003:xxxx::/24 level 1
```

(Optional) Allows a Level 1-2 router to summarize Level 1 IPv4 and IPv6 prefixes at Level 2, instead of advertising the Level 1 prefixes directly when the router advertises the summary.

- This example specifies an IPv4 address and mask.

or

- This example specifies an IPv6 prefix, and the command must be in the form documented in RFC 2373 in which the address is specified in hexadecimal using 16-bit values between colons.
- Note that IPv6 prefixes must be configured only in the IPv6 router address family configuration submode, and IPv4 prefixes in the IPv4 router address family configuration submode.

Step 9 **maximum-paths** *route-number*

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# maximum-paths 16
```

(Optional) Configures the maximum number of parallel paths allowed in a routing table.

Step 10 **distance** *weight* [*address / prefix-length* [*route-list-name*]]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# distance 90
```

(Optional) Defines the administrative distance assigned to routes discovered by the IS-IS protocol.

- A different administrative distance may be applied for IPv4 and IPv6.

Step 11 **set-attached-bit**

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# set-attached-bit
```

(Optional) Configures an IS-IS instance with an attached bit in the Level 1 LSP.

Step 12 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.

Redistributing IS-IS Routes Between Multiple Instances: Example

The following example shows usage of the **set- attached-bit** and **redistribute** commands. Two instances, instance “1” restricted to Level 1 and instance “2” restricted to Level 2, are configured.

The Level 1 instance is propagating routes to the Level 2 instance using redistribution. Note that the administrative distance is explicitly configured higher on the Level 2 instance to ensure that Level 1 routes are preferred.

Attached bit is being set for the Level 1 instance since it is redistributing routes into the Level 2 instance. Therefore, instance “1” is a suitable candidate to get from the area to the backbone.

```
  router isis 1
    is-type level-2-only
   net 49.0001.0001.0001.0001.00
   address-family ipv4 unicast
    distance 116
    redistribute isis 2 level 2
  !
  interface
   address-family ipv4 unicast
  !
  !
  router isis 2
   is-type level-1
   net 49.0002.0001.0001.0002.00
   address-family ipv4 unicast
    set
-attached-bit

  !
  interface
   address-family ipv4 unicast
```

Maximum Paths Per Algorithm

Table 1: Feature History Table

Feature Name	Release	Description
Maximum Paths Per Algorithm	Release 7.8.1	This feature introduces the new algorithm 0 command. These updates enable individual granularity for regular SPF algorithms.

A new **algorithm 0** command is introduced.

The **algorithm 0** command includes the **address-family** <*ipv4*/*ipv6*> **unicast** subcommand, and a new **maximum-paths** <*maximum-paths*> subcommand. The **maximum-paths** under **algorithm 0** configuration block applies to the standard Shortest Path First algorithm of the IS-IS instance.

Note For information on IS-IS Flex Algo Maximum Paths, refer to the "Enabling Segment Routing Flexible Algorithm" chapter in the *Segment Routing Configuration Guide for Cisco NCS 5500 Series Routers*.

The new subcommands allow for maximum number of Equal-Cost Multi-path (ECMP) to be set for individual algorithms. The value that is configured on a per-algo per address-family basis overrides any value that is configured under the IS-IS global address-family submode.

Usage Guidelines and Limitations

- The maximum-paths per algorithm takes precedence over maximum-paths per address-family.
- The maximum paths effective for each SPF algorithm are as follows:
  - For algorithm 0/Standard SPF:
    - IPv4: 1
    - IPv6: 2

Configuration Example – Max Path

This example shows how you can set the per-algo maximum path:

```
Router(config)# router isis isp
Router(config-isis)# address-family ipv4 unicast
Router(config-isis-af)# maximum-paths 12
Router(config-isis-af)# exit
Router(config-isis)# address-family ipv6 unicast
Router(config-isis-af)# maximum-paths 8
Router(config-isis-af)# exit

Router(config-isis)# algorithm 0
Router(config-isis-std-algo)# address-family ipv4 unicast
Router(config-isis-std-algo-af)# maximum-paths 1
Router(config-isis-std-algo)# exit
Router(config-isis-std-algo)# address-family ipv6 unicast
Router(config-isis-std-algo-af)# maximum-paths 2
Router(config-isis-std-algo-af)# exit
```

Maximum Paths Per Algorithm Per Prefix

Table 2: Feature History Table

Feature Name	Release	Description
Maximum Paths Per Flexible Algorithm Per Prefix	Release 7.11.1	Previously, you could configure a maximum number of Equal-Cost Multi-path (ECMP) to be set for SPF algo 0. This feature provides additional granularity to the IS-IS Maximum Paths Per-Algorithm feature by allowing you to specify a set of prefixes for SPF algo 0. Now you can achieve a balance between path diversity and computational and memory requirements by controlling the number of paths for each specific algorithm and destination prefix combination. This feature introduces these changes: **CLI** • **maximum-paths route-policy** *name* YANG Data Models: • This feature extends the native `Cisco-IOS-XR-clns-isis-cfg.yang` model See GitHub, Yang Data Models Navigator

Previously, you could set the maximum paths for a Shortest Path First (SPF) algorithm per address-family.

With this feature, you can further refine the maximum paths configuration by associating it with specific prefixes for each algorithm. The existing **maximum-paths** command is extended to include a **route-policy** qualifier to configure the maximum paths per algorithm per prefix-list.

When installing paths into the Routing Information Base (RIB), the system will check if a maximum paths value has been configured for algorithm 0 and the associated prefix. If such a configuration exists, it will be used instead of the existing address-family value to determine the number of paths to be installed.

Note Route policies that have the attribute **set maximum-paths** *number* are supported.

Note For information on IS-IS Flex Algo Maximum Paths per Prefix, refer to the "Enabling Segment Routing Flexible Algorithm" chapter in the *Segment Routing Configuration Guide for Cisco NCS 5500 Series Routers*.

Usage Guidelines and Limitations

- The commands **maximum-paths** *maximum-paths* and **maximum-paths route-policy** *rpl-name* are mutually exclusive. You can configure either an unqualified number or a route-policy for any given IS-IS instance.
- The maximum-paths per algorithm per prefix takes precedence over maximum-paths per algorithm. Likewise, the maximum-paths per algorithm takes precedence over maximum-paths per address-family. This hierarchy ensures that the most specific configuration is prioritized when determining the maximum paths for a given algorithm and prefix combination.

Example: RPL - Prefix Set

Define a Prefix Set:

```
prefix-set isis-ipv4-L1
  20.1.0.101/32
end-set
```

Create a Route Policy:

```
route-policy isis-mp-if-L1
  if destination in isis-ipv4-L1 then
    set maximum-paths 2
  endif
end-policy
```

Configure Maximum Paths Per-Prefix:

```
router isis isp
 algorithm 0
  address-family ipv4 unicast
   maximum-paths route-policy isis-mp-if-L1
  !
```

Set Priority for Adding Prefixes to RIB

This optional task describes how to set the priority (order) for which specified prefixes are added to the RIB. The prefixes can be chosen using an access list (ACL), prefix list, or by matching a tag value.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **address-family** { **ipv4** | **ipv6** } [**unicast**]
4. **metric-style wide** [**transition**] [**level** { **1** | **2** }]
5. **spf prefix-priority** [**level** { **1** | **2** }] { **critical** | **high** | **medium** } { *access-list-name* | **tag** *tag* }
6. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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 process, and places the router in router configuration mode. In this example, the IS-IS instance is called isp.

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** [**transition**] [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# metric-style wide level 1
```

Configures a router to generate and accept only wide-link metrics in the Level 1 area.

Step 5 **spf prefix-priority** [**level** { **1** | **2** }] { **critical** | **high** | **medium** } { *access-list-name* | **tag** *tag* }

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# spf prefix-priority high tag 3
```

Installs all routes tagged with the value 3 first.

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.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

IS-IS Interfaces

IS-IS interfaces can be configured as one of the following types:

- Active—advertises connected prefixes and forms adjacencies. This is the default for interfaces.
- Passive—advertises connected prefixes but does not form adjacencies. The **passive** command is used to configure interfaces as passive. Passive interfaces should be used sparingly for important prefixes such as loopback addresses that need to be injected into the IS-IS domain. If many connected prefixes need to be advertised then the redistribution of connected routes with the appropriate policy should be used instead.
- Suppressed—does not advertise connected prefixes but forms adjacencies. The **suppress** command is used to configure interfaces as suppressed.
- Shutdown—does not advertise connected prefixes and does not form adjacencies. The **shutdown** command is used to disable interfaces without removing the IS-IS configuration.

Tag IS-IS Interface Routes

This optional task describes how to associate a tag with a connected route of an IS-IS interface.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **address-family** { **ipv4** | **ipv6** } [**unicast**]
4. **metric-style wide** [**transition**] [**level** { **1** | **2** }]
5. **exit**
6. **interface** *type number*
7. **address-family** { **ipv4** | **ipv6** } [**unicast**]
8. **tag** *tag*
9. Use the **commit** or **end** command.
10. **show isis** [**ipv4** | **ipv6** | **afi-all**] [**unicast** | **safi-all**] **route** [**detail**]

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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 process, and places the router in router configuration mode. In this example, the IS-IS instance is called isp.

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** [**transition**] [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# metric-style wide level 1
```

Configures a router to generate and accept only wide link metrics in the Level 1 area.

Step 5 **exit**

Example:

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

Exits router address family configuration mode, and returns the router to router configuration mode.

Step 6 **interface** *type number*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE
```

Enters interface configuration mode.

Step 7 **address-family** { **ipv4** | **ipv6** } [**unicast**]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
```

Specifies the IPv4 or IPv6 address family, and enters address family configuration mode.

Step 8 **tag** *tag*

Example:

```
RP/0/RP0/CPU0:router(config-isis-if-af)# tag 3
```

Sets the value of the tag to associate with the advertised connected route.

Step 9 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 10 **show isis** [**ipv4** | **ipv6** | **afi-all**] [**unicast** | **safi-all**] **route** [**detail**]

Example:

```
RP/0/RP0/CPU0:router# show isis ipv4 route detail
```

Displays tag information. Verify that all tags are present in the RIB.

Tagging Routes: Example

The following example shows how to tag routes.

