Configure Segment Routing Path Computation Element

The Segment Routing Path Computation Element (SR-PCE) provides stateful PCE functionality by extending the existing IOS-XR PCEP functionality with more capabilities. SR-PCE is supported on the MPLS data plane and IPv4 control plane.

Note The Cisco IOS XRv 9000 is the recommended platform to act as the SR-PCE. Refer to the Cisco IOS XRv 9000 Router Installation and Configuration Guide for more information.

Table 1: Feature History Table

Feature Name	Release Information	Feature Description
Segment Routing Path Computation Element	Release 7.5.2	You can use a recommended platform to act as the Segment Routing Path Computation Element (SR-PCE) to calculate a suitable network path for transmitting data between a source and destination by applying metrics such as IGP, TE, and latency and restrictions such as the affinity of flexible algorithms for delay or IGP, and disjointness for LSPs. SR-PCE supports up to: • 50000 nodes • 100000 LSPs • 500000 links • 2000 PCEP sessions You can use SR-PCE for: • Disjoint Policy, on page 6 • PCE-initiated SR Policies for Traffic Management, on page 7 • PCC-initiated Policies Delegated to PCE, on page 26 • SR-PCE IPv4 Unnumbered Interface Support, on page 27

About SR-PCE

Table 2: Feature History Table

Feature Name	Release Information	Feature Description
TCP Authentication Option	Release 7.3.1	This feature introduces support for TCP Authentication Option (TCP-AO), which replaces the TCP Message Digest 5 (MD5) option, which was used for authenticating PCEP (TCP) sessions by using a clear text or encrypted password.

The path computation element protocol (PCEP) describes a set of procedures by which a path computation client (PCC) can report and delegate control of head-end label switched paths (LSPs) sourced from the PCC to a PCE peer. The PCE can request the PCC to update and modify parameters of LSPs it controls. The stateful model also enables a PCC to allow the PCE to initiate computations allowing the PCE to perform network-wide orchestration.

SR-PCE learns topology information by way of IGP (OSPF or IS-IS) or through BGP Link-State (BGP-LS).

SR-PCE is capable of computing paths using the following methods:

* TE metric—SR-PCE uses the TE metric in its path calculations to optimize cumulative TE metric.

* IGP metric—SR-PCE uses the IGP metric in its path calculations to optimize reachability.

* LSP Disjointness—SR-PCE uses the path computation algorithms to compute a pair of disjoint LSPs. The disjoint paths can originate from the same head-end or different head-ends. Disjoint level refers to the type of resources that should not be shared by the two computed paths.

   When the first request is received with a given disjoint-group ID, the first LSP is computed, encoding the shortest path from the first source to the first destination. When the second LSP request is received with the same disjoint-group ID, information received in both requests is used to compute two disjoint paths: one path from the first source to the first destination, and another path from the second source to the second destination. Both paths are computed at the same time.

TCP Authentication Option

Transmission Control Protocol (TCP) Message Digest 5 (MD5) authentication is used for authenticating PCEP (TCP) sessions by using clear text or encrypted password. This feature introduces support for TCP Authentication Option (TCP-AO), which replaces the TCP MD5 option.

TCP-AO uses Message Authentication Codes (MACs), which provides the following:

* Protection against replays for long-lived TCP connections

* More details on the security association with TCP connections than TCP MD5

* A larger set of MACs with minimal system and operational changes

TCP-AO is compatible with Primary Key Tuple (PKT) configuration. TCP-AO also protects connections when using the same PKT across repeated instances of a connection. TCP-AO protects the connections by using a traffic key that is derived from the PKT, and then coordinates changes between the endpoints.

Note TCP-AO and TCP MD5 are never permitted to use simultaneously. TCP-AO supports IPv6, and is fully compatible with the proposed requirements for the replacement of TCP MD5.

Usage Guidelines and Limitations

To ensure PCEP compatibility, we recommend that the Cisco IOS XR version on the SR-PCE be the same or later than the Cisco IOS XR version on the PCC or head-end.

These are the unsupported configurations for SR-MPLS TE Path Computation for IPv6:

Configure SR-PCE

This task explains how to configure SR-PCE.

Before you begin

The Cisco IOS XRv 9000 is the recommended platform to act as the SR-PCE.

SUMMARY STEPS

1. **configure**
2. **pce**
3. **address ipv4** *address*
4. **state-sync ipv4** *address*
5. **tcp-buffer** *size*
6. **password** {**clear** | **encrypted**} *password*
7. **tcp-ao** *key-chain* [**include-tcp-options**] [**accept-ao-mismatch-connection**]
8. **segment-routing** {**strict-sid-only** | **te-latency**}
9. **timers**
10. **keepalive** *time*
11. **minimum-peer-keepalive** *time*
12. **reoptimization** *time*
13. **exit**

DETAILED STEPS

Procedure

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

	Command or Action	**Purpose**
Step 2	**pce** **Example:** `RP/0/RP0/CPU0:router(config)# pce`	Enables PCE and enters PCE configuration mode.
Step 3	**address ipv4** *address* **Example:** `RP/0/RP0/CPU0:router(config-pce)# address ipv4 192.168.0.1`	Configures a PCE IPv4 address.
Step 4	**state-sync ipv4** *address* **Example:** `RP/0/RP0/CPU0:router(config-pce)# state-sync ipv4 192.168.0.3`	Configures the remote peer for state synchronization.
Step 5	**tcp-buffer** *size* **Example:** `RP/0/RP0/CPU0:router(config-pce)# tcp-buffer 1024000`	Configures the transmit and receive TCP buffer size for each PCEP session, in bytes. The default buffer size is 256000. The valid range is from 204800 to 1024000.
Step 6	**password** {**clear** | **encrypted**} *password* **Example:** `RP/0/RP0/CPU0:router(config-pce)# password encrypted pwd1`	Enables TCP authentication for all PCEP peers. Any TCP segment coming from the PCC that does not contain a MAC matching the configured password will be rejected. Specify if the password is encrypted or clear text. **Note** TCP-AO and TCP MD5 are never permitted to be used simultaneously.
Step 7	**tcp-ao** *key-chain* [**include-tcp-options**] [**accept-ao-mismatch-connection**] **Example:** `RP/0/RP0/CPU0:router(config-pce)# tcp-ao pce_tcp_ao include-tcp-options`	Enables TCP Authentication Option (TCP-AO) authentication for all PCEP peers. Any TCP segment coming from the PCC that does not contain a MAC matching the configured key chain will be rejected. • **include-tcp-options**—Includes other TCP options in the header for MAC calculation. • **accept-ao-mismatch-connection**—Accepts connection even if there is a mismatch of AO options between peers. **Note** TCP-AO and TCP MD5 are never permitted to be used simultaneously.

