# PLE Service Configuration

This chapter describes the Private Line Emulation (PLE) service and the configuration procedures.

# Understanding Private Line Emulation

Private Line Emulation (PLE) is a pillar of the Routed Optical Networking solution. It enables service providers and enterprises to further collapse network layers, decreasing network complexity and increasing network efficiency. PLE enables private line services to be carried over the same MPLS or Segment Routing network for non-Ethernet type services such as SONET/SDH, OTN, and Fiber Channel. PLE also supports bit-transparent Ethernet services where required.

High revenue legacy private line services exist in the network infrastructure of most service providers, often carried over a dedicated inefficient TDM OTN layer. PLE enables service providers to carry SONET/SDH, OTN, Ethernet, and Fiber Channel over a circuit-style segment routed packet network while maintaining existing service SLAs. PLE utilizes Circuit Emulation (CEM) to transparently transfer PLE client frames over MPLS or SR networks without changing the characteristics of the original signal.

### How PLE Works

Ethernet, OTN, Fiber Channel, or SONET/SDH PLE client traffic is carried on an EVPN-VPWS single homed service that is created between PLE endpoints. EVPN-VPWS signalling information is carried using BGP between the PLE circuit endpoints either through direct BGP sessions or through a BGP services route-reflector. The EVPN-VPWS pseudowire channel is set up between the endpoints when the CEM (Circuit Emulation) client interfaces are configured on each endpoint router and end-to-end transport connectivity using MPLS or Segment Routing transport is enabled.

CEM is a method through which client data can be transmitted over MPLS or Segment Routing networks in a bit-transparent manner, retaining the client L1 frame between sender and receiver. CEM over a Packet Switched Network (PSN) places the client bit streams into packet payload with appropriate pseudowire emulation headers.

The PLE initiator encapsulates the PLE client traffic and carries it over the EVPN-VPWS service running on MPLS or Segment Routing transport. The PLE terminator node extracts the bit streams from the EVPN-VPWS packets and places them onto the PLE client interface as defined by the client attribute and CEM profile.

### Supported Payloads

The following payloads are supported through PLE:

*Table 1: Supported Payloads*

| PLE Transport Type | Supported Payloads |
|---|---|
| Ethernet | 1GE and 10GE |
| OTN | OTU2 and OTU2e |
| SONET | OC-48 and OC-192 |
| SDH | STM-16 and STM64 |
| Fiber Channel | FC1, FC2, FC4, FC8, FC16, and FC32 |

The combination of payloads is also supported.

### Supported Hardware

PLE is supported on NC55-OIP-02 MPA (modular port adapter). This MPA is supported in the NC55A2-MOD-S and NC57C3-MOD-SYS routers. For more information, see Cisco NCS 5500 Modular Chassis

# Circuit-Style Segment Routing

Segment Routing provides an architecture that caters to both IP-centric transport and connection-oriented transport. IP-centric transport uses the benefits of ECMP and automated protection from TI-LFA. Connection-oriented transport, which was historically delivered over circuit-switched SONET/SDH networks, requires the following:

- End-to-end bidirectional transport that provides congruent forward and reverse paths, predictable latency, and disjointness.
- Bandwidth commitment to ensure that there is no impact on the SLA due to network load from other services.
- Monitoring and maintenance of path integrity with end-to-end 50-msec path protection.

- Persistent end-to-end paths regardless of the state of the control plane.

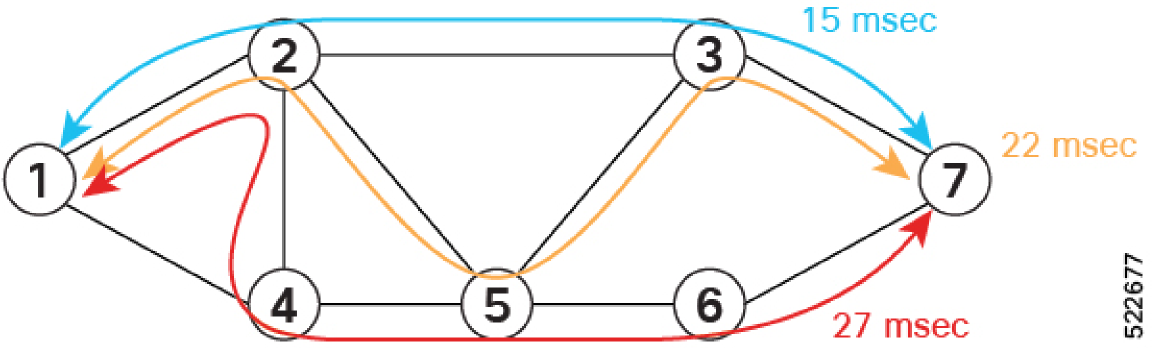
### Introduction to Circuit-Style SR Policies

Circuit-Style SR policies have the following properties:

- Guaranteed Latency over Non-ECMP Paths
- Control-Plane Independent Persistency
- Co-Routed Bidirectional Path
- Liveness Monitoring with Path Protection Switching

### Guaranteed Latency over Non-ECMP Paths

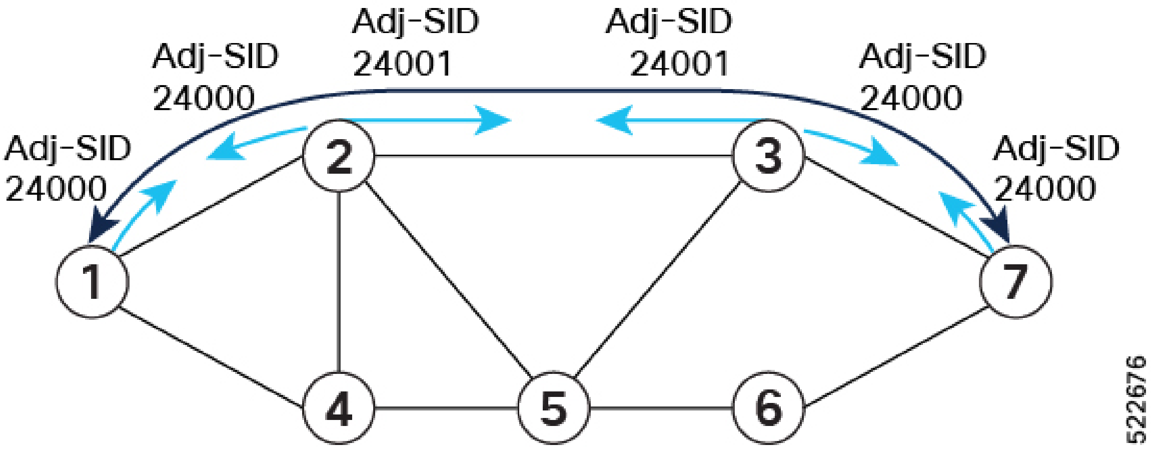
Consider the following network with three possible paths from node 1 to node 7. Of the three paths, the best end-to-end delay is provided by the blue path (1 -> 2 -> 3 -> 7). The chosen path is then encoded with adjacency SIDs corresponding to the traversed interfaces to avoid any ECMP, and therefore guarantee the latency over the path.



### Control-Plane Independent Persistency

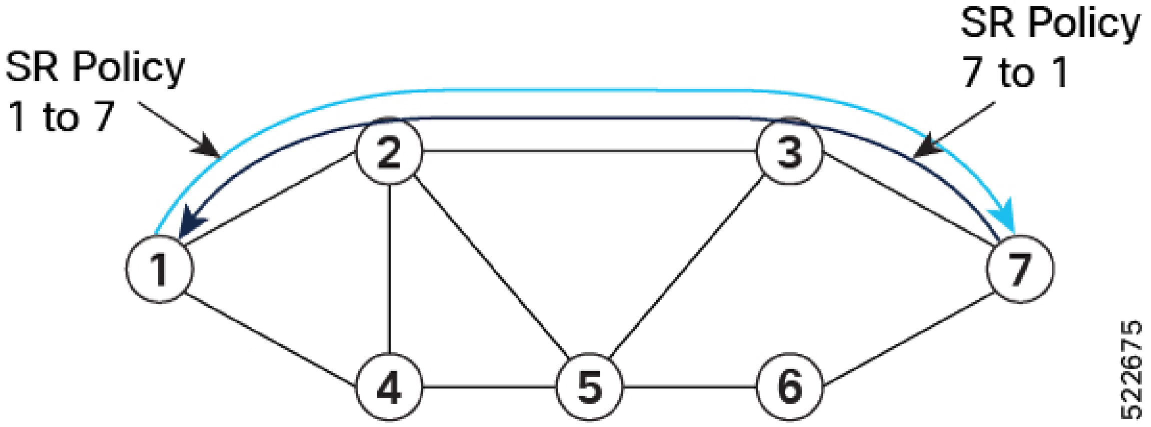
Adjacency SIDs can provide a persistent path independent from control-plane changes (such as IGP neighbor flaps), and network events (such as interface additions or interface flaps) and even the presence of IP on an interface. To achieve this, adjacency SIDs can be manually allocated to ensure persistent values, for example, after a node reload event. In addition, adjacency SIDs can be programmed as non-protected to avoid any local TI-LFA protection.

With the adjacency SIDs depicted in the following figure, the path from node 1 to node 7 is encoded with the segment list of {24000, 24001, 24000}. By manually allocating the same adjacency SID values for other direction, the path from node 7 to node 1 is encoded with the same segment list of {24000, 24001, 24000}.



