Cisco Crosswork Network Controller 5.0 Solution Workflow Guide

First Published: 2022-11-16

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

CHAPTER 1

Solution Overview

This section explains the following topics:

Description

The exponential growth of network traffic and the pressures of efficiently running network operations pose huge challenges for network operators. Providing quick intent-based service delivery with optimal network utilization and the ability to react to bandwidth and latency demand fluctuations in real time is vital to success. Migration to Software-Defined Networks (SDNs) and automation of operational tasks is the optimal way to become more efficient and competitive.

Cisco Crosswork Network Controller is a turnkey network automation solution for deploying and operating IP transport networks that delivers increased service agility, cost efficiency, and optimization for faster time-to-customer value and lower operating costs. The solution combines intent-based network automation to deliver critical capabilities for service orchestration and fulfillment, network optimization, service path computation, device deployment and management, and anomaly detection and automatic remediation. Using telemetry gathering and automated responses, Cisco Crosswork Network Controller delivers network optimization capabilities that are nearly impossible to replicate even with a highly skilled and dedicated staff operating the network.

The fully integrated solution combines functionality from multiple Crosswork components installed upon a common Crosswork infrastructure, as well as industry-leading capabilities from Cisco® Network Services Orchestrator (NSO), Cisco Segment Routing Path Computation Element (SR-PCE), and Cisco WAN Automation Engine (WAE). Its unified user interface provides a single pane of glass for real-time visualization of the network topology and services, provisioning, monitoring, and optimization.

What's New in This Release

This release of Cisco Crosswork Network Controller supports the following new features and capabilities:

- **Circuit Style Segment Routing Traffic Engineering (CS SR-TE)**:

  The Circuit Style Manager (CSM) feature pack provides a bandwidth-aware Path Computation Element (PCE) to compute CS SR-TE policy paths, provision them, and visualize them on geographic or logical maps. CS SR-TE policies guarantee allocated bandwidth services with predictable latency and persistent bidirectional path protection for critical traffic. Operators can provision CS SR-TE policies based on the operator's intent. Unlike Bandwidth on Demand, where SR policies with requested bandwidth are created on a best-effort basis, CS SR-TE reserves a percentage of bandwidth in the network and computes CS SR policy bidirectional failover paths with the requested bandwidth, metric type, and constraints. CS SR-TE also maintains a running account of all CS SR reserved bandwidth in the network. CS SR policies are typically used for high-priority services, such as crucial monetary transactions or important live video feeds, for which the operator can now offer a service level that is guaranteed to be better than simple best-effort performance.

  Crosswork Network Controller enables you to provision CS SR-TE policy configurations and easily edit policies, as needed. In addition, the ability to visualize CS SR policies in your network topology allows you to easily verify CS SR policy configurations, details, and path states. With a few clicks you can view Active and Protected paths, operational status, reserved bandwidth pool size, and monitor path failover behavior for individual CS SR policies.

- **Tree Segment Identifier (Tree-SID) policy provisioning and L3VPN service model association**:

  Operators use Tree-SID to implement multicast trees in segment-routed transport networks. Using Crosswork Network Controller, operators can:

  - Create, provision and visualize static Tree-SID policies using the UI, each representing a leaf or node along the path.
  - Create, provision and visualize dynamic Tree-SID policies directly on devices using an API.
  - Visualize dynamic Tree-SID policies using the UI Traffic Engineering page. However, there will be no mapping on the Transport tab if it is attached to an L3 point-to-multipoint VPN service.
  - Associate static mVPN Tree-SID policies associated with existing, or newly created, L3VPN service models (SR MPLS point-to-multipoint) that can also be visualized and analyzed to assist in efficient management and troubleshooting of your multicast network.
  - Modify existing static Tree-SID policies and mVPN Tree-SID policies associated with an L3VPN service model using the UI.
  - Configure link affinities used to specify the link attributes that determine which links are suitable to form a path for the Tree-SID policy and maps each bit position or attribute with a color (making it easier to refer to specific link attributes).

- **Crosswork UI Improvements**:

  Improvements to the Crosswork UI include:

  - Traffic Engineering Dashboard, which includes the TE Dashlet that provides:
    - A high-level summary of RSVP-TE tunnel, SR-MPLS, SRv6, and Tree-SID policy information (total policy count, policy state, metric types for all TE services, and specific data that is filtered upon a one-click selection).
    - Policies and Tunnels under traffic threshold for historical data by displaying RSVP-TE tunnels and SR-MPLS policies that have traffic below the defined threshold in the selected time period. This information may be used to find and filter the unused policies or tunnels.

  - Filtering the data on the dashlet based on the time range you want to view (date, 1 month, 1 week, 1 day, and 1 hour).
  - Policy and Tunnel Change Events: Displays all the policies and tunnels that have had a path or state change event ordered by the event count, within the selected time range. This information helps identify the unstable policies and tunnels.

- Traffic Engineering event and historical data information associated with a device when viewing details for a policy or tunnel. For example, the traffic rate and event history for an SR-MPLS policy can be viewed by selecting the Historical Data tab and clicking on an event. By doing so, you can view the state of the policy or tunnel at that point in time and view additional details, such as Admin and Operational state, Segment type, accumulated metric, delegated PCE, and more so to drill down on the event details.
- Enable/Disable Alarm Status Badge slider: The system allows you to enable or disable the Alarm Status Badge slider for devices and links across various topology views. By disabling the Alarm Status Badge, you can better focus the overlay on an area of interest when troubleshooting.
- Configuration of the Traffic Engineering Data Dashboard Settings (and historical data) for the collection of policy and tunnel metrics, state and path changes, data retention intervals, and the utilization threshold for underutilized LSPs.
- Global search in UI topology: You can now search within the Crosswork Network Controller topology map in the UI. This feature allows you to quickly locate devices based on the following criteria:
  - Civic Location (for example, San Jose)
  - Host/Device name (for example, NAT-01)
  - IP address (for example, 121.10.10.1.1)
- Import and Export geographical objects using Keyhole Markup Language (KML) format:
  - Using the Crosswork Network Controller UI, you can import and export KML files to exam, change, or add device geographical information and see the updates in the UI map. For example, you may use the export function to download your device's data in KML format to your system, exam and/or change the device details, and upload it into a map generator (such as Google Maps) to view your updated device information and coordinates outside of Crosswork Network Controller. You can then use the import function to upload the updated, or browse for a new, KML file back into Crosswork Network Controller. If changes were made, they will now appear in the geographical map after it refreshes. When using the import function, Crosswork Network Controller also provides a sample KML template. The sample KML template provides information on where to identify devices and their coordinates, an optional device name, and the IP address (IPv4 or IPv6) of a device with corresponding coordinates. This template can be used on your system before importing back into Crosswork Network Controller.
- Traffic Engineering device details improvements that will provide options, after selecting a device from the topology map, to select different TE tabs (such as Links, Alarms, SR-MPLS, SRv6, RSVP-TE and others) that provide associated data for the selected device's policies and prefixes.

> **Note** For more information on Crosswork UI improvements, see the Cisco Optimization Engine guide section, Visualize Traffic Engineering Services.

- **Crosswork Provisioning UI Improvements**
  - Dry Run for a deleted service: When decommissioning a service, only the configuration related to a service is deleted on the device. By implementing Dry Run, it shows the user the configuration that is deleted from multiple devices.
  - Edit in json configuration editor: Using the json configuration editor, you can highlight different details that make up the service configuration and edit them directly in the json editor before committing the configuration. For instance, go to **Services & Traffic Engineering > Provisioning (NSO)**. From the Services/Policies panel, select a service (for instance, **L2VPN > L2vpn-Service**). From the list of services available, select the Actions column for the service configuration you want to edit and click **Edit in Json Editor**. The json Configuration editor popup appears. Click within the editor to make changes on select entries or click on the icons on the left to either: **Drag to move this field**, or, **Click to open the actions menu**. If you select **Click to open the actions menu**, a drop-down list appears. Select one of the options (for example: **Insert**, **Duplicate**, or **Remove**). After you complete the configuration edits, click **Commit**.
  - Clone existing services and policies and utilize the json configuration editor to make changes to your cloned configuration. By cloning existing services and policies, you save time and ensure consistency across configurations while maintaining the ability to make specialized modifications. For instance, go to **Services & Traffic Engineering > Provisioning (NSO)**. From the Services/Policies panel, select a service (for instance, **L2VPN > L2vpn-Service**). From the list of services available, select the Actions column for the service configuration you want to copy and click **Clone**. A L2vpn-Service popup appears requiring a new vpn-id for the cloned service. Add a new name and click **Continue**. The json Configuration editor popup appears. After you complete the cloned configuration edits, click **Commit**.
  - Due to NSO Core Function Pack (CFP) model version upgrade, L2VPN, L3VPN or RSVP-TE upgrade is not supported from 4.x to 5.0. SR-TE upgrade from 4.x to 5.0 is supported. Direct upgrade from 3.x to 5.0 is not supported.
  - **Show all fields** toggle option: When editing a service configuration, you can either hide multiple fields that do not pertain to the editable service configuration or you can view all fields by using the **Show all fields** toggle option.
- **Crosswork Infrastructure and Shared Services**:
  - Support for offline licenses, solution-based licenses, and lab licenses
  - Support for user authentication via single sign-on (SSO)
  - Ability to log the user’s source IP address for auditing and accounting
  - High Availability support for Common Licensing Management Service (CLMS)
  - High Availability support for Engineering Management Functions (Inventory, Notification, Fault, and SWIM)
  - Support for visualization of device alarms and events

    - API and Notification support for alarms/events OSS integration
    - Ability to enable SMU installation using a single playbook

- **Services Overlay Visualization Enhancements**:

  Ability to select Basic View or Extended View when visualizing a service overlay. The Basic View is a minimalistic view with no additional details, edge directions, router targets, or EVI/PW IDs. The Extended View includes all details, including edge directions, router targets, and EVI/PW IDs. The services overlay visualization enhancements apply to:

    - Point-to-Point Service Visualization
    - Any-to-Any Service Visualization (L2VPN and L3VPN)
    - Hub and Spoke Service Visualization (L2VPN and L3VPN)
    - Custom Service Visualization (L2VPN and L3VPN)

- **Cisco Service Health**:

  Service Health monitoring is available for both Basic Monitoring and Advanced Monitoring.

Note For help selecting the appropriate monitoring option for your needs, see the section Basic and Advanced Monitoring Rules. In total, Basic + Advanced Monitoring provides up to 52,000 services that can be monitored.

    - Heuristic Package improvements include:
        - IPv6 support that enhances and extends the SRv6 feature support
        - New Basic and Advanced rules for L2VPN (E-LAN and E-Tree) for monitoring (including multi-point feature for E-LAN and E-Tree)
    - High Availability for all Service Health containers.
    - Assurance Graph improvements that include node aggregation and expand/collapse capabilities to view subservice summary information and associated subservices.
    - New subservices, such as:
        - Dynamic subservices implementation (also includes SR-ODN policy)
        - Reservation Protocol for Traffic Engineering (RSVP-TE) Tunnel
        - Bridge Domain
        - Mac Learning
    - Device badge feature displays an orange badge on a healthy device when viewing devices in the topological view and indicate there are down and/or degraded subservices underneath that should be identified and symptoms inspected.
    - Summary node feature summarizes the aggregated health status of child subservices and reports one consolidated health status to a service node. The Summary node feature is available in both L2VPN multipoint Basic and Advanced monitoring models.

- Basic monitoring subservices:
  - Device – Summarizes the health status of all underlying Devices participating in the given L2VPN service.
  - Bridge Domain – Summarizes the L2VPN Service’s Bridge Domain health status across all participating devices.
- Advanced monitoring subservices (in addition to what is also available with Basic monitoring)
  - EVPN – Summarizes the health status of all underlying subservices – BGP Neighbor Health & MacLearning Health across all participating PE endpoints and provides a consolidated overall EVPN health summary status.
  - Transport – Summarizes the health status of all underlying subservices – SR-ODN (dynamic), SR Policy (statically configured) and RSVP TE Tunnel, across all participating PE endpoints and provides a consolidated overall Transport health summary status.
  - SR-PCEP – Summarizes the health status of all the underlying subservices that are monitoring the PCEP sessions. Each underlying subservice monitors the PCEP session health on a particular device participating in the given VPN service.

- Dynamic subservices functionality: In contrast to other subservices, dynamic subservices will be added to or removed from the Assurance Graph according to a service’s intent and/or SR polices present on the devices. Each Summary node (Transport) has either *dynamic.l3vpn.sr.policy* or *l2vpn.sr.odn.policy.dynamic* child subservices for each device with a defined SR intent. And each dynamic subservice will have several *sr.policy.pcc.pm* subservices: one for each relevant SR policy on that device. Dynamic subservices are only for SR-policies on supported l2vpn/l3vpn services.
- Extended CLI support using new Service Health system device packages, that can derive exact sensor paths for metric health calculation, that can now be installed as a bundle when the Service Health application is deployed.

Supported Use Cases

Crosswork Network Controller supports a wide range of use cases allowing operators to manage many aspects of the network. The following describes specific use cases, with details about the Crosswork applications needed to deliver each capability.

- **Orchestrated service provisioning:** Provisioning of layer 2 VPN (L2VPN) and layer 3 VPN (L3VPN) services with underlay transport policies to define, meet, and maintain service-level agreements (SLA), using the UI or APIs. Using Segment Routing Flexible Algorithm (Flex-Algo) provisioning and visualizing to customize and compute IGP shortest paths over a network according to specified constraints.

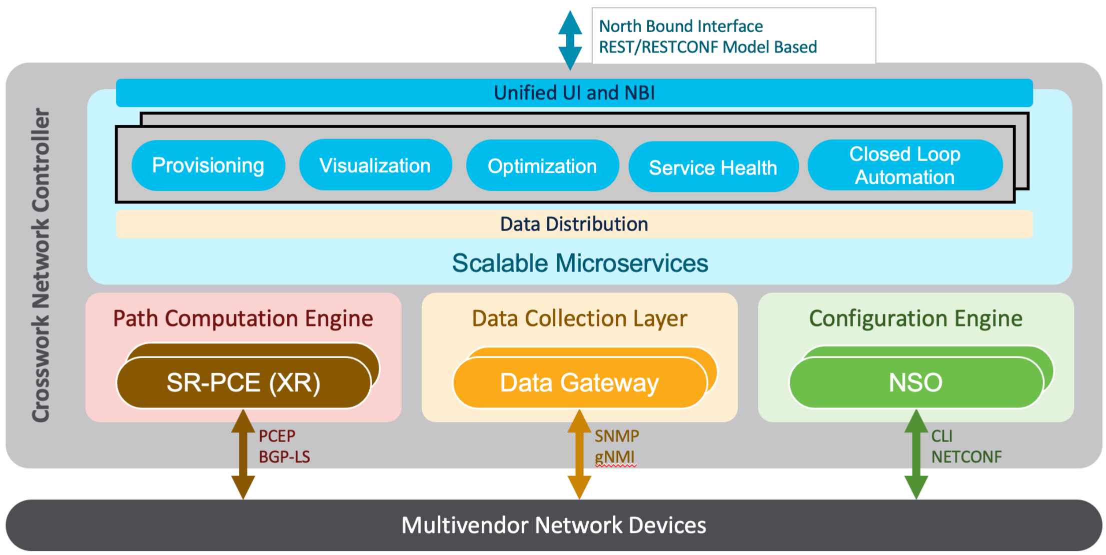
Note An SLA defines the expectations set between a service provider and a customer. The SLA details the products or services that are to be delivered, the point of contact for end-user issues, and metrics by which the effectiveness of the process is both monitored and approved.

- **Real-time network and bandwidth optimization:** Intent-based closed-loop automation, congestion mitigation, and dynamic bandwidth management based on Segment Routing and RSVP-TE. Optimization of bandwidth resource utilization by setting utilization thresholds on links and calculating tactical alternate paths when thresholds are exceeded. The ability to provision SR-Circuit Style policies and visualize them in your network topology provides:
  - Straightforward verification of SR-Circuit Style policy configurations
  - Visualization of SR-Circuit Style details, bi-directional active and candidate paths
  - Operational status details
  - Failover behavior monitoring for individual SR-Circuit Style policies
  - A percentage of bandwidth reservation for each link in the network
  - Manually triggered recalculations of existing SR-Circuit Style policy paths that may no longer be optimized due to network topology changes
- **Local Congestion Management:** Local Congestion Mitigation (LCM) provides localized mitigation recommendations within surrounding interfaces, with the use of standard protocols. Data is gathered in real-time and when congestion is detected, solutions are suggested. LCM has a "human-in-the-loop" aspect which ensures that the control of making changes in the network is in the hands of the operator.
- **Visualization of network and service topology and inventory:** Visibility into device and service inventory and visualization of devices, links, and transport or VPN services and their health status on maps with logical or geographical contexts.
- **Performance-based closed-loop automation:** Automated discovery and remediation of problems in the network by allowing Key Performance Indicator (KPI) customization and monitoring of pre-defined remediation tasks when a KPI threshold is breached. For this use case, Cisco Crosswork Health Insights and Cisco Crosswork Change Automation must be installed.
- **Planning, scheduling, and automating network maintenance tasks:** Scheduling an appropriate maintenance window for a maintenance task after evaluating the potential impact of the task (using WAE Design). Automating the execution of maintenance tasks (such as throughput checks, software upgrades, SMU installs) using playbooks. For this use case, Cisco Crosswork Health Insights and Change Automation must be installed.
- **Secured zero-touch onboarding and provisioning of devices:** Onboarding new IOS-XR devices and automatically provisioning Day0 configuration resulting in faster deployment of new hardware at lower operating costs. For this use case, Cisco Crosswork Zero Touch Provisioning must be installed.
- **Visualization of native SR paths:** Visualizing the native path using the traceroute SR-MPLS multipath command to get the actual paths between the source and the destination can be achieved using Path Query. With Cisco Crosswork Network Controller, a traceroute command runs on the source device for the destination TE-Router ID and assists in retrieving the paths.
- **Provision, Visualize, and Analyze Tree Segment Identifier Policies in Multipath Networks:** Creating and visualizing static Tree-SID policies using the UI. Static mVPN Tree-SID policies associated with existing, or newly created, L3VPN service models (SR MPLS point-to-multi-point) using the Crosswork Network Controller that can also be visualized and analyzed to assist in efficient management and troubleshooting of your multicast network. Configuring link affinities used to specify the link attributes that determine which links are suitable to form a path for the Tree-SID policy and maps each bit position or attribute with a color (making it easier to refer to specific link attributes). Modifying existing static

Tree-SID policies and mVPN Tree-SID policies associated with an L3VPN service model – both edit and delete – using the UI.

Solution Components Overview and Integrated Architecture

The following diagram provides a high-level illustration of how the solution's components work together within a single pane of glass to execute the primary supported use cases.



The following components make up the Cisco Crosswork Network Controller 5.0 solution:

Cisco Crosswork Active Topology

Cisco Crosswork Active Topology's logical and geographical maps provide real-time visibility into the physical and logical network topology, service inventory, and SR-TE policies and RSVP-TE tunnels, all within a single pane of glass. They enable operators to see, at-a-glance, the status and health of the devices, services, and policies. Services and transport policies can be visualized end-to-end as an overlay within the context of the topology map. Cisco Crosswork Active Topology provides device grouping functionality so that operators can set up their maps to monitor exactly the set of devices, services, and locations for which they are responsible. In addition, operators can save custom views for quick and easy access to the views and functionality they use on an ongoing basis.

Cisco Crosswork Optimization Engine

Cisco Crosswork Optimization Engine provides real-time network optimization allowing operators to effectively maximize network capacity utilization, as well as increase service velocity. Leveraging real-time protocols, such as BGP-LS and Path Computation Element Communication Protocol (PCEP), SR-PCE and Crosswork Optimization Engine enables closed-loop tracking of the network state, reacting quickly to changes in network conditions to support a self-healing network.

Cisco Crosswork Data Gateway

Cisco Crosswork Data Gateway is a secure, common collection platform for gathering network data from multi-vendor devices. It is an on-premise application deployed close to network devices. Crosswork Data Gateway supports multiple data collection protocols including MDT, SNMP, CLI, standards-based gNMI (dial-in), and syslog. Any type of data can be collected by Crosswork Data Gateway as long as it can be delivered over one of the supported protocols. In this way, it can provide support for a growing set of use cases and customizations.

To address scale challenges, Cisco Crosswork Data Gateway is implemented as a number of VMs and designed with a distributed architecture in mind. Each lightweight VM manages a subset of the overall network and as the network grows, additional VMs can be added horizontally to address the new demands on the compute resources. It also supports a flexible redundancy configuration based on the operator's needs. After the initial setup, Cisco Crosswork Network Controller automatically orchestrates the collection across the multiple Cisco Crosswork Data Gateway VMs.

APIs and configuration examples are available to illustrate how to add new collection jobs (outside of those built for you by Cisco Crosswork Network Controller) to gather additional information from your network. The collected data can be published to approved destinations. Supported destinations are Kafka and gRPC messaging bus.

Crosswork Common UI and API

All Cisco Crosswork Network Controller's functionality are provided within a single, common graphical user interface. This common UI brings together the features of all Crosswork Network Controller's components, including common inventory, network topology and service visualization, service and transport provisioning, and system administration and management functions. When optional add-on Crosswork components are installed, their functionalities are also fully integrated into the common UI. Having all functionality within a common UI, instead of having to separately navigate individual application UIs, enhances the operational experience and increases productivity.

A common API enables Crosswork Network Controller's programmability. The common APIs provides a single access point for all APIs exposed by various built-in components. The API provides a REST-based Northbound Interface to external systems (e.g., OSS systems) to integrate with Cisco Crosswork Network Controller. RESTCONF and YANG data models are made available for optimization use cases. For details about the APIs and examples of their usage, see the Cisco Crosswork Network Automation API Documentation on Cisco DevNet.

Crosswork Infrastructure and Shared Services

The Cisco Crosswork Infrastructure provides a resilient and scalable platform on which all Cisco Crosswork components can be deployed. This infrastructure and shared services provide:

- A single API endpoint for accessing all APIs of Crosswork applications deployed
- A shared Kafka bus to pass data between applications
- Shared database(s) (such as relational and graph) for applications to store data
- A singe shared database to store all gathered time-series data from the network
- A robust Kubernetes-based orchestration layer to provide for process-level resiliency
- Tools for monitoring the health of the infrastructure and the cluster of virtual machines (VMs) on which it resides

Cisco Crosswork Health Insights and Cisco Crosswork Change Automation

Cisco Crosswork Health Insights and Cisco Crosswork Change Automation are components that can optionally be installed with Cisco Crosswork Network Controller.