```
route-policy isis-tag-55
end-policy
!
route-policy isis-tag-555
  if destination in (5.5.5.0/24 eq 24) then
    set tag 555
    pass
  else
    drop
  endif
end-policy
!
router static
 address-family ipv4 unicast
  0.0.0.0/0 2.6.0.1
  5.5.5.0/24 Null0
 !
!
router isis uut
 net 00.0000.0000.12a5.00
 address-family ipv4 unicast
  metric-style wide
  redistribute static level-1 route-policy isis-tag-555
  spf prefix-priority critical tag 13
  spf prefix-priority high tag 444
  spf prefix-priority medium tag 777
```

Limit LSP Flooding

Limiting link-state packets (LSP) may be desirable in certain “meshy” network topologies. An example of such a network might be a highly redundant one such as a fully meshed set of point-to-point links over a nonbroadcast multiaccess (NBMA) transport. In such networks, full LSP flooding can limit network scalability. One way to restrict the size of the flooding domain is to introduce hierarchy by using multiple Level 1 areas and a Level 2 area. However, two other techniques can be used instead of or with hierarchy: Block flooding on specific interfaces and configure mesh groups.

Both techniques operate by restricting the flooding of LSPs in some fashion. A direct consequence is that although scalability of the network is improved, the reliability of the network (in the face of failures) is reduced because a series of failures may prevent LSPs from being flooded throughout the network, even though links exist that would allow flooding if blocking or mesh groups had not restricted their use. In such a case, the link-state databases of different routers in the network may no longer be synchronized. Consequences such as persistent forwarding loops can ensue. For this reason, we recommend that blocking or mesh groups be used only if specifically required, and then only after careful network design.

Control LSP Flooding for IS-IS

Flooding of LSPs can limit network scalability. You can control LSP flooding by tuning your LSP database parameters on the router globally or on the interface. This task is optional.

Many of the commands to control LSP flooding contain an option to specify the level to which they apply. Without the option, the command applies to both levels. If an option is configured for one level, the other level continues to use the default value. To configure options for both levels, use the command twice. For example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-refresh-interval 1200 level 2
RP/0/RP0/CPU0:router(config-isis)# lsp-refresh-interval 1100 level 1
```

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **lsp-refresh-interval** *seconds* [**level** { **1** | **2** }]
4. **lsp-check-interval** *seconds* [**level** { **1** | **2** }]
5. **lsp-gen-interval** { [**initial-wait** *initial* | **secondary-wait** *secondary* | **maximum-wait** *maximum*] **. . .** } [**level** { **1** | **2** }]
6. **lsp-mtu** *bytes* [**level** { **1** | **2** }]
7. **max-lsp-lifetime** *seconds* [**level** { **1** | **2** }]
8. **ignore-lsp-errors disable**
9. **interface** *type interface-path-id*
10. **lsp-interval** *milliseconds* [**level** { **1** | **2** }]
11. **csnp-interval** *seconds* [**level** { **1** | **2** }]
12. **retransmit-interval** *seconds* [**level** { **1** | **2** }]
13. **retransmit-throttle-interval** *milliseconds* [**level** { **1** | **2** }]
14. **mesh-group** { *number* | **blocked** }
15. Use the **commit** or **end** command.
16. **show isis interface** [*type interface-path-id* | **level** { **1** | **2** }] [**brief**]
17. **show isis** [**instance** *instance-id*] **database** [**level** { **1** | **2** }] [**detail** | **summary** | **verbose**] [***** | *lsp-id*]
18. **show isis** [**instance** *instance-id*] **lsp-log** [**level** { **1** | **2** }]
19. **show isis database-log** [**level** { **1** | **2** }]

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- 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 **lsp-refresh-interval** *seconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-refresh-interval 10800
```

(Optional) Sets the time between regeneration of LSPs that contain different sequence numbers

- The refresh interval should always be set lower than the **max-lsp-lifetime** command.

Step 4 **lsp-check-interval** *seconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-check-interval 240
```

(Optional) Configures the time between periodic checks of the entire database to validate the checksums of the LSPs in the database.

- This operation is costly in terms of CPU and so should be configured to occur infrequently.

Step 5 **lsp-gen-interval** { [**initial-wait** *initial* | **secondary-wait** *secondary* | **maximum-wait** *maximum*] **...** } [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-gen-interval maximum-wait 15 initial-wait 5 secondary-wait
5
```

(Optional) Reduces the rate of LSP generation during periods of instability in the network. Helps reduce the CPU load on the router and number of LSP transmissions to its IS-IS neighbors.

- During prolonged periods of network instability, repeated recalculation of LSPs can cause an increased CPU load on the local router. Further, the flooding of these recalculated LSPs to the other Intermediate Systems in the network causes increased traffic and can result in other routers having to spend more time running route calculations.

Step 6 **lsp-mtu** *bytes* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-mtu 1300
```

(Optional) Sets the maximum transmission unit (MTU) size of LSPs.

Step 7 **max-lsp-lifetime** *seconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# max-lsp-lifetime 11000
```

(Optional) Sets the initial lifetime given to an LSP originated by the router.

- This is the amount of time that the LSP persists in the database of a neighbor unless the LSP is regenerated or refreshed.

Step 8 **ignore-lsp-errors disable**

Example:

```
RP/0/RP0/CPU0:router(config-isis)# ignore-lsp-errors disable
```

(Optional) Sets the router to purge LSPs received with checksum errors.

Step 9 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE
```

Enters interface configuration mode.

Step 10 **lsp-interval** *milliseconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# lsp-interval 100
```

(Optional) Configures the amount of time between each LSP sent on an interface.

Step 11 **csnp-interval** *seconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# csnp-interval 30 level 1
```

(Optional) Configures the interval at which periodic CSNP packets are sent on broadcast interfaces.

- Sending more frequent CSNPs means that adjacent routers must work harder to receive them.
- Sending less frequent CSNP means that differences in the adjacent routers may persist longer.

Step 12 **retransmit-interval** *seconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# retransmit-interval 60
```

(Optional) Configures the amount of time that the sending router waits for an acknowledgment before it considers that the LSP was not received and subsequently resends.

```
RP/0/RP0/CPU0:router(config-isis-if)# retransmit-interval 60
```

Step 13 **retransmit-throttle-interval** *milliseconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# retransmit-throttle-interval 1000
```

(Optional) Configures the amount of time between retransmissions on each LSP on a point-to-point interface.

- This time is usually greater than or equal to the **lsp-interval** command time because the reason for lost LSPs may be that a neighboring router is busy. A longer interval gives the neighbor more time to receive transmissions.

Step 14 **mesh-group** { *number* | **blocked** }

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group blocked
```

(Optional) Optimizes LSP flooding in NBMA networks with highly meshed, point-to-point topologies.

- This command is appropriate only for an NBMA network with highly meshed, point-to-point topologies.

Step 15 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 16 **show isis** **interface** [*type* *interface-path-id* | **level** { **1** | **2** }] [**brief**]

Example:

```
RP/0/RP0/CPU0:router# show isis interface HundredGigE  brief
```

(Optional) Displays information about the IS-IS interface.

Step 17 **show isis** [**instance** *instance-id*] **database** [**level** { **1** | **2** }] [**detail** | **summary** | **verbose**] [***** | *lsp-id*]

Example:

```
RP/0/RP0/CPU0:router# show isis database level 1
```

(Optional) Displays the IS-IS LSP database.

Step 18 **show isis** [**instance** *instance-id*] **lsp-log** [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router# show isis lsp-log
```

(Optional) Displays LSP log information.

Step 19 **show isis database-log** [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router# show isis database-log level 1
```

(Optional) Display IS-IS database log information.

IS-IS Authentication

Authentication is available to limit the establishment of adjacencies by using the **hello-password** command, and to limit the exchange of LSPs by using the **lsp-password** command.

IS-IS supports plain-text authentication, which does not provide security against unauthorized users. Plain-text authentication allows you to configure a password to prevent unauthorized networking devices from forming adjacencies with the router. The password is exchanged as plain text and is potentially visible to an agent able to view the IS-IS packets.

When an HMAC-MD5 password is configured, the password is never sent over the network and is instead used to calculate a cryptographic checksum to ensure the integrity of the exchanged data.

IS-IS stores a configured password using simple encryption. However, the plain-text form of the password is used in LSPs, sequence number protocols (SNPs), and hello packets, which would be visible to a process that can view IS-IS packets. The passwords can be entered in plain text (clear) or encrypted form.

To set the domain password, configure the **lsp-password** command for Level 2; to set the area password, configure the **lsp-password** command for Level 1.

The keychain feature allows IS-IS to reference configured keychains. IS-IS key chains enable hello and LSP keychain authentication. Keychains can be configured at the router level (in the case of the **lsp-password** command) and at the interface level (in the case of the **hello-password** command) within IS-IS. These commands reference the global keychain configuration and instruct the IS-IS protocol to obtain security parameters from the global set of configured keychains.

IS-IS is able to use the keychain to implement hitless key rollover for authentication. ey rollover specification is time based, and in the event of clock skew between the peers, the rollover process is impacted. The configurable tolerance specification allows for the accept window to be extended (before and after) by that margin. This accept window facilitates a hitless key rollover for applications (for example, routing and management protocols).

Configure Authentication for IS-IS

This task explains how to configure authentication for IS-IS. This task is optional.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **lsp-password** { **hmac-md5** | **text** } { **clear** | **encrypted** } *password* [**level** { **1** | **2** }] [**send-only**] [**snp send-only**]
4. **interface** *type interface-path-id*
5. **hello-password** { **hmac-md5** | **text** } { **clear** | **encrypted** } *password* [**level** { **1** | **2** }] [**send-only**]
6. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- You can change the level of routing to be performed by a particular routing instance by using the **is-type** command.

Step 3 **lsp-password** { **hmac-md5** | **text** } { **clear** | **encrypted** } *password* [**level** { **1** | **2** }] [**send-only**] [**snp send-only**]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-password hmac-md5 clear password1 level 1
```

Configures the LSP authentication password.

- The **hmac-md5** keyword specifies that the password is used in HMAC-MD5 authentication.
- The **text** keyword specifies that the password uses cleartext password authentication.
- The **clear** keyword specifies that the password is unencrypted when entered.
- The **encrypted** keyword specifies that the password is encrypted using a two-way algorithm when entered.
- The **level 1** keyword sets a password for authentication in the area (in Level 1 LSPs and Level SNPs).
- The **level 2** keywords set a password for authentication in the backbone (the Level 2 area).

- The **send-only** keyword adds authentication to LSP and sequence number protocol data units (SNPs) when they are sent. It does not authenticate received LSPs or SNPs.
- The **snp send-only** keyword adds authentication to SNPs when they are sent. It does not authenticate received SNPs.

Note
To disable SNP password checking, the **snp send-only** keywords must be specified in the **lsp-password** command.