	Command or Action	Purpose
Step 8	**segment-routing** {**strict-sid-only** | **te-latency**} **Example:** RP/0/RP0/CPU0:router(config-pce)# **segment-routing strict-sid-only**	Configures the segment routing algorithm to use strict SID or TE latency. **Note** This setting is global and applies to all LSPs that request a path from this controller.
Step 9	**timers** **Example:** RP/0/RP0/CPU0:router(config-pce)# **timers**	Enters timer configuration mode.
Step 10	**keepalive** *time* **Example:** RP/0/RP0/CPU0:router(config-pce-timers)# **keepalive 60**	Configures the timer value for locally generated keep-alive messages. The default time is 30 seconds.
Step 11	**minimum-peer-keepalive** *time* **Example:** RP/0/RP0/CPU0:router(config-pce-timers)# **minimum-peer-keepalive 30**	Configures the minimum acceptable keep-alive timer that the remote peer may propose in the PCEP OPEN message during session establishment. The default time is 20 seconds.
Step 12	**reoptimization** *time* **Example:** RP/0/RP0/CPU0:router(config-pce-timers)# **reoptimization 30**	Configures the re-optimization timer. The default timer is 60 seconds.
Step 13	**exit** **Example:** RP/0/RP0/CPU0:router(config-pce-timers)# **exit**	Exits timer configuration mode and returns to PCE configuration mode.

Disjoint Policy

SR-PCE configuration compute disjointness for a pair of LSPs signaled by PCCs that do not include the PCEP association group-ID object in their PCEP request. This can be beneficial for deployments where PCCs do not support this PCEP object or when the network operator prefers to manage the LSP disjoint configuration centrally. The disjoint policy configuration is optional.

You can configure the disjoint group ID and the preferred level of disjointness (the type of resources that must not be shared by the two paths). SR-PCE supports the following disjoint path computations:

- link—Specifies that links are not shared on the computed paths.

- node—Specifies that nodes are not shared on the computed paths.

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

- srlg-node—Specifies that SRLG and nodes are not shared on the computed paths.

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

- If the requested disjointness level is SRLG or node, then link-disjoint paths will be computed.

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

Configure Disjoint Policy

Perform the following task to configure disjoint policy.

Configuration Example

```
Router)# configure
Router(config)# pce
Router(config-pce)# disjoint-path
Router(config-pce-disjoint)# group-id 1 type node sub-id 1
Router(config-pce-disjoint)# strict
```

strict is an optional parameter. It prevents the automatic fallback behavior of the preferred level of disjointness. If a pair of paths that meet the requested disjointness level cannot be found, the disjoint calculation is stopped and no new path is provided. The existing path is not modified.

```
Router(config-pce-disjoint)# lsp 1 pcc ipv4 192.168.0.1 lsp-name rtrA_t1 shortest-path
```

The above configuration adds LSPs to the disjoint group. The **shortest-path** forces one of the disjoint paths to follow the shortest path from the source to the destination. This option can only be applied to the first LSP specified.

```
Router(config-pce-disjoint# commit
```

PCE-initiated SR Policies for Traffic Management

An SR-TE policy can be configured on the path computation element (PCE) to reduce link congestion or to minimize the number of network touch points.

Note   The PCE-initiated SR-TE policies are entered in PCE configuration mode. For more information on configuring SR-TE policies, see the SR-TE Policy Overview.

The PCE collects network information, such as traffic demand and link utilization. When the PCE determines that a link is congested, it identifies one or more flows that are causing the congestion. The PCE finds a suitable path and deploys an SR-TE policy to divert those flows, without moving the congestion to another part of the network. When there is no more link congestion, the policy is removed.

To minimize the number of network touch points, an application, such as a Network Services Orchestrator (NSO), can request the PCE to create an SR-TE policy. PCE deploys the SR-TE policy using PCC-PCE communication protocol (PCEP).

PCEP defines the communication between PCE and PCE as specified in the following steps:

1. PCE sends a PCInitiate message to the PCC.

2. If the PCInitiate message is valid, the PCC sends a PCRpt message; otherwise, it sends PCErr message.

3. If the PCInitiate message is accepted, the PCE updates the SR-TE policy by sending PCUpd message.

You can achieve high-availability by configuring multiple PCEs with SR-TE policies. If the head-end (PCC) loses connectivity with one PCE, another PCE can assume control of the SR-TE policy.

Configure PCE-initiated SR Policy

To configure a PCE-initiated SR-TE policy, you must complete the following configurations:

1. Enter PCE configuration mode.

2. Create the segment list.

3. Create the policy.

Configure PCE-initiated SR Policy with Explicit SID List

Perform the following task to configure PCE-initiated SR policy with explicit SID list.

Configuration Example

```
/* Enter PCE configuration mode and create the SR-TE segment lists */
Router# configure
Router(config)# pce

/* Create the SR-TE segment lists */
Router(config-pce)# segment-routing
Router(config-pce-sr)# traffic-eng
Router(config-pce-sr-te)# segment-list name addr2a
Router(config-pce-sr-te-sl)# index 1 address ipv4 192.168.0.1
Router(config-pce-sr-te-sl)# exit

/* Create the SR-TE policy */
Router(config-pce-sr-te)# peer ipv4 10.1.1.1
Router(config-pce-sr-te)# policy P1
Router(config-pce-sr-te-policy)# color 2 end-point ipv4 172.16.0.1
Router(config-pce-sr-te-policy)# candidate-paths
Router(config-pce-sr-te-policy-path)# preference 50
Router(config-pce-sr-te-policy-path-preference)# explicit segment-list addr2a
Router(config-pce-sr-te-pp-info)# end
```

Running Configuration

```
pce
 segment-routing
  traffic-eng
   segment-list name addr2a
    index 1 address ipv4 192.168.0.1
   !
   peer ipv4 10.1.1.1
    policy P1
     color 2 end-point ipv4 172.16.0.1
     candidate-paths
      preference 50
       explicit segment-list addr2a
      !
     !
```

Configure PCE-initiated SR Policy with Dynamic SID List

Perform the following task to configure PCE-initiated SR policy with dynamic SID list.