Cisco Crosswork Health Insights performs real-time Key Performance Indicator (KPI) monitoring, alerting, and troubleshooting. Cisco Crosswork Health Insights enables programmable monitoring and analytics. It provides a platform dynamically for addressing changes to the network infrastructure. Cisco Crosswork Health Insights builds dynamic detection and analytics modules that allow operators to monitor and alert about network events based on user-defined logic.

Cisco Crosswork Change Automation automates the process of deploying changes to the network. Orchestration is defined via an embedded Ansible Playbook and then configuration changes are pushed to Cisco Network Services Orchestrator (NSO) to be deployed to the network.

These components within Cisco Crosswork Network Controller enable closed-loop discovery and remediation of problems in the network. Operators can match alarms to pre-defined remediation tasks, which are performed when a defined Key Performance Indicator (KPI) threshold is breached. This reduces the time it takes to discover and repair a problem while minimizing the risk of human error resulting from manual network operator intervention.

Cisco Crosswork Zero-Touch Provisioning (ZTP)

Cisco Crosswork ZTP can optionally be installed with Cisco Crosswork Network Controller.

Cisco Crosswork ZTP is an integrated turnkey solution for automatically onboarding and provisioning new IOS-XR devices, resulting in faster deployment of new hardware at lower operating costs. Operators can quickly and easily bring up devices using a Cisco-certified software image and a day-zero software configuration. After it is provisioned in this way, the new device is onboarded to the Crosswork device inventory where it can be monitored and managed along with other devices.

Cisco Crosswork ZTP offers Secure ZTP functionality in addition to the Classic ZTP functionality. Secure ZTP is based on RFC 8572 standards and uses secure transport protocols and certificates to verify devices and perform downloads. Secure ZTP is useful when public Internet resources must be traversed to reach remote network devices, or when the devices are from third-party manufacturers. With Secure ZTP, the device and the Cisco Crosswork ZTP bootstrap server authenticate each other using the device's Secure Unique Device Identifier (SUDI) and Crosswork server certificates over TLS/HTTPS. After a secure HTTPS channel is established, the Crosswork bootstrap server allows the device to request to download and apply a set of signed image and configuration artifacts adhering to the RFC 8572 YANG schema. After the image (if any) is downloaded and installed, and the device reloads with the new image, the device downloads configuration scripts and executes them.

Cisco Network Services Orchestrator

Cisco Network Services Orchestrator (NSO) is an orchestration platform that makes use of pluggable function packs to translate network-wide service intent into device-specific configuration. Cisco NSO provides flexible service orchestration and lifecycle management across physical network elements and cloud-based virtual network functions (VNFs), fulfilling the role of the Network Orchestrator (NFVO) within the ETSI (European Telecommunications Standards Institute) architecture. It provides complete support for physical and virtual network elements, with a consistent operational model across both. It can orchestrate across multi-vendor environments and support multiple technology stacks, enabling the extension of end-to-end automation to virtually any use case or device.

Cisco NSO has a rich set of APIs designed to allow developers to implement service applications. It provides the infrastructure for defining and executing the YANG data models that are needed to realize customer services. It is also responsible for providing the overall lifecycle management at the network service level.

Service and device models, written using YANG modelling language, enable Cisco NSO to efficiently 'map' service intent to device capabilities and automatically generate the minimum required configuration to be deployed in the network. This feature, facilitated by Cisco NSO's FASTMAP algorithm, is capable of comparing current configuration states with a service's intent and then generating the minimum set of changes required to instantiate the service in the network.

All Crosswork components that are included in Cisco Crosswork Network Controller or are optional add-ons, with the exception of Cisco Crosswork ZTP, require integration with Cisco NSO.

Cisco Crosswork Network Controller requires the following Cisco NSO function packs:

- SR-TE core function pack (CFP) enables provisioning of explicit and dynamic segment routing policies, including SRv6, and on-demand SR-TE policy instantiation for prefixes with a specific color.
- Sample function packs for IETF-compliant L2VPN and L3VPN provisioning. These function packs provide baseline L2VPN and L3VPN provisioning capabilities, based on IETF NM models. Prior to customization, these sample function packs enable provisioning of the following VPN services:
  - L2VPN:
    - Point-to-point VPWS using Targeted LDP
    - Point-to-point VPWS using EVPN
    - Multipoint VPLS using EVPN (with service topologies ELAN, ETREE, and Custom)
  - L3VPN
  - Sample IETF-compliant RSVP-TE function pack intended as a reference implementation for RSVP-TE tunnel provisioning, to be customized as required.

> **Note**
> - By default, the IETF-compliant NM models are used. If your organization wishes to continue to use the Flat models that were provided with the previous version, a manual setup process is required.
> - The Cisco NSO sample function packs are provided as a starting point for VPN service provisioning functionality in Cisco Crosswork Network Controller. While the samples can be used "as is" in some limited network configurations, they are intended to demonstrate the extensible design of Cisco Crosswork Network Controller. Answers to common questions can be found on Cisco Devnet and Cisco Customer Experience representatives can provide answers to general questions about the samples. Support for customization of the samples for your specific use cases can be arranged through your Cisco account team.
> - Cisco NSO currently does not support bundle ethernet (BE), route distinguisher (RD), or BGP route-target (RT) functions with L2VPN EVPN. Although it does support multihoming and L2VPN route policy, there is no option to specify an RD value in L2VPN for an EVPN ELAN/ETREE, nor is there an option to specify load balancing type. To perform these functions, contact your Cisco account team for a set of custom configuration templates and advice on configuring bundles manually.

Cisco Segment Routing Path Computation Element (SR-PCE)

Cisco SR-PCE is an IOS-XR multi-domain stateful PCE supporting both segment routing (SR) and Resource Reservation Protocol (RSVP). Cisco SR-PCE builds on the native Path Computation Engine (PCE) abilities within IOS-XR devices, and provides the ability to collect topology and segment routing IDs through BGP-LS, calculate paths that adhere to service SLAs, and program them into the source router as an ordered list of segments. A Path Computation Client (PCC) reports and delegates control of head-end tunnels sourced from the PCC to a PCE peer. The PCC and PCE establish a Path Computation Element Communication Protocol (PCEP) connection that SR-PCE uses to push updates to the network and re-optimize paths where necessary.

Cisco SR-PCE can either reside on server resources using virtualized XRv9000 , or as a converged application running within IOS-XR Routers.

> **Note** Adding static routes for auto-discovering the scale nodes from SR-PCE after 2,000 nodes is not supported.

Cisco Service Health

- Service Health substantially reduces the time required to detect and troubleshoot service quality issues. It monitors the health status of provisioned L2/L3 VPN services and enables operators to pinpoint why and where a service is degraded. It can also provide service-specific monitoring, troubleshooting, assurance, and proactive causality through a heuristic model that visualizes the:
  - Health status of sub-services (device, tunnel) to a map when a single service is selected

- Service logical dependency tree and help the operator in troubleshooting in case of degradation by locating where the problem resides, an indication of possible symptoms, and impacting metrics in case of degradation
- Historical view of service health status up to 60 days

Service Health also provides the following:

- Service Health monitoring is available for both Basic Monitoring and Advanced Monitoring options. For help selecting the appropriate monitoring option for your needs, see the section **Basic and Advanced Monitoring Rules**.
- Service Health provides **Internal Storage** of monitoring data up to a maximum limit of 50 GB. This data is stored on your system. If you exceed the limit of the internal storage, historical data will be lost. If you choose to extend Service Health storage capacity, you can optionally configure **External Storage** in the cloud using an Amazon Web Services (AWS) cloud account. By leveraging External Storage, all existing internal storage data will be automatically moved to the external cloud storage (see **Configuring Service Health External Storage Settings** appendix for more details) and your internal storage will act locally as cache storage. Configuring External Storage for Service Health ensures you will not lose historical data for services that continue to monitor a service's health, and will retain service health data for any service you choose to stop monitoring when you select the option to retain historical monitoring service for the data. For more information on Internal and External Storage, and how to retain historical monitoring service data when stopped, see the Appendix sections **Configuring Service Health External Storage Settings** and **Stopping Service Health monitoring.**

Note If you anticipate monitoring a large amount of Service Health services, Cisco recommends you configure External Storage after you install Service Health and before you begin monitoring services so to avoid exceeding the Internal Storage and losing historical data.

- To view subservices supported by Service Health L2VPN/L3VPN, see the **Service Health Supported Subservices** appendix section. Details are provided that define which subservices are supported by each VPN service flavor.
- Service Health supports point-to-point L2VPN.

Note Currently, Service Health does not support multipoint L2VPN.

- Service Health supports integration with standalone Network Services Orchestrator (NSO) or NSO Layered Service Architecture (LSA).
- NSO LSA support is limited to one CFS node and two RFS nodes. These additional NSO types serve as a high availability feature. By distributing your devices across the different types, the LSA feature in Service Health allows for dynamic configurations for assurance.

  To manage the Service Health provider Access, select **Administration > Manage Provider Access**. The Providers screen appears. See the Crosswork Administration guide and NSO documentation for additional, detailed information.

- The Service Health Collection Jobs administrative option provides the capability to view Parameterized Jobs (template-based collection jobs) that supports a greater number of jobs, adding the ability to view CLI collection jobs. This is useful when troubleshooting collection job issues by examining details of individual devices using Parameterized Jobs. Devices are identified by their Context ID (protocol) to determine if they are GMNI, SNMP, or CLI-based jobs. Additionally, you may export the collection job information to review. The information is collected at the time the export is initiated and stored in a .csv file.

Note When exporting the collection status, you must fill in the information each time an export is executed. In addition, make sure to review the Steps to Decrypt Exported File content available on the Export Collection Status pop up to ensure you can access and view the exported information.

- Service Health provides expanded redundancy/High Availability (HA) for Assurance Graph Manager, Expression Orchestrator, and Crosswork Expression Tracker microservices (two instances are now available). To view, select **Administration** > **Crosswork Manager**. In the Crosswork Summary tab, select Crosswork Service Health to view the Application Details screen and Microservices.
    - For example, if you click the Assurance Graph Manager, two redundant/high availability instances appear. In certain situations, one of the instances will be in the active-active mode while the other is in the active-standby mode. This ensures that if one instance goes down, the second acts as a redundant, HA, backup.
- Heuristic Packages: Three additional Rules have been added to assist in Basic monitoring level rules, where a rule to generate Assurance Graph information, for example Basic L2VPN NM P2P services, can be used along with two sub services:
    - Rule-L2VPN-NM- Basic
    - Rule-L2VPN-NM-P2P-Basic
    - Rule-L3VPN-NM-Basic
- Heuristic Package Metrics now has the capability for CLI based metrics and GMNI filtering customizations of packages.

Multi-Vendor Capabilities

Today's networks have typically been built up over time and incorporate multiple vendors and multiple generations of hardware and software. Furthermore, there is a lack of industry standardization, making support for these networks using a single tool challenging.

Service providers require an integrated solution to manage third-party devices that will reduce operational expenses and maintenance overhead, as well as eliminate the need to build custom applications to deploy and maintain different vendor products for a single network.

Because it uses standards-based protocols, Cisco Crosswork Network Controller has multi-vendor capabilities for:

- Network service orchestration via Cisco Network Services Orchestrator using CLI and Netconf/YANG. Cisco Network Services Orchestrator is a YANG model-driven platform for automating provisioning, monitoring, and managing applications and services across multi-vendor networks.
- Telemetry data collection using SNMP with standards-based MIBs, syslog, and gNMI with standard OpenConfig models. Cisco Crosswork Data Gateway also supports Native YANG data models for external destinations and proprietary SNMP MIBs with custom packages.
- Topology and transport discovery via SR-PCE, using IGP and BGP-LS, with link utilization and throughput collected via SNMP using standard MIBs.
- Transport path computation using PCEP.

Note For third-party network device support, use cases must be validated by Cisco Customer Experience representatives in the customer's multi-vendor environment, especially if legacy platforms and non-standard devices or services are involved.

Extensibility

The Cisco Crosswork Network Controller provisioning functionality can be extended using the application programming interfaces (APIs). For more information about the APIs, see the Cisco Crosswork Network Automation API Documentation on Cisco DevNet.

The provisioning UI is extensible as it is rendered based on the YANG model. When new services are introduced, they can be easily incorporated.

CHAPTER 2

UI Overview

This section explains the following topics:

Log In

Log into the web UI by entering the following URL in the browser’s address bar:

```
https://<Crosswork Management Network Virtual IP (IPv4)>:30603/
https://[<Crosswork Management Network Virtual IP (IPv6)>]:30603/
```

> **Note** The IPv6 address in the URL must be enclosed with brackets.

In the Log In window, enter the username and password configured during installation and click **Log In**.

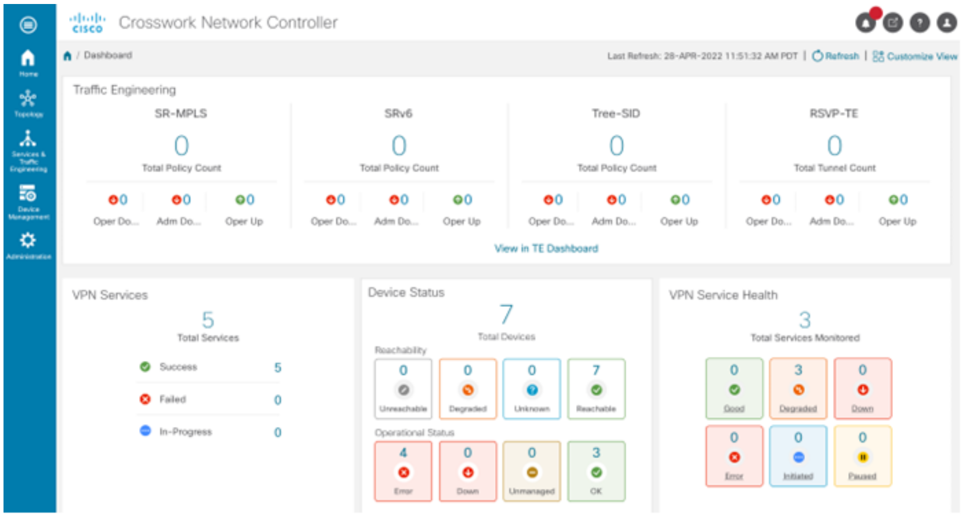
Self-signed certificate: At first-time access, some browsers display a warning that the site is untrusted. When this happens, follow the prompts to add a security exception and download the self-signed certificate from the server. After you download the certificate, the browser accepts the server as a trusted site in all future login attempts.

CA signed certificate: For production use, a CA signed certificate may be installed and is recommended to avoid a warning that the site is untrusted.

> **Note** For information on installing CA signed certificates, see the Manage Certificates topic in the *Cisco Crosswork Network Controller Administration Guide*.

Dashboard

After successful login, the Home page opens. The Home page displays the dashboard which provides an at-a-glance operational summary of the network being managed. The dashboard is made up of a series of dashlets. The specific dashlets included in your dashboard depend on which Cisco Crosswork applications you have installed. Links in each dashlet allow you to drill down for more details.



> **Note** Your Dashboard may differ from this screen capture, which displays optional components you may not have installed.

Navigation

The main menu along the left side of the window provides access to all features and functionality in Cisco Crosswork Network Controller, as well as to device management and administrative tasks. The Home, Topology, Services & Traffic Engineering, Device Management and Administration menu options are available when all native components of Cisco Crosswork Network Controller are installed. Additional menu options are available in the main menu depending on which Cisco Crosswork add-on applications are installed.

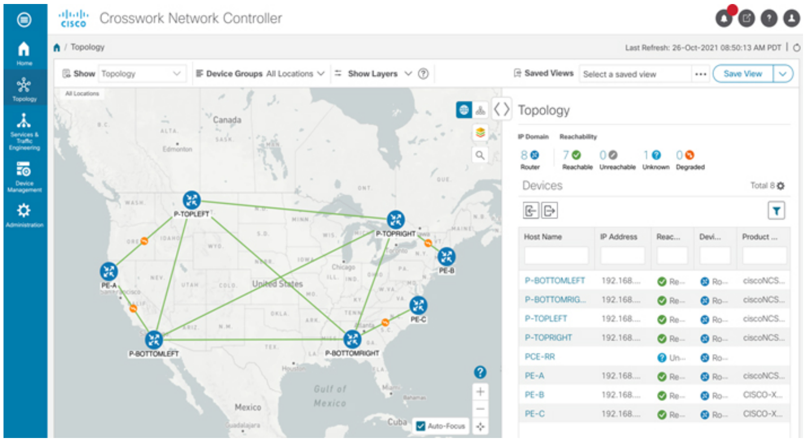
Home

The home page contains the dashboard, as described in the Dashboard topic.

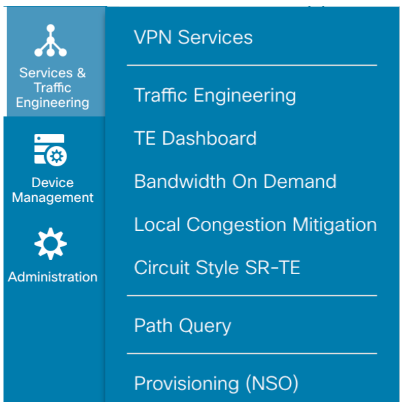
Topology

Users can display the network device and link topology on a logical map or a geographical (geo) map. The logical map shows devices and their links, positioned according to an automatic layout algorithm. The geo map shows single devices, device groups, device clusters, links, and tunnels, superimposed on a map of the world. Each device location on the map reflects the device's GPS coordinates (longitude and latitude). Operators supply this location information in CSV or KML files uploaded using the Crosswork UI.

The Topology page consists of a map showing managed devices and the links between them, along with a device table listing managed devices. In the map you can see the status and health of the devices at a glance. Clicking on a device in the table highlights the device on the map and shows details of the device and its associated links. Use the toggle buttons to switch between the geographical map (shown below) and the logical map. Clicking on the question mark in the map provides a detailed legend of the various symbols and their meaning.



Services & Traffic Engineering

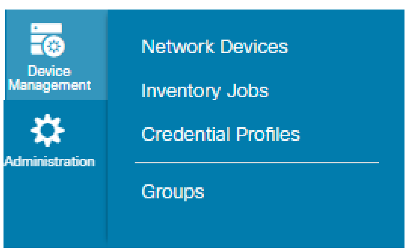


The Services & Traffic Engineering menu provides access to VPN and transport provisioning and visualization functionality, bandwidth management functionality, as well as access to the configuration pages used to enable Feature Packs. For more information, click here to see the Crosswork Optimization Engine 5.0 User Guide.

Choose **VPN services** or **Traffic Engineering** to see managed VPN services, SRv6 policies, or SR-TE policies/RSVP-TE tunnels within the context of a logical or geographical map.

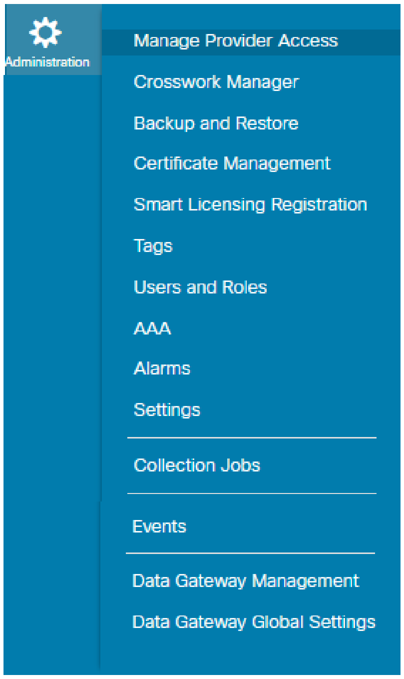
Choose **Provisioning (NSO)** to access the provisioning UI rendered from the Cisco Network Services Orchestrator models. Here you can create L2VPN and L3VPN services, SR-TE policies, SRv6 policies, SR ODN templates, and RSVP-TE tunnels. You can also create the resources required for these services and policies, such as resource pools, route policies for L2VPN and L3VPN services, and SID lists for SR-TE policies. SR-TE policies and RSVP-TE tunnels can be attached to VPN services to define and maintain SLAs by tracking network changes and automatically reacting to optimize the network.

Device Management



The Device Management menu provides access to device-related functionality, including adding, managing, and grouping devices, creating and managing credential profiles, and viewing a history of device-related jobs.

Administration



The Administration menu provides access to all system management functions, data gateway management, Crosswork cluster and application health, backup and restore, smart licensing and other setup and maintenance functions that are typically performed by an administrator.

Click here to see the Crosswork Network Controller 5.0 Administration Guide for information about these functions.

CHAPTER 3

Orchestrated Service Provisioning

This section explains the following topics:

Overview

By using the scenario workflows described in this section, we are providing examples of how to configure the system to deliver the operator’s intended configuration. These scenarios do not fully demonstrate all of the capabilities of Crosswork Network Controller. They are intended to demonstrate the flexibility of the platform. Additional customization is possible either by leveraging the resources available on Cisco DevNet or through engagement with Cisco Customer Experience.

Objective

Provision a set of VPN services with underlay transport policies that will meet and maintain service-level agreements (SLAs) between the service provider and the customer. An SLA defines the service-delivery expectations agreed upon between the service provider and the customer. The SLA details the products or services that the provider is to deliver to the customer, the provider's point of contact to which the customer will bring service issues, and the metrics the provider and customer both use to monitor compliance with the SLA.

Challenge

The service-provider network state changes continuously and so quickly that it is difficult to track and react to network problems fast enough to avoid congestion and maintain SLA compliance. In a typical lifecycle, there is a feedback loop that traditionally requires manual monitoring and intervention, which is time- and resource-intensive.

Solution

With network automation, the objective is to automate the feedback loop to enable quicker reaction to and remediation of network events. With Crosswork Network Controller, network operators can orchestrate L2VPN and L3VPN services across the transport network, via a programmable interface, in a very quick and efficient manner. Segment routing traffic engineering (SR-TE) polices can be configured to continuously track network changes and automatically react to optimize the network. These SR-TE polices can serve as the underlay configuration for the VPN services to automatically maintain the SLAs.

