

# 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 additional 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.


- About SR-PCE, on page 1
- Configure SR-PCE, on page 2
- PCE-Initiated SR Policies, on page 5
- ACL Support for PCEP Connection, on page 7


## About SR-PCE

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. 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.

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.

# 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** *size*
6. **password** {**clear** | **encrypted**} *password*
7. **segment-routing** {**strict-sid-only** | **te-latency**}
8. **timers**
9. **keepalive** *time*
10. **minimum-peer-keepalive** *time*
11. **reoptimization** *time*
12. **exit**

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **configure**<br>**Example:**<br>`RP/0/RP0/CPU0:router# configure` | Enters global configuration mode. |
| **Step 2** | **pce**<br>**Example:**<br>`RP/0/RP0/CPU0:router(config)# pce` | Enables PCE and enters PCE configuration mode. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 3** | **address ipv4** *address*<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce)# address ipv4 192.168.0.1` | Configures a PCE IPv4 address. |
| **Step 4** | **state-sync ipv4** *address*<br>**Example:**<br>`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** *size*<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce)# tcp-buffer size 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*<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce)# password encrypted pwd1` | Enables TCP MD5 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. |
| **Step 7** | **segment-routing** {**strict-sid-only** \| **te-latency**}<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce)# segment-routing strict-sid-only` | Configures the segment routing algorithm to use strict SID or TE latency.<br>**Note** This setting is global and applies to all LSPs that request a path from this controller. |
| **Step 8** | **timers**<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce)# timers` | Enters timer configuration mode. |
| **Step 9** | **keepalive** *time*<br>**Example:**<br>`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. |

| | Command or Action | Purpose |
|---|---|---|
| **Step 10** | **minimum-peer-keepalive** *time*<br>**Example:**<br>`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 11** | **reoptimization** *time*<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce-timers)# reoptimization 600` | Configures the re-optimization timer. The default timer is 1800 seconds. |
| **Step 12** | **exit**<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce-timers)# exit` | Exits timer configuration mode and returns to PCE configuration mode. |

# Configure the Disjoint Policy (Optional)

This task explains how to configure the SR-PCE to 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.

### SUMMARY STEPS

1. **disjoint-path**
2. **group-id** *value* **type** {**link** | **node** | **srlg** | **srlg-node**} [**sub-id** *value*]
3. **strict**
4. **lsp** {**1** | **2**} **pcc ipv4** *address* **lsp-name** *lsp_name* [**shortest-path**]

### DETAILED STEPS

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | **disjoint-path**<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce)# disjoint-path` | Enters disjoint configuration mode. |
| **Step 2** | **group-id** *value* **type** {**link** | **node** | **srlg** | **srlg-node**} [**sub-id** *value*]<br>**Example:** | Configures the disjoint group ID and defines the preferred level of disjointness (the type of resources that should not be shared by the two paths): |

| | Command or Action | Purpose |
|---|---|---|
| | `RP/0/RP0/CPU0:router(config-pce-disjoint)# group-id 1 type node sub-id 1` | • **link**—Specifies that links are not shared on the computed paths.<br>• **node**—Specifies that nodes are not shared on the computed paths.<br>• **srlg**—Specifies that links with the same SRLG value are not shared on the computed paths.<br>• **srlg-node**—Specifies that SRLG and nodes are not shared on the computed paths.<br>If a pair of paths that meet the requested disjointness level cannot be found, then the paths will automatically fallback to a lower level:<br>• If the requested disjointness level is SRLG or node, then link-disjoint paths will be computed.<br>• 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. |
| **Step 3** | **strict**<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce-disjoint)# strict` | (Optional) 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 terminates and no new path is provided. The existing path is not modified. |
| **Step 4** | **lsp** {**1** \| **2**} **pcc ipv4** *address* **lsp-name** *lsp_name* [**shortest-path**]<br>**Example:**<br>`RP/0/RP0/CPU0:router(config-pce-disjoint)# lsp 1 pcc ipv4 192.168.0.1 lsp-name rtrA_t1 shortest-path`<br>`RP/0/RP0/CPU0:router(config-pce-disjoint)# lsp 2 pcc ipv4 192.168.0.5 lsp-name rtrE_t2` | Adds LSPs to the disjoint group.<br>The **shortest-path** keyword 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 the first LSP specified. |

# PCE-Initiated SR Policies

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.

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).

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.

### Configuration Example: PCE-Initiated SR Policy with Explicit SID List

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

1. Enter PCE configuration mode.
2. Create the segment list.



**Note** When configuring an explicit path using IP addresses of intermediate links, the SR-TE process selects either the protected or the unprotected Adj-SID of the link, depending on the order in which the Adj-SIDs were received.

3. Create the policy.

```
/* 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 10 address ipv4 1.1.1.2
Router(config-pce-sr-te-sl)# index 20 address ipv4 10.2.3.2
Router(config-pce-sr-te-sl)# index 30 address ipv4 1.1.1.4
Router(config-pce-sr-te-sl)# exit

/* Create the SR-TE policy */
Router(config-pce-sr-te)# peer ipv4 1.1.1.1
Router(config-pce-sr-te)# policy P1
Router(config-pce-sr-te-policy)# color 2 end-point ipv4 2.2.2.2
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)# commit
Router(config-pce-sr-te-pp-info)# end
Router(config)#
```

### Running Config

```
pce
```

```
segment-routing
 traffic-eng
  segment-list name addr2a
   index 1 address ipv4 14.14.14.4
  !
  peer ipv4 1.1.1.1
   policy P1
    color 2 end-point ipv4 2.2.2.2
    candidate-paths
     preference 50
      explicit segment-list addr2a
     !
    !
```

# ACL Support for PCEP Connection

PCE protocol (PCEP) (RFC5440) is a client-server model running over TCP/IP, where the server (PCE) opens a port and the clients (PCC) initiate connections. After the peers establish a TCP connection, they create a PCE session on top of it.

The ACL Support for PCEP Connection feature provides a way to protect a PCE server using an Access Control List (ACL) to restrict IPv4 PCC peers at the time the TCP connection is created based on the source address of a client. When a client initiates the TCP connection, the ACL is referenced, and the client source address is compared. The ACL can either permit or deny the address and the TCP connection will proceed or not.

Refer to the Implementing Access Lists and Prefix Lists chapter in the *IP Addresses and Services Configuration Guide for Cisco NCS 5500 Series Routers* for detailed ACL configuration information.

To apply an ACL to the PCE, use the **pce peer-filter ipv4 access-list** *acl_name* command.

The following example shows how to configure an ACL and apply it to the PCE:

```
pce
 address ipv4 1.1.1.5
 peer-filter ipv4 access-list sample-peer-filter
!
ipv4 access-list sample-peer-filter
 10 permit ipv4 host 1.1.1.6 any
 20 permit ipv4 host 1.1.1.7 any
 30 deny ipv4 any any
!
```