Step 4 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet
```

Enters interface configuration mode.

Step 5 **hello-password** { **hmac-md5** | **text** } { **clear** | **encrypted** } *password* [**level** { **1** | **2** }] [**send-only**]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)#hello-password text clear mypassword
```

Configures the authentication password for an IS-IS interface.

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.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

Configure Keychains for IS-IS

This task explains how to configure keychains for IS-IS. This task is optional.

Keychains can be configured at the router level (**lsp-password** command) and at the interface level (**hello-password** command) within IS-IS. These commands reference the global keychain configuration and instruct the IS-IS protocol to obtain security parameters from the global set of configured keychains. The router-level configuration (**lsp-password** command) sets the keychain to be used for all IS-IS LSPs generated by this router, as well as for all Sequence Number Protocol Data Units (SN PDUs). The keychain used for HELLO PDUs is set at the interface level, and may be set differently for each interface configured for IS-IS.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **l sp-password keychain** *keychain-name* [**level** { **1** | **2** }] [**send-only**] [**snp send-only**]

4. **interface** *type interface-path-id*
5. **hello-password keychain** *keychain-name* [**level** { **1** | **2** }] [**send-only**]
6. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- You can change the level of routing to be performed by a particular routing instance by using the **is-type** command.

Step 3 **l sp-password keychain** *keychain-name* [**level** { **1** | **2** }] [**send-only**] [**snp send-only**]

Example:

```
RP/0/RP0/CPU0:router(config-isis)# lsp-password keychain isis_a level 1
```

Configures the keychain.

Step 4 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE
```

Enters interface configuration mode.

Step 5 **hello-password keychain** *keychain-name* [**level** { **1** | **2** }] [**send-only**]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)#hello-password keychain isis_b
```

Configures the authentication password for an IS-IS interface.

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.

- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

Conditional Default Route Originate in IS-IS based on BGP Neighbor Status

Table 3: Feature History Table

Feature Name	Release Information	Feature Description
Conditional Default Route Originating in IS-IS	Release 7.4.1	The Conditional Default Route Originating in IS-IS feature allows you to enhance the granularity of the default route the IS-IS originates based on a condition. It enables IS-IS to originate the default route based on the presence of a specific route in the RIB originated by a particular BGP speaker. This feature improves the reaction time of the watched route in the RIB by avoiding periodical queries of the routing policy. This feature enables you to respond to the client in a timely fashion when the watched route changes in the RIB.

Table 4: Feature History Table

Feature Name	Release Information	Feature Description
Conditional Default Route Originating in IS-IS	Release 7.3.2	The Conditional Default Route Originating in IS-IS feature allows you to enhance the granularity of the default route the IS-IS originates based on a condition. It enables IS-IS to originate the default route based on the presence of a specific route in the RIB originated by a particular BGP speaker. This feature improves the reaction time of the watched route in the RIB by avoiding periodical queries of the routing policy. This feature enables you to respond to the client in a timely fashion when the watched route changes in the RIB.

Conditional Default Route Originating in IS-IS featureis based on BGP Neighbor Status feature allows you to enhance the granularity in the way IS-IS originates the default route based on certain specific conditions.

This feature improves the reaction time on the changes of the watched route in the RIB. With the **async** keyword in RPL, it avoids periodical query of the given policy. However, this feature allows you to callback to the client when the watched route changes in the RIB.

Configuration Example

```
Router(config)#router isis 1
Router(config-isis)# is-type level-2-only
Router(config-isis)# net 47.0000.0000.0005.00
Router(config-isis)# address-family ipv4 unicast
Router(config-isis-af)# metric-style wide
Router(config-isis-af)# mpls traffic-eng level-2-only
Router(config-isis-af)# mpls traffic-eng router-id 10.5.5.5
Router(config-isis-af)# default-information originate route-policy
Router(config-isis-af)# segment-routing mpls sr-prefer
Router(config-isis-af)# exit
Router(config-isis)# address-family ipv6 unicast
Router(config-isis-af)# metric-style wide
Router(config-isis-af)# default-information originate route-policy
Router(config-isis-af)# segment-routing mpls sr-prefer
Router(config-isis-af)# exit
Router(config-isis)# exit

/* Configure originate default route in ISIS based on BGP Neighbor Status */
Router(config)# route-policy track_bgp_neighbor
Router(config-rpl)# if track track-bgp-neighbors is up then
Router(config-rpl-if)# pass
Router(config-rpl-if)# endif
Router(config-rpl)# end-policy

/* Configure originate default route in ISIS based on BGP Route Status in RIB. */
Router(config)# route-policy track-bgp-neighbors
Router(config-rpl)# if rib-has-route async (192.1.1.0/24, 192.1.2.0/24) and source in
(10.2.35.1) and track track-bgp-neighbors is up then
Router(config-rpl-if)# pass
Router(config-rpl-if)# endif
Router(config-rpl-if)# end-policy

/* Track BGP neighbors */
Router(config)# track track-bgp-neighbors
Router(config-track)# type bgp neighbor address-family state
Router(config-track)# exit
Router(config)# address-family ipv4 unicast
Router(config)# neighbor 10.2.35.1

/* Configure the prefix-set in RPL */
Router(config)# prefix-set bgp_ipv6_neighbor_id
Router(config-pfx)# 10:2:35::1
Router(config-pfx)# end-set

Router(config)# prefix-set bgp_ipv6_watched_routes
Router(config-pfx)# 192:1:1::/112
Router(config-pfx)# 192:1:2::/112
Router(config-pfx)# end-set

Router(config)# route-policy default_route_policy_ipv6
Router(config-rpl)# if rib-has-route async bgp_ipv6_watched_routes and protocol is bgp 100
 and source in bgp_ipv6_neighbor_id then
Router(config-rpl-if)# pass
Router(config-rpl-if)# else
Router(config-rpl-if)# drop
Router(config-rpl-if)# endif
Router(config-rpl)# end-policy


router isis 1
```

```
 is-type level-2-only
 net 47.0000.0000.0005.00
.
.

address-family ipv4 unicast
  metric-style wide
  mpls traffic-eng level-2-only
  mpls traffic-eng router-id 5.5.5.5
  default-information originate route-policy <policy name - track-bgp-neighbors>
  segment-routing mpls sr-prefer
 !
 address-family ipv6 unicast
  metric-style wide
  default-information originate route-policy <policy name - default_route_policy_ipv6>
  segment-routing mpls sr-prefer

/* Configure originate default route in ISIS based on BGP Neighbor Status */

Tue May  4 11:02:22.031 IST
route-policy track_bgp_neighbor
  if track track-bgp-neighbors is up then
    pass
  endif
end-policy

/* Configure originate default route in ISIS based on BGP Route Status in RIB */
Mon Mar  8 13:25:26.263 IST
route-policy track-bgp-neighbors
  if rib-has-route async (192.1.1.0/24, 192.1.2.0/24) and source in (10.2.35.1) and track
track-bgp-neighbors is up then
    pass
  endif
end-policy

/* Configure tracking the status of the BGP neighbor */
show run track track-bgp-neighbors
Mon Mar  8 13:39:49.489 IST
track track-bgp-neighbors
 type bgp neighbor address-family state
  address-family ipv4 unicast
   neighbor 10.2.35.1
  !
 !
!
/* Configure prefix-set in RPL */
show rpl route-policy default_route_policy_ipv6 detail
Mon Mar  8 13:25:48.631 IST
prefix-set bgp_ipv6_neighbor_id
  10:2:35::1
end-set
!
prefix-set bgp_ipv6_watched_routes
  192:1:1::/112,
  192:1:2::/112
end-set
!
route-policy default_route_policy_ipv6
  if rib-has-route async bgp_ipv6_watched_routes and protocol is bgp 100 and source in
bgp_ipv6_neighbor_id then
    pass
  else
    drop
  endif
```

```
end-policy
!
```

Verification

```
/* Verify the status of the BGP neighbor */
Router(config)# show bgp neighbor brief
Mon Mar  8 13:30:27.312 IST
Neighbor        Spk   AS Description  Up/Down  NBRState
10.2.35.1         0   100             02:18:39 Established
10:2:35::1        0   100             02:18:40 Established

/* Verify the IPv4 RIB route */
Router# show route ipv4 192.1.1.0/24
Mon Mar  8 13:33:14.726 IST
Routing entry for 192.1.1.0/24
  Known via "bgp 100", distance 200, metric 0, type internal
  Installed Mar  8 11:11:52.738 for 02:21:22
  Routing Descriptor Blocks
    10.2.35.1, from 10.2.35.1
      Route metric is 0
  No advertising protos.

/* Verify the IPv6 RIB route */
Router# show route ipv6 192:1:1::/112
Mon Mar  8 13:33:31.340 IST
Routing entry for 192:1:1::/112
  Known via "bgp 100", distance 200, metric 0, type internal
  Installed Mar  8 11:11:52.738 for 02:21:38
  Routing Descriptor Blocks
    10:2:35::1, from 10:2:35::1
      Route metric is 0
  No advertising protos.

/* Verify tracking the status of the BGP neighbor */
Router# show track track-bgp-neighbors
Mon Mar  8 13:52:16.746 IST
Track track-bgp-neighbors
        BGP Neighbor AF IPv4 Unicast NBR 10.2.35.1 vrf default
        Reachability is UP
                Neighbor Address Reachablity is Up
                BGP Neighbor Address-family state is Up
        12 changes, last change IST Mon Mar 08 2021 11:11:52.741
        Delay up 0 secs(default), down 0 secs(default)

/* Verify the default route status in IS-IS address family */
Router# show isis
Mon Mar  8 13:34:39.412 IST

IS-IS Router: 1
  System Id: 0000.0000.0005
  Instance Id: 0
  IS Levels: level-2-only
  Manual area address(es):
    47
  Routing for area address(es):
    47
!!  .
  .
  Topologies supported by IS-IS:
    IPv4 Unicast
    .
```

```
    .
      Originating default route active since Mar 08 2021 11:12:05.914 IST
    IPv6 Unicast
    .
    .
      Originating default route active since Mar 08 2021 11:12:05.917 IST
!!
/* Verify the IS-IS database */
Router# show isis database detail verbose r5 | i 0.0.0.0/0
Mon Mar  8 13:47:10.624 IST
  Metric: 0          IP-Extended 0.0.0.0/0

Router# show isis database detail verbose r5 | i ::/0
Mon Mar  8 13:47:10.727 IST
  Metric: 0          MT (IPv6 Unicast) IPv6 ::/0

/* Verify the IPv4 IS-IS routes */
Router# show isis ipv4 route 0.0.0.0/0

Mon Mar  8 13:44:58.226 IST

L2 0.0.0.0/0 [10/115]
     via 10.1.35.2, TenGigE0/0/0/31, r5, SRGB Base: 16000, Weight: 0

/* Verify the IPv6 IS-IS routes */
Router# show isis ipv6 route 0::0/0
Mon Mar  8 13:45:02.699 IST

L2 ::/0 [10/115]
     via fe80::28a:96ff:fee7:f418, TenGigE0/0/0/31, r5, SRGB Base: 16000, Weight: 0
```

Nonstop Forwarding

On Cisco IOS XR software, IS-IS NSF minimizes the amount of time a network is unavailable to its users following the restart of the IS-IS process.