The services required for this solution can be created and managed using the Crosswork Network Controller UI. L2/L3 VPN Yang model-based service intents are implemented using the Cisco Network Services Orchestrator sample function packs, which provide sample service models that can be extended and fine-tuned to meet customer needs. Optionally, Service Health monitoring can be enabled to see which services are working as provisioned, if issues have been flagged, and what symptoms are detailed so to quickly address and fix.

Note The Network Services Orchestrator sample function packs are provided as a starting point for VPN service provisioning functionality in Crosswork Network Controller. While the samples can be used “as is” in some limited network configurations, they are intended to demonstrate the extensible design of Crosswork Network Controller. Answers to common questions can be found on Cisco Devnet and Cisco Customer Experience can provide answers to general questions about the samples. Support for customization of the samples for your specific use cases can be arranged through your Cisco account team.

How Does it Work?

1. User creates an SR-TE policy/On-Demand Next Hop (ODN) template with intent (e.g., bandwidth, latency) using the Cisco Crosswork Network Controller UI or APIs.
2. User creates a VPN service using the UI or APIs and specifies the following:
   - The endpoints participating in the VPN
   - Other required VPN parameters
   - The SR-TE policy/ODN template that is to be associated with the VPN service
3. During the provisioning process for the above steps, Cisco Network Services Orchestrator configures the SR-TE policy and the VPN service on the specified endpoints.
4. When the service is active, the network interacts with the SR-PCE to dynamically program the path that meets the intent in the configured SR-TE policy/ODN template. The headend device requests a path from the SR-PCE via PCEP (for dynamic SR-TE policies). If the request specifies bandwidth, the SR-PCE gets the path from Cisco Crosswork Optimization Engine.
5. The SR-PCE sends the path to the headend device via PCEP and updates the headend if path changes are required.

Usage Scenarios

We will walk you through the following usage scenarios that illustrate the execution of the orchestrated service provisioning use case using the Cisco Crosswork Network Controller UI:

- Scenario 1 – Implement and Maintain SLA for an L3VPN Service for SR-MPLS (using ODN)
- Scenario 2 – Implement and Maintain SLA for an L3VPN Service for SRv6 (using ODN)

- Scenario 3 – Mandate a Static Path for an EVPN-VPWS Service using an Explicit MPLS SR-TE Policy
- Scenario 4 – Provision an L2VPN service over an RSVP-TE tunnel with reserved bandwidth
- Scenario 5 – Provision a Soft Bandwidth Guarantee with Optimization Constraints

Additional Resources

- For information about segment routing and segment routing policies, click here to see the Crosswork Optimization Engine 5.0 User Guide.
- Cisco Network Services Orchestrator documentation is included in the latest Network Services Orchestrator image here.

Scenario 1 – Implement and Maintain SLA for an L3VPN Service for SR-MPLS (using ODN)

This scenario walks you through the procedure for provisioning an L3VPN service with a specific SLA objective: all traffic for this service must take the lowest-latency path. The customer requires this low-latency path for this service, as all of this service's traffic is high priority. The customer also wants to use disjoint paths; that is, two unique paths that steer traffic from the same source but to two unique destinations, avoiding common links so that there is no single point of failure.

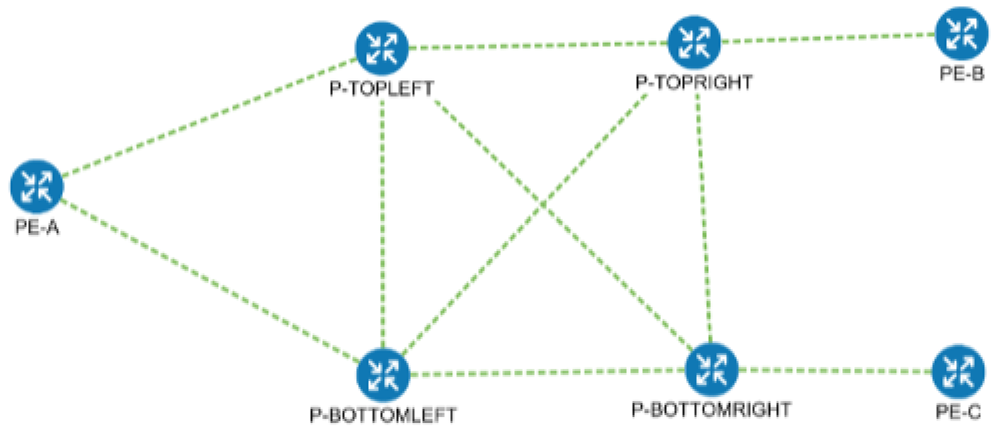
We'll achieve this using Segment Routing (SR) On-Demand Next Hop (ODN). SR ODN allows a service headend router to automatically instantiate an SR-TE policy to a BGP next-hop when required (on-demand). We configure the headend with an ODN template with a specific color that identifies the SLA. Crosswork will optimize the traffic path when it receives a prefix with that SLA-specific color. We define prefixes in a route policy that is associated with the L3VPN.

Crosswork Network Controller continues to monitor the network and will automatically optimize the network based on the defined SLA, in a closed loop.

Within this workflow, we also have the option to enable Crosswork's Service Health monitoring, and to use Flex-Algo as a constraint on how paths are computed and visualized. With Service Health monitoring, operators can gather quick insights into degraded and down services and then use these insights to visualize, inspect, and troubleshoot for improved network optimization.

With Flex-Algo, we can customize IGP shortest-path computations using algorithms we define. IGP will compute paths based on a user-defined combination of metric types and constraints, and present a filtered topology view based on our specific Flex-Algo definitions.

The following topology provides the base for this scenario:



In this scenario, we will:

- Create a segment routing ODN template with a specific color on the endpoints to ensure that traffic is transported within an LSP (underlay) and that a best-path tunnel is created dynamically when a prefix with the specified color is received. The ODN template defines the SLA on which you want to optimize the path. In this case, we will optimize on latency.
- Specify that the computed paths be disjoint: they will not share the same link.
- Create a route policy on each endpoint to be used to bind the L3VPN to the ODN template. This route policy adds a color attribute to the customer prefixes and advertises via BGP to other endpoints. This color attribute is used to indicate the SLA required for these prefixes.
- Create an L3VPN service with 3 endpoints and enable Service Health monitoring.
- Visualize how this overlay/underlay configuration optimizes the traffic path and automatically maintains the SLA while monitoring your service's health.

Assumptions and Prerequisites

- To use ODN, BGP peering for the prefixes must be configured between the endpoints or PEs. Usually for L3VPN, this is the VPNv4 and VPNv6 address family peering.
- For Service Health enablement, Service Health must be installed. See the Crosswork Network Controller Installation Guide chapter, Install Crosswork Applications.
- Before using Service Health's Assurance Graph, ensure that topology map nodes have been fully configured and created with a profile associated to the service. If not, Subservice Details metrics will show that no value has yet to be reported.
- L3VPN service monitoring supports XR devices and does not support XD devices. Thus, after an L3VPN service is created and Service Health monitoring is enabled, if a provider and devices are removed, and then added back, service monitoring remains in a degraded state with a METRIC_SCHEDULER error. To recover, service monitoring must be stopped and restarted.
- (Optional) Flexible Algorithms, and the IDs that are used, must be configured in your network.

> **Note** Screen captures, showing services and data, are for example purposes only and may not always reflect the devices or data described in the workflow content.

Step 1 Create an ODN template to map color to an SLA objective and constraints

Disjointness constraints work by associating a disjoint group ID with the ODN template, and all tunnels with the same disjoint group ID will be disjoint, i.e., they will use different links, nodes and shared risk link groups depending on how the disjoint groups are configured.

We will create the following ODN templates:

- Headend PE-A, color 72, latency, disjoint path (link), group ID 16 - L3VPN_NM-SRTE-ODN_72-a
- Headend PE-A, color 71, latency, disjoint path (link), group ID 16 - L3VPN_NM-SRTE-ODN_71-a
- Headend PE-B and PE-C, color 70, latency - L3VPN_NM-SRTE-ODN_70
- Headend PE-B, color 72, latency - L3VPN_NM-SRTE-ODN_72-b
- Headend PE-C, color 71, latency - L3VPN_NM-SRTE-ODN_71-c

For example purposes, we will show how to create the first ODN template - L3VPN_NM-SRTE-ODN_72-a. The other ODN templates can be created using the same procedure.

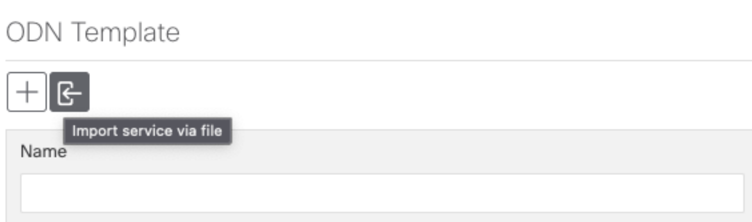
Before you begin

In this step, we will create an ODN template on each endpoint. The ODN template specifies the color and the intent; in this case, latency and disjointness. This ODN template will be used to dynamically create tunnels (on-demand) when prefixes with matching colors are received via BGP. Traffic to these prefixes will be automatically steered into the newly created tunnels, thereby meeting the SLA objective and constraints intended for these prefixes and signaled using colors in the BGP routes.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **SR-TE** > **ODN-Template**.

Step 2 Click **+** to create a new template and give it a unique name. In this case, the name is **L3VPN_NM-SRTE-ODN_72-a**. Click **Continue**.

You may also browse for an existing template on your system so to import the file. The information from the imported file is populated into the form.



Step 3 Choose the head-end device, **PE-A**, and specify the color **72**.

Step 4 Under dynamic, select **“latency”** as the metric-type. This is the SLA objective on which we are optimizing.

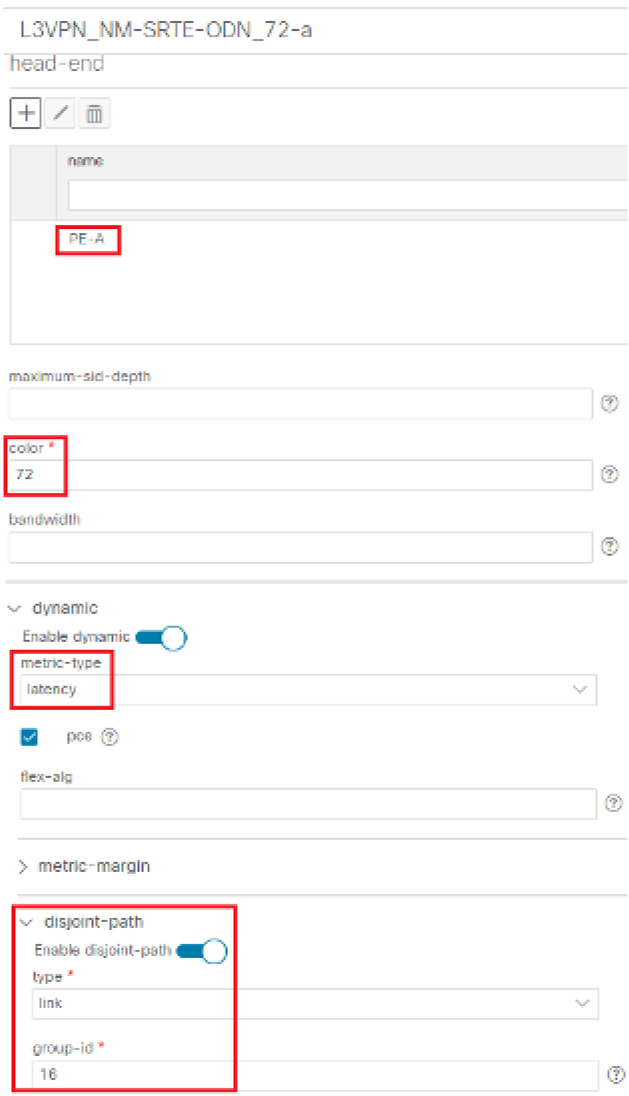
Step 5 Select the **pce** check box to specify that the path should be computed by the SR-PCE, not by the Path Computation Client (PCC).

Step 6 Define the required constraints. In this case, we want the computed paths to be disjoint in that they must not share a link.

Under disjoint-path, choose **link** as the type, and specify a numeric group ID, in this case, **16**, as the group-id.

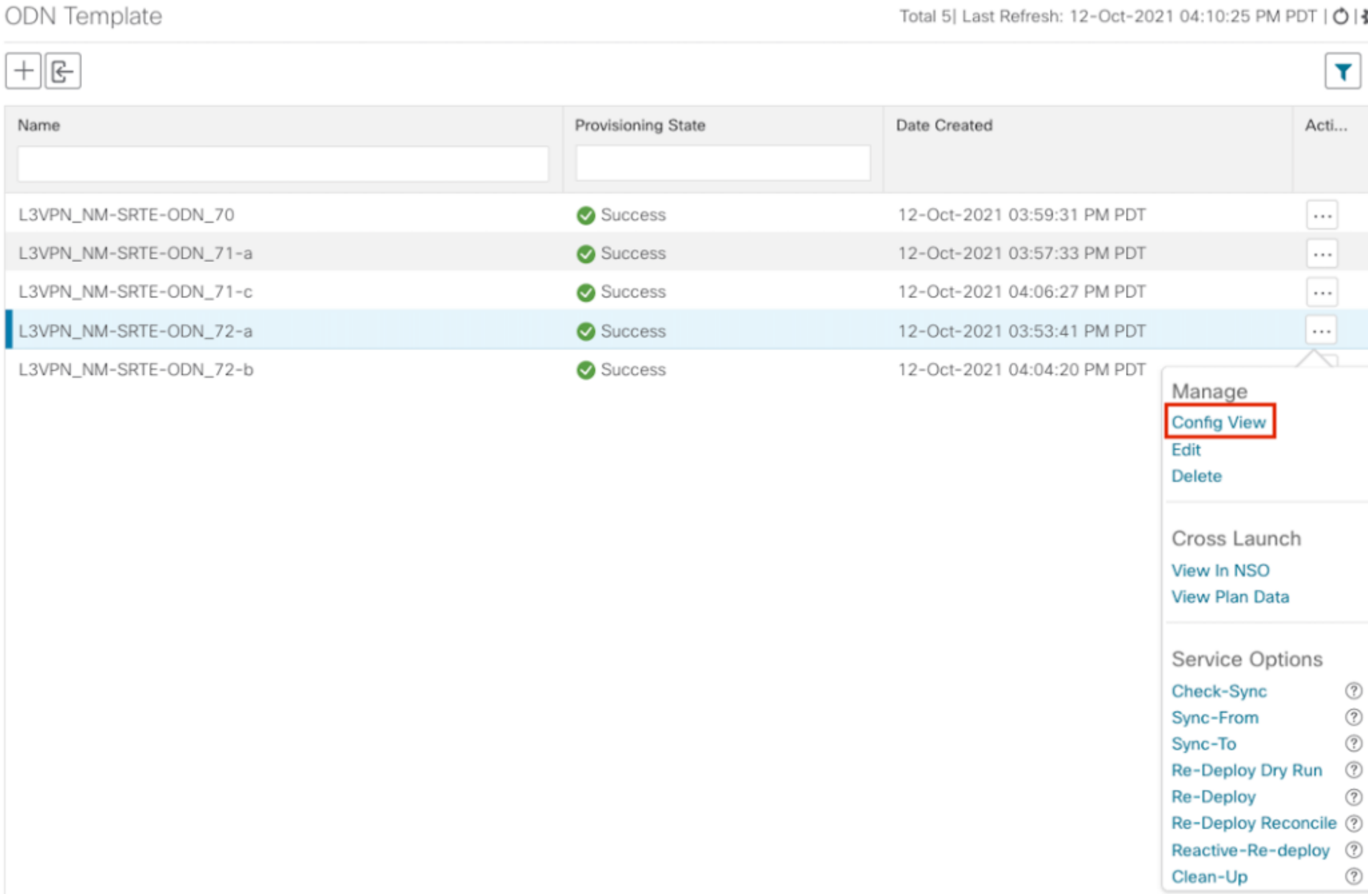
Note You may choose the group ID. All paths requested with the same group-id will be disjoint from each other.

Note Optionally, you may configure Flex-Algo as a constraint.



Step 7 Commit your changes or click **Dry Run** to check what will be configured on the devices before you commit.

Step 8 Check that the new ODN template appears in the table and its provisioning state is **Success**. Click ⋯ in the Actions column and choose **Config View** to see the Yang model-based service intent data that details the ODN template you created.



Step 9 Create the other ODN templates listed above in the same manner.

Step 2 Create an L3VPN Route Policy

In this step, we will create a route policy for each endpoint, and we will specify the same color as defined in the ODN template for that endpoint. The route policy defines the prefixes to which the SLA applies. When traffic from the specified network with a matching color is received, paths are computed based on the SLA defined in the ODN template. We will create the following route policies:

- Color 70, IPv4 prefix 70.70.70.0/30 - L3VPN_NM-SRTE-RP-PE-A-7
- Color 71, IPv4 prefix 70.70.71.0/30 - L3VPN_NM-SRTE-RP-PE-B-7
- Color 72, IPv4 prefix 70.70.72.0/30 - L3VPN_NM-SRTE-RP-PE-C-7

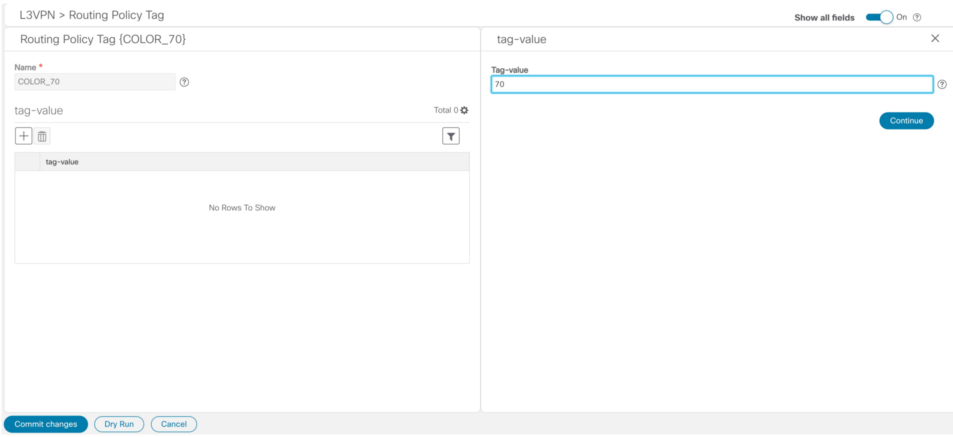
First, we will create the routing policy tag and routing policy destination prefix. The routing policy prefixes should match with the subnet prefix configured on the PE devices in the service.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **Routing Policy Tag**.

Step 2 Click + to create a new routing policy tag and type the name of the tag set: **COLOR_70**. Click **Continue**.

This is used as a label to reference the set in actions and conditions.

Step 3 Under tag-value, click + and type the Tag-value: **70**.



The tag value may be a number between **1 – 4294967295** and should match to a color value.

Step 4 Click **Continue**. The new routing policy tag name with the new tag value is visible. Click **Commit changes**.

Create the other two routing policy tags (**COLOR_71** and **COLOR_72**) and tag values (**71** and **72**) by following the same steps above.

Now create the routing policy destination prefixes.

Step 5 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **Routing Policy Destination Prefix.**

Step 6 Click + to create a new routing policy destination prefix and type the name: **DEST_PREFIX_SET_70**.

The name of the prefix set will reference the set in match conditions.

Step 7 For Mode, select **ipv4**.

Step 8 Expand prefixes and click + to add the ip-prefix to the prefix-list.

Step 9 or Ip-prefix, type **70.70.70.0/30** and click **Continue**.

Create the other two routing policy destination prefixes (**DEST_PREFIX_SET_71** and **DEST_PREFIX_SET_72**) by following the same steps.

Now we are ready to create the first route policy - L3VPN_NM-SRTE-RP-PE-A-7. The other route policies can be created using the same procedure.

Step 10 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3vpn** > **Routing Policy**.

Step 11 Click **+** to create a new route policy and type a unique name for the top-level policy definition: **L3VPN_NM-SRTE-RP-PE-A-7**. Click **Continue**. The statements section appears.

Note The Route Policy statement defines the condition and action taken by the system.

Step 12 Expand statements and click + to add the name of the policy statement (such as **stmt1**) and click **Continue**. The statement{stmt1} panel appears showing **conditions** and **actions** sections.

Step 13 Expand conditions and then expand match-dest-prefix-set before selecting the Prefix-set list and select **DEST_PREFIX_SET_70**. This is what references a defined prefix set.

Note Once selected, the **Enable match-dest-prefix-set** toggle, which will match a referenced prefix-set according to the logic defined in the match-set-options list, switches on.

Step 14 Expand actions and then expand bgp-actions.

Step 15 For bgp-actions, slide the Enable bgp-actions toggle to the on position. By toggling bgp-actions on, it defines the top-level container for BGP-specific actions.

Step 16 Now expand set-ext-community. Slide the Enable-set-ext-community toggle to the on position. By toggling set-ext-community on, it sets the extended community attributes.

Step 17 For Method and reference, select the Ext-community-set-ref list and select **COLOR_70**. The Ext-community-set-ref references a defined extended community set by name.

Note Creating routing-policy tag-set is mandatory and needs to be mapped here.

Step 18 Click **X** in the top-right corner to close the statement{stmnt1} panel and click **Commit changes**.

Step 19 Create the other route policies (**L3VPN_NM-SRTE-RP-PE-B-7** and **L3VPN_NM-SRTE-RP-PE-C-7**) in the same manner prior to creating the L3VPN service.

After creating the L3VPN route policies, create the VPN profile for each route policy and then create and provision the L3VPN service. The VPN profile will be referenced from the L3VPN service. This will bind the route policy to the L3VPN service.

Step 3 Create and provision the L3VPN service

In this step, we will create the L3VPN service with three endpoints: PE-A, PE-B, and PE-C. Each endpoint will be associated with a vpn-instance-profile, which in turn points to a VPN profile that contains the route policy with the same color as specified in the ODN template. In this way, traffic that matches the specified prefixes and color will be treated according to the SLA specifications.

First, we will create the VPN profiles. The newly created VPN profiles will have the same names as the L3VPN routing policy names.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **VPN Profiles**.