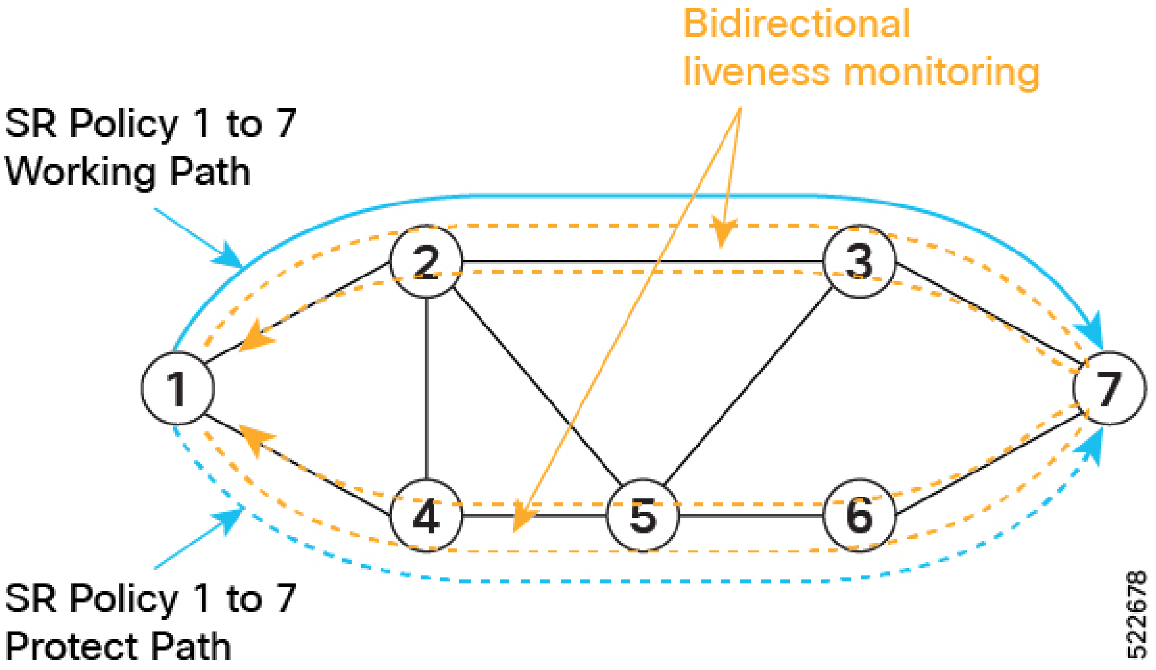
### Co-Routed Bidirectional Path

Forward and return SR policies with congruent paths are routed along the same nodes.



### Liveness Monitoring with Path Protection Switching

Bidirectional liveness monitoring on the working and protect paths ensures fast and consistent switchover, while a protect path is pre-programmed over disjoint facilities.



### Usage Guidelines and Limitations

- Candidate path (CP) behavior:
  - The working path has the highest CP preference value.
  - The protect path has the second highest CP preference value.
  - The restore path has the lowest CP preference value and is configured as "backup ineligible."
  - Paths with the same role in both directions (working, protect, and restore) must have the same CP preference value.
- Bidirectional path behavior:
  - All paths must be configured as corouted.
  - All paths with the same role in both directions (working, protect, and restore) must have the same bidirectional association ID value.
  - The bidirectional association ID value must be globally unique.

### Guaranteed Bandwidth

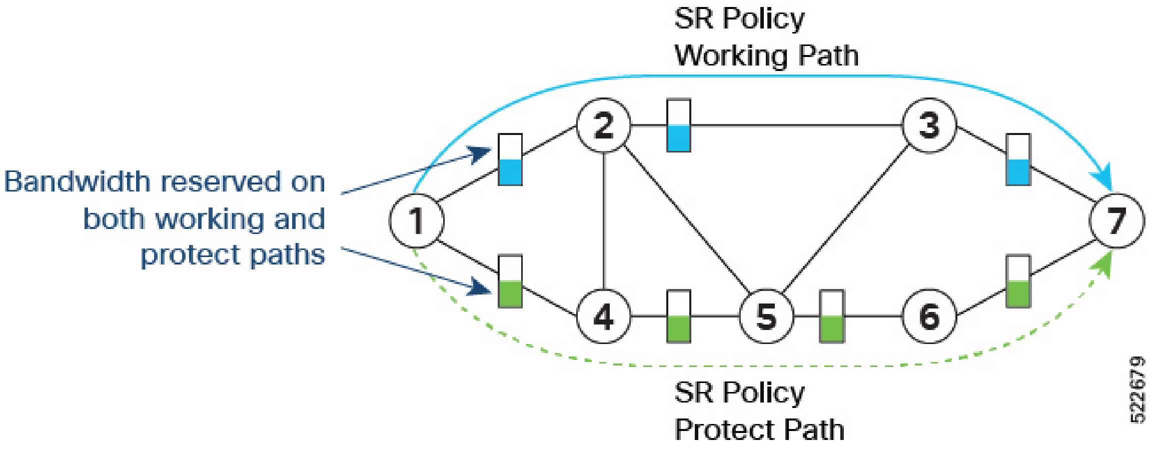
Most services carried over the CS-SR policy are constant-rate traffic streams. Any packet loss due to temporary congestion leads to bit errors at the service layer. Therefore, bandwidth must be managed very tightly and guaranteed to the services mapped to CS-SR policies.

A centralized controller manages the bandwidth reservation. The controller maintains the reserved bandwidth on each link based on the requested bandwidth:

- Monitors amount of traffic

- Uses knowledge of the active path used by the policy
- Computes the per-link reservable bandwidth accordingly

Bandwidth is reserved on both the working and protect paths.



# Configure PLE over EVPN-VPWS

Configuring PLE over the EVPN-VPWS service procedure involves the following tasks:

1. Enable Frequency Synchronization, on page 6
2. Configure Optics Controller with Payloads, on page 7
3. Configure QoS Policy, on page 8
4. Configure CEM Interface for L2VPN Service, on page 8
5. Configure Performance Measurement, on page 9
6. Configure Segment Routing Adjacency SIDs, on page 9
7. Define Paths Between PLE Endpoints, on page 10
8. Configure Static Circuit-Style SR-TE Policies, on page 12
9. Configure PLE over EVPN-VPWS, on page 22

See the Private Line Emulation over EVPN-VPWS Single Homed section for pre-requisites and other details.

# Enable Frequency Synchronization

Use this task to synchronize the clocks between PE1 and PE2.

**Enable Frequency Synchronization on PLE-PE1**

The following exaple enables frequency synchronization on PLE-PE1.

```
Router(config)# frequency synchronization
Router(config-freqsync)# quality itu-t option 1
Router(config-freqsync)# exit
Router(config)# interface TwentyFiveGigE0/0/0/24
Router(config-if)# frequency synchronization
Router(config-if-freqsync)# quality transmit exact itu-t option 1 PRC
```

Use the **show frequency synchronization interfaces** command to verify that the clock is transmitted.

**Enable Frequency Synchronization on PLE-PE2**

The following exanple enables frequency synchronization on PLE-PE2.

```
Router(config)# frequency synchronization
Router(config-freqsync)# quality itu-t option 1
Router(config-freqsync)# exit
Router(config)# interface TwentyFiveGigE0/0/0/32
Router(config-if)# frequency synchronization
Router(config-if-freqsync)# selection input
Router(config-if-freqsync)# priority 1
Router(config-if-freqsync)# wait-to-restore 0
```

Use the **show frequency synchronization selection** command to verify whether PLE-PE2 is LOCKED to PLE-PE1's clock.

# Configure Optics Controller with Payloads

Use this task to configure the Optics controller with different payloads.

Each port on the NC55-OIP-2 (PID of PLE Modular Port Adapter) can be independently configured for a specific PLE transport type. This configuration is done by configuring the **port-mode** command under the optics controller.

The **port-mode** command has several parameters. The first parameter sets the PLE transport type: Ethernet, OTN, SONET, SDH, or Fibre Channel. The **framing** parameter is required and must be set to **cem-packetize**, specifying this as a PLE CEM configuration. The **rate** parameter is used to configure the specific payload.

See the **Supported Payloads** section for information on all the payloads supported.

The following examples show how to configure the Optics controller for different payloads.

**Ethernet:**

Router(config)# **controller Optics0/0/3/1**

Router(config-Optics)# **port-mode Ethernet framing cem-packetize rate 10GE**

**OTN:**

Router(config)# **controller Optics0/0/3/2**

Router(config-Optics)# **port-mode otn framing cem-packetize rate otu2**

**SONET/SDH:**

Router(config)# **controller Optics0/0/3/3**

Router(config-Optics)# **port-mode sonet framing cem-packetize rate OC48**

Router(config)# **controller Optics0/0/3/4**

Router(config-Optics)# **port-mode sdh framing cem-packetize rate STM16**

**Fiber Channel:**

Router(config)# **controller Optics0/0/3/5**

Router(config-Optics)# **port-mode FC framing cem-packetize rate FC1**

# Configure QoS Policy

Use this task to configure QoS policy to prioritize PLE traffic. This configuration is optional but is recommended so that higher priority PLE traffic is not dropped during network congestion.

**Step 1** Router(config)# **policy-map ple-policy**

Defines the parent QoS policy to be applied to the CEM interface to prioritize the PLE traffic.