Configuration Example

In this example, the PCE creates the policy and computes the path.

```
/* Enter PCE configuration mode */
Router# configure
Router(config)# pce

/* Create the SR-TE policy */
Router(config-pce)# segment-routing
Router(config-pce-sr)# traffic-eng
Router(config-pce-sr-te)# peer ipv4 10.1.1.1
Router(config-pce-sr-te)# policy P1
Router(config-pce-sr-te-policy)# color 2 end-point ipv4 172.16.0.1
Router(config-pce-sr-te-policy)# binding-sid mpls 10001
Router(config-pce-sr-te-policy)# candidate-paths
Router(config-pce-sr-te-policy-path)# preference 50
Router(config-pce-sr-te-policy-path-preference)# dynamic mpls
Router(config-pce-sr-te-pp-info)# metric type igp
Router(config-pce-sr-te-pp-info)# commit
```

Running Configuration

```
pce
 segment-routing
  traffic-eng
   peer ipv4 10.1.1.1
   policy P1
    binding-sid mpls 10001
    color 2 end-point ipv4 172.16.0.1
    candidate-paths
     preference 50
      dynamic mpls
       metric
        type igp
       !
      !
     !
```

```
                    !
                   !
                  !
                 !
                !
```

Exclude Network Resources during Path Computation over SR-TE Policies

Table 3: Feature History Table

Feature Name	Release Information	Feature Description
Exclude Network Resources during Path Computation over SR-TE Policies	Release 24.1.1	

Feature Name	Release Information	Feature Description
		You can reduce the risk of attacks due to less secure network resources and IP addresses, improve network performance affected by overloaded or congested paths, and ensure higher levels of network stability and availability that could be impacted by resources experiencing issues and requiring maintenance. These improvements are possible because you can now exclude specific network resources using their IP addresses during path computation for traffic over SR-TE policies. Previously, you could not exclude network resources during path computation. The feature introduces these changes: **CLI:** • The **resources** option is introduced in the **segment-routing traffic-eng policy** and **segment-routing traffic-eng on-demand color** commands. • The **resource-list** keyword is introduced in the **segment-routing traffic-eng** command. **YANG Data Models:** • `Cisco-IOS-XR-infra-xtc-oper.yang` • `Cisco-IOS-XR-infra-xtc-agent-oper.yang` • `Cisco-IOS-XR-infra-xtc-agent-cfg.yang` See (GitHub, Yang Data Models Navigator)

You can configure the SR-TE policies as either single or disjoint candidate paths that do not include certain IP addresses or subnets when traffic is routed through the network.

The key benefits of the feature are:

- **Security** - Excluding certain IP addresses or network resources can help protect sensitive resources from unauthorized access or attacks.

- **Performance** - By excluding certain resources, traffic can be directed away from congested parts of the network.

- **Isolation** - Excluding resources can help isolate different parts of the network, preventing issues in one part of the network from affecting others.

Configuration to exclude network resources during Path Computation

To exclude network resources during path computation for traffic, you must complete these high-level tasks in order:

1. Configure the network resources or IP addresses that you want to exclude from the network list.

2. Associate the excluded network resources to candidate paths for SR-TE or ODN SR-TE policies.

Configure the network resources or IP addresses to exclude from the network list

Perform the following task to configure a list of IPv4 addresses that you want to exclude from the network resource list:

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

Running Configuration

```
!
segment-routing
 traffic-eng
  resource-list node_resc_list
   index 1 ipv4 10.10.10.1
   index 2 ipv4 10.10.10.8
  !
 !
!
```

Associate the excluded network resources to candidate paths for SR-TE policies

Perform the following task to associate the excluded IPv4 addresses to one or more candidate paths for SR-TE policies:

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

Running Configuration

```
!
segment-routing
 traffic-eng
  policy dynamic_pcep_policy
   candidate-paths
```

```
        preference 100
         constraints
          resources
           exclude resource-list node_resc_list
          !
        !
      !
    !
  !
 !
!
```

Verification

```
Router#show segment-routing traffic-eng policy endpoint ipv4 100.2.1.1 color 8001

Wed Aug 30 22:21:50.014 UTC
SR-TE policy database
--------------------
Color: 8001, End-point: 100.2.1.1
Name: srte_c_8001_ep_100.2.1.1
Status:
Admin: up Operational: up for 00:00:50 (since Aug 30 22:20:59.341)
Candidate-paths:
Preference: 100 (configuration) (active)
Name: dynamic_pcep_policy
Requested BSID: 8001
PCC info:
Symbolic name: cfg_dynamic_pcep_policy_discr_100
PLSP-ID: 14042
Constraints:
Protection Type: protected-preferred
Maximum SID Depth: 12
Exclude Resources: node_resc_list
10.10.10.1
10.10.10.8
Dynamic (pce 100.7.1.1) (valid)
Metric Type: TE, Path Accumulated Metric: 440
SID[0]: 41111 [Adjacency-SID, 101.1.5.1 - 101.1.5.2]
SID[1]: 21600 [Prefix-SID, 100.6.1.1]
SID[2]: 41600 [Adjacency-SID, 101.2.6.2 - 101.2.6.1]
Attributes:
Binding SID: 8001
Forward Class: Not Configured
Steering labeled-services disabled: yes
Steering BGP disabled: no
IPv6 caps enable: yes
Invalidation drop enabled: no
Max Install Standby Candidate Paths: 0
```

Associate the excluded network resources to candidate paths for ODN SR-TE policies

Perform the following task to associate the excluded IPv4 addresses for ODN SR-TE policies:

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

Running Configuration

```
!
segment-routing
 traffic-eng
  on-demand color 7001
```

```
   constraints
    resources
     exclude resource-list node_resc_list
    !
   !
  !
 !
!
```

Verification

Router#**show segment-routing traffic-eng policy endpoint ipv4 100.2.1.1 color 7001**

```
Wed Aug 30 22:53:01.079 UTC

SR-TE policy database
---------------------

Color: 7001, End-point: 100.2.1.1
  Name: srte_c_7001_ep_100.2.1.1
  Status:
    Admin: up  Operational: up for 00:31:56 (since Aug 30 22:21:04.869)
  Candidate-paths:
    Preference: 200 (BGP ODN) (inactive) (shutdown)
      Requested BSID: dynamic
      Constraints:
        Protection Type: protected-preferred
        Maximum SID Depth: 12
        Exclude Resources: node_resc_list
          10.10.10.1
          10.10.10.8
      Dynamic (inactive)
        Metric Type: IGP,   Path Accumulated Metric: 0
    Preference: 100 (BGP ODN) (active)
      Requested BSID: dynamic
      PCC info:
        Symbolic name: bgp_c_7001_ep_100.2.1.1_discr_100
        PLSP-ID: 14044
      Constraints:
        Protection Type: protected-preferred
        Maximum SID Depth: 12
        Exclude Resources: node_resc_list
          10.10.10.1
          10.10.10.8
      Dynamic (pce 100.7.1.1) (valid)
        Metric Type: IGP,   Path Accumulated Metric: 440
          SID[0]: 41111 [Adjacency-SID, 101.1.5.1 - 101.1.5.2]
          SID[1]: 21600 [Prefix-SID, 100.6.1.1]
          SID[2]: 41600 [Adjacency-SID, 101.2.6.2 - 101.2.6.1]
  Attributes:
    Binding SID: 51679
    Forward Class: Not Configured
    Steering labeled-services disabled: yes
    Steering BGP disabled: no
    IPv6 caps enable: yes
    Invalidation drop enabled: no
    Max Install Standby Candidate Paths: 0
```

Enable Strict Disjointness for SR-TE policies in the PCE

Table 4: Feature History Table

Feature Name	Release Information	Feature Description
Enable Strict Disjointness for SR-TE Policies in the PCE	Release 24.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC:K100])(select variants only*). Enabling strict disjointness for Segment Routing Traffic Engineering (SR-TE) policies in the Path Computation Element (PCE) ensures that computed paths are completely disjoint, enhancing network resiliency. This feature is crucial for scenarios requiring high availability, as it prevents shared risk link groups by ensuring paths do not overlap, thus safeguarding against simultaneous path failures. *Previously this feature was supported on Q200 and Q100. It is now extended to Cisco 8712-MOD-M routers.

