



# Configure Segment Routing Microloop Avoidance

The Segment Routing Microloop Avoidance feature enables link-state routing protocols, such as IS-IS and OSPF, to prevent or avoid microloops during network convergence after a topology change.

- [About Segment Routing Microloop Avoidance, on page 1](#)
- [Configure Segment Routing Microloop Avoidance for IS-IS, on page 3](#)
- [Configure Segment Routing Microloop Avoidance for OSPF, on page 4](#)

## About Segment Routing Microloop Avoidance

IP hop-by-hop routing may induce microloops (uLoops) at any topology transition. Microloops are a day-one IP challenge. Microloops are brief packet loops that occur in the network following a topology change (link down, link up, or metric change events). Microloops are caused by the non-simultaneous convergence of different nodes in the network. If a node converges and sends traffic to a neighbor node that has not converged yet, traffic may be looped between these two nodes, resulting in packet loss, jitter, and out-of-order packets.

Segment Routing resolves the microloop problem. A router with the Segment Routing Microloop Avoidance feature detects if microloops are possible for a destination on the post-convergence path following a topology change associated with a remote link event.

If a node determines that a microloop could occur on the new topology, the IGP computes a microloop-avoidant path to steer the traffic to that destination loop-free over the post-convergence path.

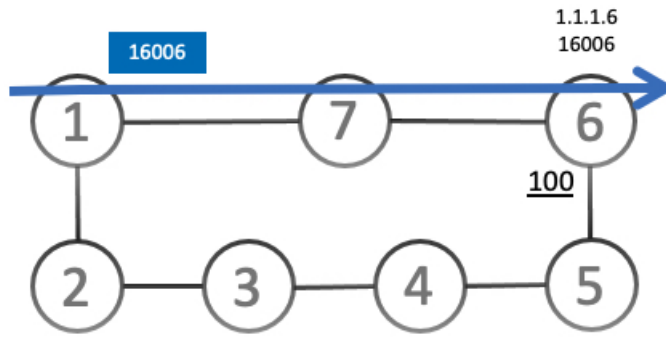
The IGP updates the forwarding table and temporarily (based on a RIB update delay timer) installs the SID-list imposition entries associated with the microloop-avoidant path for the destination with possible microloops.

After the RIB update delay timer expires, IGP updates the forwarding table, removing the microloop-avoidant SID list and traffic now natively follows the post-convergence path.

SR microloop avoidance is a local behavior and therefore not all nodes need to implement it to get the benefits.

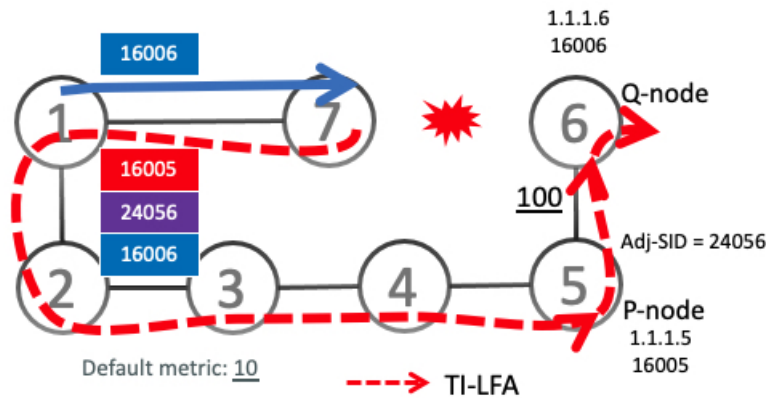
In the topology below, microloops can occur after the failure of the link between Node6 and Node7.

At steady state, Node1 sends traffic to node 6 (16006) via Node7. Node 7 is configured with TI-LFA to protect traffic to Node6.