**Step 2** Router(config-pmap)# **class class-default**

There is only one traffic flow type in a PLE service. Hence, the default class is used to match all the ingress packets on the PLE CEM interface.

**Step 3** Router(config-pmap-c)# **set traffic-class 6**

Cisco NCS 5500 and NCS 5700 platforms use traffic class within the router to identify specific traffic flows and differentiate how they are treated on egress. In this example, traffic class 6 is set on ingress. The egress QoS policy matches traffic class 6 and sets appropriate queuing behavior. This value is used as an example; it must be set to the high priority EXP value used across the network. The egress QoS policy is outside the scope of this guide. For egress QoS policy, see *Modular QoS Configuration Guide for Cisco NCS 5500 Series Routers*.

**Step 4** Router(config-pmap-c)# **set mpls experimental topmost 6**

Sets the topmost MPLS EXP (TC) value to 6. The value can be used at subsequent nodes in the path to classify PLE traffic and set appropriate queuing behavior. This value is used as an example; it must be set to the high priority EXP value used across the network.

**Step 5** Router(config-pmap-c)# **end-policy-map**

# Configure CEM Interface for L2VPN Service

Use this task to configure the CEM interface for the L2VPN EVPN-VPWS service.

**Step 1** Router(config)# **interface CEM0/0/3/1**

Configures the appropriate CEM interface.

**Step 2** Router(config-if)# **l2transport**

Enables the CEM interface to be used in a L2VPN EVPN-VPWS service.

# Configure Performance Measurement

Use this task to configure the performance measurement to enable the liveness monitoring of the Segment Routing policy.

**Step 1** Router(config)# **performance-measurement**

Enters the performance measurement configuration.

**Step 2** Router(config-perf-meas)# **liveness-profile name liveness-check**

Creates a SR Policy liveness profile.

**Step 3** Router(config-perf-meas)# **liveness-detection multiplier 3**

Configures liveness detection parameters and configures the number of missed liveness probes to determine whether the SR Policy is down.

**Step 4** Router(config-pm-ld-srpolicy)# **probe**

Enters the probe parameter configuration mode.

**Note** By default, probe measurment mode is set to loopback. In this mode, probe packets from the sender node are looped back to the sender node from the receiver node to test end-to-end liveness.

**Step 5** Router(config-pm-ld-srpolicy-probe)# **tx-interval 30000**

Sets the interval of probe packets that are sent by the sender. The interval value is set in microseconds, and set to 30 seconds in this example.

**Note** Using a **tx-interval** value lower than 30 seconds requires hw-offload support. NCS-55A2-MOD supports this in R7.9.1.

**Step 6** Router(config-pm-ld-srpolicy-probe)# **tos dscp 48**

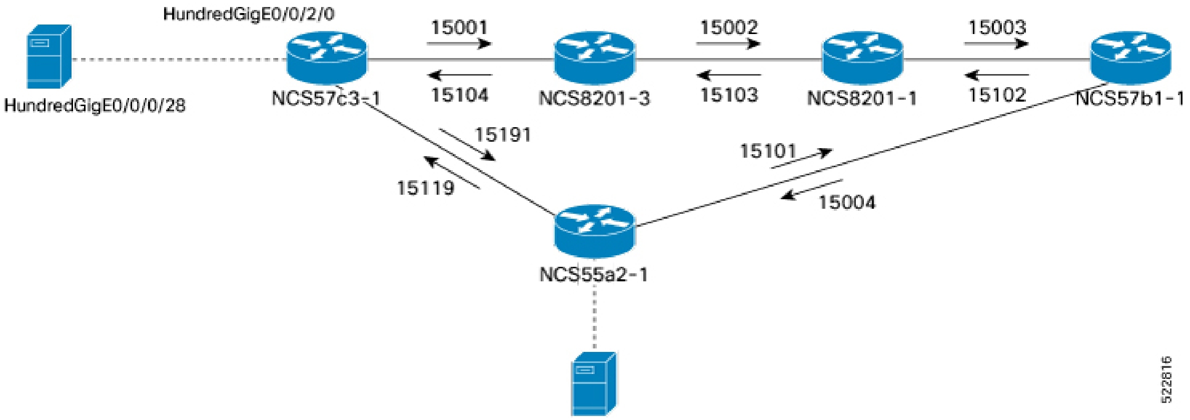
The *tos* parameter sets the Differentiated Services Code Point (DSCP) value on the probe packets to the specified DSCP. In this example it is 48, but must be set to the appropriate QoS value on the provider network to treat the liveness probe packets at high priority.

# Configure Segment Routing Adjacency SIDs

Use this task to configure SR Adjacency SIDs between routers. Adjacency SIDs are unidirectional and are used to define the explicit path between endpoints. Adjacency SIDs must set on each interface participating in the SR policy path.

The following figure illustrates the SR adjacency SIDs configured between routers.

*Figure 1: Segment Routing Adjacency SIDs*



**Step 1** Router(config)# **router isis core**

Enters IS-IS configuration mode for the appropriate instance. In this example, the instance is core.

**Step 2** Router(config-isis)# **interface HundredGigE0/0/2/0**

Configures specific IS-IS interface.

**Step 3** Router(config-isis-if)# **address-family ipv4 unicast**

Enters the IPv4 address family configuration if MPLS segment routing is used.

**Step 4** Router(config-isis-if-af)# **adjacency-sid absolute 15001**

Configures a persistent SR adjacency SID on the interface. This value must be assigned from the Segment Routing Local Block (SRLB) of the router. The default SRLB range in IOS-XR is 15000–15999.

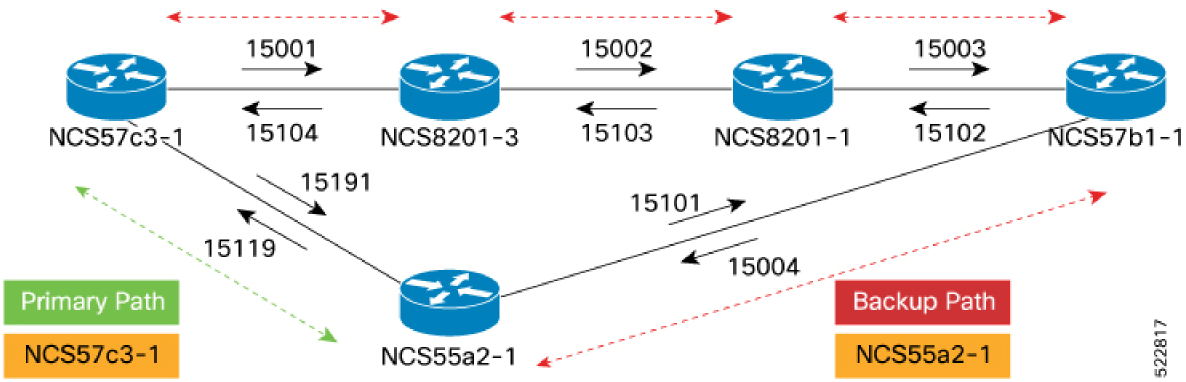
Configure other SR adjacency SIDs as appropriate.

# Define Paths Between PLE Endpoints

Use this task to define end-to-end paths between PLE endpoints.

**Note** This task applies only for static circuit-style SR-TE policies.

*Figure 2: Working and Reverse Path Definitions*



Perform the following configurations on the NCS57C3-1 router to define the working forward path. In the illustration, the working forward path is a single hop between directly connected routers.

**Step 1** Router(config)# **segment-routing**

Enters segment routing configuration.

**Step 2** Router(config-sr)# **traffic-engineering**

Enters segment routing TE configuration.

**Step 3** Router(config-sr-te)# **segment-list working-forward-path**

Creates segment list with name *working-forward-path*.

**Step 4** Router(config-sr-te-sl)# **index 1 mpls label 15191**

Sets index 1 to the first hop interface adjacency SID.

Perform the following configurations on the NCS57C3-1 router to define the working reverse path. In the illustration, the working reverse path is a single hop between directly connected routers.

```
Router(config-sr-te)# segment-list working-reverse-path
Router(config-sr-te-sl)# index 1 mpls label 15119
```

Perform the following configurations on the NCS57C3-1 router to define the protect forward path. In the illustration, the protect forward path has four interface hops between the near-end and far-end router.

```
Router(config-sr-te)# segment-list protect-forward-path
Router(config-sr-te-sl)# index 1 mpls label 15001
Router(config-sr-te-sl)# index 2 mpls label 15002
Router(config-sr-te-sl)# index 3 mpls label 15003
Router(config-sr-te-sl)# index 4 mpls label 15004
```

Perform the following configurations on the NCS57C3-1 router to define the protect reverse path. In the illustration, the protect reverse path has four interface hops between the near-end and far-end router.

```
Router(config-sr-te)# segment-list protect-reversepath
Router(config-sr-te-sl)# index 1 mpls label 15101
Router(config-sr-te-sl)# index 2 mpls label 15102
Router(config-sr-te-sl)# index 3 mpls label 15103
Router(config-sr-te-sl)# index 4 mpls label 15104
```