Feature Name	Release Information	Feature Description
Enable Strict Disjointness for SR-TE Policies in the PCE	Release 24.1.1	You can now enforce strict disjoint constraints for Label Switched Paths (LSPs) and minimize the risk of a single point of failure that affects multiple LSPs. With this feature, if disjoint paths cannot be found for LSPs, the Path Computation Element (PCE) does not return any path. Previously, enforcing strict disjoint constraints for disjoint paths for LSPs was not possible. The feature introduces these changes: **CLI:** The **fallback disable** keyword is introduced in the **segment-routing traffic-eng policy** and **segment-routing traffic-eng on-demand color** commands. **YANG Data Models:** • `Cisco-IOS-XR-infra-xtc-oper.yang` • `Cisco-IOS-XR-infra-xtc-agent-oper.yang` • `Cisco-IOS-XR-infra-xtc-agent-cfg.yang` See (GitHub, Yang Data Models Navigator)

When you enable fallback disable, the Path Computation Element (PCE) does not return any path for LSPs that are not disjoint if there are no disjoint candidate paths. In other words, when the PCE does not find multiple LSPs that satisfy the disjointness requirement, it returns no path. This indicates that there might be a networking issue that prevents the establishment of the desired redundancy, and you can investigate and address the underlying problem.

Without the option, the PCE can relax disjointness by applying an objective function or use a local policy when no objective function is requested.

Enforcing strict disjointness ensures that LSPs do not share any common links or nodes along their paths, minimizing the risk of a single point of failure affecting multiple LSPs. The feature is critical to maintain network reliability and ensures that traffic is properly routed in the event of a network failure.

Configure Strict Disjointness in the PCE

Enable strict disjointness for ODN SR-TE policies

Perform the following steps to enable strict disjointness for ODN SR-TE policies:

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

Running Configuration

```
segment-routing
 traffic-eng
  on-demand color 4
   dynamic
    disjoint-path group-id 1 type node fallback disable
   !
  !
 !
!
```

Enable strict disjointness for SR-TE policies

Perform the following steps to enable strict disjointness for SR-TE policies:

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

Running Configuration

```
segment-routing
 traffic-eng
  policy foo
   color 1 end-point ipv4 10.10.10.1
   candidate-paths
    preference 100
     dynamic
      pcep
      !
      metric
       type latency
      !
     !
     constraints
      disjoint-path group-id 1 type node fallback disable
     !
    !
   !
  !
 !
!
```

Verification

```
Router#sh pce association
Wed Mar 13 14:38:27.173 PDT
```

```
PCE's association database:
---------------------
Association: Type Node-Disjoint, Group 1, Strict
 Associated LSPs:
  LSP[0]:
    PCC 10.10.10.1, tunnel name cfg_foo_discr_100,  PLSP ID 4, tunnel ID 8, LSP ID 2,
Configured on PCC
  LSP[1]:
    PCC 10.10.10.1, tunnel name bgp_c_4_ep_10.10.10.2_discr_100,  PLSP ID 1, tunnel ID 5,
LSP ID 3, Configured on PCC
 Status: Satisfied
```

Anycast SID-Aware Path Computation

Table 5: Feature History Table

Feature Name	Release Information	Feature Description
Anycast SID-Aware Path Computation	Release 24.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100]) (select variants only*) This feature allows the SR-TE head-end or SR-PCE to compute a path that is encoded using Anycast prefix SIDs of nodes along the path. * This feature is supported on Cisco 8712-MOD-M routers.

This feature allows the SR-TE head-end or SR-PCE to compute a path that is encoded using Anycast prefix SIDs of nodes along the path.

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

Note For information on configuring Anycast SID, see Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface and Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface.

This example shows how Anycast SIDs are inserted into a computed SID list.

The following figure shows 3 isolated IGP domains without redistribution and without BGP 3107. Each Area Border Router (ABR) 1 through 4 is configured with a node SID. ABRs 1 and 2 share Anycast SID 16012 and ABRs 3 and 4 share Anycast SID 16034.
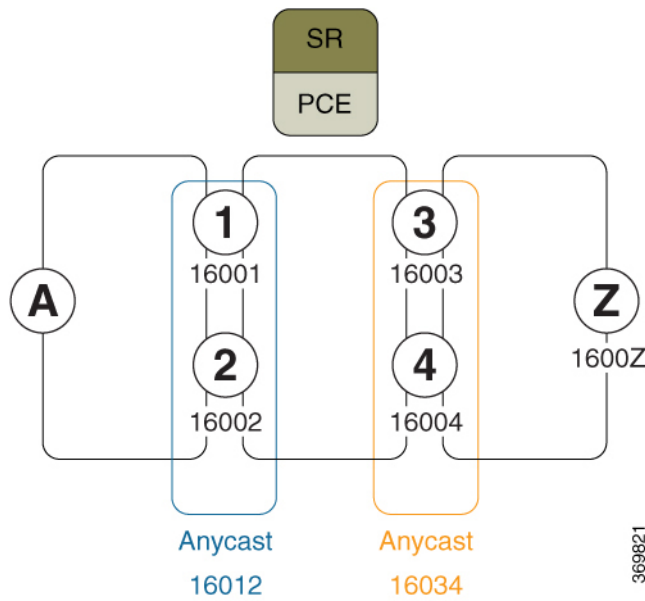
Consider the case where nodes A and Z are provider edge (PE) routers in the same VPN. Node A receives a VPN route with BGP next-hop to node Z. Node A resolves the SR path to node Z based on ODN behaviors with delegation of path computation to SR-PCE.