Step 2 Click + to create a valid VPN profile to be referenced in the VPN service.

Step 3 Select the Id list and select **L3VPN_NM-SRTE-RP-PE-A-7**.

Now create and provision the L3VPN service.

Step 4 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3vpn** > **L3vpn-Service**.

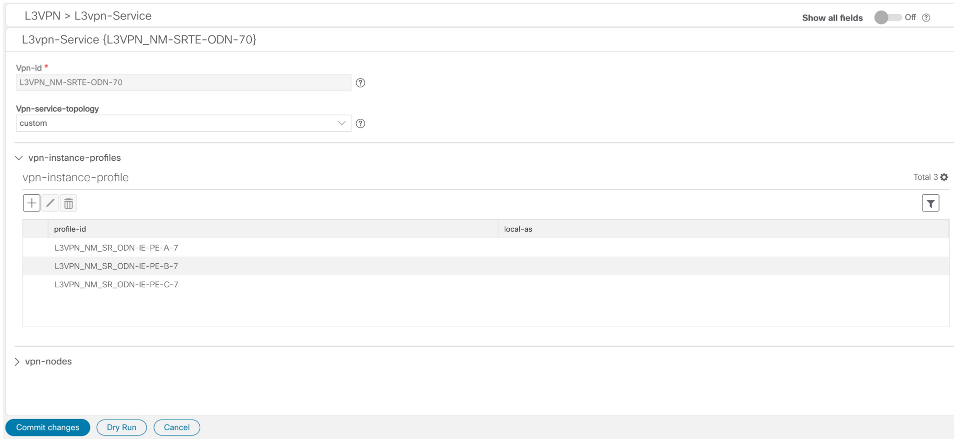
Step 5 Click **+** to create a new service and type a new Vpn-id: **L3VPN_NM-SRTE-ODN-70**.

A VPN identifier uniquely identifies a VPN and has a local meaning (for example, within a service provider network).

Step 6 Click **Continue**.

Step 7 Create vpn-instance-profiles, which is a container that defines the route distinguisher (RD), route targets, and the export/import route policy. We will create vpn-instance-profiles for each endpoint, as follows:

- L3VPN_NM_SR_ODN-IE-PE-A-7 with route distinguisher 0:70:70
- L3VPN_NM_SR_ODN-IE-PE-B-7 with route distinguisher 0:70:71
- L3VPN_NM_SR_ODN-IE-PE-C-7 with route distinguisher 0:70:72



a. Expand vpn-instance-profiles and click + to create a new vpn-instance-profile profile-id: **L3VPN_NM_SR_ODN-I-PE-A-7**. Click **Continue**.

b. Enter the route distinguisher (Rd) that will differentiate the IP prefixes and make them unique: **0:70:70**.

c. For address-family, click + and select **ipv4** from the list. Click **Continue**.

d. Define the required VPN targets, including route targets and route target types (import/export/both).

e. Under vpn-policies, in the Export-policy list, choose the relevant VPN profile (which contains the route policy: **L3VPN_NM-SRTE-RP-PE-A-7**). This forms the association between the VPN and the ODN template that defines the SLA.

f. Click **X** in the top-right corner when you are done.

g. Similarly, create the other vpn-instance-profiles.

Step 8 Define each VPN endpoint individually: PE-A, PE-B, and PE-C.

a) Expand vpn-nodes and click + to select the relevant device from the list: **PE-A**. Click **Continue**.
b) Enter the Local-as number for network identification: **200**.
c) Expand active-vpn-instance-profiles and click + to select the Profile-id you created in the previously: **L3VPN_NM-SRTE-RP-PE-A-7**. Click **Continue**.
d) Define the network access parameters for communication from the PE towards the CE:

- Under vpn-network-accesses, click + to create a new set of VPN access parameters and provide a unique ID. Click **Continue**.
- In the Interface-id field, type **Loopback70**. This is the identifier for the physical or logical interface. The identification of the sub-interface is provided at the connection level and/or the IP connection level.
- Expand ip-connection > ipv4 and enter a Local-address (**70.70.70.1**) and the Prefix-length (**30**).
- For routing-protocols, define BGP routing protocol parameters, including the Peer-as number (**70**), Address-family (**ipv4**) IP address of the BGP neighbor (**70.70.70.2**), and Multihop number (for example, **11**) that indicates the number of hops allowed between the BGP neighbor and the PE device.
- Click **X** in the top-right corner until you are back on the Create L3VPN screen.
- Similarly, create the other VPN nodes: **PE-B** and **PE-C**.

Step 9 Commit your changes or click **Dry Run** to check what will be configured on the devices before you commit.

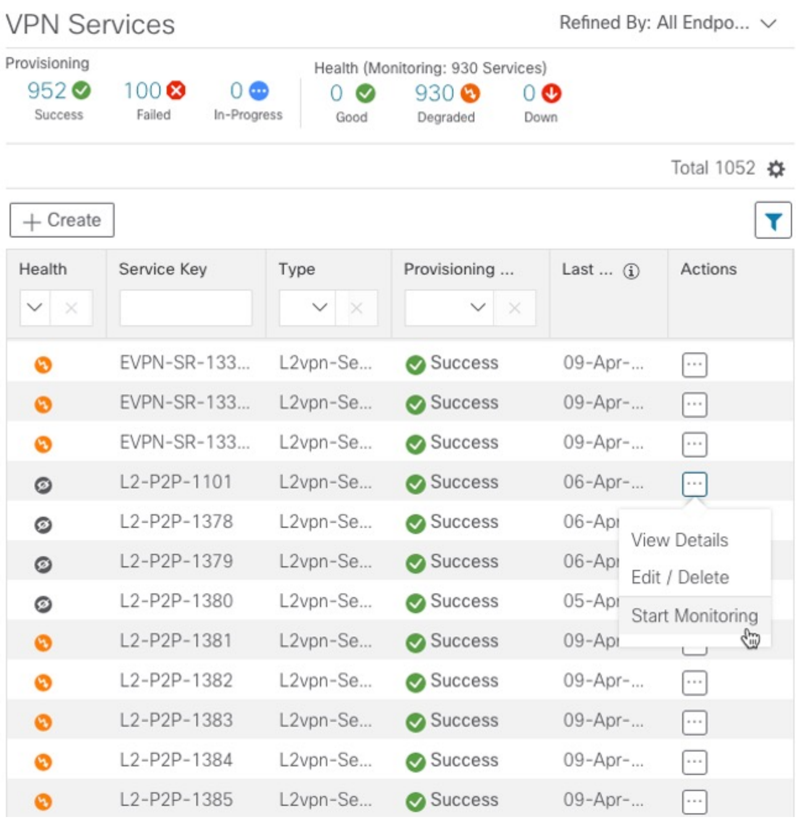
Step 10 Check that the new L3VPN service appears in the table and its provisioning state is **Success**.

Step 4 Enable Service Health monitoring

Step 1 Go to **Services & Traffic Engineering** > **VPN Services**. The map opens on the left side of the screen and the table opens on the right side of the screen.

Step 2 In the Actions column, click for the new service you want to start monitoring health.

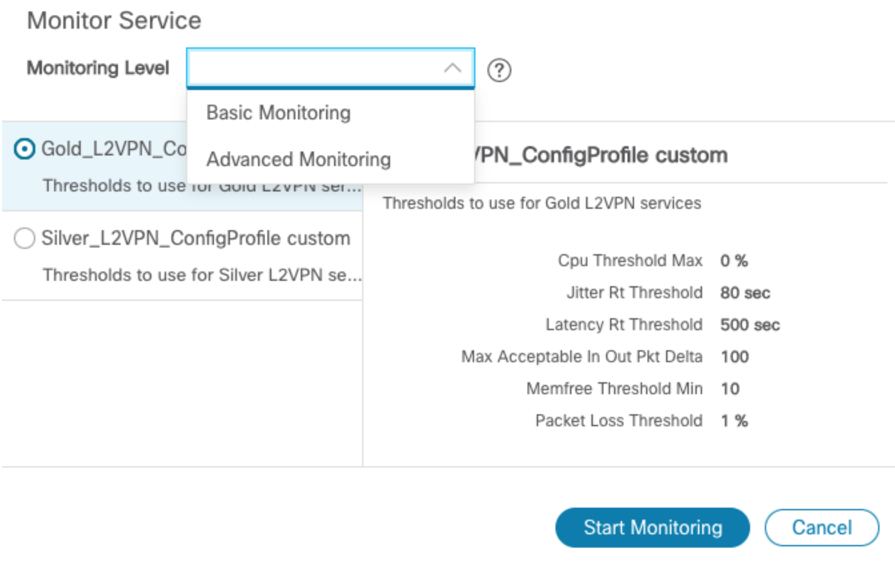
Step 3 Click **Start Monitoring**.



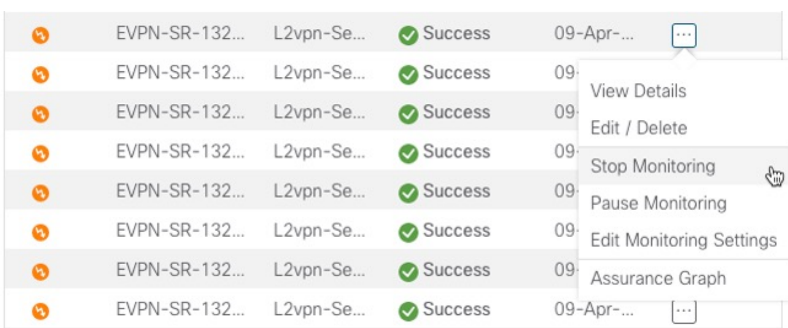
Note The Health column color coding indicates the health of the service:

- Blue = Initiated
- Green = Good
- Orange = Degraded
- Red = Down
- Gray = Not Monitoring

Step 4 In the Monitor Service pop-up, select the Monitoring Level. For help selecting the appropriate monitoring level for your needs, see the section Basic and Advanced Monitoring Rules.



Note Once you have started monitoring the health of this service, if you select the Actions column and click to view additional Service Health options, you will see: **Stop Monitoring**, **Pause Monitoring**, **Edit Monitoring Settings, Assurance Graph**.



Note If you select Edit Monitoring Settings, you may update the Monitoring Level setting from Basic Monitoring to Advanced Monitoring, or from Advanced Monitoring to Basic Monitoring, at any time. You may also update to a different Configuration Profile (from Gold profile to Silver profile or from Silver profile to Gold profile).

Note If you later decide to Stop Monitoring a service that has already been started, you have the option to retain the historical service data for that stopped service. See Stopping Service Health Monitoring in the Appendix for additional steps and details.

Step 5 Click **Start Monitoring**.

Step 6 Repeat this step for each service you wish to start health monitoring.

Step 7 Click **X** in the top-right corner when you are done.

Step 5 Visualize the New VPN Service on the Map to See the Traffic Path

Step 1 In the L3VPN Service table, click on the service name or click ⋯ in the Actions column and choose **View Details** from the menu.

The map opens and the service details are shown to the right of the map.

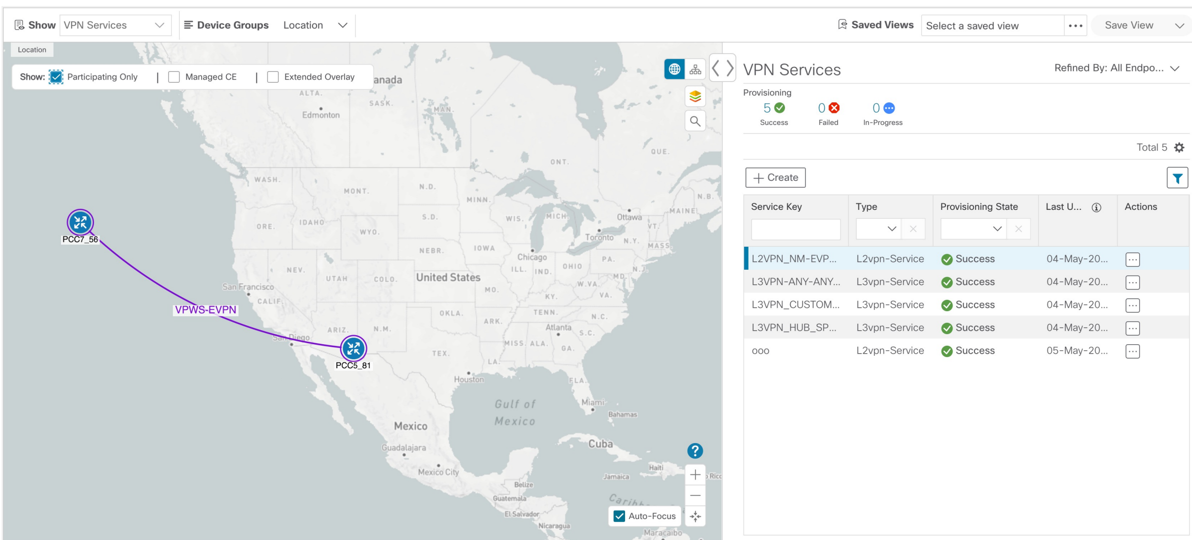
or

Go to **Services & Traffic Engineering** > **VPN Services**.

The map opens and a table of VPN services is displayed to the right of the map.

Click on the VPN in the Services table. If there are many services in the table, you can filter by name, type, or provisioning state to help locate the VPN.

In the map, you will see the VPN as an overlay on the topology. It shows a representation of the three endpoints and a dashed line that indicates that it is a virtual path.

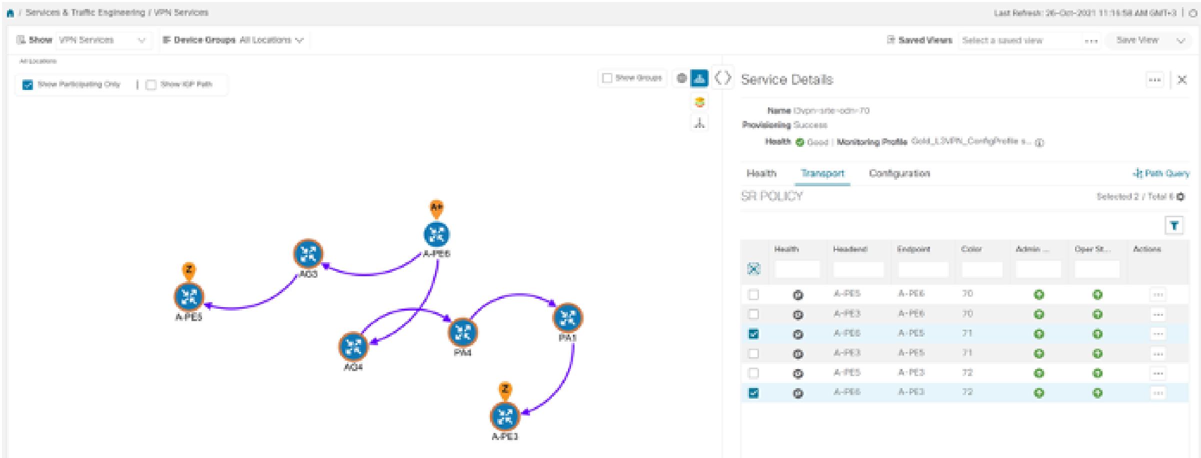


Select the **Show Participating Only** check box if you do not want to see the devices that are not involved in the selected VPN.

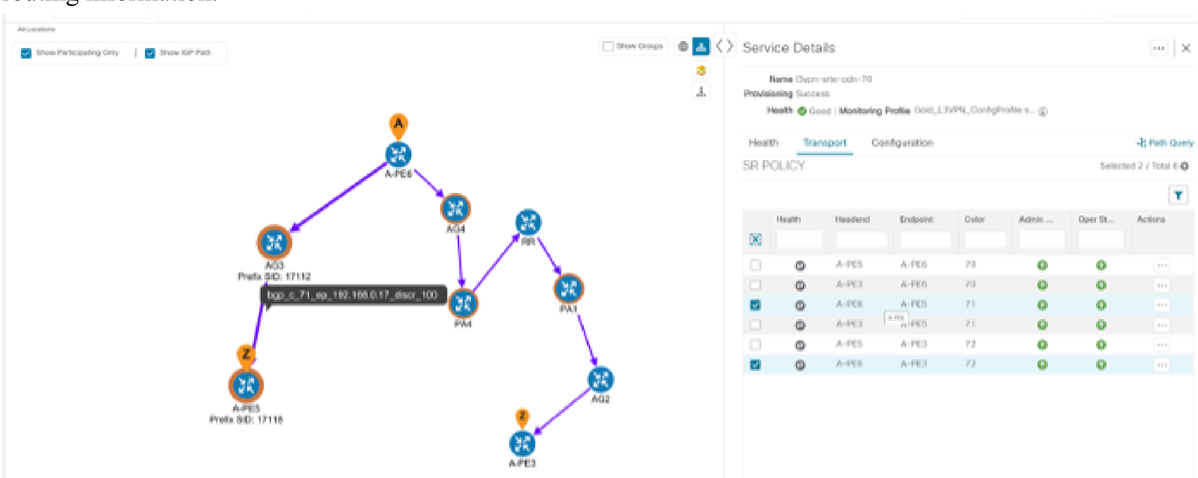
Step 2 In the Actions column, click ⋯ to drill down to a detailed view of the VPN service, including the device configurations and the computed transport paths.

Step 3 To see the computed paths for this VPN, click on the Transport tab in the Service Details pane. All the dynamically created SR-TE policies are listed in the Transport tab. Select one or more SR-TE policies to see the path from endpoint to endpoint on the map.

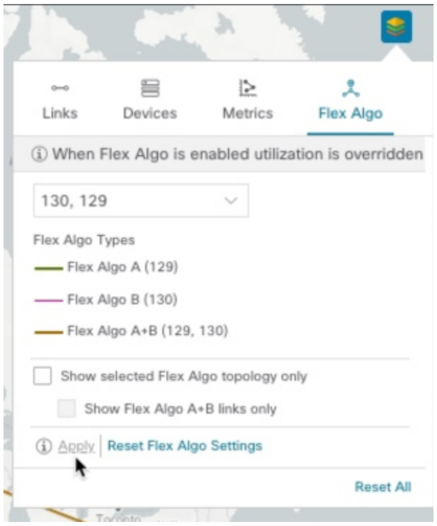
In this example, we are looking at the disjoint paths computed from PE-A to PE-B and from PE-A to PE-C.



Step 4 To see the physical path between the endpoints, select the **Show IGP Path** check box in the top-left corner of the map. Hover with your mouse over a selected policy in the table to highlight the path in the map and show prefix SID and routing information.



Step 5 To filter the topology to a specific Flex-Algo constraint and visualize nodes and links you have configured manually in your network, click the button at the top right of the map and do the following:



a) Click the **Flex Algo** tab.
b) From the drop-down list, choose up to 2 Flex-Algo IDs.
c) View the Flex-Algo Types and confirm that the selection is correct. Also, note the color assignments for each Flex-Algo ID.
d) (Optional) Check the **Show selected Flex Algo topology only** check box to isolate the Flex-Algo IDs on the topology map. When this option is enabled, SR policy selection is disabled.
e) Check the **Show Flex Algo A+B links only** to show only those links and nodes that participate in both Flex-Algos.

   If a selected Flexible Algorithm is defined with criteria but there are no links and node combinations that match it (for example, a defined affinity to include all nodes or links with the color blue), then the topology map will be blank. If a selected Flexible Algorithm is not configured on a node or link, then the default blue link or node color appears.

f) Click **Apply**. You must click **Apply** for any additional changes to your Flex-Algo selections to see the update on the topology map.
g) (Optional) Click **Save View** to save the topology view and Flexible Algorithm selections.

Step 6 Observe automatic network optimization

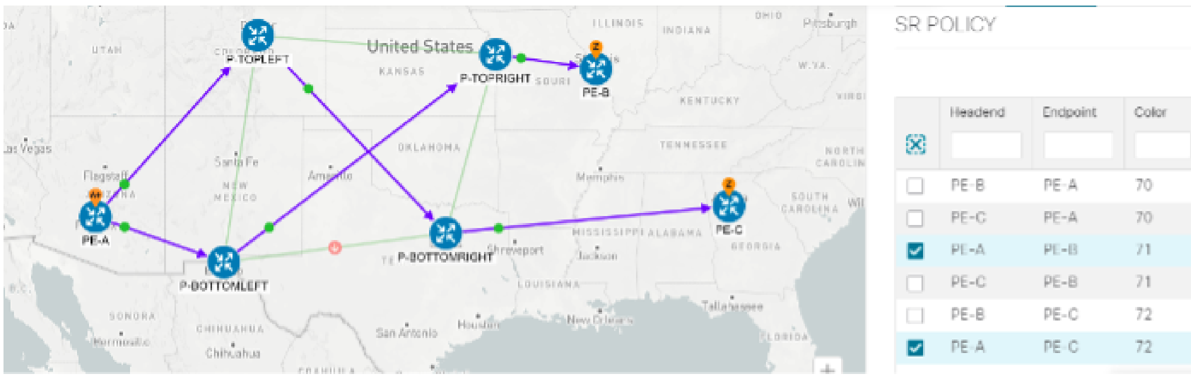
Observe automatic network optimization

The SR-PCE constantly monitors the network and automatically optimizes the traffic path based on the defined SLA. For illustration purposes, let’s look at what happens when one of the links goes down, in this case, the link between P-BOTTOMLEFT and P-BOTTOMRIGHT. This means that the previous path from PE-A to PE-C is no longer viable. Therefore, the SR-PCE computes an alternative path, both from PE-A to PE-C and from PE-A to PE-B, to compensate for the link that is down and to maintain the disjoint paths.

Recomputed paths:

Source and Destination	Old path	New path

PE-A > PE-C	PE-A > P-BOTTOMLEFT > P-BOTTOMRIGHT > PE-C	PE-A > P-TOPLEFT > P-BOTTOMRIGHT > PE-C
PE-A > PE-B	PE-A > P-TOPLEFT > P-TOPRIGHT > PE-B	PE-A > P-BOTTOMLEFT > P-TOPRIGHT > PE-B



Step 7 Inspect a degraded service using Service Health to determine active symptoms

In this step, we will monitor the VPN services using Assurance Graph capabilities and inspect any services or related nodes that are degraded. By inspecting the root causes that lead to reported active symptoms and impacted services, you can determine what issues must be addressed first to maintain a healthy setup and what requires further inspection and troubleshooting.