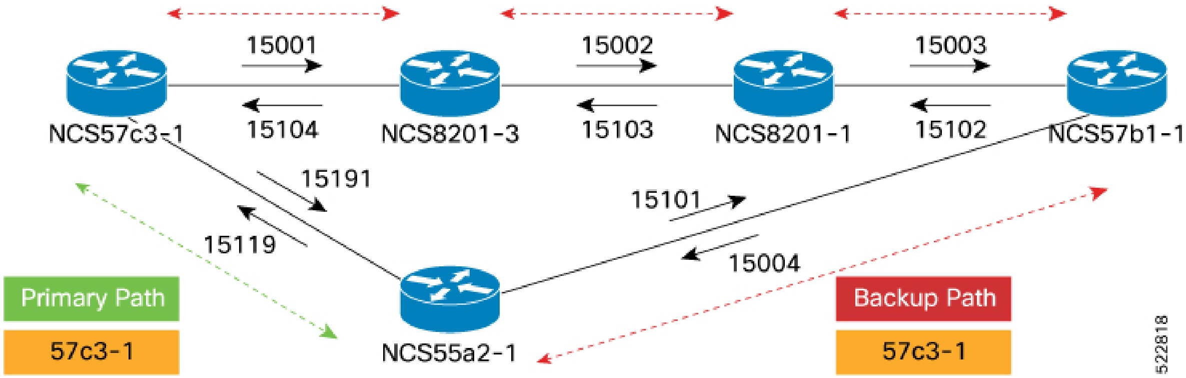
Perform the following configurations on the NCS55A2-1 router to define all the paths according to the illustration.

```
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# segment-list working-forward-path
Router(config-sr-te-sl)# index 1 mpls label 15119
!
Router(config-sr-te)# segment-list working-reverse-path
Router(config-sr-te-sl)# index 1 mpls label 15191
!
Router(config-sr-te)# segment-list protect-forward-path
Router(config-sr-te-sl)# index 1 mpls label 15101
Router(config-sr-te-sl)# index 2 mpls label 15102
Router(config-sr-te-sl)# index 3 mpls label 15103
Router(config-sr-te-sl)# index 4 mpls label 15104
!
Router(config-sr-te)# segment-list protect-reversepath
Router(config-sr-te-sl)# index 1 mpls label 15001
Router(config-sr-te-sl)# index 2 mpls label 15002
Router(config-sr-te-sl)# index 3 mpls label 15003
Router(config-sr-te-sl)# index 4 mpls label 15004
```

# Configure Static Circuit-Style SR-TE Policies

Use this task to configure static circuit-style SR-TE policies.

***Figure 3: Segment Routing Traffic Engineering Policy***



Perform the following configurations on the NCS57C3-1 router.

### Before you begin

Static Circuit-Style SR-TE policies requires IOS XR 7.7.1 or later software.

**Step 1** Router(config)# **segment-routing**

Enters segment routing configuration.

**Step 2** Router(config-sr)# **traffic-engineering**

Enters segment routing TE configuration.

**Step 3** Router(config-sr-te)# **policy to-55a2-1**

Creates a SR-TE policy with the name *to-55a2-1*.

**Step 4** Router(config-sr-te-policy)# **color 1001 end-point ipv4 10.0.0.44**

Configures the SR-TE policy with the user-defined color and loopback address of the remote PLE far-end router.

**Step 5** Router(config-sr-te-policy)# **path-protection**

Enables path protection on the SR-TE policy. When configured, the near-end router keeps the protect path in warm-standby state to quickly transition to it if the working path is down.

**Step 6** Router(config-sr-te-policy)# **candidate-paths**

Configures the SR working and protect candidate paths.

**Step 7** Router(config-sr-te-policy-path)# **preference 50**

Sets the user-defined preference on the protect path. Higher preference candidate paths are preferred over lower preference paths.

**Note** The protect path must always be set with lower preference value.

**Step 8** Router(config-sr-te-policy-path-pref)# **explicit segment-list protect-forward-path**

Configures the candidate path to use an explicit segment list. The previously defined *protect-forward-path* explicit segment list is used.

**Step 9** Router(config-sr-te-policy-path-pref)# **reverse-path segment-list protect-reverse-path**

Configures the reverse path to ensure that the SR-TE Policy uses a corouted bidirectional path. The previously defined *protect-reverse-path* explicit segment list is used.

Configure the higher priority working path similar to the protect path using the following commands:

```
Router(config-sr-te-policy-path)# preference 100
Router(config-sr-te-policy-path-pref)# explicit segment-list working-forward-path
Router(config-sr-te-policy-path-pref)# reverse-path segment-list working-reverse-path
```

**Step 10** Router(config)# **performance-measurement**

Configures performance-measurement for the SR-TE policy.

**Step 11** Router(config-perf-meas)# **liveness-detection**

Configures liveness detection parameters.

**Step 12** Router(config-perf-meas)# **liveness-profile name liveness-check**

Configures a user-defined liveness policy to be used. In this case, the previously defined liveness profile *liveness-check* is used.

Perform the following configurations on the NCS55A2-1 router:

```
Router(config)# segment-routing
Router(config-sr)# traffic-eng
Router(config-sr-te)# policy to-57c3-1
Router(config-sr-te-policy)# color 1001 end-point ipv4 10.0.0.42
Router(config-sr-te-policy)# path-protection
Router(config-sr-te-policy)# candidate-paths
Router(config-sr-te-policy-path)# preference 50
Router(config-sr-te-policy-path-pref)# explicit segment-list protect-forward-path
Router(config-sr-te-policy-path-pref)# reverse-path segment-list protect-reverse-path
Router(config-sr-te-policy-path)# preference 100
Router(config-sr-te-policy-path-pref)# explicit segment-list working-forward-path
Router(config-sr-te-policy-path-pref)# reverse-path segment-list working-reverse-path
Router(config)# performance-measurement
Router(config-perf-meas)# liveness-detection
Router(config-perf-meas)# liveness-profile name liveness-check
```

# Configure Dynamic Circuit-Style SR-TE Policies Without Bandwidth CAC

Use this task to configure dynamic circuit-style SR-TE policies without bandwidth Call Admission Control (CAC).

### Before you begin

Dynamic Circuit-Style SR-TE policies without bandwidth CAC requires the following software:

- IOS XR R7.8.1 or later
- SR-PCE 7.8.1 or later

**Step 1** Router(config)# **segment-routing**

Enters segment routing configuration.

**Step 2** Router(config-sr)# **traffic-engineering**

Enters segment routing traffic engineering configuration.

**Step 3** Router(config-sr-te)# **policy** *policy-name*

Creates a SR-TE policy with the specified name.

**Step 4** Router(config-sr-te-policy)# **color 119 end-point ipv4 10.0.0.44**

Configures the SR-TE policy with the user-defined color and loopback address of the remote PLE far-end router.

**Step 5** Router(config-sr-te-policy)# **path-protection**

Enables path protection on the SR-TE policy. When configured, the near-end router keeps the protect path in standby state to quickly transition to it if the working path is down.

**Step 6** Router(config-sr-te-policy)# **candidate-paths dynamic pcep**

Configures the SR working, protect, and restoration candidate paths.

The **candidate-paths** parameter is set to **dynamic->pcep** to enable SR-PCE delegation. The dynamic computation can be based on standard metrics such as traffic engineering, IGP, or latency.

**Step 7** Router(config-sr-te-policy-path)# **preference 50**

Sets the user-defined preference on the protect path. Higher preference candidate paths are preferred over lower preference paths.

**Note** The protect path must always be set with lower preference value.

The third candidate path with a preference lower than 100 can be defined as a restoration path. The restoration path is not presignaled and must be configured with **backup-ineligible**. The restoration path does not have a defined disjoint-path ID, but has a bidirectional association-id matching the reverse policy.

The **lock-duration** parameter determines the reversion behavior. If a higher preference path is available, the policy switches to it after the expiry of lock duration. The default value is 300 seconds; the value 0 disables dynamic reversion.

**Step 8** Router(config-sr-te-policy-path)# **constraints segments protection {unprotected-only | adjacency-sid-only}**

Configures the protection behavior constraints of adjacency-SID.

Circuit-Style uses segments constraints of **unprotected-only** and **adjacency-sid-only**. Manual adjacency-SIDs are used if they are defined; otherwise, will use dynamic adjacency-SIDs. Manual adjacency-SIDs are recommended.

**Step 9** Router(config-sr-te-path-pref-const)# **disjoint-path group-id 10 type {link | node | srlg | srlg-node}**

Configures the disjoint-path constraints.

The **group-id** parameter of the **disjoint-path** must be the same for both active and backup paths. It must be unique for nodes using the same SR-PCE.

The co-routed association ID must be the same for both bidirectional active and backup paths.

**Step 10** Router(config)# **performance-measurement**

Configures performance-measurement for the SR-TE policy.

**Step 11** Router(config-perf-meas)# **liveness-detection**

Configures liveness detection parameters. The liveness detection is enabled on CS-SR policy to trigger path switchover.

**Step 12** Router(config-perf-meas)# **liveness-profile name liveness-check**

Configures a user-defined liveness policy to be used.

**Step 13** Router(config-perf-meas)# **invalidation-action down**

Brings down the candidate path when it becomes invalid.