When the IS-IS process restarts, all routing peers of that device usually detect that the device went down and then came back up. This transition results in what is called a *routing flap*, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. NSF helps to suppress routing flaps, thus reducing network instability.

NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored following the process restarts. When the NSF feature is configured, peer networking devices do not experience routing flaps. To preserve routing across RP failover events, NSR must be configured in addition to NSF.

When the Cisco IOS XR router running IS-IS routing performs the process restarts, the router must perform two tasks to resynchronize its link-state database with that of its IS-IS neighbors. First, it must relearn the available IS-IS neighbors on the network without causing a reset of the neighbor relationship. Second, it must reacquire the contents of the link-state database for the network.

The IS-IS NSF feature offers two options when configuring NSF:

- IETF NSF
- Cisco NSF

If neighbor routers on a network segment are NSF-aware, meaning that they are running a software version that supports RFC5306, they assist a router configured with **nsf ietf** command that is restarting. IETF NSF

enables the neighbor routers provide adjacency and link-state information to help rebuild the routing information following a failover.

In Cisco IOS XR software, Cisco NSF checkpoints (stores persistently) all the state necessary to recover from a restart without requiring any special cooperation from neighboring routers. The state is recovered from the neighboring routers, but only using the standard features of the IS-IS routing protocol. This capability makes Cisco NSF suitable for use in networks in which other routers have not used the IETF standard implementation of NSF.

Note If you configure IETF NSF on the Cisco IOS XR router and a neighbor router does not support IETF NSF, the affected adjacencies flap, but nonstop forwarding is maintained to all neighbors that do support IETF NSF. A restart reverts to a cold start if no neighbors support IETF NSF.

Note ISIS, as a routing protocol, supports shorter hello intervals for neighbor adjacency management. However, the nonstop forwarding or routing feature on Cisco routers only work for default hello interval. When a failover occurs, there is a delay before the hello messages are sent and before the system is ready to send or receive the packets. The nonstop forwarding or routing feature is supported with a default hello interval or multiplier to overcome this delay. BFD, which runs on line cards, is responsible to fast-detect the loss of connectivity so there is no need to shorten hello interval.

Note Currently, a user can configure an aggressive hello-interval (lower than the default of 10 seconds for peer-to-peer session). But, if NSF is configured as a recovery for RP switchover, the default hello interval has to be used so that the sessions do not run into the risk of flapping during switchover.

Using LAN adjacencies in high availability (HA) scenarios is not recommended, since there is no designated intermediate system (DIS) redundancy in the protocol and traffic will either drop or be rerouted temporarily during DIS re-election.

Configure Nonstop Forwarding for IS-IS

This task explains how to configure your router with NSF that allows the software to resynchronize the IS-IS link-state database with its IS-IS neighbors after a process restart. The process restart could be due to an:

- RP failover (for a warm restart)
- Simple process restart (due to an IS-IS reload or other administrative request to restart the process)
- IS-IS software upgrade

In all cases, NSF mitigates link flaps and loss of user sessions. This task is optional.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **nsf** { **cisco** | **ietf** }

4. **nsf interface-expires** *number*
5. **nsf interface-timer** *seconds*
6. **nsf lifetime** *seconds*
7. Use the **commit** or **end** command.
8. **show running-config** [*command*]

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- 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 **nsf** { **cisco** | **ietf** }

Example:

```
RP/0/RP0/CPU0:router(config-isis)# nsf ietf
```

Enables NSF on the next restart.

- Enter the **cisco** keyword to run IS-IS in heterogeneous networks that might not have adjacent NSF-aware networking devices.
- Enter the **ietf** keyword to enable IS-IS in homogeneous networks where *all* adjacent networking devices support IETF draft-based restartability.

Step 4 **nsf interface-expires** *number*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# nsf interface-expires 1
```

Configures the number of resends of an acknowledged NSF-restart acknowledgment.

- If the resend limit is reached during the NSF restart, the restart falls back to a cold restart.

Step 5 **nsf interface-timer** *seconds*

Example:

```
RP/0/RP0/CPU0:router(config-isis) nsf interface-timer 15
```

Configures the number of seconds to wait for each restart acknowledgment.

Step 6 **nsf lifetime** *seconds*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# nsf lifetime 20
```

Configures the maximum route lifetime following an NSF restart.

- This command should be configured to the length of time required to perform a full NSF restart because it is the amount of time that the Routing Information Base (RIB) retains the routes during the restart.
- Setting this value too high results in stale routes.
- Setting this value too low could result in routes purged too soon.

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 **show running-config** [*command*]

Example:

```
RP/0/RP0/CPU0:router# show running-config router isis isp
```

(Optional) Displays the entire contents of the currently running configuration file or a subset of that file.

- Verify that “nsf” appears in the IS-IS configuration of the NSF-aware device.
- This example shows the contents of the configuration file for the “isp” instance only.

IS-IS Restart Signaling Support

The IS-IS Restart Signaling Support feature enables a restarting router to signal to its neighbors that it is restarting. This signaling allows neighboring routers to reestablish their adjacencies without going through the down state. At the same time, the neighboring routers initiate the synchronization of the database.

When an IS-IS router restarts, there is a temporary disruption of routing due to events in both the restarting router and the neighbors of the restarting router. The router that has restarted computes its own routes before it synchronizes the database with its neighbors.

The restarting router sends Suppress Adjacency (SA) advertisement toward the neighbor. The restarting router sends Intermediate-to-Intermediate Hello (IIH) messages to its neighbor to suppress the advertisement of the adjacency until the router is able to propagate newer versions of LSPs. The neighbor continues to suppress the advertisement of adjacency until it receives the SA bit clear message.

The IS-IS Restart Signaling Support conforms to the specifications detailed in RFC 5306.

ISIS NSR

Non Stop Routing (NSR) suppresses IS-IS routing changes for devices with redundant route processors during processor switchover events (RP failover or ISSU), reducing network instability and downtime. When Non Stop Routing is used, switching from the active to standby RP have no impact on the other IS-IS routers in the network. All information needed to continue the routing protocol peering state is transferred to the standby processor prior to the switchover, so it can continue immediately upon a switchover.

To preserve routing across process restarts, NSF must be configured in addition to NSR.

Note NSR applies to dual RP platforms only. To see which platforms are modular and dual RP, check this link: https://www.cisco.com/c/en/us/products/routers/network-convergence-system-5500-series/models-comparison.html.

Note Currently, a user can configure an aggressive hello-interval (lower than the default of 10 seconds for peer-to-peer session). But, if Non Stop Routing (NSR) is configured, the default hello interval has to be used so that the sessions do not run into the risk of flapping during switchover.

Using LAN adjacencies in high availability (HA) scenarios is not recommended, since there is no designated intermediate system (DIS) redundancy in the protocol and traffic will either drop or be rerouted temporarily during DIS re-election.

Configuring ISIS-NSR

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

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.

Step 3 **nsr**

Example:

```
RP/0/RP0/CPU0:router(config-isis)# nsr
```

Configures the NSR feature.

Step 4 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 5 **show isis nsr adjacency**

Example:

```
RP/0/RP0/CPU0:router
# show isis nsr adjacency
System Id Interface SNPA State Hold Changed NSF IPv4 BFD IPv6 BFD
  R1-v1S   Nii0     *PtoP* Up   83  00:00:33 Yes  None    None
```

Displays adjacency information.

Step 6 **show isis nsr status**

Example:

```
RP/0/RP0/CPU0:router
router#show isis nsr status
IS-IS test NSR(v1a) STATUS (HA Ready):
                           V1 Standby V2 Active V2 Standby
SYNC STATUS:                  TRUE      FALSE(0) FALSE(0)
PEER CHG COUNT:                1           0        0
UP TIME:                     00:03:12     not up    not up
```

Displays the NSR status information.

Step 7 **show isis nsr statistics**

Example:

```
RP/0/RP0/CPU0:router
router#show isis nsr statistics
IS-IS test NSR(v1a) MANDATORY STATS :
                                 V1 Active               V1 Standby         V2 Active           V2
 Standby
L1 ADJ:                               0                    0                  0
   0
L2 ADJ:                               2                    2                  0
   0
LIVE INTERFACE:                       4                    4                  0
   0
```

```
PTP INTERFACE:                  1                   1                   0
   0
LAN INTERFACE:                  2                   2                   0
   0
LOOPBACK INTERFACE:             1                   1                   0
   0
TE Tunnel:                      1                   1                   0
   0
TE LINK:                        2                   2                   0
   0
NSR OPTIONAL STATS :
L1 LSP:                         0                   0                   0
   0
L2 LSP:                         4                   4                   0
   0
IPV4 ROUTES:                    3                   3                   0
   0
IPV6 ROUTES:                    4                   4                   0
   0
```

Shows number of ISIS adjacencies, lsps, routes, tunnels, Te links on active and standby routers.

Configuring IS-IS Adjacency Stagger

Certain events like process restart or reload can involve a significant processing overhead. Updating routing tables with all adjacencies, maintaining them, and synchronizing the database with each adjacent router requires a lot of bandwidth. These processes may require large number of packets being sent and/or received, depending on the state of the database on the routers. If packets are dropped in any direction, it can lead to an unstable state.

We cannot prevent events like process restart or reload, but we can handle such events better by limiting the number of adjacencies that area being established simultaneously. To limit the number of adjacencies from getting established simultaneously, you can configure adjacency stagger. By configuring IS-IS adjacency stagger, you can specify the initial number neighbourhood routers from which adjacencies can fully form after a process restart or reload. If you configure IS-IS adjacency stagger, you can also specify the subsequent number of simultaneous neighbors that are allowed to form adjacency.