Default metric: 10

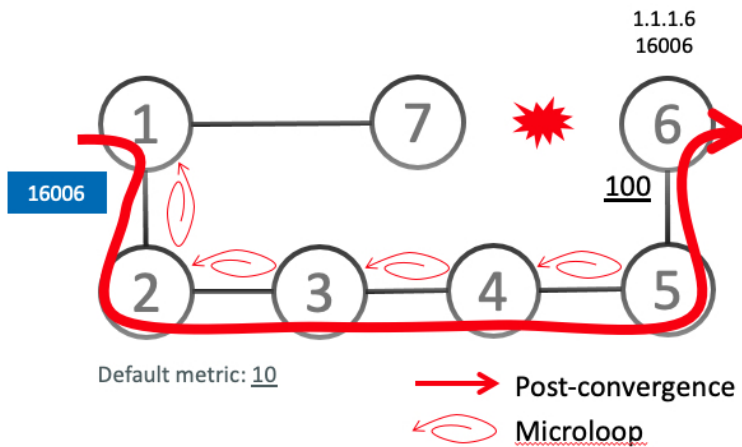
TI-LFA on Node7 pre-computes a backup path for traffic to Node6 (prefix SID 16006) that will be activated if the link between Node7 and Node6 goes down. In this network, the backup path would steer traffic toward Node5 (prefix SID 16005) and then via link between Node5 and Node6 (adj-SID 24056). All nodes are notified of the topology change due to the link failure.



Default metric: 10

-----> TI-LFA

However, if nodes along the path do not converge at the same time, microloops can be introduced. For example, if Node2 converged before Node3, Node3 would send traffic back to Node2 as the shortest IGP path to Node6. The traffic between Node2 and Node3 creates a microloop.



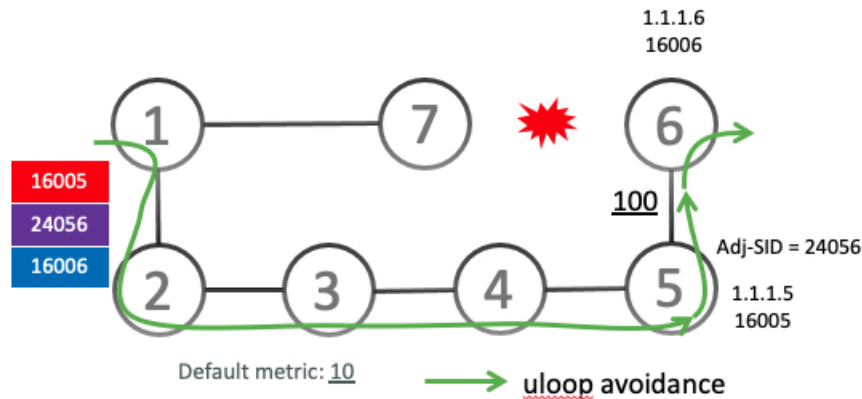
Default metric: 10

→ Post-convergence  
 ↻ Microloop

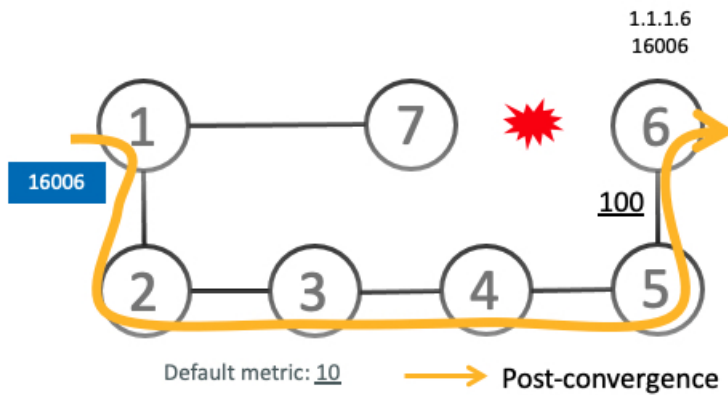
With microloop avoidance configured on Node1, a post-convergence path is computed and possible microloops on the post-convergence path for any destination are detected.

If microloops are possible on the post-convergence path to Node6, a microloop-avoidant path is constructed to steer the traffic to Node6 loop-free over the microloop-avoidant path {16005, 24056, 16006}.

Node1 updates the forwarding table and installs the SID-list imposition entries for those destinations with possible microloops, such as Node6. All nodes converge and update their forwarding tables, using SID lists where needed.