The following is an example of dynamic circuit-style SR-TE policy configuration without bandwidth Call Admission Control (CAC).

```
policy dynamic-cs-srte-to-55a2-p2
   color 119 end-point ipv4 10.0.0.44
   path-protection
   !
   candidate-paths
    preference 100
     dynamic
```

```
  pcep
  !
  metric
   type igp
  !
 !
 constraints
  segments
   protection unprotected-only
   adjacency-sid-only
  !
  disjoint-path group-id 10 type link
 !
 bidirectional
  co-routed
  association-id 101
 !
!
preference 200
 dynamic
  pcep
  !
  metric
   type igp
  !
 !
 lock
   duration 30
 !
 constraints
  segments
   protection unprotected-only
   adjacency-sid-only
  !
  disjoint-path group-id 10 type link
 !
 bidirectional
  co-routed
  association-id 201
 !
!
preference 50
 dynamic
  pcep
  !
  metric
   type igp
  !
 !
 backup-ineligible
 lock
   duration 60
 !
 constraints
  segments
   protection unprotected-only
   adjacency-sid-only
  !
 !
 bidirectional
  co-routed
  association-id 301
 !
!
```

```
!
performance-measurement
 liveness-detection
  liveness-profile backup name CS-PROTECT
  liveness-profile name CS-WORKING
  invalidation-action down
```

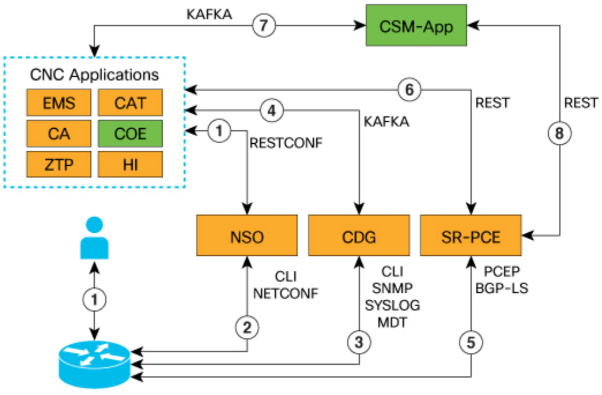
# Configure Dynamic Circuit-Style SR-TE Policies With Bandwidth CAC

Dynamic PLE with bandwidth Call Admission Control (CAC) is supported through Crosswork Network Controller. Bandwidth constraint is specified for the circuit-style policy. Cisco Crosswork Optimization Engine application is part of Circuit Style Manager.

The Circuit Style Segment Routing Traffic Engineering (CS-SR TE) feature pack provides a bandwidth-aware Path Computation Element (PCE) to compute CS-SR policy paths that you can visualize in your network. CS-SR policies guarantee allocated bandwidth services with predictable latency and persistent bidirectional path protection of critical traffic. CS-SR TE reserves a percentage of bandwidth in the network and computes CS-SR policy bidirectional failover paths with the requested bandwidth.

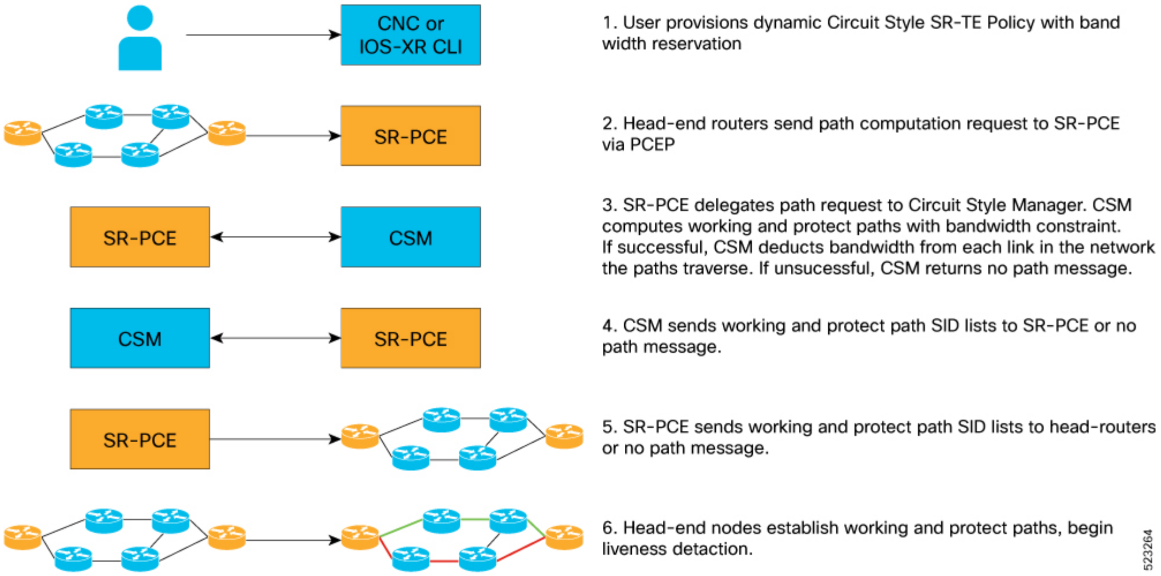
Circuit Style Manager requires policies be configured on each head-end node with matching association IDs, bandwidth values, constraints, computation type, and source/endpoint IP addresses.

*Figure 4: Crosswork Optimization Engine - SR—PCE Communication Flow Diagram*



1. User provisions CS-SRTE policy via CNC CS-SRTE Function Packor on the router directly via CLI.
2. (Optional) CS-SRTE is provisioned to device via NSO from CNC
3. CDG collects data from network for use by COE to build network model
4. CDG utilizes Kafka bus to send collected information to CW for app consumption
5. SR-PCE collects IGP/TE topology and SR Policy information from network
6. CNC/COE learns TE topology and CS-SRTE Policies with BW reservations from SR-PCE
7. Network data is consumed by CSM application
8. CSM subscribes to SR-PCE BWoD API or SR-PCE can delegate CS-SRTE Policies with BW reservations to CSM for path computation

*Figure 5: Crosswork Optimization Engine - SR—PCE Call Flow Diagram*


CNC or IOS-XR CLI
1. User provisions dynamic Circuit Style SR-TE Policy with band width reservation
SR-PCE
2. Head-end routers send path computation request to SR-PCE via PCEP
SR-PCE
CSM
3. SR-PCE delegates path request to Circuit Style Manager. CSM computes working and protect paths with bandwidth constraint. If successful, CSM deducts bandwidth from each link in the network the paths traverse. If unsucessful, CSM returns no path message.
CSM
SR-PCE
4. CSM sends working and protect path SID lists to SR-PCE or no path message.
SR-PCE
5. SR-PCE sends working and protect path SID lists to head-routers or no path message.
6. Head-end nodes establish working and protect paths, begin liveness detaction.
523264

### Before you begin

Dynamic Circuit-Style SR-TE policies with bandwidth CAC requires the following software:

- IOS XR R7.9.1 or later
- SR-PCE 7.9.1 or later
- Crosswork Optimization Engine 5.0 or later

**Step 1** Enable the Circuit Style Segment Routing Traffic Engineering (CS-SR TE) feature pack.

**Step 2** Configure CS-SR policies on the devices.

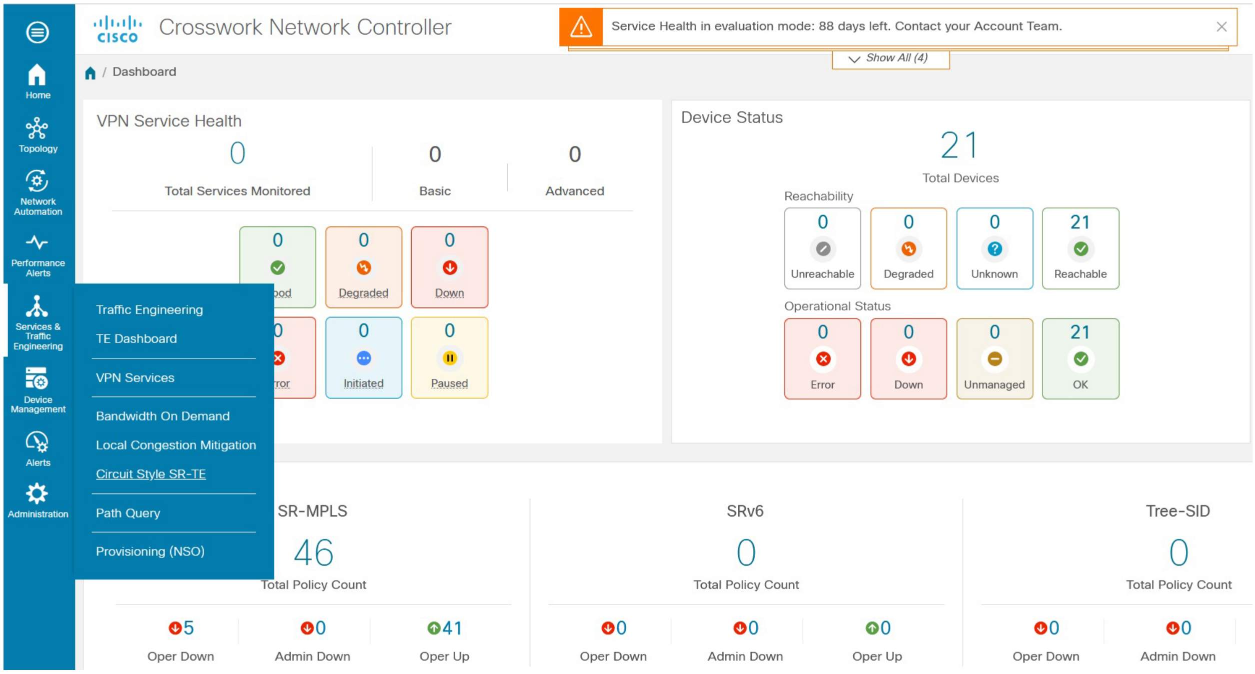
**Step 3** Verify that the CS-SR policies appear in the SR Policy table.