Restrictions

- IS-IS adjacency stagger is only supported on point-to-point interfaces and not on LAN interfaces.
- IS-IS adjacency stagger is not supported with NSF (non-stop forwarding) mechanisms.

Configuration Example

To configure IS-IS adjacency stagger on a point-to-point interface, you must use the following configuration steps:

1. Configure IS-IS.
2. Configure adjacency stagger.

Configuration

```
/* Enter the global configuration mode and configure IS-IS */
Router# config
Router(config)# router isis 1

/* Configure IS-IS adjacency stagger */
Router(config-isis)# adjacency stagger 2 3
Router(config-isis)# commit
```

Multiprotocol Label Switching Traffic Engineering

Table 5: Feature History Table

Feature Name	Release Information	Feature Description
MPLS TE Preference for Tunnels	Release 7.6.1	You can now configure the MPLS TE traffic for equal-cost multipath (ECMP) such that it flows only through TE tunnels. This is useful in scenarios where the hardware has resource constraints that limit the number of mixed ECMP routes. In earlier releases, IS-IS installed multiple ECMPs for a route in the Routing Information Base (RIB) through TE tunnels and physical interfaces by default. This feature introduces the following command: **mpls traffic-eng tunnel preferred** **mpls traffic-eng tunnel preferred** **mpls traffic-eng tunnel preferred**

The MPLS TE feature enables an MPLS backbone to replicate and expand the traffic engineering capabilities of Layer 2 ATM and Frame Relay networks. MPLS is an integration of Layer 2 and Layer 3 technologies.

For IS-IS, MPLS TE automatically establishes and maintains MPLS TE label-switched paths across the backbone by using Resource Reservation Protocol (RSVP). The route that a label-switched path uses is determined by the label-switched paths resource requirements and network resources, such as bandwidth. Available resources are flooded by using special IS-IS TLV extensions in the IS-IS. The label-switched paths are explicit routes and are referred to as traffic engineering (TE) tunnels.

Configure MPLS Traffic Engineering for IS-IS

This task explains how to configure IS-IS for MPLS TE. This task is optional.

Before you begin

Your network must support the MPLS software feature before you enable MPLS TE for IS-IS on your router.

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

Note MPLS traffic engineering currently does not support routing and signaling of LSPs over unnumbered IP links. Therefore, do not configure the feature over those links.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **address-family** { **ipv4** | **ipv6** } [**unicast**]
4. **mpls traffic-eng level** { **1** | **2** }
5. **mpls traffic-eng router-id** { *ip-address* | *interface-name interface-instance* }
6. **mpls traffic-eng tunnel preferred**
7. **metric-style wide** [**level** { **1** | **2** }]
8. Use the **commit** or **end** command.
9. **show isis** [**instance** *instance-id*] **mpls traffic-eng tunnel**
10. **show isis** [**instance** *instance-id*] **mpls traffic-eng adjacency-log**
11. **show isis** [**instance** *instance-id*] **mpls traffic-eng advertisements**

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- 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 **mpls traffic-eng level** { **1** | **2** }

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# mpls traffic-eng level 1
```

Configures a router running IS-IS to flood MPLS TE link information into the indicated IS-IS level.

Step 5 **mpls traffic-eng router-id** { *ip-address* | *interface-name interface-instance* }

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# mpls traffic-eng router-id loopback0
```

Specifies that the MPLS TE router identifier for the node is the given IP address or an IP address that is associated with the given interface.

Step 6 **mpls traffic-eng tunnel preferred**

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# mpls traffic-eng tunnel preferred
```

(optional) Prevents IS-IS from installing routes that use both MPLS TE tunnels and physical interfaces, and limits IS-IS to use only MPLS TE tunnels for ECMP.

Step 7 **metric-style wide** [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-af)# metric-style wide level 1
```

Configures a router to generate and accept only wide link metrics in the Level 1 area.

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.

Step 9 **show isis** [**instance** *instance-id*] **mpls traffic-eng tunnel**

Example:

```
RP/0/RP0/CPU0:router# show isis instance isp mpls traffic-eng tunnel
```

(Optional) Displays MPLS TE tunnel information.

Step 10 **show isis** [**instance** *instance-id*] **mpls traffic-eng adjacency-log**

Example:

```
RP/0/RP0/CPU0:router# show isis instance isp mpls traffic-eng adjacency-log
```

(Optional) Displays a log of MPLS TE IS-IS adjacency changes.

Step 11 **show isis** [**instance** *instance-id*] **mpls traffic-eng advertisements**

Example:

```
RP/0/RP0/CPU0:router# show isis instance isp mpls traffic-eng advertisements
```

(Optional) Displays the latest flooded record from Cisco Multiprotocol Label Switching Traffic Engineering.

MPLS TE Forwarding Adjacency

MPLS TE forwarding adjacency allows a network administrator to handle a traffic engineering, label switch path (LSP) tunnel as a link in an Interior Gateway Protocol (IGP) network, based on the Shortest Path First (SPF) algorithm. A forwarding adjacency can be created between routers in the same IS-IS level. The routers can be located multiple hops from each other. As a result, a TE tunnel is advertised as a link in an IGP network, with the cost of the link associated with it. Routers outside of the TE domain see the TE tunnel and use it to compute the shortest path for routing traffic throughout the network.

MPLS TE forwarding adjacency is considered in IS-IS SPF only if a two-way connectivity check is achieved. This is possible if the forwarding adjacency is bidirectional or the head end and tail end routers of the MPLS TE tunnel are adjacent.

The MPLS TE forwarding adjacency feature is supported by IS-IS. For details on configuring MPLS TE forwarding adjacency, see the MPLS Configuration Guide.

Tune Adjacencies for IS-IS

This task explains how to enable logging of adjacency state changes, alter the timers for IS-IS adjacency packets, and display various aspects of adjacency state. Tuning your IS-IS adjacencies increases network stability when links are congested. This task is optional.

For point-to-point links, IS-IS sends only a single hello for Level 1 and Level 2, which means that the level modifiers are meaningless on point-to-point links. To modify hello parameters for a point-to-point interface, omit the specification of the level options.

The options configurable in the interface submode apply only to that interface. By default, the values are applied to both Level 1 and Level 2.

The **hello-password** command can be used to prevent adjacency formation with unauthorized or undesired routers. This ability is particularly useful on a LAN, where connections to routers with which you have no desire to establish adjacencies are commonly found.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **log adjacency changes**
4. **interface** *type interface-path-id*
5. **hello-padding** { **disable** | **sometimes** } [**level** { **1** | **2** }]

6. **hello-interval** *seconds* [**level** { **1** | **2** }]
7. **hello-multiplier** *multiplier* [**level** { **1** | **2** }]
8. **h ello-password** { **hmac-md5** | **text** } { **clear** | **encrypted** } *password* [**level** { **1** | **2** }] [**send-only**]
9. Use the **commit** or **end** command.
10. **show isis** [**instance** *instance-id*] **adjacency** *t ype interface- path-id*] [**detail**] [**systemid** *system-id*]
11. show isis adjacency-log
12. **show isis** [**instance** *instance-id*] **interface** [*type interface-path-id*] [**brief** | **detail**] [**level** { **1** | **2** }]
13. **show isis** [**instance** *instance-id*] **neighbors** [*interface-type interface-instance*] [**summary**] [**detail**] [**systemid** *system-id*]

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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.

- You can change the level of routing to be performed by a particular routing instance by using the **is-type** command.

Step 3 **log adjacency changes**

Example:

```
RP/0/RP0/CPU0:router(config-isis)# log adjacency changes
```

Generates a log message when an IS-IS adjacency changes state (up or down).

Step 4 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE
```

Enters interface configuration mode.

Step 5 **hello-padding** { **disable** | **sometimes** } [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# hello-padding sometimes
```

Configures padding on IS-IS hello PDUs for an IS-IS interface on the router.

- Hello padding applies to only this interface and not to all interfaces.

Step 6 **hello-interval** *seconds* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)#hello-interval 6
```

Specifies the length of time between hello packets that the software sends.

Step 7 **hello-multiplier** *multiplier* [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# hello-multiplier 10
```

Specifies the number of IS-IS hello packets a neighbor must miss before the router should declare the adjacency as down.

- A higher value increases the networks tolerance for dropped packets, but also may increase the amount of time required to detect the failure of an adjacent router.
- Conversely, not detecting the failure of an adjacent router can result in greater packet loss.

Step 8 **h ello-password** { **hmac-md5** | **text** } { **clear** | **encrypted** } *password* [**level** { **1** | **2** }] [**send-only**]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# hello-password text clear mypassword
```

Specifies that this system include authentication in the hello packets and requires successful authentication of the hello packet from the neighbor to establish an adjacency.

Step 9 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 10 **show isis** [**instance** *instance-id*] **adjacency** *t ype interface- path-id*] [**detail**] [**systemid** *system-id*]

Example:

```
RP/0/RP0/CPU0:router# show isis instance isp adjacency
```

(Optional) Displays IS-IS adjacencies.

Step 11 show isis adjacency-log

Example:

```
RP/0/RP0/CPU0:router# show isis adjacency-log
```

(Optional) Displays a log of the most recent adjacency state transitions.

Step 12 **show isis** [**instance** *instance-id*] **interface** [*type interface-path-id*] [**brief** | **detail**] [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router# show isis interface HundredGigE  brief
```

(Optional) Displays information about the IS-IS interface.

Step 13 **show isis** [**instance** *instance-id*] **neighbors** [*interface-type interface-instance*] [**summary**] [**detail**] [**systemid** *system-id*]

Example:

```
RP/0/RP0/CPU0:router# show isis neighbors summary
```

(Optional) Displays information about IS-IS neighbors.

MPLS Label Distribution Protocol IGP Synchronization

Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP) Interior Gateway Protocol (IGP) Synchronization ensures that LDP has completed label exchange before the IGP path is used for switching. MPLS traffic loss can occur in the following two situations:

- When an IGP adjacency is established, the router begins forwarding packets using the new adjacency before LDP has exchanged labels with peers on that link.
- When an LDP session closes, the router continues to forward traffic using the link associated with the LDP peer rather than using an alternate path with an established LDP session.

This feature provides a mechanism to synchronize LDP and IS-IS to minimize MPLS packet loss. The synchronization is accomplished by changing the link metric for a neighbor IS-IS link-state packet (LSP), based on the state of the LDP session.

When an IS-IS adjacency is established on a link but the LDP session is lost or LDP has not yet completed exchanging labels, IS-IS advertises the maximum metric on that link. In this instance, LDP IS-IS synchronization is not yet achieved.