Step 1 Click **X** in the top-right corner to return to the VPN Services list.

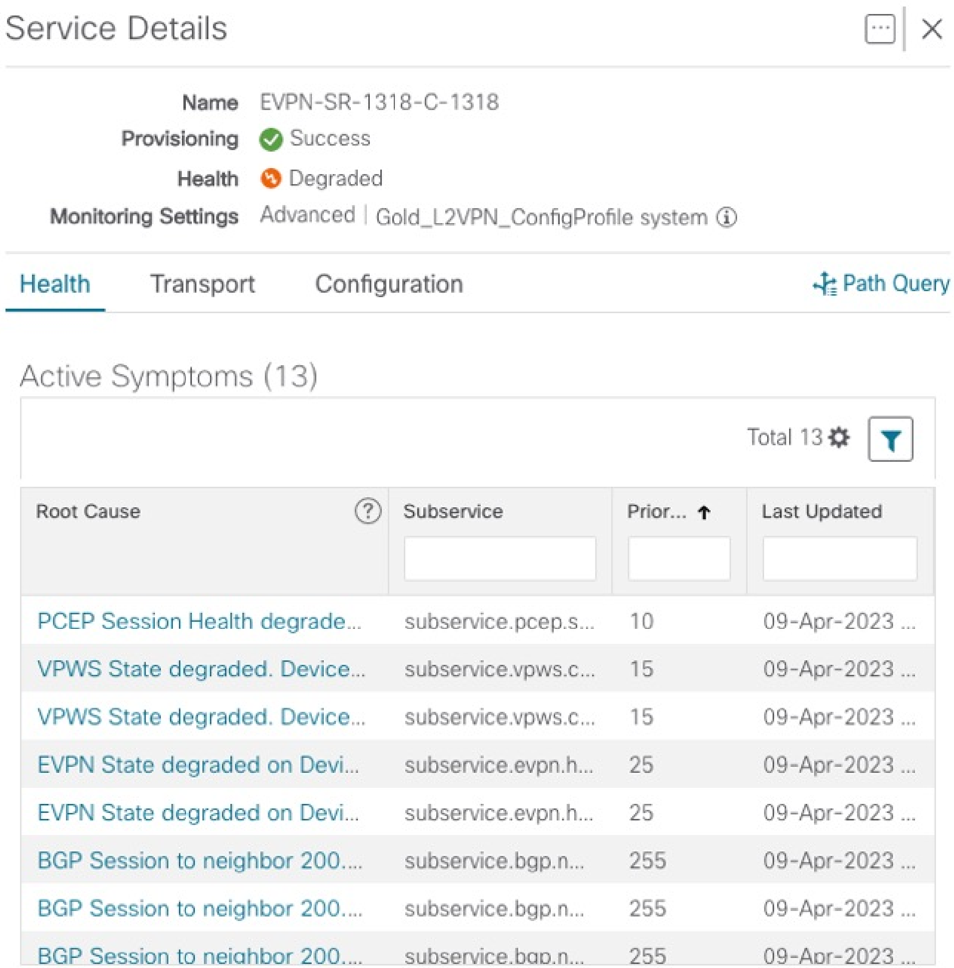
Step 2 Click on the name of a service that shows as being degraded. The map will update to highlight the service you selected.

Degraded services show an orange icon in the Health column. You can filter by health state (Down, Degraded, Good) by clicking in the space at the top of the column and selecting the appropriate filter. To clear the filter, click the X next to the designated filter appearing in the space at the top of the column and it will remove all filtering and default to showing all VPN Services in the list.

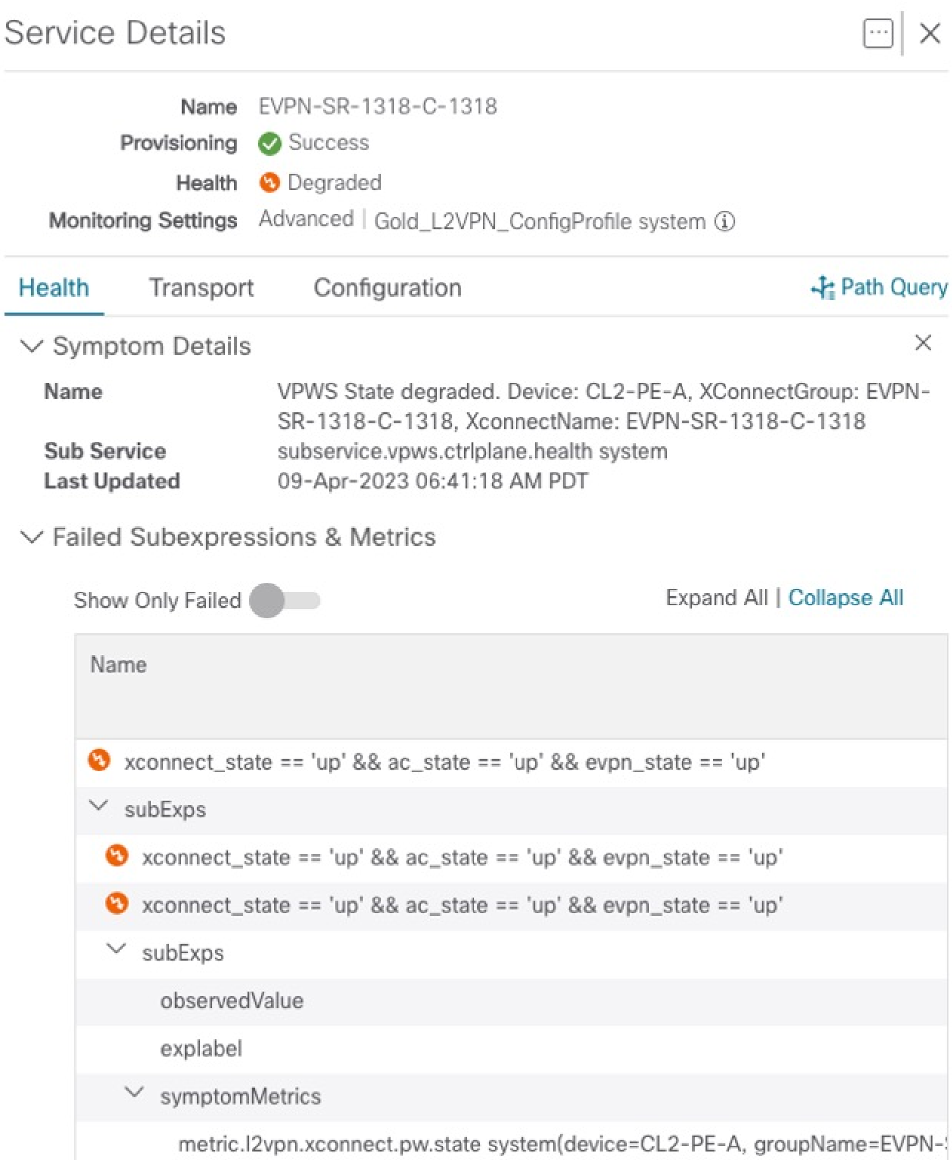
Note If a service is not yet being monitored, the icon in the Health column will show as the color grey. To enable monitoring for such a service, click ⋯ and select **Start Monitoring.**

Step 3 In the Actions column, click ⋯ and click **View Details**. The Service Details screen appears on the right side.

Step 4 With the Health tab selected, review Active Symptoms for the degraded service (including the Root Cause, Subservice, Priority, and Last Updated details) present in the Health tab if the service is currently being monitored.



Step 5 Click on a Root Cause and view both the Symptom Details and the Failed Sub expressions & Metrics information. As needed, you can expand or collapse all of the symptoms listed in the tree. In addition, use the **Show Only Failed** toggle to focus on only failed expression values.



Step 6 Select the Transport and Configuration tabs and review the details provided.

Step 7 To further isolate the degraded service details, click **X** in the top-right corner to return to the VPN Services list.

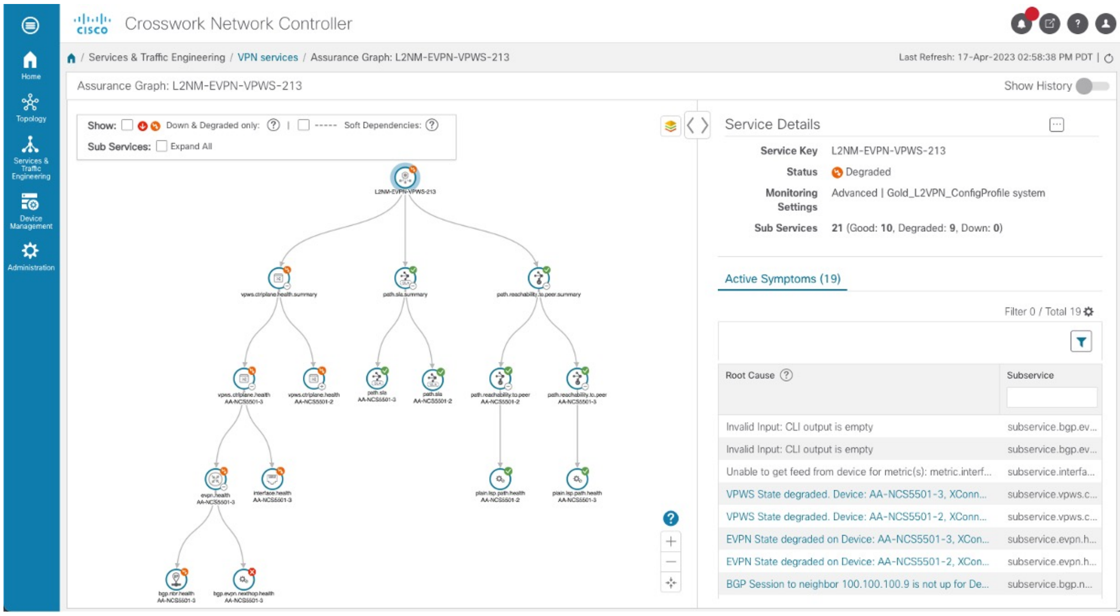
Step 8 Again, click on the name of the degraded service in the list. The Service Details panel appears and the map updates, isolating the corresponding devices participating with that service.

Step 9 Within the map, view further service health details doing the following:

a) At the top-left of the map, select the Show Participating Only check box so the map only shows the participating services.

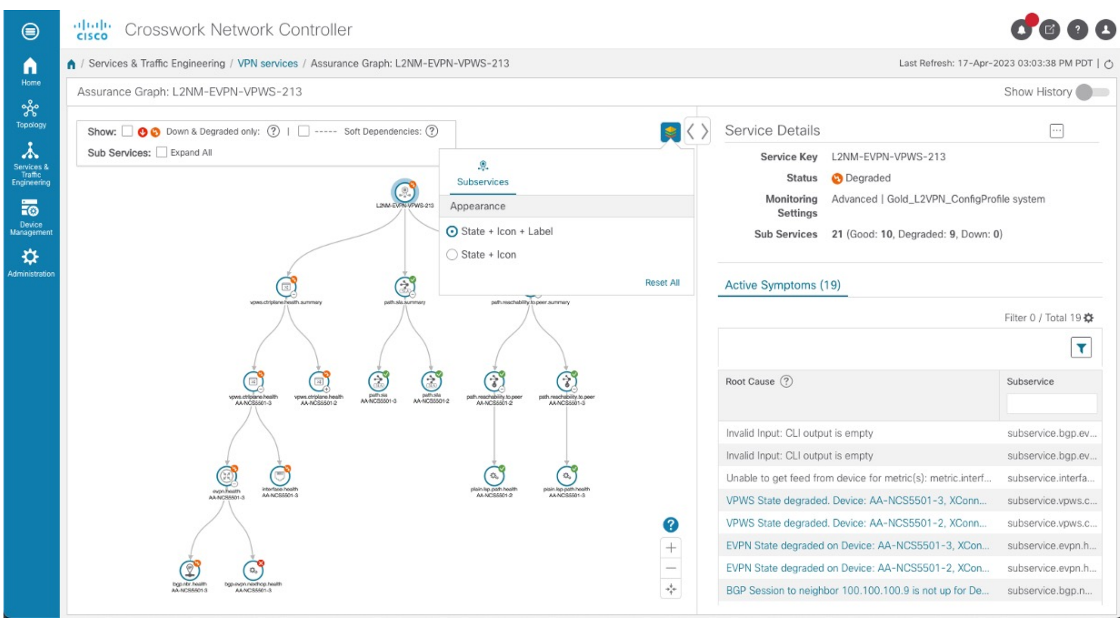
b) In the map, hover your mouse over one of the devices and smaller badges that indicate health status and review the pop-up information relating to its Reachability State, Host Name, Node IP, and Type.

Step 10 In the Actions column, click for the degraded service in the list and click **Assurance Graph**. The topology map of services and subservices appear with the Service Details panel showing Service Key, Status, Sub Services, and Active Symptoms details.



Note This will take time to update after a service has been enabled for monitoring, and may take up to 5-10 minutes.

At the top-right of the map, select the stack icon to select the appearance option for the Subservices: **State + Icon + Label** or **State + Icon**. In addition, in the middle section of the Service Details panel, KPI metrics details are displayed such as jitter, latency, and packet loss (information collected using Y.1731 probes). For example:

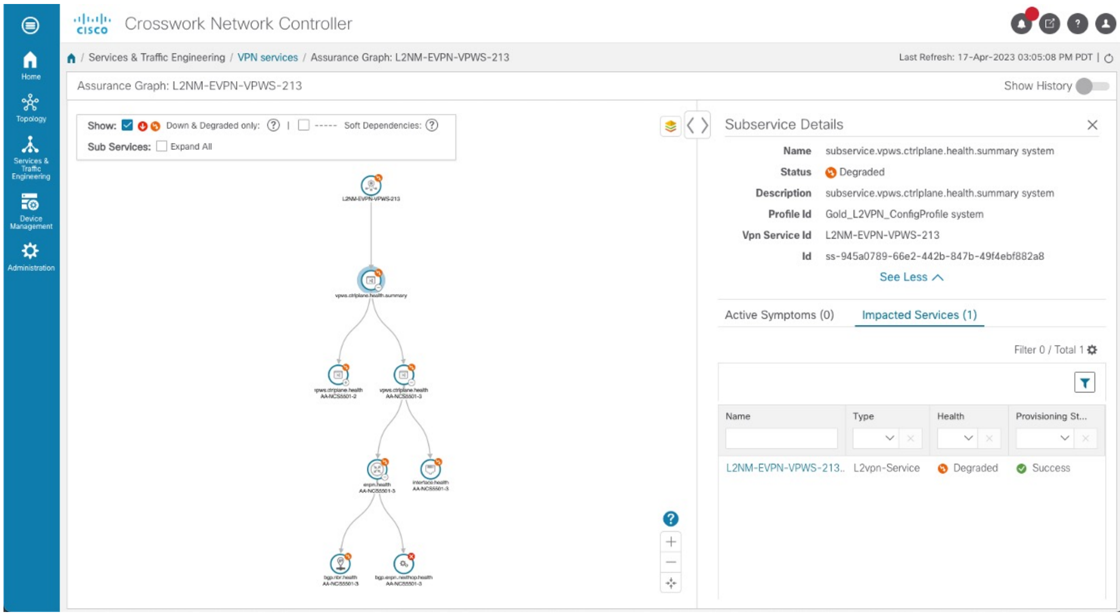


Step 11 In the topology map, select a degraded subservice. The Subservice Details panel appears with subservice metrics, as well as subservice specific Active Symptoms and Impacted Services details.

- **Active Symptoms**: Provides symptom details for nodes actively being monitored.

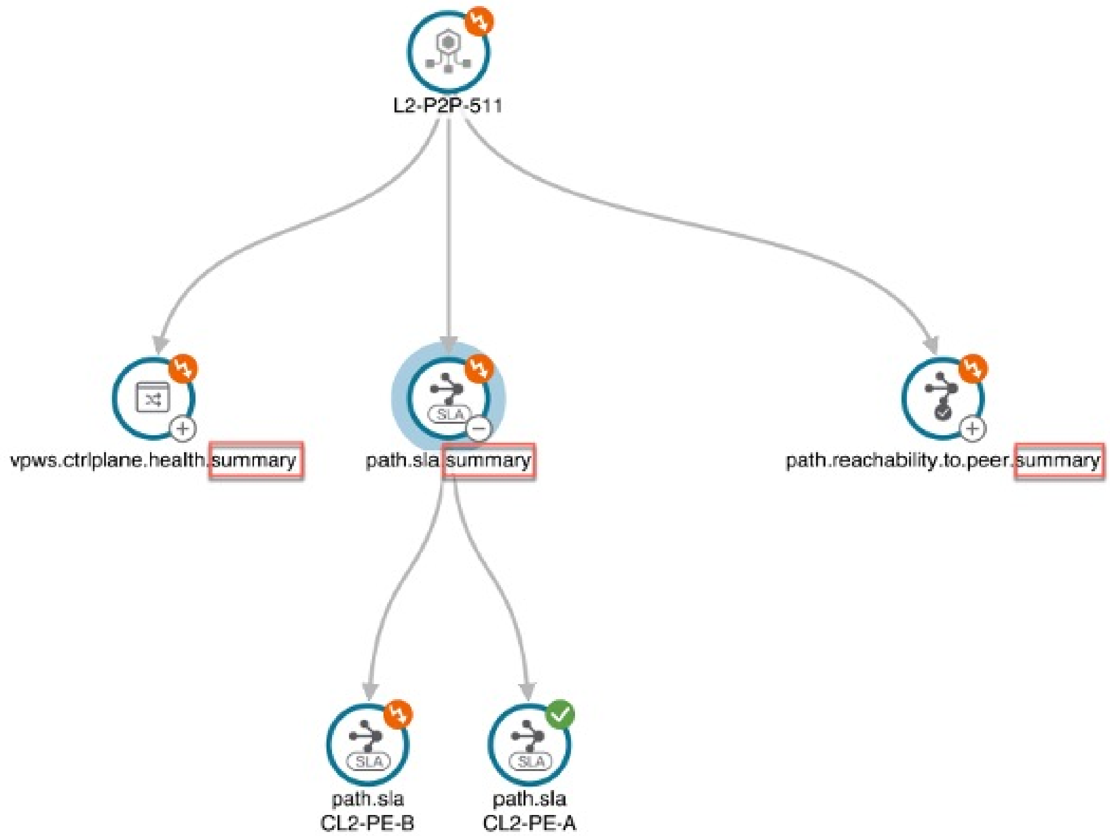
- **Impacted Services**: Provides information for services that are impacted by issues based on historical monitoring of health status.

Note Use your mouse to on subservices in the map for details on the degraded health. At the top left of the map, select Down & Degraded Only or Soft Dependencies to further isolate subservices.



Note In some cases, the Summary node feature is available and summarizes the aggregated health status of child subservices and reports one consolidated health status to a service node. The Summary node feature is available in both L2VPN multipoint Basic and Advanced monitoring models.

- Basic monitoring subservices:
  - Device – Summarizes the health status of all underlying Devices participating in the given L2VPN service.
  - Bridge Domain – Summarizes the L2VPN Service's Bridge Domain health status across all participating devices.
- Advanced monitoring subservices (in addition to what is also available with Basic monitoring)
  - EVPN – Summarizes the health status of all underlying subservices – BGP Neighbor Health & MacLearning Health across all participating PE endpoints and provides a consolidated overall EVPN health summary status.
  - Transport – Summarizes the health status of all underlying subservices – SR-ODN (dynamic), SR Policy (statically configured) and RSVP TE Tunnel, across all participating PE endpoints and provides a consolidated overall Transport health summary status.
  - SR-PCEP – Summarizes the health status of all the underlying subservices that are monitoring the PCEP sessions. Each underlying subservice monitors the PCEP session health on a particular device participating in the given VPN service.



Step 12 Inspect the Active Symptoms and Impacted Services information and the root causes associated with the degraded service so to determine what issues may need to be addressed to maintain a healthy setup.

To further troubleshoot a service health issue (such as a device that is degraded due to not properly fetching data), continue with the following steps to examine if the issue is associated with a collection job.

Step 13 Select **Administration** > **Collection Jobs**.

The Collection Jobs screen appears.

Step 14 Select the Parameterized Jobs tab.

Step 15 Review the Parameterized Jobs list to pinpoint devices that may have service health degradation issues. By reviewing Parameterized Jobs, you can identify and focus on GMNI, SNMP, and CLI-based jobs by their Context ID (protocol) for further troubleshooting purposes.

Step 16 In the Job Details panel, select the collection job you want to export and click the **export** button to download the status of collection jobs for further examination. The information provided is collected at the time the export is initiated in a .csv file.

The Export Collection Status pop up appears.

Note When exporting the collection status, you must fill in the information each time an export is executed. In addition, make sure to review the Steps to Decrypt Exported File content available on the Export Collection Status pop up to ensure you can access and view the exported information.

Step 17 Click **Export**.

Step 18 To check the status of the exported collection job data, click **View Export Status** at the top right of the Job Details panel. The Export Status Jobs panel appears providing the status of the export request.

Step 19 Review the exported .csv file for collection job details and the possible cause of the degraded device.

Summary and Conclusion

As we observed in this example, operators can use Cisco Crosswork Network Controller to orchestrate L3VPNs with SLAs and to maintain these SLAs using SR-TE policies that continuously track network conditions and automatically react to optimize the network. This automation increases efficiency and reduces human error that is generally unavoidable with manual tasks. Enabling Service Health to monitor provisioned services allows for more detailed symptoms, metrics, and analyzation of each service.