# Enable Circuit Style Segment Routing Traffic Engineering Using Crosswork Network Controller

In order to manage and visualize CS-SR policies on the topology map, you must first enable CS-SR TE and set bandwidth reservation settings.

When CS-SR TE is enabled, it computes the best failover bidirectional paths with the requested bandwidth and other constraints defined in the CS-SR policy configuration between two nodes.

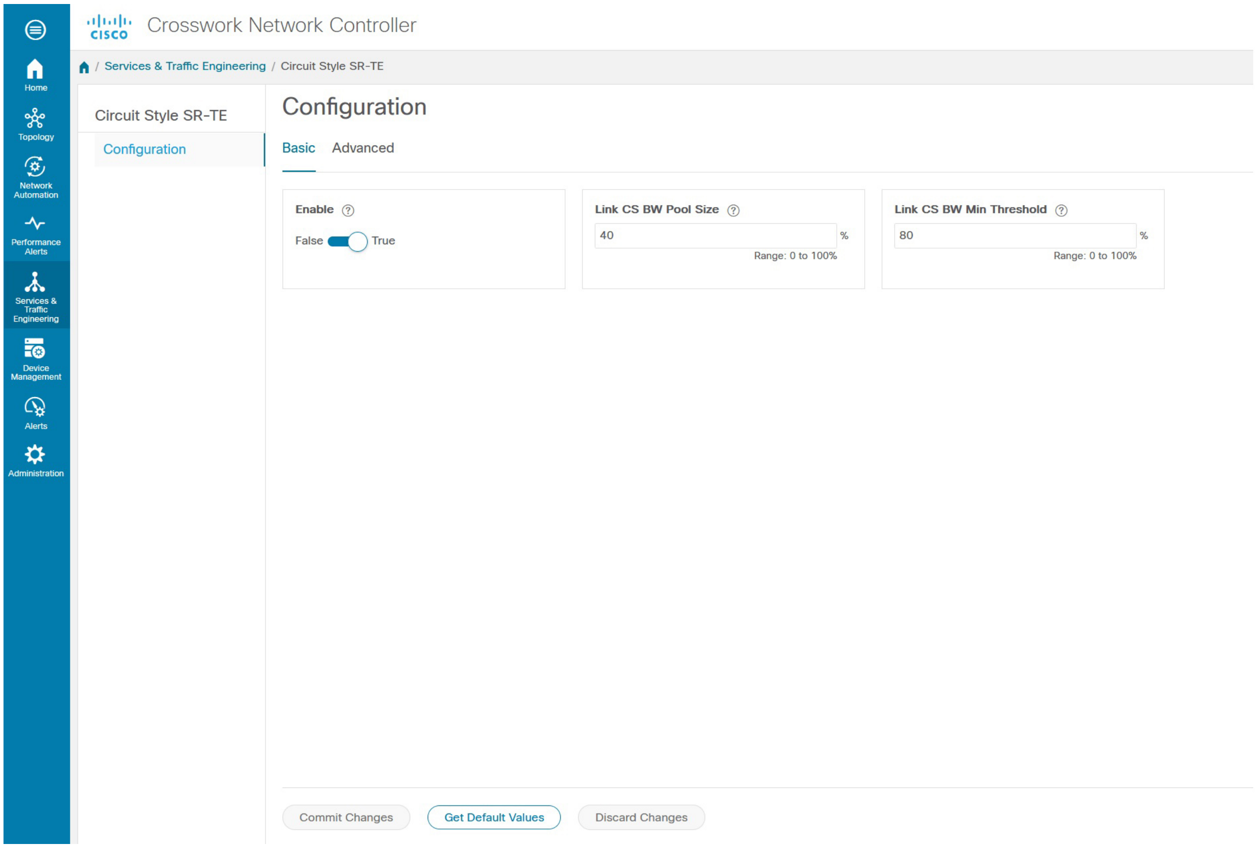
**Step 1** From the main menu, choose **Services & Traffic Engineering** > **Circuit Style SR-TE** > **Configuration** .



523258

**Step 2** Toggle the **Enable** switch to **True**.

**Step 3** Enter the required bandwidth pool size and threshold information.



Configuration

Enable: False / True (set to True)

Link CS BW Pool Size: 40 % (Range: 0 to 100%)

Link CS BW Min Threshold: 80 % (Range: 0 to 100%)

523259

| Option | Description |
|---|---|
| **Basic Pane** | |
| **Link CS BW Pool Size** | The percentage of bandwidth that each link reserves for CS-SR policies. |
| **Link CS BW Min Threshold** | An alert is generated when the bandwidth utilization percentage on a link used for a CS-SR policy exceeds this value. |
| **Advanced Pane** | |
| **Validation Interval** | The interval that CS-SR policy will wait before the bandwidth that is reserved for an undelegated policy is returned to the CS-SR policy bandwidth pool. |
| **Timeout** | The duration until which CS-SR policy will wait for the delegation request, to generate a notification. |
| **Restore Delegation Delay** | The duration until which CS-SR policy will pause before processing a restore path delegation. |

**Step 4** Click **Commit Changes** to save the configuration.

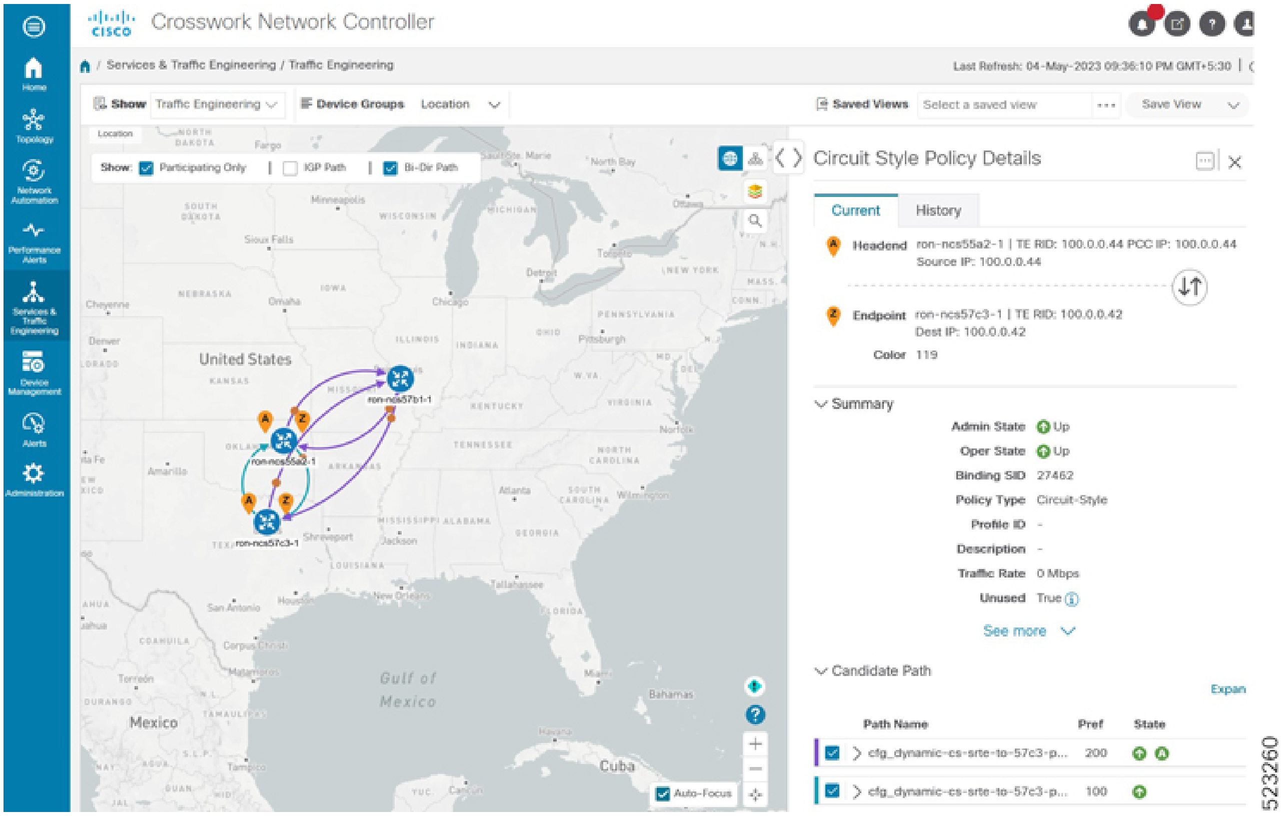
## View CS-SR Policy Details Using Crosswork Network Controller

**Step 1** From the main menu, choose **Services & Traffic Engineering** > **SR-MPLS** and click **Circuit Style**.

The **SR Policy** table lists all the CS-SR policies.

**Step 2** From the **Actions** column, click **View Details** for one of the CS-SR policies.

The **Circuit Style Policy Details** window is displayed in the side panel. By default, the active path is displayed and shows the bidirectional paths (**Bi-Dir Path** checkbox is checked) on the topology map.