Note In IS-IS, a link with a maximum wide metric (0xFFFFFF) is not considered for shortest path first (SPF). Therefore, the maximum wide metric of -1 (0XFFFFFE) is used with MPLS LDP IGP synchronization.

When LDP IS-IS synchronization is achieved, IS-IS advertises a regular (configured or default) metric on that link.

Configuring MPLS LDP IS-IS Synchronization

This task explains how to enable Multiprotocol Label Switching (MPLS) Label Distribution Protocol (LDP) IS-IS synchronization. MPLS LDP synchronization can be enabled for an address family under interface configuration mode. Only IPv4 unicast address family is supported. This task is optional.

SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **interface** *type interface-path-id*
4. **address-family ipv4 unicast**
5. **mpls ldp sync** [**level** { **1** | **2** }]
6. Use the **commit** or **end** command.

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration 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 process, and places the router in router configuration mode.

- By default, all IS-IS instances are automatically Level 1 and Level 2. You can change the level of routing to be performed by a particular routing instance by using the **is-type** command.

Step 3 **interface** *type interface-path-id*

Example:

```
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE
```

Enters interface configuration mode.

Step 4 **address-family ipv4 unicast**

Example:

```
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
```

Specifies the IPv4 address family and enters router address family configuration mode.

Step 5 **mpls ldp sync** [**level** { **1** | **2** }]

Example:

```
RP/0/RP0/CPU0:router(config-isis-if-af)# mpls ldp sync level 1
```

Enables MPLS LDP synchronization for the IPv4 address family under interface HundredGigE .

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.
- **No** —Exits the configuration session without committing the configuration changes.
- **Cancel** —Remains in the configuration session, without committing the configuration changes.

Disable IID-TLV of IS-IS Protocol Instance

Table 6: Feature History Table

Feature Name	Release Information	Feature Description

Disable IID-TLV of IS-IS Protocol Instance	Release 7.10.1	Introduced in this release on: NCS 5500 fixed port routers; NCS 5700 fixed port routers; NCS 5500 modular routers (NCS 5500 line cards; NCS 5700 line cards [Mode: Compatibility; Native]) You can now disable Instance Identifier Type-Length-Value (IID-TLV) in the Hello and LSP packets when multiple IS-IS protocol instances are configured on the router. Each IS-IS instance has a unique instance-ID set, the TLV of which is sent in the Hello and LSP packets. The IID-TLV attribute helps in uniquely identifying the IS-IS protocol instance as well as the topologies to which the Protocol Data Units (PDUs) apply. The feature introduces these changes: **CLI** New Command: • **iid disable** Modified Commands: • The **hello-padding** command is extended to IS-IS process configuration mode • The **disable (IS-IS)** command is modified with a new **level** keyword, and also extended to interface configuration mode. **YANG Data Model** • New XPaths for `openconfig-isis.yang` (see GitHub, YANG Data Models Navigator)

IS-IS Overload Bit Avoidance

The IS-IS overload bit avoidance feature allows network administrators to prevent label switched paths (LSPs) from being disabled when a router in that path has its Intermediate System-to-Intermediate System (IS-IS) overload bit set.

When the IS-IS overload bit avoidance feature is activated, all nodes with the overload bit set, including head nodes, mid nodes, and tail nodes, are ignored, which means that they are still available for use with label switched paths (LSPs).

Note The IS-IS overload bit avoidance feature does *not* change the default behavior on nodes that have their overload bit set if those nodes are not included in the path calculation (PCALC).

The IS-IS overload bit avoidance feature is activated using the following command:

mpls traffic-eng path-selection ignore overload

The IS-IS overload bit avoidance feature is deactivated using the **no** form of this command:

no mpls traffic-eng path-selection ignore overload

When the IS-IS overload bit avoidance feature is deactivated, nodes with the overload bit set cannot be used as nodes of last resort.

Configure IS-IS Overload Bit Avoidance

This task describes how to activate IS-IS overload bit avoidance.

Before you begin

The IS-IS overload bit avoidance feature is valid only on networks that support the following features:

- MPLS
- IS-IS

SUMMARY STEPS

1. **configure**
2. **mpls traffic-eng path-selection ignore overload**

DETAILED STEPS

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router# configure
```

Enters global configuration mode.

Step 2 **mpls traffic-eng path-selection ignore overload**

Example:

```
RP/0/RP0/CPU0:router(config)# mpls traffic-eng path-selection ignore overload
```

Activates IS-IS overload bit avoidance.

Configuring IS-IS Overload Bit Avoidance: Example

The following example shows how to activate IS-IS overload bit avoidance:

```
config
 mpls traffic-eng path-selection ignore overload
```

The following example shows how to deactivate IS-IS overload bit avoidance:

```
config
 no mpls traffic-eng path-selection ignore overload
```

Configuring Global Weighted SRLG Protection

A shared risk link group (SRLG) is a set of links sharing a common resource and thus shares the same risk of failure. The existing loop-free alternate (LFA) implementations in interior gateway protocols (IGPs) support SRLG protection. However, the existing implementation considers only the directly connected links while computing the backup path. Hence, SRLG protection may fail if a link that is not directly connected but shares the same SRLG is included while computing the backup path. Global weighted SRLG protection feature provides better path selection for the SRLG by associating a weight with the SRLG value and using the weights of the SRLG values while computing the backup path.

To support global weighted SRLG protection, you need information about SRLGs on all links in the area topology. You can flood SRLGs for remote links using ISIS or manually configuring SRLGS on remote links.

Configuration Examples: Global Weighted SRLG Protection

There are three types of configurations that are supported for the global weighted SRLG protection feature.

- local SRLG with global weighted SRLG protection
- remote SRLG flooding
- remote SRLG static provisioning

This example shows how to configure the local SRLG with global weighted SRLG protection feature.

```
RP/0/RP0/CPU0:router(config)# srlg
RP/0/RP0/CPU0:router(config-srlg)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-srlg-if)# name group1
RP/0/RP0/CPU0:router(config-srlg-if)# exit
RP/0/RP0/CPU0:router(config-srlg)# interface TenGigE0/0/0/1
RP/0/RP0/CPU0:router(config-srlg-if)# name group1
RP/0/RP0/CPU0:router(config-srlg)# name group value 100
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix srlg-protection
weighted-global
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix tiebreaker srlg-disjoint
 index 1
RP/0/RP0/CPU0:router(config-isis)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-isis-if)# point-to-point
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix ti-lfa
RP/0/RP0/CPU0:router(config-isis)# srlg
RP/0/RP0/CPU0:router(config-isis-srlg)# name group1
RP/0/RP0/CPU0:router(config-isis-srlg-name)# admin-weight 5000
```

This example shows how to configure the global weighted SRLG protection feature with remote SRLG flooding.The configuration includes local and remote router configuration. On the local router, the global weighted SRLG protection is enabled by using the **fast-reroute per-prefix srlg-protection weighted-global** command. In the remote router configuration, you can control the SRLG value flooding by using the **advertise application lfa link-attributes srlg** command. You should also globally configure SRLG on the remote router.

The local router configuration for global weighted SRLG protection with remote SRLG flooding is as follows:

```
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix srlg-protection
weighted-global
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix tiebreaker srlg-disjoint
 index 1
RP/0/RP0/CPU0:router(config-isis-if-af)# exit
RP/0/RP0/CPU0:router(config-isis)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-isis-if)# point-to-point
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix ti-lfa
RP/0/RP0/CPU0:router(config-isis-if-af)# exit
RP/0/RP0/CPU0:router(config-isis)# srlg
RP/0/RP0/CPU0:router(config-isis-srlg)# name group1
RP/0/RP0/CPU0:router(config-isis-srlg-name)# admin-weight 5000
```

The remote router configuration for global weighted SRLG protection with remote SRLG flooding is as follows:

```
RP/0/RP0/CPU0:router(config)# srlg
RP/0/RP0/CPU0:router(config-srlg)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-srlg-if)# name group1
RP/0/RP0/CPU0:router(config-srlg-if)# exit
RP/0/RP0/CPU0:router(config-srlg)# interface TenGigE0/0/0/1
RP/0/RP0/CPU0:router(config-srlg-if)# name group1
RP/0/RP0/CPU0:router(config-srlg)# name group value 100
RP/0/RP0/CPU0:router(config-srlg)# exit
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# advertise application lfa link-attributes srlg
```

This example shows configuring the global weighted SRLG protection feature with static provisioning of SRLG values for remote links. You should perform these configurations on the local router.

```
RP/0/RP0/CPU0:router(config)# srlg
RP/0/RP0/CPU0:router(config-srlg)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-srlg-if)# name group1
RP/0/RP0/CPU0:router(config-srlg-if)# exit
RP/0/RP0/CPU0:router(config-srlg)# interface TenGigE0/0/0/1
RP/0/RP0/CPU0:router(config-srlg-if)# name group1
RP/0/RP0/CPU0:router(config-srlg)# name group value 100
RP/0/RP0/CPU0:router(config-srlg)# exit
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix srlg-protection
weighted-global
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix tiebreaker srlg-disjoint
 index 1
RP/0/RP0/CPU0:router(config-isis)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-isis-if)# point-to-point
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
```

```
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix
RP/0/RP0/CPU0:router(config-isis-if-af)# fast-reroute per-prefix ti-lfa
RP/0/RP0/CPU0:router(config-isis)# srlg
RP/0/RP0/CPU0:router(config-isis-srlg)# name group1
RP/0/RP0/CPU0:router(config-isis-srlg-name)# admin-weight 5000
RP/0/RP0/CPU0:router(config-isis-srlg-name)# static ipv4 address 10.0.4.1 next-hop ipv4
address 10.0.4.2
RP/0/RP0/CPU0:router(config-isis-srlg-name)# static ipv4 address 10.0.4.2 next-hop ipv4
address 10.0.4.1
```

IS-IS Penalty for Link Delay Anomaly

Note For information on configuring the link delay anomaly threshold values, refer to Link Anomaly Detection with IGP Penalty in the Segment Routing Configuration Guide.

When you configure Link Anomaly Detection in SR-PM, PM sets an anomaly bit (A-bit). When IGP receives the A-bit, IGP can automatically increase the IGP metric of the link by a user-defined amount (IGP penalty). This updated IGP metric is advertised in the network to make this link undesirable or unusable. When the link recovers, PM resets the A-bit.

Note When node is reloaded, the default or configured IGP metric (without penalty) is advertised until a new measurement is available.

Configuration