Before considering Anycast SIDs, the head-end router or SR-PCE computes the SID list.
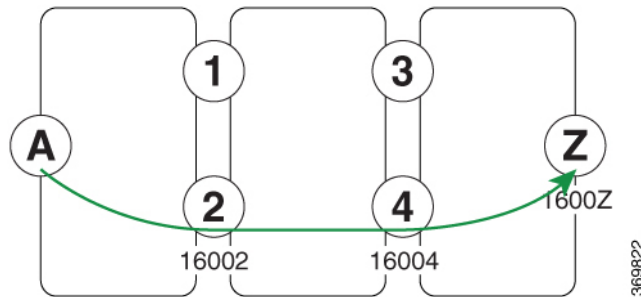
Assume that the computed path from node A to node Z traverses node 2 and node 4. This translates to SID list {16002, 16004, 1600Z} when node SIDs are leveraged to encode the path.

When an Anycast SID-aware path is requested, the path computation algorithm performs the following:

- **Path Computation**—Computes the path according to optimization objectives and constraints

- **Path Encoding**—Encodes the path in a SID list leveraging node-SIDs and adj-SIDs as applicable

- **Anycast SID Replacement**—Reiterates the original SID list by replacing node SIDs with Anycast SIDs present on the nodes along the computed path.

 If a node has multiple Anycast SIDs, the algorithm considers them according to their weights.

- **Optimality Validation**—The new paths are validated against the original optimization objectives and constraints (maintain same cumulative metric as original SID list and do not violate path constraints).

- **Anycast SID Promotion**—If the optimality validation is successful, then the Anycast-encoded SID list is signaled and instantiated in the forwarding.

The following figure depicts cumulative metrics between nodes in the network.

Under these conditions, the optimality check is met, and therefore, the Anycast-encoded SID list from node A to node Z is {16012,16034,1600Z}.
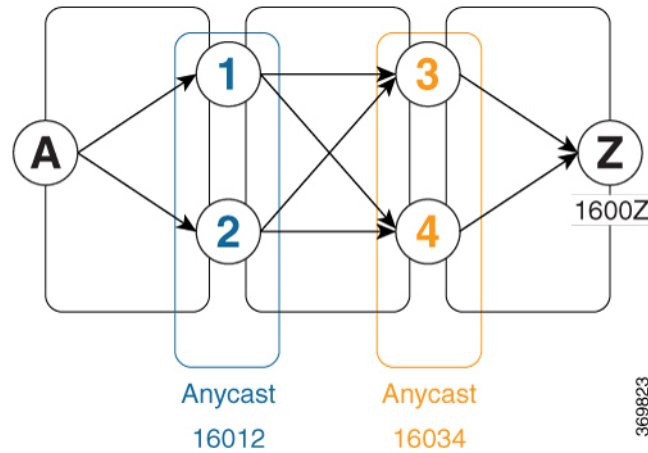
The Anycast SID aware path computation also provides resiliency. For example, if one of the ABRs (in this case, ABR 1) becomes unavailable or unreachable, the path from node A to node Z {16012,16034,1600Z} will still be valid and usable.
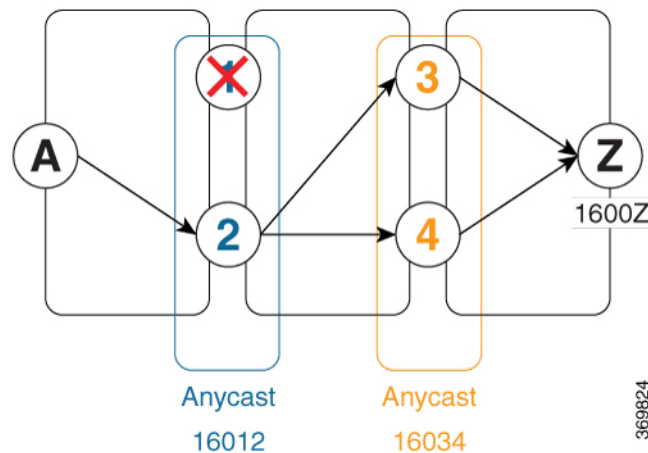
Configuration Examples

1. Configure Prefix SIDs on the ABR nodes.

    a. Configure each node with a node SID.

    b. Configure each group of nodes with a shared Anycast SID.

    See Configuring a Prefix-SID on the IS-IS Enabled Loopback Interface and Configuring a Prefix-SID on the OSPF-Enabled Loopback Interface.

2. Configure SR policies to include Anycast SIDs for path computation using the **anycast-sid-inclusion** command.

This example shows how to configure a local SR policy to include Anycast SIDs for PCC-initiated path computation at the head-end router:

```
Router(config)# segment-routing traffic-eng
Router(config-sr-te)# policy FOO
Router(config-sr-te-policy)# color 10 end-point ipv4 1.1.1.10
Router(config-sr-te-policy)# candidate-paths
Router(config-sr-te-policy-path)# preference 100
Router(config-sr-te-policy-path-pref)# dynamic
Router(config-sr-te-pp-info)# anycast-sid-inclusion
```

Running Configuration

Use the **anycast-sid-inclusion** command to request Anycast SID-aware path computation for the following SR policy types:

- Local SR policy with PCC-initiated path computation at the head-end router:

```
segment-routing
  traffic-eng
    policy FOO
      color 10 end-point ipv4 1.1.1.10
      candidate-paths
        preference 100
          dynamic
            anycast-sid-inclusion
```

- Local SR policy with PCC-initiated/PCE-delegated path computation at the SR-PCE:

```
segment-routing
 traffic-eng
  policy BAR
   color 20 end-point ipv4 1.1.1.20
    candidate-paths
     preference 100
      dynamic
        pcep
        anycast-sid-inclusion
```

- On-demand SR policies with a locally computed dynamic path at the head-end, or centrally computed dynamic path at the SR-PCE:

```
segment-routing
  traffic-eng
   on-demand color 10
    dynamic
     anycast-sid-inclusion
```

- On-demand SR policies with centrally computed dynamic path at the SR-PCE:

```
segment-routing
 traffic-eng
  on-demand color 20
   dynamic
    pcep
    anycast-sid-inclusion
```

Weighted Anycast SIDs

Table 6: Feature History Table

Feature Name	Release Information	Feature Description
Weighted Anycast SID-Aware Path Computation	Release 7.3.1	This feature extends Anycast SIDs with weighted nodes. Weighted Anycast nodes advertise a cost (weight) along with the Anycast SID. Traffic is then distributed according to the weights. Weighted Anycast SIDs allow for highly available paths with node redundancy and path optimality that provide Fast Re-Route (FRR) for node failure of service provider edge (PE) routers and ABR/ASBRs nodes in multi-domain networks.

Weighted Anycast nodes advertise a cost along with the Anycast SID. This cost serves as a weight. The native SR path computation algorithms are augmented to compute optimum paths relying on Weighted Anycast SIDs during path encoding. Traffic to the SID is then distributed according to the weights.