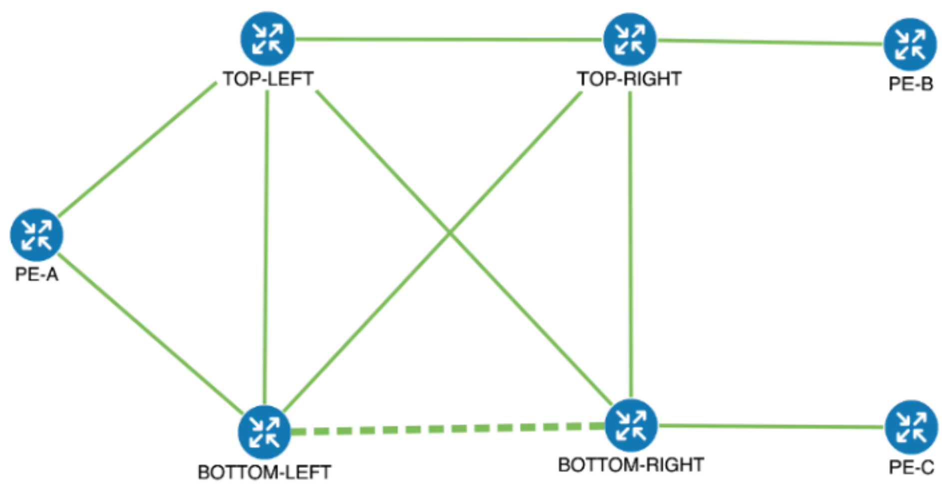
Scenario 2 – Implement and Maintain SLA for an L3VPN Service for SRv6 (using ODN)

This scenario walks you through the procedure for provisioning an L3VPN service that requires a specific SLA objective. In this example, the lowest latency path is the SLA objective. The customer requires a low latency path for high priority traffic. The customer wants to use disjoint paths, i.e., two unique paths that steer traffic from the same source and to the same destination, avoiding common links so that there is no single point of failure. The customer also wants to enable SRv6, which utilizes the IPv6 protocol to handle packets with more efficiency, increase security and performance, allowing for a significantly larger number of possible addresses.

This is achieved using Segment Routing (SR) On-Demand Next Hop (ODN). ODN allows a service head-end router to automatically instantiate an SR-TE policy to a BGP next-hop when required (on-demand). The headend is configured with an ODN template with a specific color that defines the SLA upon which the traffic path will be optimized when a prefix with the specified color is received. Prefixes are defined in a route policy that is associated with the L3VPN.

Cisco Crosswork Network Controller continues to monitor the network and will automatically optimize the network based on the defined SLA, in a closed loop.

The following topology provides the base for this scenario:



In this scenario, we will:

- Create a segment routing ODN template with a specific color on the endpoints to ensure that traffic is transported within an LSP (underlay) and that a best-path tunnel is created dynamically when a prefix with the specified color is received. Enable SRv6 (IPv6) for service and link details. The ODN template defines the SLA on which you want to optimize the path. In this case, we will optimize on latency.
- Specify that the computed paths be disjoint: they will not share the same link.
- Create a route policy on each endpoint to be used to bind the L3VPN to the ODN template. This route policy adds a color attribute to the customer prefixes and advertises via BGP to other endpoints. This color attribute is used to indicate the SLA required for these prefixes.
- Create an L3VPN service with 3 endpoints: PE-A, PE-B, and PE-C. This is the overlay configuration.
- Visualize how this overlay/underlay configuration optimizes the traffic path and automatically maintains the SLA.

Assumptions and Prerequisites

- To use ODN with SRv6, BGP peering for the prefixes must be configured between the endpoints/PEs. Usually for L3VPN, this is the VPNv4 and VPNv6 address family peering, and this BGP peering is required to be over IPv6.

Procedure to Implement and Maintain SLA for an L3VPN Service for SRv6 Using ODN is detailed in this section.

Step 1 Create an ODN template to map color to an SLA objective and constraints

We will create the following ODN templates:

- Headend PE-A, color 72, latency, disjoint path (link), group ID 16 - L3VPN_NM-SRTE-ODN_72-a
- Headend PE-A, color 71, latency, disjoint path (link), group ID 16 - L3VPN_NM-SRTE-ODN_71-a
- Headend PE-B and PE-C, color 70, latency - L3VPN_NM-SRTE-ODN_70

    - With multiple headends in the SRv6 enabled ODN template, the same locator name should be configured on the headend routers. Otherwise, different ODN templates should be created for each headend.

- Headend PE-B, color 72, latency - L3VPN_NM-SRTE-ODN_72-b
- Headend PE-C, color 71, latency - L3VPN_NM-SRTE-ODN_71-c

For example purposes, we will show how to create the first ODN template - L3VPN_NM-SRTE-ODN_72-a. The other ODN templates can be created using the same procedure.

Before you begin

In this step, we will create an ODN template on each endpoint. The ODN template specifies the color and the intent; in this case, latency and disjointness. This ODN template will be used to dynamically create tunnels (on-demand) when prefixes with matching colors are received via BGP. Traffic to these prefixes will be automatically steered into the newly created tunnels, thereby meeting the SLA objective and constraints intended for these prefixes and signaled using colors in the BGP routes.

Disjointness constraints work by associating a disjoint group ID with the ODN template, and all tunnels with the same disjoint group ID will be disjoint, i.e., they will use different links, nodes and shared risk link groups depending on how the disjoint groups are configured.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **SR-TE** > **ODN-Template**.

Step 2 Click **+** to create a new template and give it a unique name.

In this case, the name is **L3VPN_NM-SRTE-ODN_72-a**.

Step 3 Choose the headend device, **PE-A**, and specify the color **72**.

Step 4 Under srv6, select the **Enable srv6** toggle.

Step 5 Under locator, enter the required SRv6 **locator-name**.

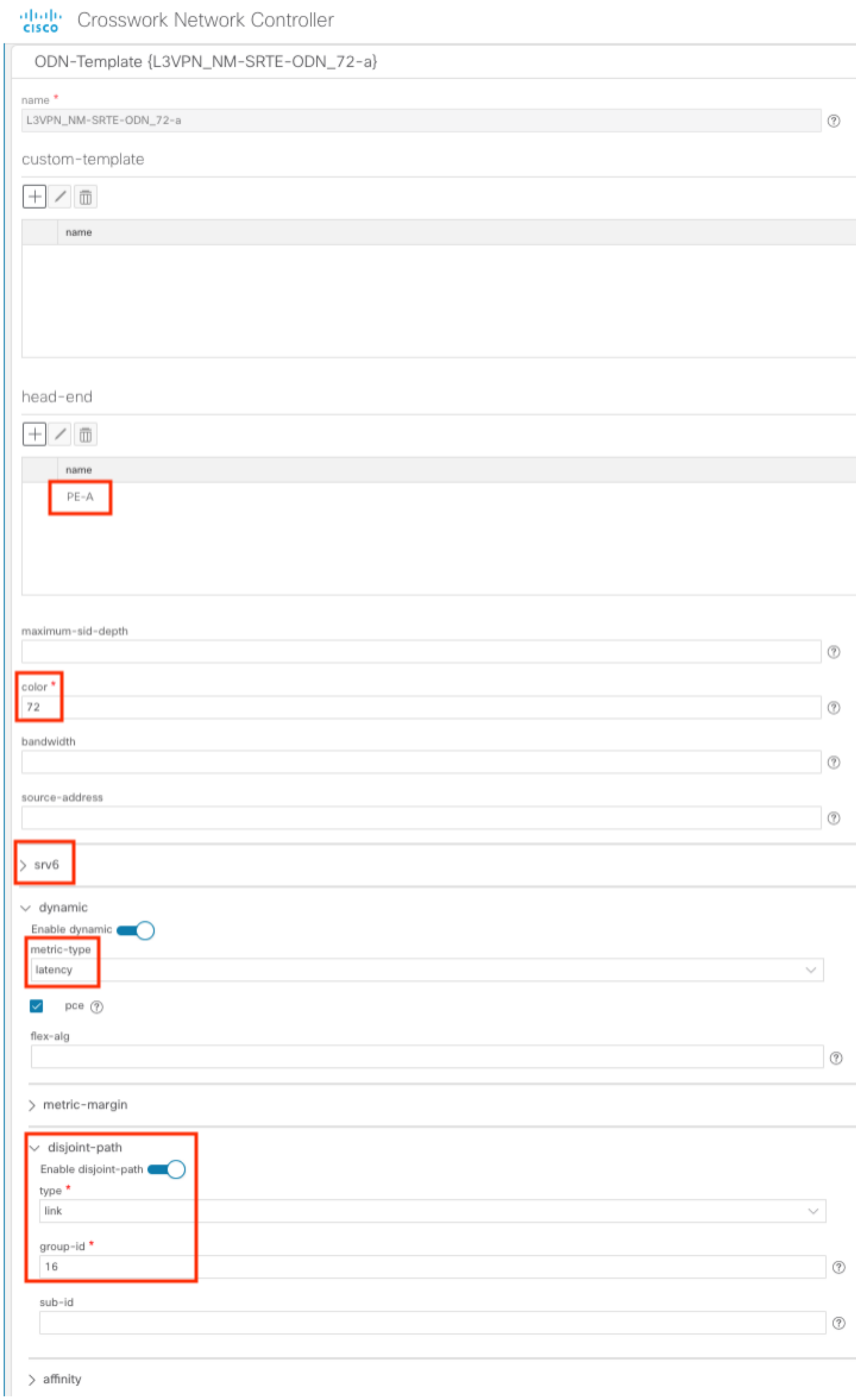
The locator name should match what is configured on the router.

Step 6 Under dynamic, select **“latency”** as the metric type. This is the SLA objective on which we are optimizing.

Step 7 Select the **pce** check box to specify that the path should be computed by the SR-PCE, not by the Path Computation Client (PCC).

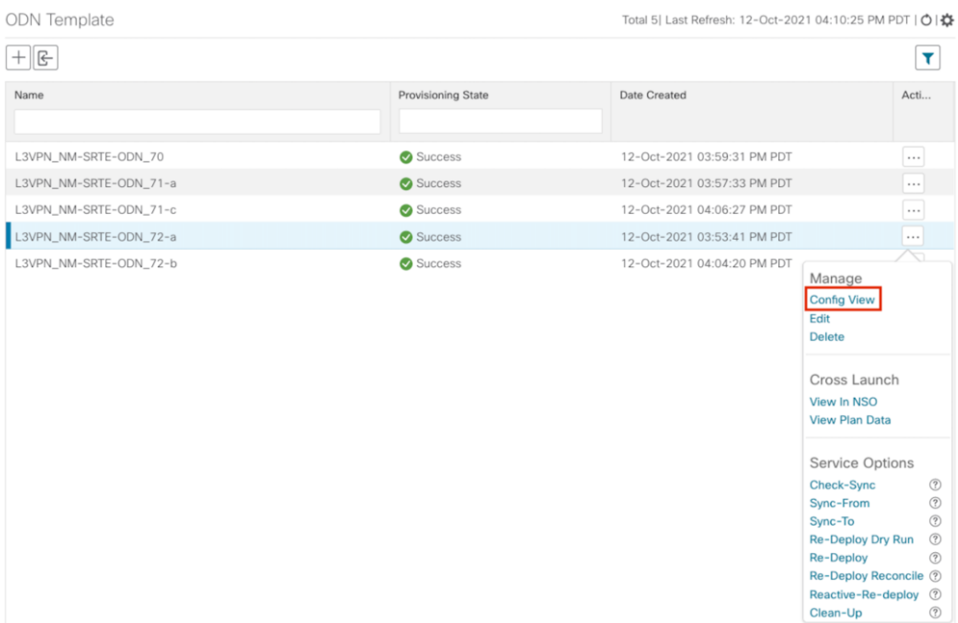
Step 8 Define the required constraints. In this case, we want the computed paths to be disjoint in that they must not share a link.

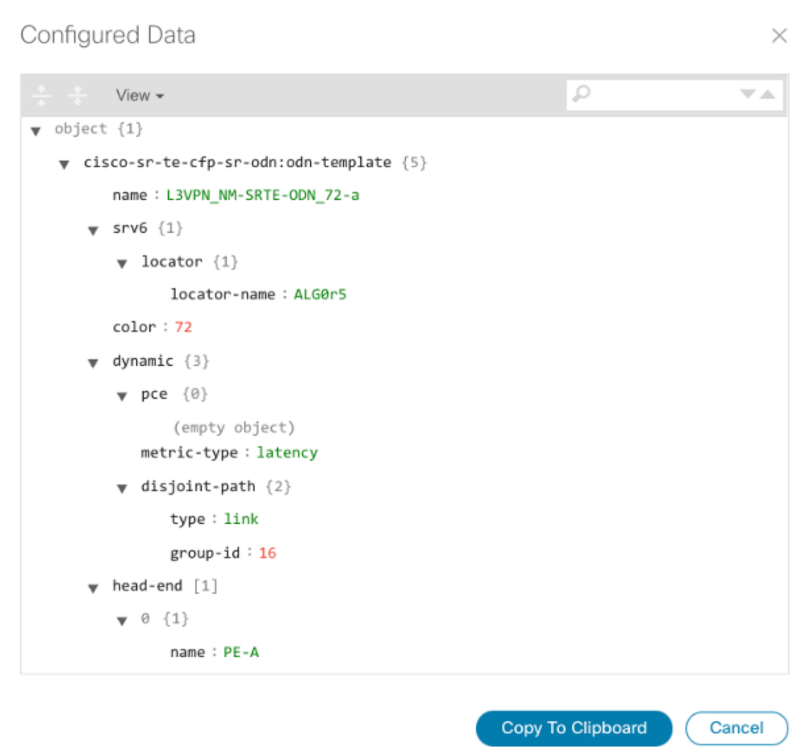
Under disjoint-path, choose **link** as the type, and specify a numeric group ID, in this case, 16.


Crosswork Network Controller
ODN-Template {L3VPN_NM-SRTE-ODN_72-a}
name *
L3VPN_NM-SRTE-ODN_72-a
custom-template
name
head-end
name
PE-A
maximum-sid-depth
color *
72
bandwidth
source-address
srv6
dynamic
Enable dynamic
metric-type
latency
pce
flex-alg
metric-margin
disjoint-path
Enable disjoint-path
type *
link
group-id *
16
sub-id
affinity

Commit your changes or click **Dry Run** to check what will be configured on the devices before you commit.

Step 9 Check that the new ODN template appears in the table and its provisioning state is **Success**. Click ⋯ in the Actions column and choose **Config View** to see the Yang model-based service intent data that details the ODN template you created.





Step 10 Create the other ODN templates listed above in the same manner.

Step 2 Create an L3VPN Route Policy

In this step, we will create a route policy for each endpoint, and we will specify the same color as defined in the ODN template for that endpoint. The route policy defines the prefixes to which the SLA applies. When traffic from the specified network with a matching color is received, paths are computed based on the SLA defined in the ODN template. We will create the following route policies:

- Color 70, IPv6 prefix 70:70:70::0/64 - L3VPN_NM-SRTE-RP-PE-A-7
- Color 71, IPv6 prefix 70:70:71::0/64 - L3VPN_NM-SRTE-RP-PE-B-7
- Color 72, IPv6 prefix 70:70:72::0/64 - L3VPN_NM-SRTE-RP-PE-C-7

For example purposes, we will show how to create the first route policy - L3VPN_NM-SRTE-RP-PE-A-7. The other route policies can be created using the same procedure.

First, we will create the routing policy tag and routing policy destination prefix. The routing policy prefixes should match with the subnet prefix configured on the PE devices in the service.

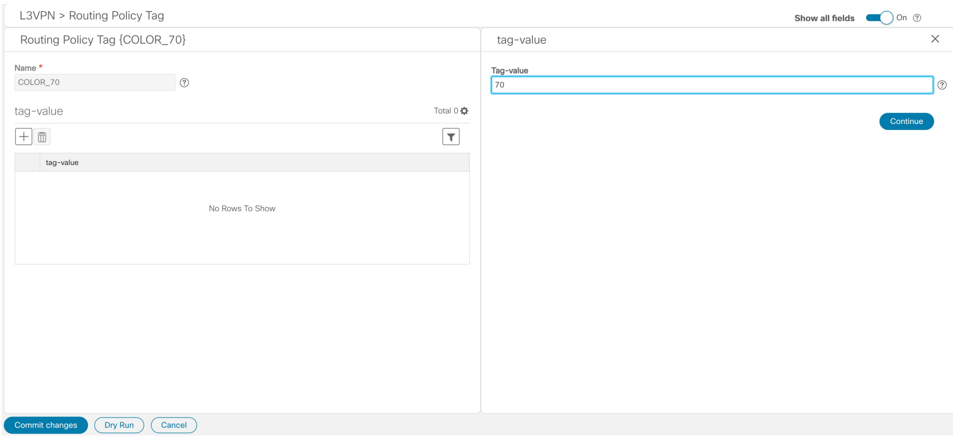
Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **Routing Policy Tag**.

Step 2 Click + to create a new routing policy tag and type the name of the tag set: **COLOR_70**. Click **Continue**.

This is used as a label to reference the set in actions and conditions.

Step 3 Under tag-value, click + and type the Tag-value: **70**.

The tag value may be a number between **1 – 4294967295** and should match to a color value.



Step 4 Click **Continue**. The new routing policy tag name with the new tag value is visible. Click **Commit changes**.

Create the other two routing policy tags (**COLOR_71** and **COLOR_72**) and tag values (**71** and **72**) by following the same steps above.

Now create the routing policy destination prefixes.

Step 5 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **Routing Policy Destination Prefix**.

Step 6 Click + to create a new routing policy destination prefix and type the name: **DEST_PREFIX_SET_70**.

The name of the prefix set will reference the set in match conditions.

Step 7 For Mode, select **ipv6**.

Step 8 Expand prefixes and click + to add the ip-prefix to the prefix-list.

Step 9 For Ip-prefix, type **70:70:70::0/64** and click **Continue**.

Create the other two routing policy destination prefixes (**DEST_PREFIX_SET_71** and **DEST_PREFIX_SET_72**) by following the same steps.

Now we are ready to create the first route policy L3VPN_NM-SRTE-RP-PE-A-7. The other route policies can be created using the same procedure.

Step 10 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **Routing Policy**.

Step 11 Click **+** to create a new route policy and type a unique name for the top-level policy definition: **L3VPN_NM-SRTE-RP-PE-A-7**. Click **Continue**. The statements section appears.

Note The Route Policy statement defines the condition and action taken by the system.

Step 12 Expand statements and click + to add the name of the policy statement (such as **stmt1**) and click **Continue**. The statement{stmt1} panel appears showing **conditions** and **actions** sections.

Step 13 Expand conditions and then expand match-dest-prefix-set before selecting the Prefix-set list and select **DEST_PREFIX_SET_70**. This is what references a defined prefix set.

Note Once selected, the **Enable match-dest-prefix-set** toggle, which will match a referenced prefix-set according to the logic defined in the match-set-options list, switches on.

Step 14 Expand actions and then expand bgp-actions.

Step 15 For bgp-actions, slide the Enable bgp-actions toggle to the on position. By toggling bgp-actions on, it defines the top-level container for BGP-specific actions.

Step 16 Now expand set-ext-community. Slide the Enable-set-ext-community toggle to the on position. By toggling set-ext-community on, it sets the extended community attributes.

Step 17 For Method and reference, select the Ext-community-set-ref list and select **COLOR_70**. The Ext-community-set-ref references a defined extended community set by name.

Note Creating routing-policy tag-set is mandatory and needs to be mapped here.

Step 18 Click **X** in the top-right corner to close the statement{stmnt1} panel and click Commit changes.

Step 19 Create the other route policies (L3VPN_NM-SRTE-RP-PE-B-7 and L3VPN_NM-SRTE-RP-PE-C-7) in the same manner.

After creating the L3VPN route policies, create the VPN profile for each route policy and then create and provision the L3VPN service. The VPN profile will be referenced from the L3VPN service. This will bind the route policy to the L3VPN service.

Step 3 Create and provision the L3VPN service

In this step, we will create the L3VPN service with three endpoints: PE-A, PE-B, and PE-C. Each endpoint will be associated with a vpn-instance-profile, which in turn points to a VPN profile that contains the route policy with the same color as specified in the ODN template. In this way, traffic that matches the specified prefixes and color will be treated according to the SLA specifications.

First, we will create the VPN profiles. The newly created VPN profiles will have the same names as the L3VPN routing policy names.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3VPN** > **VPN Profiles**.

Step 2 Click + to create a valid VPN profile to be referenced in the VPN service.

Step 3 Select the Id list and select **L3VPN_NM-SRTE-RP-PE-A-7**.

Now create and provision the L3VPN service.

Step 4 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L3vpn** > **L3vpn-Service**..

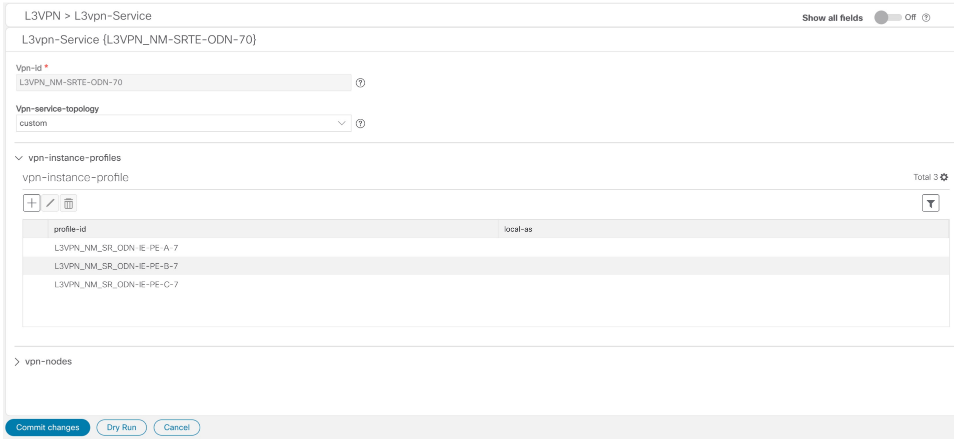
Step 5 Click **+** to create a new service and type a new Vpn-id: **L3VPN_NM-SRTE-ODN-70**.

A VPN identifier uniquely identifies a VPN and has a local meaning (for example, within a service provider network).

Step 6 Click **Continue**.