```
RP/0/RSP0/CPU0:ios(config)# router isis 100
RP/0/RSP0/CPU0:ios(config-isis)# interface GigabitEthernet 0/1/0/1
RP/0/RSP0/CPU0:ios(config-isis-if)# address-family ipv4 unicast
RP/0/RSP0/CPU0:ios(config-isis-if-af)# metric fallback anomaly delay increment 25
RP/0/RSP0/CPU0:ios(config-isis-if-af)# exit
RP/0/RSP0/CPU0:ios(config-isis-if)# exit
RP/0/RSP0/CPU0:ios(config-isis)# interface GigabitEthernet 0/1/0/2
RP/0/RSP0/CPU0:ios(config-isis-if)# address-family ipv4 unicast
RP/0/RSP0/CPU0:ios(config-isis-if-af)# metric fallback anomaly delay multiplier 2
```

Running Configuration

```
router isis 100
 interface GigabitEthernet0/1/0/1
  address-family ipv4 unicast
   metric fallback anomaly delay increment 25
  !
 !
 interface GigabitEthernet0/1/0/2
  address-family ipv4 unicast
   metric fallback anomaly delay multiplier 2
  !
 !
!
```

Setting an SPF interval for delaying the IS-IS SPF computations

Table 7: Feature History Table

Feature Name	Release	Description
Setting SPF interval in IS-IS to postpone the IS-IS SPF computations	Release 7.7.1	You can now define a standard algorithm to postpone the IS-IS SPF computations by setting an SPF interval. This reduces the computational load and churn on IGP nodes when multiple temporally close network events trigger multiple SPF computations. This algorithm also reduces the probability and the duration of transient forwarding loops during native IS-IS convergence when the protocol reacts to multiple temporally close events. This feature complies with RFC 8405. This feature introduces the **spf-interval ietf** command.

You can set an SPF interval in IS-IS to define a standard algorithm to postpone the IS-IS SPF computations off. This reduces the computational load and churn on IGP nodes when multiple temporally close network events trigger multiple SPF computations.

This algorithm reduces the probability and the duration of transient forwarding loops during native IS-IS convergence when the protocol reacts to multiple temporally close events.

To do this, you can use the algorithm specified by RFC 8405 to temporarily postpone the IS-IS SPF computation.

This task is optional.

Setting IETF for postponing SPF calculations

Configuration

1. Enter to the Cisco IOS XR configuration mode.

   For example,

   ```
   Router# configure
   ```

2. Enable IS-IS routing for the specified routing instance and place the router in router configuration mode.

   For example,

```
Router(config)# router isis <tag>
```

3. Specify the IPv4 or IPv6 address family, and then enters router address family configuration mode.

   For example,

   ```
   Router(config-isis)# address-family {ipv4 | ipv6} unicast
   ```

4. Set the interval type (IETF) for SPF calculations.

   For example,

   ```
   Router(config-isis-af)# spf-interval ietf
   ```

5. Commit the changes.

   For example,

   ```
   Router(config-isis-af)# commit
   ```

Configuration Example

```
Router# configure
Router(config)# router isis isp
Router(config-isis)# address-family ipv4 unicast
Router(config-isis-af)# spf-interval ietf?
initial-wait   Initial delay before running a route calculation [50]
short-wait     Short delay before running a route calculation [200]
long-wait      Long delay before running a route calculation [5000]
learn-interval Time To Learn interval for running a route calculation [500]
holddown-interval  Holddown interval for running a route calculation [10000]
level             Set SPF interval for one level only
Router(config-isis-af)# spf-interval ietf
Router(config-isis-af)# commit
```

Verification Example

```
Router# show run router isis
router isis 1
 net 49.0001.0000.0000.0100.00
 log adjacency changes
 address-family ipv4 unicast
  metric-style wide
  spf-interval ietf
 !
 address-family ipv6 unicast
  metric-style wide
  spf-interval ietf
 !

Router(config-isis-af)# spf-interval ietf?
initial-wait   Initial delay before running a route calculation [50]
short-wait     Short delay before running a route calculation [200]
long-wait      Long delay before running a route calculation [5000]
learn-interval Time To Learn interval for running a route calculation [500]
holddown-interval  Holddown interval for running a route calculation [10000]
level             Set SPF interval for one level only
```

The following **show** command displays the output with the new spf-interval algorithm. The output displays the actual delay taken to compute the SPF.

```
Router# show isis ipv4 spf-log last 5 detail
   IS-IS 1 Level 2 IPv4 Unicast Route Calculation Log
                   Time Total Trig.
```

```
Timestamp    Type    (ms) Nodes Count First Trigger LSP    Triggers
------------ ----- ----- ----- ----- -------------------- ------------------------
--- Wed Mar 16 2022 ---
15:31:49.763  FSPF     1     6     3         tb5-r4.00-00 LINKBAD PREFIXBAD
  Delay:                  101ms (since first trigger)
                          261177ms (since end of last calculation)
  Trigger Link:           tb5-r2.00
  Trigger Prefix:         34.1.24.0/24
  New LSP Arrivals:       0
  SR uloop:               No
  Next Wait Interval:     200ms
  RIB Batches:            1 (0 critical, 0 high, 0 medium, 1 low)
  Timings (ms):           +--Total--+
                           Real   CPU
    SPT Calculation:          1     1
    Route Update:             0     0
                          ----- -----
```

It is recommended to use the default delay values, which are listed in Syntax description. These default parameters are suggetsed by RFC 8405. These should be appropriate for most networks.

However, you can configure different values if required.

For example,