The following example shows how node SID, Anycast SID, and Weighted Anycast SID are applied on node 1:

```
router isis 1
 interface Loopback0
  address-family ipv4 unicast
   prefix-sid absolute 16001 // Node SID
  !
 !
 interface Loopback1
  prefix-attributes anycast
  address-family ipv4 unicast
   prefix-sid absolute 16012 // Anycast SID - (prefer node 1 or 2)
  !
 !
 interface Loopback2
  prefix-attributes anycast
  address-family ipv4 unicast
   weight 1
   prefix-sid absolute 17012 // Weighted Anycast SID (prefer node 1)
  !
 !
 interface Loopback3
  prefix-attributes anycast
  address-family ipv4 unicast
   weight 100000
   prefix-sid absolute 18012 // Weighted Anycast SID (prefer node 2)
  !
 !
!
```

Configure the Shortest Path for Disjoint Candidate Paths

Table 7: Feature History Table

Feature Name	Release Information	Feature Description
Configure the Shortest Path for Disjoint Candidate Paths	Release 24.1.1	You can now configure the available disjoint paths for Label Switched Paths (LSPs) to prefer the shortest path between two points in the network. This configuration ensures that traffic is routed along the most efficient route in the network. Previously, you could not configure the shortest path preference for disjoint LSPs. The feature introduces these changes: **CLI:** The **shortest-path** keyword is introduced in the **policy candidate-paths constraints disjoint-path** command. **YANG Data Models:** • `Cisco-IOS-XR-infra-xtc-oper.yang` • `Cisco-IOS-XR-infra-xtc-agent-oper.yang` • `Cisco-IOS-XR-infra-xtc-agent-cfg.yang` See (GitHub, Yang Data Models Navigator)

The configuration enables the available disjoint paths to indicate preference for the shortest path between two points in the network. This way the traffic is contained within specific network planes as per the affinity constraints while still provisioning Label Switched Paths (LSPs) that are diverse from each other to meet the path disjointness requirement.

In earlier releases, if path disjointness was configured for LSPs, the Path Computation Element (PCE) prioritized finding disjoint paths for the LSPs over adhering to the affinity constraints.

Configure the shortest path for disjoint candidate paths

Perform the following task to indicate the disjoint path preference for the shortest path in the network:

```
Router(config)#segment-routing traffic-eng
Router(config-sr-te)#policy dynamic_pcep_policy_disjoint
```

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

Running Configuration

```
!
segment-routing
 traffic-eng
   policy dynamic_pcep_policy
   candidate-paths
    preference 100
      constraints
       disjoint-path group-id 1 type link shortest-path


     !
    !
   !
  !
 !
!
```

Verification

```
Router#show pce lsp name cfg_dynamic_pcep_policy_disjoint_discr_100 detail
Output received:
Wed Aug 30 18:33:57.807 UTC

PCE's tunnel database:
----------------------
PCC 100.1.1.1:

Tunnel Name: cfg_dynamic_pcep_policy_disjoint_discr_100
Color: 30115
Interface Name: srte_c_30115_ep_100.2.1.1
 LSPs:
  LSP[0]:
    source 100.1.1.1, destination 100.2.1.1, tunnel ID 9031, LSP ID 24
    State: Admin up, Operation active
    Setup type: Segment Routing
    Binding SID: 30115
    Maximum SID Depth: 12
    Preference: 100
    Bandwidth: requested 0 kbps, applied 0 kbps
    Protection type: protected-preferred
    Prefix-SID algorithm: 0 (set: 0)
    PCEP information:
      PLSP-ID 0x36e9, flags: D:1 S:0 R:0 A:1 O:2 C:0
    LSP Role: Exclude LSP
    State-sync PCE: None
    PCC: 100.1.1.1
    LSP is subdelegated to: None
    Reported path:
      Metric type: TE, Accumulated Metric 30
       SID[0]: Node, Label 21200, Address 100.2.1.1
    Computed path: (Local PCE)
      Computed Time: Wed Aug 30 18:31:12 UTC 2023 (00:02:45 ago)
      Metric type: TE, Accumulated Metric 30
       SID[0]: Node, Label 21200, Address 100.2.1.1
    Reverse path: (Local PCE)
      None
      Computed Time: Wed Aug 30 18:31:12 UTC 2023 (00:02:45 ago)
    Recorded path:
      None
```

```
        Disjoint Group Information:
          Type Link-Disjoint, Group 1 (SP)
        SR Policy Association Group:
          Color: 30115, Endpoint: 100.2.1.1
          Policy Name: srte_c_30115_ep_100.2.1.1
          Preference: 100
          CP Name: dynamic_pcep_policy_disjoint
```

PCC-initiated Policies Delegated to PCE

Policies are created on PCC and a path is requested from PCE. PCEP connection must be up and functional for PCE to perform the path computation.

Configure PCC-initiated Policies Delegated to PCE

Perform the following task to configure the SR policies at PCC and delegate them to PCE after establishing PCEP connection.

Configuration Example

```
Router # configure
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy local_dynamic_disj_1
Router(config-sr-te-policy)# binding-sid mpls 19002
Router(config-sr-te-policy)# color 19002 end-point ipv4 192.168.0.1
Router(config-sr-te-policy)# candidate-paths
Router(config-sr-te-policy-path)# preference 100
Router(config-sr-te-policy-path-preference)# dynamic
Router(config-sr-te-pp-info)# pcep
Router(config-sr-te-path-pcep)# exit
Router(config-sr-te-pp-info)# metric
Router(config-sr-te-path-metric)# type te
Router(config-sr-te-path-metric)# exit
Router(config-sr-te-pp-info)# exit
Router(config-sr-te-policy-path-preference)# constraints
Router(config-sr-te-path-pref-const)# affinity
Router(config-sr-te-path-pref-const-aff)# include-any
Router(config-sr-te-path-pref-const-aff-rule)# name blue
Router(config-sr-te-path-pref-const-aff-rule)# name green
Router(config-sr-te-path-pref-const-aff-rule)# commit
```

Running Configuration

```
Router # show running-config segment-routing traffic-eng policy local_dynamic_disj_1
segment-routing
 traffic-eng
  policy local_dynamic_disj_1
   binding-sid mpls 19002
   color 19002 end-point ipv4 192.168.0.1
   candidate-paths
    preference 100
     dynamic
      pcep
      !
      metric
       type te
      !
```

```
         !
     constraints
      affinity
       include-any
        name blue
        name green
       !
      !
     !
    !
   !
  !
 !
```

SR-PCE IPv4 Unnumbered Interface Support

This feature allows IPv4 unnumbered interfaces to be part of an SR-PCE topology database.