After the RIB update delay timer expires, the microloop-avoidant path is replaced with regular forwarding paths; traffic now natively follows the post-convergence path.



## Configure Segment Routing Microloop Avoidance for IS-IS

This task describes how to enable Segment Routing Microloop Avoidance and set the Routing Information Base (RIB) update delay value for IS-IS.

### Before you begin

Ensure that the following topology requirements are met:

- Router interfaces are configured as per the topology.
- Routers are configured with IS-IS.

- Segment routing for IS-IS is configured. See [Enabling Segment Routing for IS-IS Protocol](#).

## SUMMARY STEPS

1. **configure**
2. **router isis** *instance-id*
3. **address-family** { **ipv4** | **ipv6** } [ **unicast** ]
4. **microloop avoidance segment-routing**
5. **microloop avoidance rib-update-delay** *delay-time*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure</b> <b>Example:</b>  RP/0/RP0/CPU0:router# <b>configure</b>	Enters mode.
<b>Step 2</b>	<b>router isis</b> <i>instance-id</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config)# <b>router isis</b> 1	Enables IS-IS routing for the specified routing instance, and places the router in router configuration mode.  You can change the level of routing to be performed by a particular routing instance by using the <b>is-type</b> router configuration command.
<b>Step 3</b>	<b>address-family</b> { <b>ipv4</b>   <b>ipv6</b> } [ <b>unicast</b> ] <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis)# <b>address-family</b> <b>ipv4 unicast</b>	Specifies the IPv4 or IPv6 address family and enters router address family configuration mode.
<b>Step 4</b>	<b>microloop avoidance segment-routing</b> <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis-af)# <b>microloop</b> <b>avoidance segment-routing</b>	Enables Segment Routing Microloop Avoidance.
<b>Step 5</b>	<b>microloop avoidance rib-update-delay</b> <i>delay-time</i> <b>Example:</b>  RP/0/RP0/CPU0:router(config-isis-af)# <b>microloop</b> <b>avoidance rib-update-delay</b> 3000	Specifies the amount of time the node uses the microloop avoidance policy before updating its forwarding table. The <i>delay-time</i> is in milliseconds. The range is from 1-60000. The default value is 5000.

# Configure Segment Routing Microloop Avoidance for OSPF

This task describes how to enable Segment Routing Microloop Avoidance and set the Routing Information Base (RIB) update delay value for OSPF.

**Before you begin**

Ensure that the following topology requirements are met:

- Router interfaces are configured as per the topology.
- Routers are configured with OSPF.
- Segment routing for OSPF is configured. See [Enabling Segment Routing for OSPF Protocol](#).

**SUMMARY STEPS**

1. **configure**
2. **router ospf** *process-name*
3. **microloop avoidance segment-routing**
4. **microloop avoidance rib-update-delay** *delay-time*

**DETAILED STEPS**

	<b>Command or Action</b>	<b>Purpose</b>
<b>Step 1</b>	<b>configure</b> <b>Example:</b> RP/0/RP0/CPU0:router# <code>configure</code>	Enters mode.
<b>Step 2</b>	<b>router ospf</b> <i>process-name</i> <b>Example:</b> RP/0/RP0/CPU0:router(config)# <code>router ospf 1</code>	Enables OSPF routing for the specified routing process, and places the router in router configuration mode.
<b>Step 3</b>	<b>microloop avoidance segment-routing</b> <b>Example:</b> RP/0/RP0/CPU0:router(config-ospf)# <code>microloop avoidance segment-routing</code>	Enables Segment Routing Microloop Avoidance.
<b>Step 4</b>	<b>microloop avoidance rib-update-delay</b> <i>delay-time</i> <b>Example:</b> RP/0/RP0/CPU0:router(config-ospf)# <code>microloop avoidance rib-update-delay 3000</code>	Specifies the amount of time the node uses the microloop avoidance policy before updating its forwarding table. The <i>delay-time</i> is in milliseconds. The range is from 1-60000. The default value is 5000.