```
Router# configure
Router(config)# router isis isp
Router(config-isis)# address-family ipv4 unicast
Router(config-isis-af)# spf-interval ietf
Router(config-isis-af)# commit
Router(config-isis-af)# spf-interval ietf short-wait 500
Router(config-isis-af)# commit
```

LSP Fast-Flooding on IS-IS Networks

Table 8: Feature History Table

Feature Name	Release Name	Description
LSP Fast-Flooding on IS-IS Networks	Release 24.3.1	You can now accelerate the rate at which Link State Packets (LSPs) are distributed across an IS-IS network. Faster LSP distribution means faster network convergence. This faster convergence ensures that the most accurate topology information is quickly available across all routers on the network, reducing the chances of routing loops or misrouting. The feature introduces these changes: **CLI:** • **lsp-fast-flooding** • **max-lsp-tx** • **psnp-interval** • **remote-psnp-delay** **YANG Data Model:** • `Cisco-IOS-XR-um-router-isis-cfg` (see GitHub, YANG Data Models Navigator)

IS-IS Fast Flooding of LSPs

The IS-IS Fast Flooding of LSPs feature increases the transmission of Link State Packets (LSPs) within an IS-IS domain. By boosting the flooding rate of LSPs, this feature facilitates quicker distribution of network topology data, aiding the network in adapting more swiftly to modifications, such as link failures or restorations, node failures or addtions, and configuration changes. You can set up the feature to dispatch LSPs in bursts, targeting a default rate of 1000 LSPs per second, which is significantly higher than the conventional rate.

Flexibility in Adjustable Parameters and Activation

Unlike traditional IS-IS flooding, which operates at a fixed rate of 30 LSPs per second, the IS-IS Fast Flooding of LSPs feature offers adaptability and control. It encompasses several adjustable parameters accessible through commands, such as settings for the local Partial Sequence Number PDU (PSNP) interval and the maximum flooding rate, which can be modified according to particular network requirements. The feature also modulates the sending rate based on the neighbor's capacity to process and acknowledge the LSPs,

ensuring effective communication and preventing congestion.This feature is inactive by default, permitting you to activate it selectively in scenarios where the network infrastructure will benefit from improved LSP flooding speed, such as disaster recovery, real-time applications requiring low latency, edge applications closer to customer networks, dynamic network environments, high availability and redundancy setups, and large-scale networks. This selective activation is essential for maintaining optimal performance across various network topologies.

Dynamic Flooding Rate Optimization

The IS-IS Fast Flooding of LSPs feature incorporates real-time adjustment of the LSP flooding rate. This dynamic modification is based on persistent monitoring of the acknowledgment rates from neighboring routers through PSNP receipts. When the device on which the feature enabled identifies delays in acknowledgment, it automatically decreases the flooding rate to avoid overloading the neighbor's processing capabilities. Conversely, if acknowledgments are received promptly at the device, the device may elevate the flooding rate up to the configured maximum, enhancing the speed of topology distribution. This adaptive method ensures that the device reacts appropriately to the operational conditions of the network, providing an optimal balance between fast convergence and network stability. You can utilize this information to fine-tune the feature's parameters, ensuring that the flooding rate is both efficient and sustainable.

Role of the Commands in Configuring IS-IS Fast Flooding of LSPs

Configuring the **lsp-fast-flooding** command enables the fast flooding of Link State Packets (LSPs) to improve network response to topology changes. This command activates the IS-IS Fast Flooding of LSPs feature, which increases the transmission rate of LSPs, allowing for quicker distribution of network topology data. This helps the network adapt more swiftly to modifications such as link failures, restorations, node failures, additions, and configuration changes.

Configuring the **max-lsp-tx** command enables you to set the maximum number of Link State Packets (LSPs) that can be transmitted per second in an IS-IS routing process. This command allows you to control and limit the rate at which LSPs are sent, ensuring that the network can handle the increased traffic without becoming congested.

Configuring the **psnp-interval** command enables you to set the interval at which the IS-IS protocol sends Partial Sequence Number PDUs (PSNPs) to acknowledge received LSPs. This command helps manage the timing of acknowledgments, ensuring that the network can efficiently process and confirm the receipt of LSPs.

Configuring the **remote-psnp-delay** command enables you to specify the maximum delay, in milliseconds, that the router should wait for a Partial Sequence Number Protocol (PSNP) acknowledgment from neighbors after sending a LSP. This command helps manage the timing and responsiveness of the IS-IS Fast Flooding of LSPs, ensuring that the network can handle the increased rate of LSP transmission without causing congestion or delays.

References for IS-IS

This section provides additional conceptual information on IS-IS. It includes the following topics:

IS-IS Functional Overview

Small IS-IS networks are typically built as a single area that includes all routers in the network. As the network grows larger, it may be reorganized into a backbone area made up of the connected set of all Level 2 routers from all areas, which is in turn connected to local areas. Within a local area, routers know how to reach all

system IDs. Between areas, routers know how to reach the backbone, and the backbone routers know how to reach other areas.

The IS-IS routing protocol supports the configuration of backbone Level 2 and Level 1 areas and the necessary support for moving routing information between the areas. Routers establish Level 1 adjacencies to perform routing within a local area (intra-area routing). Routers establish Level 2 adjacencies to perform routing between Level 1 areas (interarea routing).

Each IS-IS instance can support either a single Level 1 or Level 2 area, or one of each. By default, all IS-IS instances automatically support Level 1 and Level 2 routing. You can change the level of routing to be performed by a particular routing instance using the **is-type** command.

Multiple IS-IS instances can exist on the same physical interface. However, you must configure different instance-id for every instance that shares the same physical interface.

Alternatively, you can also create dot1q sub-interfaces and configure each dot1q sub-interface to different IS-IS instances.

Default Routes

You can force a default route into an IS-IS routing domain. Whenever you specifically configure redistribution of routes into an IS-IS routing domain, the software does not, by default, redistribute the default route into the IS-IS routing domain. The **default-information originate** command generates a *default route* into IS-IS, which can be controlled by a route policy. You can use the route policy to identify the level into which the default route is to be announced, and you can specify other filtering options configurable under a route policy. You can use a route policy to conditionally advertise the default route, depending on the existence of another route in the routing table of the router.

Overload Bit on Router

The overload bit is a special bit of state information that is included in an LSP of the router. If the bit is set on the router, it notifies routers in the area that the router is not available for transit traffic. This capability is useful in four situations:

1. During a serious but nonfatal error, such as limited memory.
2. During the startup and restart of the process. The overload bit can be set until the routing protocol has converged. However, it is not employed during a normal NSF restart or failover because doing so causes a routing flap.
3. During a trial deployment of a new router. The overload bit can be set until deployment is verified, then cleared.
4. During the shutdown of a router. The overload bit can be set to remove the router from the topology before the router is removed from service.

Overload Bit Configuration During Multitopology Operation

Because the overload bit applies to forwarding for a single topology, it may be configured and cleared independently for IPv4 and IPv6 during multitopology operation. For this reason, the overload is set from the router address family configuration mode. If the IPv4 overload bit is set, all routers in the area do not use the router for IPv4 transit traffic. However, they can still use the router for IPv6 transit traffic.

Attached Bit on an IS-IS Instance

The attached bit is set in a router that is configured with the **is-type** command and **level-1-2** keyword. The attached bit indicates that the router is connected to other areas (typically through the backbone). This functionality means that the router can be used by Level 1 routers in the area as the default route to the backbone. The attached bit is usually set automatically as the router discovers other areas while computing its Level 2 SPF route. The bit is automatically cleared when the router becomes detached from the backbone.

Note If the connectivity for the Level 2 instance is lost, the attached bit in the Level 1 instance LSP would continue sending traffic to the Level 2 instance and cause the traffic to be dropped.

To simulate this behavior when using multiple processes to represent the **level-1-2** keyword functionality, you would manually configure the attached bit on the Level 1 process.

IS-IS Support for Route Tags

The IS-IS Support for route tags feature provides the capability to associate and advertise a tag with an IS-IS route prefix. Additionally, the feature allows you to prioritize the order of installation of route prefixes in the RIB based on a tag of a route. Route tags may also be used in route policy to match route prefixes (for example, to select certain route prefixes for redistribution).

Flood Blocking on Specific Interfaces

With this technique, certain interfaces are blocked from being used for flooding LSPs, but the remaining interfaces operate normally for flooding. This technique is simple to understand and configure, but may be more difficult to maintain and more error prone than mesh groups in the long run. The flooding topology that IS-IS uses is fine-tuned rather than restricted. Restricting the topology too much (blocking too many interfaces) makes the network unreliable in the face of failures. Restricting the topology too little (blocking too few interfaces) may fail to achieve the desired scalability.

To improve the robustness of the network in the event that all nonblocked interfaces drop, use the **csnp-interval** command in interface configuration mode to force periodic complete sequence number PDUs (CSNPs) packets to be used on blocked point-to-point links. The use of periodic CSNPs enables the network to become synchronized.

Maximum LSP Lifetime and Refresh Interval

By default, the router sends a periodic LSP refresh every 15 minutes. LSPs remain in a database for 20 minutes by default. If they are not refreshed by that time, they are deleted. You can change the LSP refresh interval or maximum LSP lifetime. The LSP interval should be less than the LSP lifetime or else LSPs time out before they are refreshed. In the absence of a configured refresh interval, the software adjusts the LSP refresh interval, if necessary, to prevent the LSPs from timing out.

Minimum Remaining Lifetime

The Minimum Remaining Lifetime feature prevents premature purging and unnecessary flooding of LSPs. If the Remaining Lifetime field gets corrupted during flooding, this corruption is undetectable. The consequences of such corruption depend on how the Remaining Lifetime value is altered. This feature resolves this problem by enabling IS-IS to reset the Remaining Lifetime value of the received LSP, to the maximum LSP lifetime.

By default, the maximum LSP lifetime is configured as 1200 seconds and you can configure it to a different value using the **max-lsp-lifetime** *seconds* command. This action ensures that whatever be the value of Remaining Lifetime that is received, a system other than the originator of an LSP will never purge the LSP, until the LSP has existed in the database at least for maximum LSP lifetime.

If the remaining lifetime for the LSP reaches 0, the LSP is kept in the link state database for an additional 60 seconds. This additional lifetime is known as Zero Age Lifetime. If the corresponding router does not update the LSP even after the Zero Age Lifetime, the LSP is deleted from the link state database.

The Remaining Lifetime field is also useful in identifying a problem in the network. If the received LSP lifetime value is less than the Zero Age Lifetime, which is 60 seconds, IS-IS generates an error message indicating that it's a corrupted lifetime event. The sample error message is as follows:

```
Dec 14 15:36:45.663 : isis[1011]: RECV L2 LSP 1111.1111.1112.03-00 from 1111.1111.1112.03:
 possible corrupted lifetime 59 secs for L2 lsp 1111.1111.1112.03-00 from SNPA 02e9.4522.5326
 detected.
```

IS-IS saves the received remaining lifetime value in LSP database. The value is shown in the **show isis database** command output under the **Rcvd** field.

Mesh Group Configuration

Configuring mesh groups (a set of interfaces on a router) can help to limit flooding. All routers reachable over the interfaces in a particular mesh group are assumed to be densely connected with each router having at least one link to every other router. Many links can fail without isolating one or more routers from the network.

In normal flooding, a new LSP is received on an interface and is flooded out over all other interfaces on the router. With mesh groups, when a new LSP is received over an interface that is part of a mesh group, the new LSP is not flooded over the other interfaces that are part of that mesh group.

Multi-Instance IS-IS

Table 9: Feature History Table

Feature Name	Release Information	Feature Description
32 IS-IS Instances	Release 7.6.1	You can now configure up to 32 IS-IS instances, thus enhancing the ability to isolate resources within your router and on the network. This ability enables you to configure more instances consuming network-wide resources at different rates, giving you more flexibility to manage your networks efficiently. In earlier releases, you could configure up to16 IS-IS instances.

You can configure up to 32 IS-IS instances per router, as long as **segment-routing mpls** is not configured under **router isis**. Multiple IS-IS instances can share a single interface if the instances you configure are with different instance IDs.

If **segment-routing mpls** is configured under **router isis**, then IS-IS takes the connection to Label Switching Database (LSD) component within the router. Configuring 32 IS-IS instances may exceed the 32 connections that are allowed to LSD. These connections are given out on a first-come-first-serve basis. It is possible for the IS-IS instances to take them all and prevent other clients like LDP, BGP, and so on, from getting connections. This may have adverse effects on the system.

Note IS-IS does not connect to LSD unless it needs one due to **segment-routing mpls** configuration. So, you can configure any number of connections up to 32, as long as the **segment-routing mpls** is not configured. Ensure caution if SR-MPLS is in use.

Use the **show mpls lsd clients** command to determine how many IS-IS instances you can configure. To do this, bring up the system without any IS-IS configuration and observe the number of LSD clients.

For example:

```
RP/0/0/CPU0:r100#show mpls lsd clients
Wed Mar 16 08:10:32.646 PDT
ID Services                                 Location
-- ---------------------------------------- ------------
0  LSD(A)                                   0/0/CPU0
1  Static(A)                                0/0/CPU0
2  L2VPN(A)                                 0/0/CPU0
3  PIM6:pim6(A)                             0/0/CPU0
4  Application-Controller:XTC(A)            0/0/CPU0
5  PIM:pim(A)                               0/0/CPU0
6  BFD(A)                                   0/0/CPU0
7  TE-Control(A)                            0/0/CPU0
8  LDP(A)                                   0/0/CPU0
```

In the example, nine client connections are being used, leaving 23 for use by IS-IS instances or other clients.

Because the Routing Information Base (RIB) treats each of the IS-IS instances as equal routing clients, you must be careful when redistributing routes between IS-IS instances. The RIB does not know to prefer Level 1 routes over Level 2 routes. For this reason, if you are running Level 1 and Level 2 instances, you must enforce the preference by configuring different administrative distances for the two instances.

Label Distribution Protocol IGP Auto-configuration

Label Distribution Protocol (LDP) Interior Gateway Protocol (IGP) auto-configuration simplifies the procedure to enable LDP on a set of interfaces used by an IGP instance. LDP IGP auto-configuration can be used on a large number interfaces (for example, when LDP is used for transport in the core) and on multiple IGP instances simultaneously.

This feature supports the IPv4 address family for the default VPN routing and forwarding (VRF) instance.

LDP IGP auto-configuration can also be explicitly disabled on individual interfaces under LDP using the **igp auto-config disable** command. This allows LDP to receive all IGP interfaces except the ones explicitly disabled.

See the MPLS configuration guide for information on configuring LDP IGP auto-configuration.

MPLS LDP-IGP Synchronization Compatibility with LDP Graceful Restart

LDP graceful restart protects traffic when an LDP session is lost. If a graceful restart-enabled LDP session fails, MPLS LDP IS-IS synchronization is still achieved on the interface while it is protected by graceful restart. MPLS LDP IGP synchronization is eventually lost under the following circumstances:

- LDP fails to restart before the LDP graceful restart reconnect timer expires.
- The LDP session on the protected interface fails to recover before the LDP graceful restart recovery timer expires.

MPLS LDP-IGP Synchronization Compatibility with IGP Nonstop Forwarding

IS-IS nonstop forwarding (NSF) protects traffic during IS-IS process restarts and route processor (RP) failovers. LDP IS-IS synchronization is supported with IS-IS NSF only if LDP graceful restart is also enabled over the interface. If IS-IS NSF is not enabled, the LDP synchronization state is not retained across restarts and failovers.