An unnumbered IPv4 interface is not identified by its own unique IPv4 address. Instead, it is identified by the router ID of the node where this interfaces resides and the local SNMP index assigned for this interface.

This feature provides enhancements to the following components:

- IGPs (IS-IS and OSPF):

  - Support the IPv4 unnumbered interfaces in the SR-TE context by flooding the necessary interface information in the topology

- SR-PCE:

Note SR-PCE and path computation clients (PCCs) need to be running Cisco IOS XR software release 7.5.2 or later.

  - Compute and return paths from a topology containing IPv4 unnumbered interfaces.

  - Process reported SR policies from a head-end router that contain hops with IPv4 unnumbered adjacencies.

    PCEP extensions for IPv4 unnumbered interfaces adhere to IETF RFC8664 "PCEP Extensions for Segment Routing" (https://datatracker.ietf.org/doc/rfc8664/). The unnumbered hops use a Node or Adjacency Identifier (NAI) of type 5. This indicates that the segment in the explicit routing object (ERO) is an unnumbered adjacency with an IPv4 ID and an interface index.

- SR-TE process at the head-end router:

  - Compute its own local path over a topology, including unnumbered interfaces.

  - Process PCE-computed paths that contain hops with IPv4 unnumbered interfaces.

  - Report a path that contains hops with IPv4 unnumbered interfaces to the PCE.

Configure SR-PCE IPv4 Unnumbered Interface

This section provides the commands to configure the SR-PCE IPv4 unnumbered interface.

Configuration Example

The following example shows how to configure an IPv4 unnumbered interface:

```
Router # configure
Router(config)# interface GigabitEthernet0/0/0/0
Router(config-if)# ipv4 point-to-point
Router(config-if)# ipv4 unnumbered Loopback0
```

To bring up the IPv4 unnumbered adjacency under the IGP, configure the link as point-to-point under the IGP configuration. The following example shows how to configure the link as point-to-point under the IGP configuration:

```
Router # configure
Router(config)# router ospf one
Router(config-ospf)# area 0
Router(config-ospf-ar)# interface GigabitEthernet0/0/0/0
Router(config-ospf-ar-if)# network point-to-point
```

Verification

Use the **show ipv4 interface** command to display information about the interface:

```
Router # show ipv4 interface GigabitEthernet0/0/0/0 brief
Tue Apr  2 12:59:53.140 EDT
Interface                IP-Address      Status          Protocol
GigabitEthernet0/0/0/0   192.168.0.1     Up              Up
```

This interface shows the IPv4 address of Loopback0.

Use the **show snmp interface** command to find the SNMP index for this interface:

```
Router# show snmp interface
Tue Apr  2 13:02:49.190 EDT
ifName : Null0                 ifIndex : 3
ifName : Loopback0             ifIndex : 10
ifName : GigabitEthernet0/0/0/0  ifIndex : 6
```

The interface is identified with the pair (IPv4:192.168.0.1, index:6).

Use the **show ospf neighbor** command to display the adjacency:

```
Router# show ospf neighbor gigabitEthernet 0/0/0/0 detail
…
Neighbor 192.168.0.4, interface address 192.168.0.4
   In the area 0 via interface GigabitEthernet0/0/0/0
   Neighbor priority is 1, State is FULL, 6 state changes
   …
   Adjacency SIDs:
       Label: 24001,    Dynamic, Unprotected
   Neighbor Interface ID: 4
```

The output of the **show pce ipv4 topology** command is enhanced to display the interface index instead of the IP address for unnumbered interfaces:

```
Router# show pce ipv4 topology
…
  Link[2]: unnumbered local index 6, remote index 4
```

```
  Local node:
    OSPF router ID: 192.168.0.1 area ID: 0 ASN: 0
  Remote node:
    TE router ID: 192.168.0.4
    OSPF router ID: 192.168.0.4 area ID: 0 ASN: 0
  Metric: IGP 1, TE 1, Latency 1 microseconds
  Bandwidth: Total 125000000 Bps, Reservable 0 Bps
  Admin-groups: 0x00000000
  Adj SID: 24001 (unprotected)
```

The output of **show pce lsp detail** command includes unnumbered hops:

```
Router# show pce lsp detail
…
  Reported path:
    Metric type: TE, Accumulated Metric 3
      SID[0]: Adj unnumbered, Label 24001, local 192.168.0.1(6), remote 192.168.0.4(4)
      SID[1]: Adj unnumbered, Label 24002, local 192.168.0.4(7), remote 192.168.0.3(7)
      SID[2]: Adj unnumbered, Label 24000, local 192.168.0.3(5), remote 192.168.0.2(5)
  Computed path: (Local PCE)
    Computed Time: Wed Apr 03 11:01:46 EDT 2019 (00:01:06 ago)
    Metric type: TE, Accumulated Metric 3
      SID[0]: Adj unnumbered, Label 24001, local 192.168.0.1(6), remote 192.168.0.4(4)
      SID[1]: Adj unnumbered, Label 24002, local 192.168.0.4(7), remote 192.168.0.3(7)
      SID[2]: Adj unnumbered, Label 24000, local 192.168.0.3(5), remote 192.168.0.2(5)
```

Inter-Domain Path Computation Using Redistributed SID

Table 8: Feature History Table

| Feature Name	Release Information	Feature Description
Inter-Domain Path Computation Using Redistributed SID	Release 24.4.1	Introduced in this release on: Fixed Systems (8700 [ASIC: K100]) (select variants only*) This feature adds new functionality to the SR-PCE that enables it to compute a path for remote non-SR end-point device distributed by BGP-LU. * This feature is supported on Cisco 8712-MOD-M routers.

A Path Computation Element (PCE) computes SR-TE paths based on SR topology database that stores connectivity, state, and TE attributes of SR network nodes and links. BGP Labeled Unicast (BGP-LU) provides MPLS transport across IGP boundaries by advertising loopbacks and label binding of impact edge and border routers across IGP boundaries.

This feature adds new functionality to the SR-PCE that enables it to compute a path for remote non-SR end-point device distributed by BGP-LU.

The remote end-point device in the BGP-LU domain is unknown to the SR-PCE. For the SR-PCE to know about the end-point device, the gateway ABR/ASBR learns the end-point prefix via BGP-LU. The prefix is

then redistributed to SR-PCE topology database from the gateway ABR/ASBR. SR-PCE then can compute the best path from the head-end device to the selected gateway router.