Crosswork Network Controller
Home
Topology
Network Automation
Performance Alerts
Services & Traffic Engineering
Device Management
Alerts
Administration
Services & Traffic Engineering / Traffic Engineering
Last Refresh: 04-May-2023 09:36:10 PM GMT+5:30
Show Traffic Engineering
Device Groups Location
Saved Views Select a saved view
Save View
Show: ☑ Participating Only | □ IGP Path | ☑ Bi-Dir Path
ron-ncs57b1-1
ron-ncs55a2-1
ron-ncs57c3-1
Auto-Focus
523260

Circuit Style Policy Details

Current | History

Headend ron-ncs55a2-1 | TE RID: 100.0.0.44 PCC IP: 100.0.0.44
Source IP: 100.0.0.44

Endpoint ron-ncs57c3-1 | TE RID: 100.0.0.42
Dest IP: 100.0.0.42

Color 119

Summary

| | |
|---|---|
| Admin State | Up |
| Oper State | Up |
| Binding SID | 27462 |
| Policy Type | Circuit-Style |
| Profile ID | - |
| Description | - |
| Traffic Rate | 0 Mbps |
| Unused | True |

See more

Candidate Path

Expan

| | Path Name | Pref | State |
|---|---|---|---|
| ☑ | cfg_dynamic-cs-srte-to-57c3-p... | 200 | Up A |
| ☑ | cfg_dynamic-cs-srte-to-57c3-p... | 100 | Up |

If bandwidth of a new policy exceeds bandwidth pool size, the policy will be down. The requested bandwidth is set to the configured bandwidth, but the reserved bandwidth will be 0.

If an existing policy is configured to a new bandwidth exceeding the bandwidth pool size, the policy will stay up but the reserved bandwidth stays at the previous value.

The following sample output shows a policy with requested bandwidth and current bandwidth.

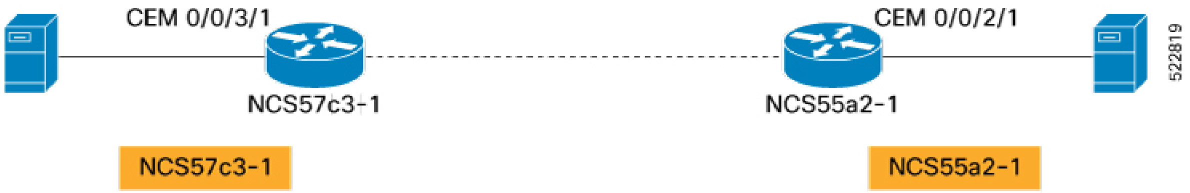
```
RP/0/RP0/CPU0:ron-ncs57c3-1#show segment-routing traffic-eng policy color 119
Thu May 4 09:01:57.797 PDT

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

Color: 119, End-point: 10.0.0.44
Name: srte_c_119_ep_10.0.0.44
Status:
Admin: up Operational: up for 11:46:42 (since May 3 21:15:14.898)
Candidate-paths:
Preference: 200 (configuration) (active)
Name: dynamic-cs-srte-to-55a2-p2
Requested BSID: dynamic
PCC info:
Symbolic name: cfg_dynamic-cs-srte-to-55a2-p2_discr_200
PLSP-ID: 7
Constraints:
Protection Type: unprotected-only
Maximum SID Depth: 12
Adjacency SIDs Only: True
Dynamic (pce 201.0.0.106) (valid)
Metric Type: IGP, Path Accumulated Metric: 20
```

```
SID[0]: 15739 [Adjacency-SID, 10.139.1.1 - 10.139.1.2]
SID[1]: 15720 [Adjacency-SID, 10.1.26.0 - 10.1.26.1]
Reverse path:
SID[0]: 15227 [Adjacency-SID, 10.1.26.1 - 10.1.26.0]
SID[1]: 15727 [Adjacency-SID, 10.139.1.2 - 10.139.1.1]
Protection Information:
Role: WORKING
Path Lock: Timed
Lock Duration: 300(s)
State: ACTIVE
Preference: 100 (configuration) (protect)
Name: dynamic-cs-srte-to-55a2-p2
Requested BSID: dynamic
PCC info:
Symbolic name: cfg_dynamic-cs-srte-to-55a2-p2_discr_100
PLSP-ID: 6
Constraints:
Protection Type: unprotected-only
Maximum SID Depth: 12
Adjacency SIDs Only: True
Dynamic (pce 201.0.0.106) (valid)
Metric Type: IGP, Path Accumulated Metric: 30
SID[0]: 15734 [Adjacency-SID, 10.1.42.0 - 10.1.42.1]
Reverse path:
SID[0]: 15523 [Adjacency-SID, 10.1.42.1 - 10.1.42.0]
Protection Information:
Role: PROTECT
Path Lock: Timed
Lock Duration: 300(s)
State: STANDBY
Attributes:
Binding SID: 24047
Forward Class: Not Configured
Steering labeled-services disabled: no
Steering BGP disabled: no
IPv6 caps enable: yes
Bandwidth Requested: 35.000 Gbps
Bandwidth Current: 35.000 Gbps
Invalidation drop enabled: no
Max Install Standby Candidate Paths: 0
```

# Configure PLE over EVPN-VPWS

Use this task to configure PLE over EVPN-VPWS. The two core components are the Pseudowire class specifying the transport type and xconnect service to configure the EVPN-VPWS service parameters.

***Figure 6: PLE EVPN-VPWS Service***



Perform the following configurations on the NCS57C3-1 router.

**Step 1** Router(config)# **l2vpn**

Enters l2vpn configuration.

**Step 2** Router(config-l2vpn)# **pw-class circuit-style-srte**

Defines the Pseudowire class to be used with the PLE service. The same Pseudowire class can be used for multiple PLE services between the same far-end routers, or a unique class can be used for each service. The Pseudowire class defines the underlying transport for the service, in this case MPLS using a specific SR-TE policy. The **pw-class** command is followed by a user-defined name, in this example, *circuit-style-srte*.

**Step 3** Router(config-l2vpn-pwc)# **encapsulation mpls**

Configures transport encapsulation.

**Step 4** Router(config-l2vpn-pwc-mpls)# **preferred-path sr-te policy srte_c_1001_ep_10.0.0.42**

The preferred-path is the circuit-style policy. The preferred-path is used to configure an explicit MPLS path to be used for the l2vpn service. In this case, the preferred-path is set to the SR-TE policy created from the source node to the 10.0.0.42 endpoint node. The path is not the configured name of the policy; it is the computed name.

**Step 5** Router(config-l2vpn)# **xconnect group ple**

Enters xconnect configuration mode. The **xconnect group** command is used to administratively group similar L2VPN services and can be set to any user-defined value.

**Step 6** Router(config-l2vpn-xc)# **p2p ple-cs-1**

Creates point-to-point L2VPN service with user-defined name *ple-cs-1*.

**Step 7** Router(config-l2vpn-xc-p2p)# **interface cem 0/0/2/1**

Specifes the client interface. This interface is the CEM interface type for PLE.

**Step 8** Router(config-l2vpn-xc-p2p)# **neighbor evpn evi 100 target 4201 source 4401**

Configures EVPN-VPWS parameters. The user-defined *evi* value must be the same on each far-end router. The target must be the “source” value that is configured on the remote endpoint.

**Step 9** Router(config-l2vpn-xc-p2p-pw)# **pw-class circuit-style-srte**

Attaches the previously defined Pseudowire class *circuit-style-te* to the L2VPN service.

Perform the following configurations on the NCS55a2-1 router.

```
Router(config)# l2vpn
Router(config-l2vpn)# pw-class circuit-style-srte
Router(config-l2vpn-pwc)# encapsulation mpls
Router(config-l2vpn-pwc-mpls)# preferred-path sr-te policy srte_c_1001_ep_10.0.0.44
Router(config-l2vpn)# xconnect group ple
Router(config-l2vpn-xc)# p2p ple-cs-1
Router(config-l2vpn-xc-p2p)# interface cem0/0/2/1
Router(config-l2vpn-xc-p2p)# neighbor evpn evi 100 target 4201 source 4401
Router(config-l2vpn-xc-p2p-pw)# pw-class circuit-style-srte
```

# Verification

The following example shows the general CEM statistics.

```
RP/0/RP0/CPU0:ron-ncs57c3-1#show controllers CEM 0/0/3/1

Sat Sep 24 11:34:22.533 PDT
Interface : CEM0/0/3/1
Admin state : Up
Oper state : Up
Port bandwidth : 10312500 kbps
Dejitter buffer (cfg/oper/in-use) : 0/813/3432 usec
Payload size (cfg/oper) : 1280/1024 bytes
PDV (min/max/avg) : 980/2710/1845 usec
Dummy mode : last-frame
Dummy pattern : 0xaa
Idle pattern : 0xff
Signalling : No CAS
RTP : Enabled
Clock type : Differential
Detected Alarms : None
Statistics Info
---------------
Ingress packets : 517617426962, Ingress packets drop : 0
Egress packets : 517277124278, Egress packets drop : 0
Total error : 0
Missing packets : 0, Malformed packets : 0
Jitter buffer underrun : 0, Jitter buffer overrun : 0
Misorder drops : 0
Reordered packets : 0, Frames fragmented : 0
Error seconds : 0, Severely error seconds : 0
Unavailable seconds : 0, Failure counts : 0
Generated L bits : 0, Received L bits : 0
Generated R bits : 339885178, Received R bits : 17
Endpoint Info
-------------
Passthrough : No
```