Step 7 Create vpn-instance-profiles, which is a container that defines the route distinguisher (RD), route targets, and the export/import route policy. We will create vpn-instance-profiles for each endpoint, as follows:

- L3VPN_NM_SR_ODN-IE-PE-A-7 with route distinguisher 0:70:70
- L3VPN_NM_SR_ODN-IE-PE-B-7 with route distinguisher 0:70:71
- L3VPN_NM_SR_ODN-IE-PE-C-7 with route distinguisher 0:70:72



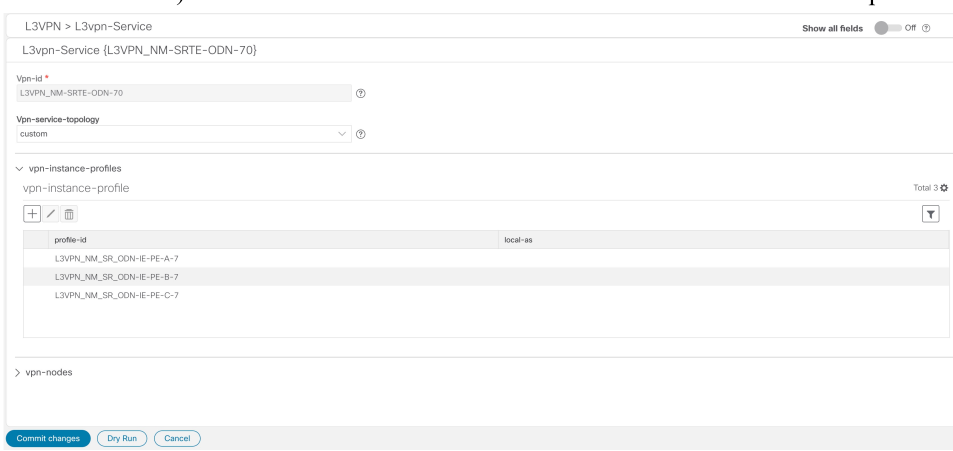
a. Expand vpn-instance-profiles and click + to create a new vpn-instance-profile profile-id: **L3VPN_NM_SR_ODN-I-PE-A-7**. Click **Continue**.

b. Enter the route distinguisher (Rd) that will differentiate the IP prefixes and make them unique: **0:70:70**.

c. For address-family, click + and select **ipv6** from the list. Click **Continue**.

d. Define the required VPN targets, including route targets and route target types (import/export/both).

e. Under vpn-policies, in the Export-policy list, choose the relevant VPN profile (which contains the route policy: **L3VPN_NM-SRTE-RP-PE-A-7**). This forms the association between the VPN and the ODN template that



defines the SLA.

f. Click **X** in the top-right corner when you are done.

g. Expand srv6 and slide the Enable srv6 toggle to the on position and then click + under address-family.

h. Select **ipv6** from address family list and click **Continue**.

i. For Locator-name, type **ALG0r5**. The SRv6 locator name should match locators configured at a node-global level on each router. Click **X** in the top-right corner until you are back on the Create L3VPN screen.

j. Similarly, create the other vpn-instance-profiles.

Step 8 Define each VPN endpoint individually: PE-A, PE-B, and PE-C.

a) Expand vpn-nodes and click + to select the relevant device from the list: **PE-A**. Click **Continue**.

b) Enter the local autonomous system number for network identification: **200**.

c) Expand active-vpn-instance-profiles and click + to select the Profile-id you created in the previously: **L3VPN_NM-SRTE-RP-PE-A-7**. Click **Continue**.

d) Define the network access parameters for communication from the PE towards the CE:

- Under vpn-network-accesses, click + to create a new set of VPN access parameters and provide a unique ID. Click **Continue**.
- In the Interface-id field, type **Loopback70**. This is the identifier for the physical or logical interface. The identification of the sub-interface is provided at the connection level and/or the IP connection level.
- Expand ip-connection > ipv6 and enter a Local-address (**70:70:70::1**) and the Prefix-length (**64**).
- Expand routing-protocols and click + before typing a unique identifier for the routing protocol: **EBGP**. Click **Continue**.
- From the routing protocol Type list, select **bgp-routing**.
- Expand bgp and for Peer-as, type **70**. This information indicates the customer's ASN when the customer requests BGP routing.
- From the Address-family list, select **ipv6**.
- Under neighbor, click + to create a neighbor IP address and type **70:70:70::2**. Click **Continue**.
- Type the Multihop number: **11**. This describes the number of IP hops allowed between a given BGP neighbor and the PE.
- For redistribute-connected, click + and select **ipv6** from the Address-family list. Click **Continue**.
- Click **X** in the top-right corner until you are back on the Create L3VPN screen.
- Similarly, create the other VPN nodes: **PE-B** and **PE-C**.

Step 9 Commit your changes or click **Dry Run** to check what will be configured on the devices before you commit.

Step 10 Check that the new L3VPN service appears in the table and its provisioning state is **Success**.

Step 4 Visualize the New VPN Service on the Map to See the Traffic Path

Step 1 In the L3VPN Service table, click on the service name or click [...] in the Actions column and choose **View Details** from the menu.

The map opens and the service details are shown to the right of the map.

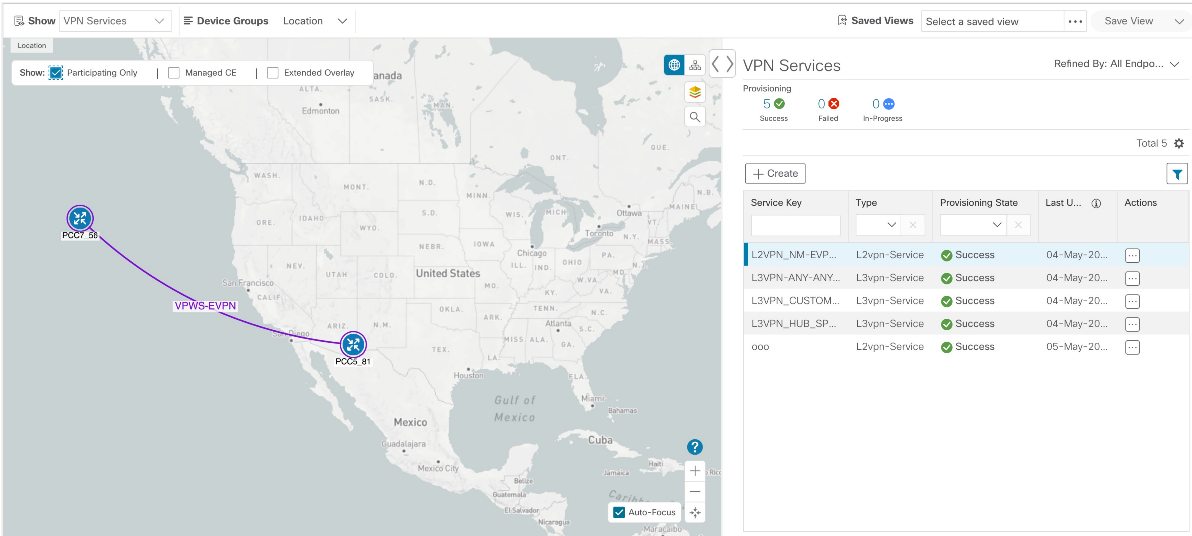
or

a) Go to **Services & Traffic Engineering > VPN Services**.

The map opens and a table of VPN services is displayed to the right of the map.

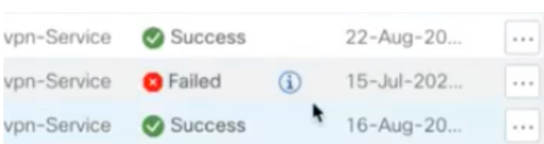
b) Click on the VPN in the Services table. If there are many services in the table, you can filter by name, type, or provisioning state to help locate the VPN.

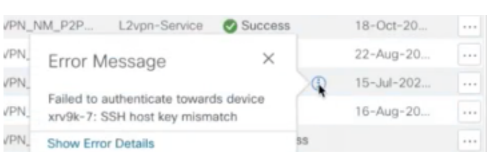
In the map, you will see the VPN as an overlay on the topology. It shows a representation of the three endpoints and a dashed line that indicates that it is a virtual path.

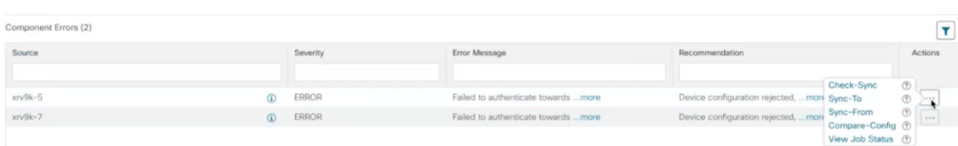


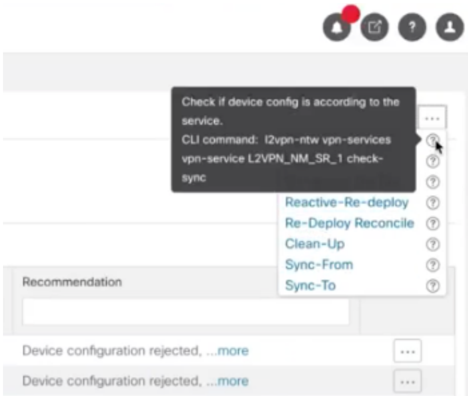
Select the **Show Participating Only** check box if you do not want to see the devices that are not involved in the selected VPN.

Note When a Provision State shows a Failed state, an information icon appears. This is true whether you are on the VPN Services, Service Details, and many of the Provisioning screens that show a table of services and their Provisioning status. If you select the icon, Error Message details appear describing the failure. You can also click the **Show Error Details** link to view the Component Errors screen and take action to fix the error. Each failed source provides further error message details and recommendations. For example, in the Action column for the failed source on the component Errors screen, you may click [...] for different options (such as, **Check-Sync**, **Sync-To**, **Sync-From**, **Compare-Config**, **View Job Status**) that will assist in fixing the error. Service level actions are also available for additional options (such as, **Re-Deploy**, **Reactive-Re-deploy**, **Re-Deploy Reconcile**, **Clean-up**, etc.) that will assist in fixing the service level error. Use the information icons that appear next to these options, as well, for further fix details.

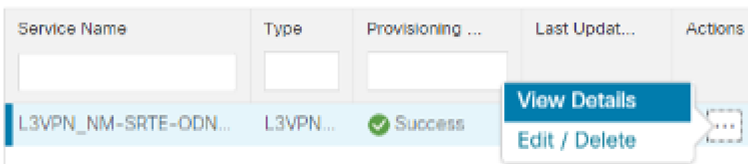






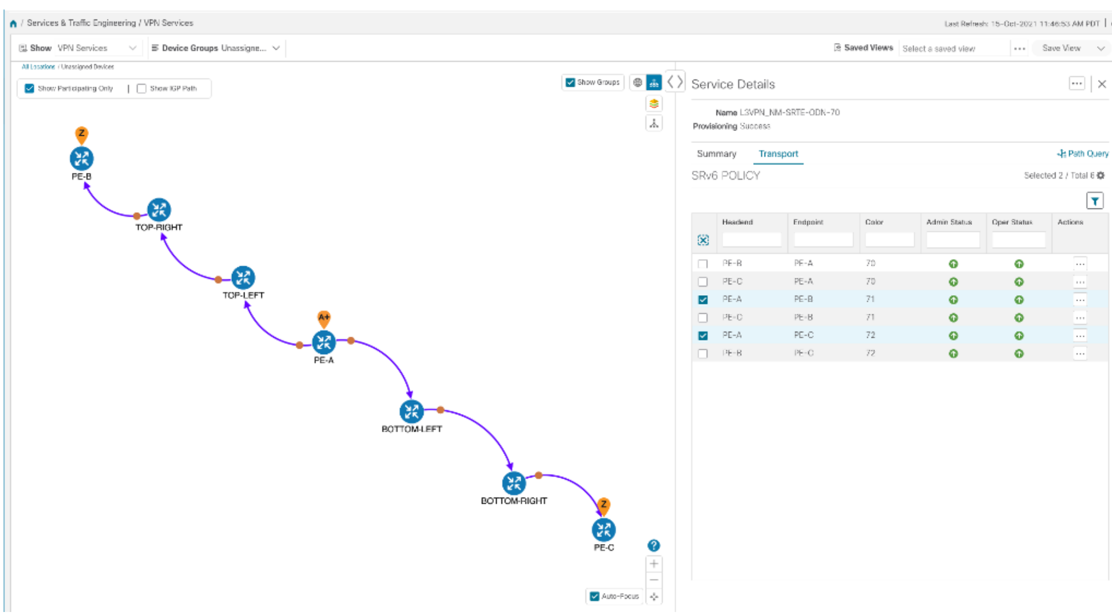


Step 2 In the Actions column, click to drill down to a detailed view of the VPN service, including the device configurations and the computed transport paths.

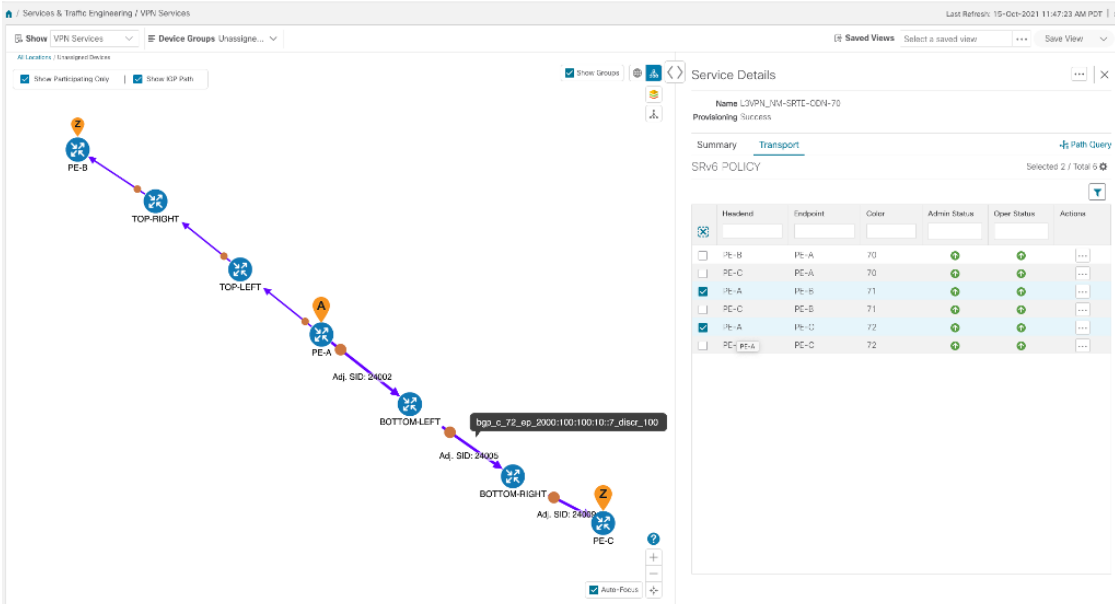


Step 3 To see the computed paths for this VPN, click on the Transport tab in the Service Details pane. All the dynamically created SR-TE policies are listed in the Transport tab. Select one or more SR-TE policies to see the path from endpoint to endpoint on the map.

In this example, we are looking at the disjoint paths computed from PE-A to PE-B and from PE-A to PE-C.



Step 4 To see the physical path between the endpoints, select the **Show IGP Path** check box in the top-left corner of the map. Hover with your mouse over a selected policy in the table to highlight the path in the map and show prefix SID and routing information.

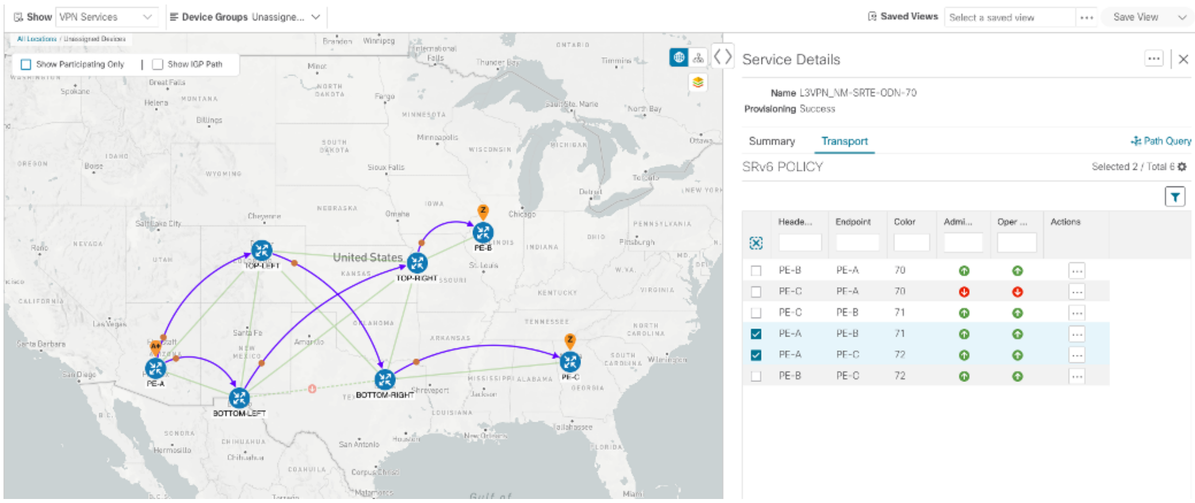


Step 5 Observe automatic network optimization

The SR-PCE constantly monitors the network and automatically optimizes the traffic path based on the defined SLA. For illustration purposes, let’s take a look at what happens when one of the links goes down, in this case, the link between P-BOTTOMLEFT and P-BOTTOMRIGHT. This means that the previous path from PE-A to PE-C is no longer viable. Therefore, the SR-PCE computes an alternative path, both from PE-A to PE-C and from PE-A to PE-B, in order to compensate for the link that is down and to maintain the disjoint paths.

Recomputed paths:

Source and Destination	Old path	New path
PE-A > PE-C	PE-A > BOTTOM-LEFT > BOTTOM-RIGHT > PE-C	PE-A > TOP-LEFT > BOTTOM-RIGHT > PE-C
PE-A > PE-B	PE-A > TOP-LEFT > TOP-RIGHT > PE-B	PE-A > BOTTOM-LEFT > TOP-RIGHT > PE-B



Summary and Conclusion

As we observed in this example, operators can use Cisco Crosswork Network Controller to orchestrate L3VPNs for SRv6 with SLAs and to maintain these SLAs using SR-TE policies that continuously track network conditions and automatically react to optimize the network. This automation increases efficiency and reduces human error that is generally unavoidable with manual tasks.

Scenario 3 – Mandate a Static Path for an EVPN-VPWS Service using an Explicit MPLS SR-TE Policy

To ensure that mission-critical traffic within a VPN traverses the higher capacity interfaces, rather than the lower capacity interfaces, we will create a point-to-point EVPN-VPWS service and associate a preferred path (explicit) MPLS SR-TE policy on both endpoints for service instantiation. In this way, we will mandate a static path for the mission-critical traffic.

In this scenario, we will see how quick and easy it is to create SR-TE policies and VPN services by uploading a file containing all the required configurations. We will download sample files (templates) from the provisioning UI, fill in the required data, and then import the file via the UI. Lastly, we will use the Service Health functionality to review the health of the services and view the Assurance Graph and Last 24Hr Metrics to better analyze our service's health details.

Note In this scenario, reference to SR-TE specifically means SR-TE over MPLS.

In this scenario, we will:

- Create a SID list - a list of prefix or adjacency Segment IDs, each representing a device or link along the path.

- Provision an explicit SR-TE policy, which will reference the SID list, thus creating a predefined path into which the EVPN prefix will be routed.
- Provision a point-to-point EVPN-VPWS service from PE-A to PE-C and attach the explicit SR-TE policy.
- Visualize the path of the service and review the health of the services.

Assumptions and Prerequisites

- For transport mapping to L2VPN service, devices must be configured with the **l2vpn all** command.
- For Service Health enablement, Service Health must be installed. See the Crosswork Network Controller Installation Guide chapter, Install Crosswork Applications.
- (Optional) Service Health provides **Internal Storage** of monitoring data up to a maximum limit of 50 GB. This data is stored on your system. If you exceed the limit of the internal storage, historical data will be lost. If you choose to extend Service Health storage capacity, you can configure **External Storage** in the cloud using an Amazon Web Services (AWS) cloud account. By leveraging External Storage, all existing internal storage data will be automatically moved to the external cloud storage (see **Configuring Service Health External Storage Settings** appendix for more details) and your internal storage will act locally as cache storage. Configuring External Storage for Service Health ensures you will not lose historical data for services that continue to monitor a service's health, and will retain service health data for any service you choose to stop monitoring when you select the option to retain historical monitoring service for the data. For more information on Internal and External Storage, and how to retain historical monitoring service data when stopped, see the Appendix sections **Configuring Service Health External Storage Settings** and **Stopping Service Health monitoring**.
- Before using Service Health's Assurance Graph, ensure that topology map nodes have been fully configured and created with a profile associated to the service. If not, Subservice Details metrics will show that no value has yet to be reported.
- For Service Health, you must configure 2 buckets on the Y1731 profile associated with the device. If you have fewer than 2 buckets configured, Service Health cannot report the Y1731 probes/KPIs on the service details page.

Step 1 Prepare for Creating a SID List

Before you begin

The SID list consists of a series of prefix or adjacency SIDs, each representing a node or link along on the path. Each segment is an end-to-end path from the source to the destination, and it instructs the routers in the network to follow the specified path instead of the shortest path calculated by the IGP.

To build the SID list, you will need the MPLS labels of the desired traversing path. You can get these labels from the devices themselves or you can invoke the northbound Cisco Crosswork Optimization Engine API to retrieve this information.

Refer to Cisco Crosswork Network Automation API Documentation on Cisco Devnet for more information about the API.