The following topology shows an SR domain and a BGP-LU domain, with a gateway ABR/ASBR between the two domains.
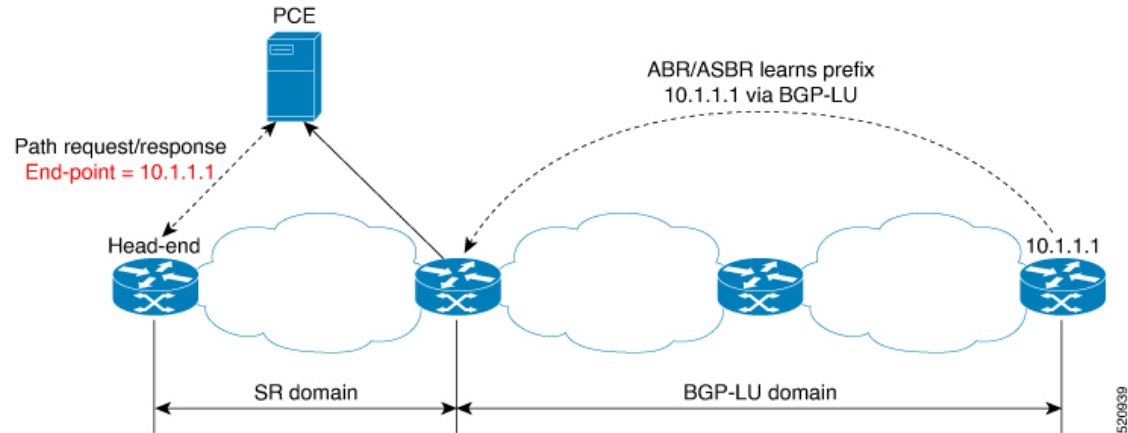
1. The gateway ABR/ASBR is configured with BGP/IGP helper to learn the remote prefix through BGP-LU and redistribute the remote prefix to the IGP helper, then to SR-PCE.

2. The SR-PCE selects the best gateway node to BGP-LU domain and computes the path to reach the remote prefix through the gateway node.

3. The head-end device in the SR domain requests a path to the remote destination and signals the SR profile interworking with the BGP-LU domain.

The BGP-LU prefix advertisement to SR-PCE Traffic Engineer Database (TED) is done by creating an IGP helper on the ABR/ASBR to redistribute BGP-LU prefix information to IGP. IGP then sends the prefix information to the SR-PCE via BGP-LS.

If there are multiple ABR/ASBRs advertising the same remote BGP-LU prefix, the SR-PCE selects the best gateway node to the BGP-LU domain using the accumulative metric from the head-end device to the gateway and the advertised metric from the gateway to the destination.
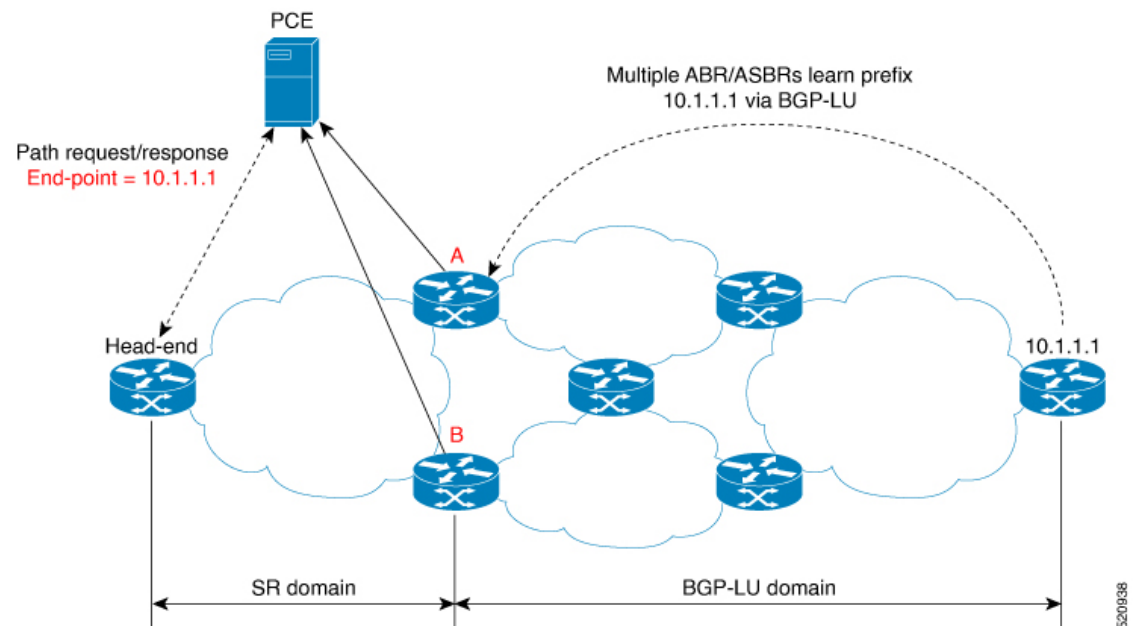
Example: Inter-Domain Path Computation Using Redistributed SID

The following examples show the configurations for the IGP helper, BGP-LU, and proxy BGP-SR:

Configuration on the End-Point Device

Configure the end-point device to allocate a label for the BGP-LU prefix on the end-point device:

```
router bgp 3107
bgp router-id 1.0.0.8
address-family ipv4 unicast
  network 1.0.0.8/32 route-policy bgplu-com
  allocate-label all

route-policy bgplu-com
  set community (65002:999)
end-policy
```

Configuration on the Gateway ABR/ASBR

1. Configure the remote prefix set and create the route policy for the BGP-LU domain:

```
prefix-set bgplu
  1.0.0.7/32,
  1.0.0.8/32,
  1.0.0.101/32,
  1.0.0.102/32
end-set
!

route-policy bgp2isis
  if destination in bgplu then
    pass
  else
    drop
  endif
```

```
end-policy
!
end
```

2. Configure the helper IGP instance on the Loopback interface:

```
router isis 101
 is-type level-2-only
 net 49.0001.0000.1010.1010.00
 distribute link-state instance-id 9999
 nsf cisco
 nsf lifetime 120
 address-family ipv4 unicast
  metric-style wide
  maximum-paths 64
  router-id Loopback10
  redistribute bgp 3107 metric 200 route-policy bgp2isis
  segment-routing mpls sr-prefer
!
interface Loopback10 >>> this loopback is for gateway SR-TE node-id
  passive
  address-family ipv4 unicast
   prefix-sid index 2001 explicit-null
```

3. Configure the gateway proxy BGP-SR and SR Mapping Server to allocate SR labels:

```
router bgp 3107
 address-family ipv4 unicast
  segment-routing prefix-sid-map
  allocate-label all

segment-routing
global-block 16000 23999
mapping-server
  prefix-sid-map
   address-family ipv4
    1.0.0.7/32 2007
    1.0.0.8/32 2008
    1.0.0.101/32 2101
    1.0.0.102/32 2102
```