The following example shows the PM statistics for 30 seconds.

```
RP/0/RP0/CPU0:ron-ncs57c3-1#show controllers CEM 0/0/3/1 pm current 30-sec cem

Sat Sep 24 11:37:02.374 PDT
CEM in the current interval [11:37:00 - 11:37:02 Sat Sep 24 2022]
CEM current bucket type : Valid
INGRESS-PKTS : 2521591 Threshold : 0 TCA(enable) : NO
EGRESS-PKTS : 2521595 Threshold : 0 TCA(enable) : NO
INGRESS-PKTS-DROPPED : 0 Threshold : 0 TCA(enable) : NO
EGRESS-PKTS-DROPPED : 0 Threshold : 0 TCA(enable) : NO
INPUT-ERRORS : 0 Threshold : 0 TCA(enable) : NO
OUTPUT-ERRORS : 0 Threshold : 0 TCA(enable) : NO
MISSING-PKTS : 0 Threshold : 0 TCA(enable) : NO
PKTS-REORDER : 0 Threshold : 0 TCA(enable) : NO
JTR-BFR-UNDERRUNS : 0 Threshold : 0 TCA(enable) : NO
JTR-BFR-OVERRUNS : 0 Threshold : 0 TCA(enable) : NO
MIS-ORDER-DROPPED : 0 Threshold : 0 TCA(enable) : NO
MALFORMED-PKT : 0 Threshold : 0 TCA(enable) : NO
ES : 0 Threshold : 0 TCA(enable) : NO
SES : 0 Threshold : 0 TCA(enable) : NO
UAS : 0 Threshold : 0 TCA(enable) : NO
FC : 0 Threshold : 0 TCA(enable) : NO
TX-LBITS : 0 Threshold : 0 TCA(enable) : NO
```

```
TX-RBITS : 0 Threshold : 0 TCA(enable) : NO
RX-LBITS : 0 Threshold : 0 TCA(enable) : NO
RX-RBITS : 0 Threshold : 0 TCA(enable) : NO
```

The following example shows the controller information for Ethernet 10GE.

```
RP/0/RP0/CPU0:ron-ncs57c3-1#show controllers TenGigECtrlr0/0/3/2

Sat Sep 24 11:43:23.164 PDT
Operational data for interface TenGigECtrlr0/0/3/2:
State:
Administrative state: enabled
Operational state: Up
LED state: Green On
PRBS:
Status: Locked
Mode: Source-sink
Pattern: User-defined
Direction: System
Error-inject: None
Framing: Unframed
User-pattern: 0xabcdef0123456789
Phy:
Media type: Not known
Autonegotiation disabled.
Operational values:
Speed: 10Gbps
Duplex: Full Duplex
Flowcontrol: None
Loopback: None (or external)
Inter-packet gap: standard (12)
BER monitoring:
Not supported
```

The following example shows the PM statistics for Ethernet controller.

```
RP/0/RP0/CPU0:ron-ncs57c3-1#show controllers TenGigECtrlr0/0/3/2 pm current 30-sec ether

Sat Sep 24 11:45:39.919 PDT
ETHER in the current interval [11:45:30 - 11:45:39 Sat Sep 24 2022]
ETHER current bucket type : Valid
RX-PKT : 4528985 Threshold : 0 TCA(enable) : NO
STAT-PKT : 9057971 Threshold : 0 TCA(enable) : NO
OCTET-STAT : 2318840576 Threshold : 0 TCA(enable) : NO
OVERSIZE-PKT : 0 Threshold : 0 TCA(enable) : NO
FCS-ERR : 0 Threshold : 0 TCA(enable) : NO
LONG-FRAME : 0 Threshold : 0 TCA(enable) : NO
JABBER-STATS : 0 Threshold : 0 TCA(enable) : NO
64-OCTET : 0 Threshold : 0 TCA(enable) : NO
65-127-OCTET : 0 Threshold : 0 TCA(enable) : N
128-255-OCTET : 0 Threshold : 0 TCA(enable) : NO
256-511-OCTET : 9057971 Threshold : 0 TCA(enable) : NO
512-1023-OCTET : 0 Threshold : 0 TCA(enable) : NO
1024-1518-OCTET : 0 Threshold : 0 TCA(enable) : NO
IN-UCAST : 0 Threshold : 0 TCA(enable) : NO
IN-MCAST : 0 Threshold : 0 TCA(enable) : NO
IN-BCAST : 0 Threshold : 0 TCA(enable) : NO
OUT-UCAST : 0 Threshold : 0 TCA(enable) : NO
OUT-BCAST : 0 Threshold : 0 TCA(enable) : NO
OUT-MCAST : 0 Threshold : 0 TCA(enable) : NO
TX-PKT : 4528986 Threshold : 0 TCA(enable) : NO
OUT-OCTET : 1159420416 Threshold : 0 TCA(enable) : NO
IFIN-ERRORS : 0 Threshold : 0 TCA(enable) : NO
IFIN-OCTETS : 0 Threshold : 0 TCA(enable) : NO
STAT-MULTICAST-PKT : 0 Threshold : 0 TCA(enable) : NO
```

```
STAT-BROADCAST-PKT : 0 Threshold : 0 TCA(enable) : NO
STAT-UNDERSIZED-PKT : 0 Threshold : 0 TCA(enable) : NO
IN_GOOD_BYTES : 1159420160 Threshold : 0 TCA(enable) : NO
IN_802_1Q_FRAMES : 0 Threshold : 0 TCA(enable) : NO
IN_GOOD_PKTS : 4528985 Threshold : 0 TCA(enable) : NO
IN_DROP_OTHER : 0 Threshold : 0 TCA(enable) : NO
OUT_GOOD_BYTES : 1159420416 Threshold : 0 TCA(enable) : No
OUT_802_1Q_FRAMES : 0 Threshold : 0 TCA(enable) : NO
OUT_GOOD_PKTS : 4528986 Threshold : 0 TCA(enable) : NO
IN_ERROR_FRAGMENTS : 0 Threshold : 0 TCA(enable) : NO
IN_PKT_64_OCTET : 0 Threshold : 0 TCA(enable) : NO
IN_PKTS_65_127_OCTETS : 0 Threshold : 0 TCA(enable) : NO
IN_PKTS_128_255_OCTETS : 0 Threshold : 0 TCA(enable) : NO
IN_PKTS_256_511_OCTETS : 4528985 Threshold : 0 TCA(enable) : NO
IN_PKTS_512_1023_OCTETS : 0 Threshold : 0 TCA(enable) : NO
IN_PKTS_1024_1518_OCTETS : 0 Threshold : 0 TCA(enable) : NO
OUT_PKT_64_OCTET : 0 Threshold : 0 TCA(enable) : NO
OUT_PKTS_65_127_OCTETS : 0 Threshold : 0 TCA(enable) : NO
OUT_PKTS_128_255_OCTETS : 0 Threshold : 0 TCA(enable) : NO
OUT_PKTS_256_511_OCTETS : 4528986 Threshold : 0 TCA(enable) : NO
OUT_PKTS_512_1023_OCTETS : 0 Threshold : 0 TCA(enable) : NO
OUT_PKTS_1024_1518_OCTETS : 0 Threshold : 0 TCA(enable) : NO
TX_UNDERSIZED_PKT : 0 Threshold : 0 TCA(enable) : NO
TX_OVERSIZED_PKT : 0 Threshold : 0 TCA(enable) : NO
TX_FRAGMENTS : 0 Threshold : 0 TCA(enable) : NO
TX_JABBER : 0 Threshold : 0 TCA(enable) : NO
TX_BAD_FCS : 0 Threshold : 0 TCA(enable) : NO

Last clearing of "show controllers ETHERNET" counters never
```

The following example shows the controller information for Fiber Channel 8G.

```
RP/0/RP0/CPU0:ron-ncs57c3-1#show controllers EightGigFibreChanCtrlr0/0/3/4

Sat Sep 24 11:53:09.820 PDT
Operational data for Fibre Channel controller EightGigFibreChanCtrlr0/0/3/4
State:
Admin State : Up
Operational state : Up
LED state : Green On
Secondary admin state : Normal
Laser Squelch : Disabled
Performance Monitoring is enabled
Operational values:
Speed : 8 Gbps
Loopback : None
BER monitoring:
Signal Degrade : 1e-0
Signal Fail : 1e-0
Hold-off Time : 0 ms
Forward Error Correction : Not Configured
```

The following example shows the PM statistics for Fiber Channel.

```
RP/0/RP0/CPU0:ron-ncs57c3-1#show controllers EightGigFibreChanCtrlr0/0/3/4 pm current 30-sec
 fc
Sat Sep 24 11:51:55.168 PDT
FC in the current interval [11:51:30 - 11:51:55 Sat Sep 24 2022]
FC current bucket type : Valid
IFIN-OCTETS : 16527749196 Threshold : 0 TCA(enable) : NO
RX-PKT : 196758919 Threshold : 0 TCA(enable) : NO
IFIN-ERRORS : 0 Threshold : 0 TCA(enable) : NO
RX-BAD-FCS : 0 Threshold : 0 TCA(enable) : NO
IFOUT-OCTETS : 0 Threshold : 0 TCA(enable) : NO
TX-PKT : 0 Threshold : 0 TCA(enable) : NO
```

```
TX-BAD-FCS : 0 Threshold : 0 TCA(enable) : NO
RX-FRAMES-TOO-LONG : 0 Threshold : 0 TCA(enable) : NO
RX-FRAMES-TRUNC : 0 Threshold : 0 TCA(enable) : NO
TX-FRAMES-TOO-LONG : 0 Threshold : 0 TCA(enable) : NO
TX-FRAMES-TRUNC : 0 Threshold : 0 TCA(enable) : NO
```