Step 1 Prepare the input required to produce the SID list for the path from endpoint to endpoint. You will need the router ID of each endpoint, as follows:

```
{
  "input": {
    "head-end": "100.100.100.7",
    "end-point": "100.100.100.5",
    "sr-policy-path": {
      "path-optimization-objective": "igp-metric"
    }
  }
}'
```

Step 2 Invoke the API on the Cisco Crosswork Network Controller server by using the input prepared in the previous step. For example:

```
curl --location --request POST
'https://10.194.63.198:30603/crosswork/nbi/optima/v1/restconf/operations/cisco-crosswork-
optimization-engine-sr-policy-operations:sr-policy-dryrun' \
--header 'Content-Type: application/yang-data+json' \
--header 'Accept: application/yang-data+json' \
--header 'Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pbiIsImlzRnJvbU5ld0xvZ2luIjoidHJ1ZSIsInBvbGljeV9pZCI6ImFkb
WluIiwiYXV0aGVudGljYXRpb25EYXRlIjoiMjAyMS0wMy0yMlQxNjozODozNy43NDY2MTZaW0dNVF0iLCJzdWNjZXNzZnV
sQXV0aGVudGljYXRpb25IYW5kbGVycyI6IlF1ZXJ5RGF0YWJhc2VBdXRoZW50aWNhdGlvbkhhbmRsZXIiLCJpc3MiOiJod
HRwOlwvXC9sb2NhbGhvc3Q6NTQ4OVwvU1NPIiwibGFzdF9uYW1lIjoic21pdGgiLCJjcmVkZW50aWFsVHlwZSI6IlVzZXJ
uYW1lUGFzc3dvcmRDcmVkZW50aWFsIiwiYXVkIjoiaHR0cHM6XC9cLzEwLjE5NC42My4xOTg6MzA2MDNcL2FwcC1kYXNoY
m9hcmQiLCJhdXRoZW50aWNhdGlvbk1ldGhvZCI6IlF1ZXJ5RGF0YWJhc2VBdXRoZW50aWNhdGlvbkhhbmRsZXIiLCJsb25
nVGVybUF1dGhlbnRpY2F0aW9uUmVxdWVzdFRva2VuVXNlZCI6ImZhbHNlIiwiY2hhbmdlX3B3ZCI6ImZhbHNlIiwiZXhwI
joxNjE2NDU5OTIwLCJpYXQiOjE2MTY0MzExMjAsImZpcnN0X25hbWUiOiJqb2huIiwianRpIjoiU1QtODQtOFVlWXMybEt
3R2d1Z3RIYjl4MzVmTFlNTGVVRlp6OURyNGpoeFcxakhsV01VYXdXSWgxbUdTd01aRC1tOEk1S2Z0amI2ZmlWTUhlYnBYY
jBMMFZqRFc2WVppUFVUbHRpNFVpZnNUeG9aQ284WWpPWEc2VlFjS0Mwb29lWjJhc3BWanMzYnA3bHo5VkhySlBCTzl5TDN
GcFRIWXRPeWJtVi1jYXMtMSJ9.Vi4k0w8KsOv5M_O8zBqWochT3k9V9Pn2NjSn5ES9c5Pf-
4ds0o4kk6xuZx5_cggauiEICuUMnzmRzneST-oAuA' \
--data-raw '{
  "input": {
    "head-end": "100.100.100.5",
    "end-point": "100.100.100.7",
    "sr-policy-path": {
      "path-optimization-objective": "igp-metric"
    }
  }
}'
```

Step 3 Note the SID list ID in the API response. You will use this when creating the SID list in the next step. For example:

```
{
  "cisco-crosswork-optimization-engine-sr-policy-operations:output": {
    "segment-list-hops": [
      {
        "step": 0,
        "sid": 23002,
        "ip-address": "100.100.100.7",
        "type": "node-ipv4"
      }
    ],
    "igp-route": [
      {
        "node": "PE-A",
        "interface": "GigabitEthernet0/0/0/0"
      },
      {
        "node": "P-TOPLEFT",
        "interface": "GigabitEthernet0/0/0/2"
      },
      {
        "node": "P-BOTTOMRIGHT",
```

```
          "interface": "GigabitEthernet0/0/0/3"
        }
      ],
      "state": "success",
      "message": ""
    }
}
```


Step 2 Create the SID List in the Provisioning UI

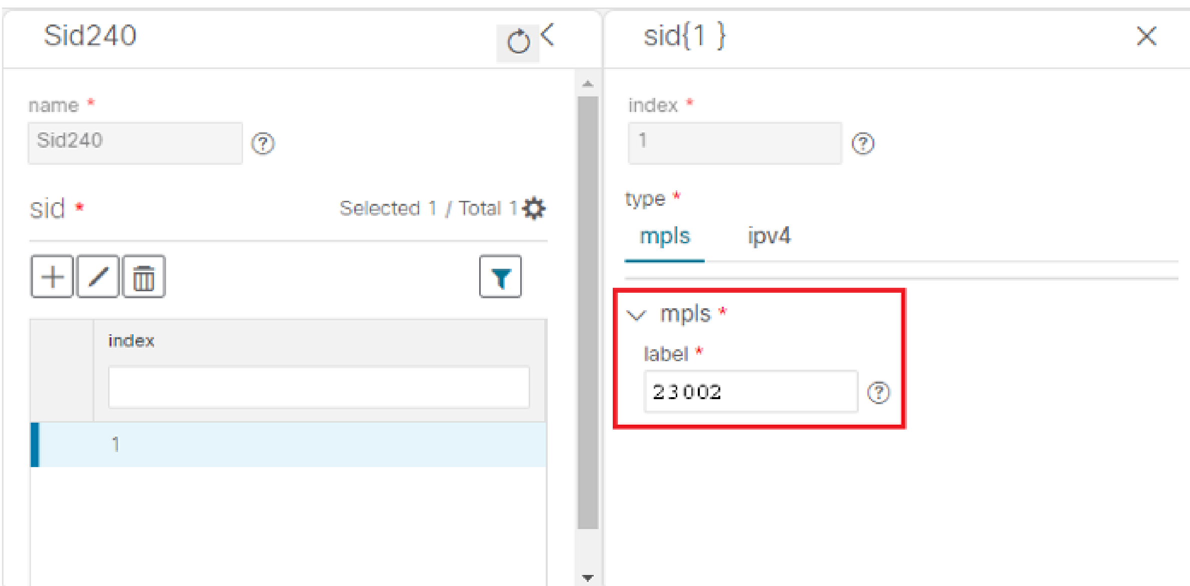
In this scenario, we will create a SID list for traffic from PE-C to PE-A and another SID list for traffic in the opposite direction.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **SR-TE** > **SID-List**.

Step 2 Click **+** to create a new SID list and give it a unique name. For this example, the SID list name is **L2VPN_NM-P2P-SRTE-PE-C-240**. Click **Continue**.

Step 3 Under sid, click + to create a new SID index and give it a numeric value. Click **Continue**.

Step 4 Under mpls, enter the SID ID that was received in the API response in Step 1.



Step 5 Click **X** in the top-right corner to return to the SID list. Your new SID appears in the index table.

Step 6 Repeat these steps to create additional SID indexes, as required.

Step 7 Commit your changes.

Step 8 Check that the new SID list appears in the table.

Step 9 Create another SID list for the traffic from PE-A to PE-C. For this example, the SID list name is **L2VPN_NM-P2P-SRTE-PE-A-240**.

Step 3 Create an explicit SR-TE policy for each VPN endpoint by importing a file

In this step, we will provision two explicit SR-TE policies which will reference the SID lists created in Step 1.

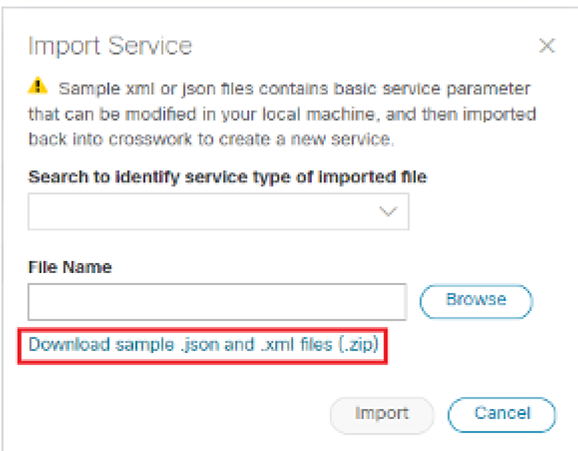
The first SR-TE policy specifies PE-C as the headend and provides the IP address of PE-A as the tail end. The second SR-TE policy specifies PE-A as the headend and provides the IP address of PE-C as the tail end.

Instead of manually filling in each field in the provisioning UI, we will import an xml file containing all the configurations required to create the SR-TE policy.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **SR-TE** > **Policy**.

Step 2 Click Import button above the table .

Step 3 Download the sample .json or .xml file which will serve as a template for the required configuration. In the Import Service dialog, click the **Download sample .json and .xml files (.zip)** link



Step 4 Unzip the downloaded file and open sr-Policy.xml in an XML editor.

Step 5 Edit the xml file as required. Provide a name for the SR-TE policy, and specify the SID list to be associated with this policy. Save the xml file.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
    <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te-sr-policies">
      <policy>
        <name>SR-Policy-1</name>
        <head-end>
          <name>iosxrv-5</name>
        </head-end>
        <tail-end>7.7.7.7</tail-end>
        <color>100</color>
        <binding-sid>100</binding-sid>
        <path>
          <preference>100</preference>
          <dynamic>
            <metric-type>te</metric-type>
            <metric-margin>
             <relative>40</relative>
            </metric-margin>
            <constraints>
              <sid-limit>10</sid-limit>
            </constraints>
          </dynamic>
        </path>
        <path>
          <preference>200</preference>
          <explicit>
            <sid-list>
              <name>mysidlist</name>
              <weight>10</weight>
            </sid-list>
            <constraints>
              <affinity>
                <rule>
                  <action>include-all</action>
                  <color>GREEN</color>
                  <color>RED</color>
                </rule>
              </affinity>
            </constraints>
          </explicit>
        </path>
      </policy>
      <sid-list>
        <name>mysidlist</name>
        <sid>
          <index>1</index>
          <mpls>
            <label>17001</label>
          </mpls>
```

Step 6 In the Import Service dialog, select **Policy** as the type of file to import, browse to the edited xml file, and click **Import**. If there are any errors in the file, you will be notified. If there are no errors, the file will be imported. The policy will be created and the devices will be configured accordingly.

Step 7 Check whether the new SR-TE policy appears in the Policy table and its Provisioning State is **Success**.

Step 8 Click in the Actions column and choose **Config View** to see to see the Yang model-based service intent data that details the SR-TE policy you created. You can also check the devices themselves to make sure that they were provisioned correctly.

Step 4 Create and provision the L2VPN service

In this step, we will create and provision a P2P VPN service with PE-A and PE-C as the endpoints. The VPN service will reference the SR-TE policies we created in the previous step to ensure that the traffic traversing the VPN will follow the path defined in the SID lists.

As we did with the SR-TE policy, we will create the VPN service by importing an xml file containing all the required configurations. Once we have provisioned the VPN service, we will edit it in the provisioning UI in order to associate the SR-TE policies.

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L2vpn** > **L2vpn-Service**.

Step 2 Click Import button above the table.

Step 3 If you did not download the sample .json or .xml files in Step 3, do so now.

Step 4 Open l2nm.xml in an XML editor.

Step 5 Edit the xml file as required. Provide a name for the L2VPN, configure each endpoint, and define the VPN parameters.

This is the configuration for PE-A in our example:


```
    <vpn-node-id>xrv9k-22</vpn-node-id>
    <signaling-option>
      <ldp-or-l2tp>
        <pw-peer-list>
          <peer-addr>192.168.0.22</peer-addr>
          <vc-id>100</vc-id>
          <mpls-label xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">100</mpls-label>
        </pw-peer-list>
      </ldp-or-l2tp>
    </signaling-option>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>300</id>
        <interface-id>GigabitEthernet0/0/0/1</interface-id>
        <connection>
          <encapsulation>
            <encap-type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:dot1q</encap-type>
            <dot1q>
              <cvlan-id>100</cvlan-id>
            </dot1q>
          </encapsulation>
        </connection>
      </vpn-network-access>
    </vpn-network-accesses>
    <te-service-mapping xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
      <te-mapping>
        <sr-policy>
          <policy-type>policy</policy-type>
          <policy>SR-300</policy>
        </sr-policy>
      </te-mapping>
    </te-service-mapping>
  </vpn-node>
  <vpn-node>
    <vpn-node-id>xrv9k-23</vpn-node-id>
```


Step 6 Save the xml file.

Step 7 In the Import Service dialog, select **l2vpn service** as the type of file to import, browse to the edited xml file, and click **Import**. If there are any errors in the file, you will be notified. If there are no errors, the file will be imported. The service will be created and the devices will be configured accordingly.

Step 8 Check that the new L2VPN service appears in the L2VPN Service table and its Provisioning State is **Success**.

Step 9 Click in the Actions column and choose **Config View** to see the Yang model-based service intent data that details the VPN service you created. You can also check the devices themselves to make sure that they were provisioned correctly.

Step 5 Attach the SR-TE policies to the L2VPN Service

At this stage, the provisioned L2VPN service you created does not have associated SR-TE policies that define the transport path. In this step, we will edit the L2VPN service in the provisioning GUI, attach the relevant SR-TE policies to each endpoint, and re-provision it.

Step 1 Locate the L2VPN in the VPN Service table.

Step 2 Click  in the Actions column and choose **Edit**.

Step 3 Under vpn-nodes, select **PE-A** and click the **Edit** button above the table.

Step 4 In the pane that opens on the right, open the **te-service-mapping** > **te-mapping** section.

Step 5 In the sr-policy tab, in the policy field, enter the name of the SR-TE policy created for PE-A:**L2VPN_NM-P2P-SRTE-PE-A-240**.

Step 6 Click **X** in the top-right corner to close the PE-A pane.

Step 7 Repeat the above steps for PE-C and attach the SR-TE policy: **L2VPN_NM-P2P-SRTE-PE-C-240**.

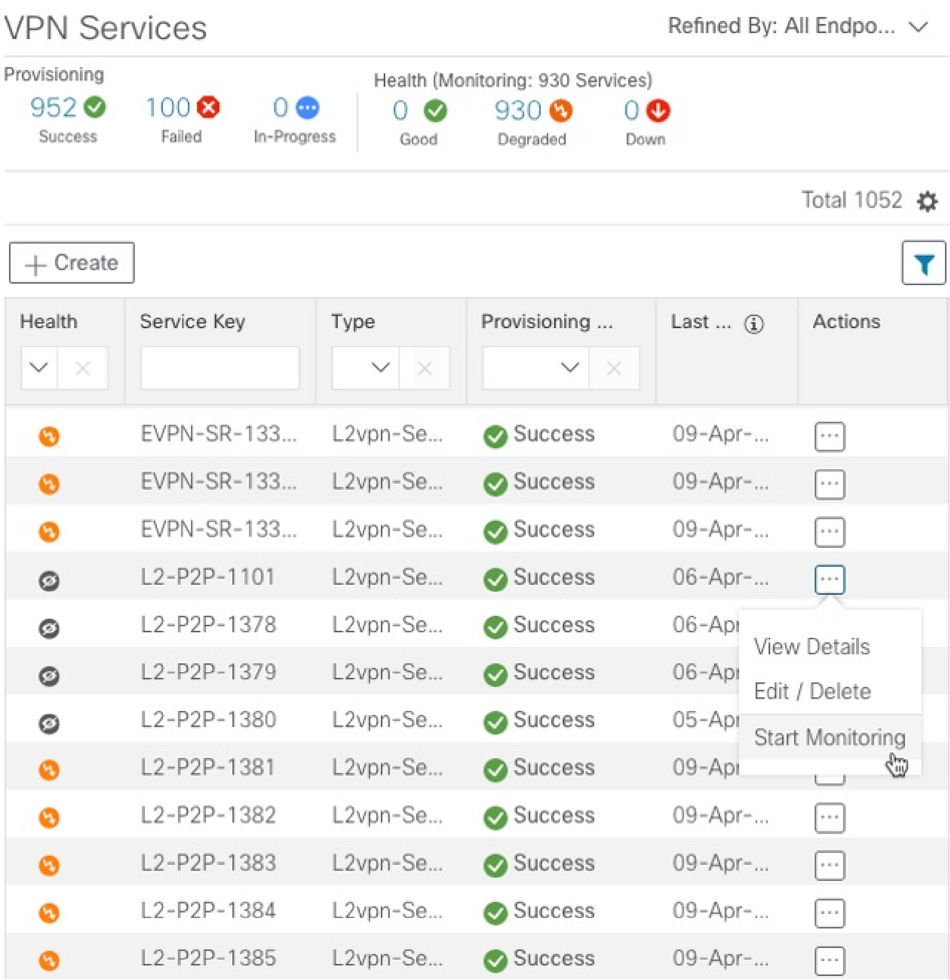
Step 8 Click **Commit Changes**.

Step 6 Enable Service Health monitoring

Step 1 Go to **Services & Traffic Engineering** > **VPN Services**. The map opens and a table of VPN Services is displayed to the right of the map.

Step 2 In the Actions column, click  for the new service you want to start monitoring health.

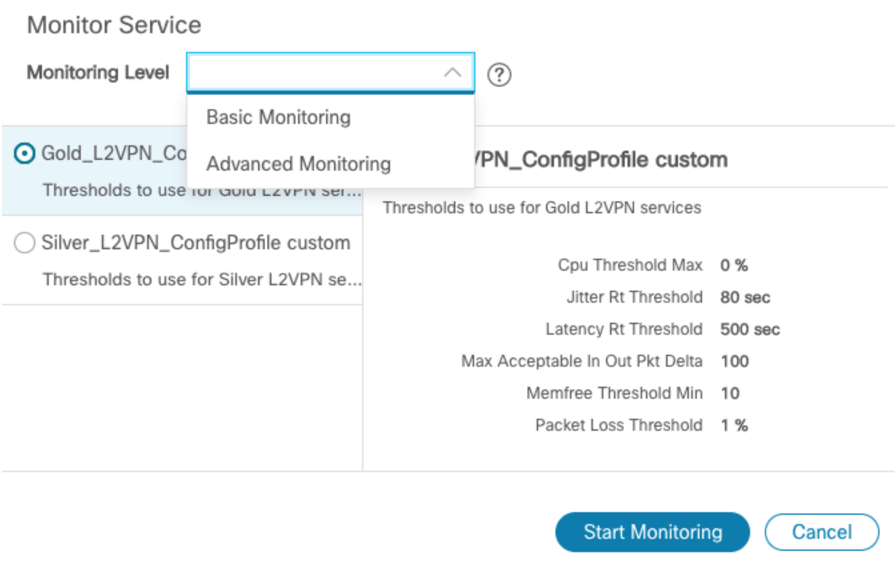
Step 3 Click **Start Monitoring**.



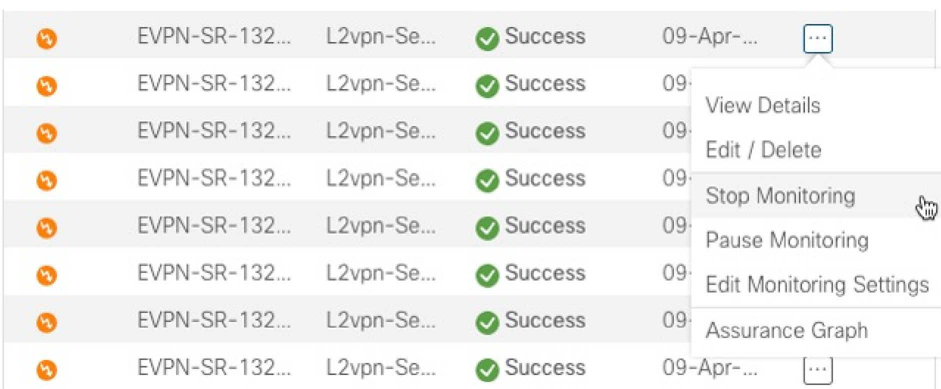
Note The Health column color coding indicates the health of the service:

- Blue = Initiated
- Green = Good
- Orange = Degraded
- Red = Down
- Gray = Not Monitoring

Step 4 In the Monitor Service pop-up, select the Monitoring Level. For help selecting the appropriate monitoring level option for your needs, see the section Basic and Advanced Monitoring Rules



Once you have started monitoring the health of this service, if you select the Actions column and click [...] to view additional Service Health options, you will see: **Stop Monitoring**, **Pause Monitoring**, **Edit Monitoring Settings, Assurance Graph**.



Note If you select **Edit Monitoring Settings**, you may update the Monitoring Level setting from Basic Monitoring to Advanced Monitoring, or from Advanced Monitoring to Basic Monitoring, at any time.

Note If you later decide to **Stop Monitoring** a service that has already been started, you have the option to retain the historical service data for that stopped service. See Stopping Service Health Monitoring in the Appendix for additional steps and details.

Step 5 Click **Start Monitoring**.

Step 6 Repeat this step for each service you wish to start health monitoring.

Step 7 Click **X** in the top-right corner when you are done.

Step 7 Visualize the L2VPN on the Map

In this step we will take a look at the representation of the L2VPN on the map, and we'll see the paths the traffic will take from PE-A to PE-C and vice versa, based on the explicit SR-TE policies we created.

Step 1 In the L2VPN Service table, in the Actions column for the new VPN, click and choose **ViewDetails** from the menu. The map opens and the service details are shown to the right of the map.

or

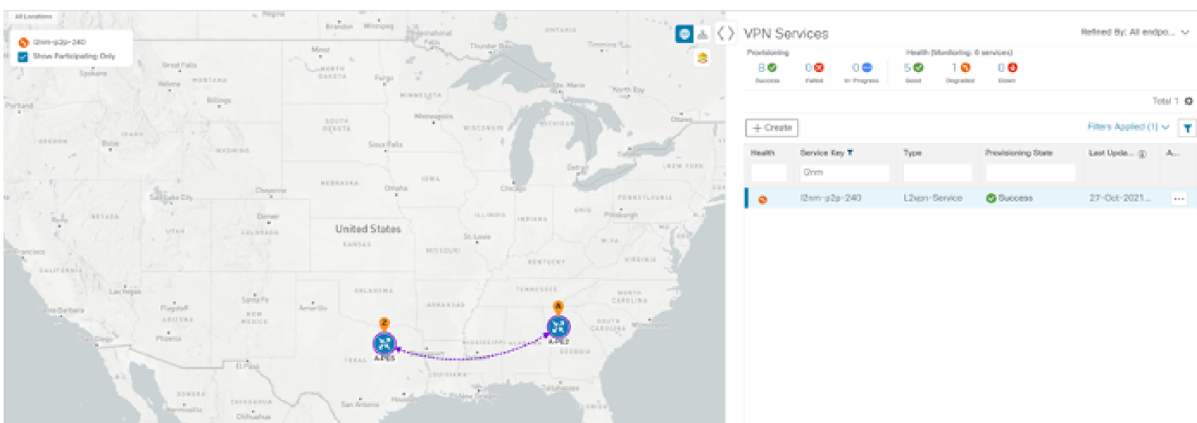
Go to **Services & Traffic Engineering** > **VPN Services**.

The map opens and a table of VPN services is displayed to the right of the map.

a) Click on the VPN in the Services table. If there are many services in the table, you can filter by name, type, or provisioning state to help locate the VPN.

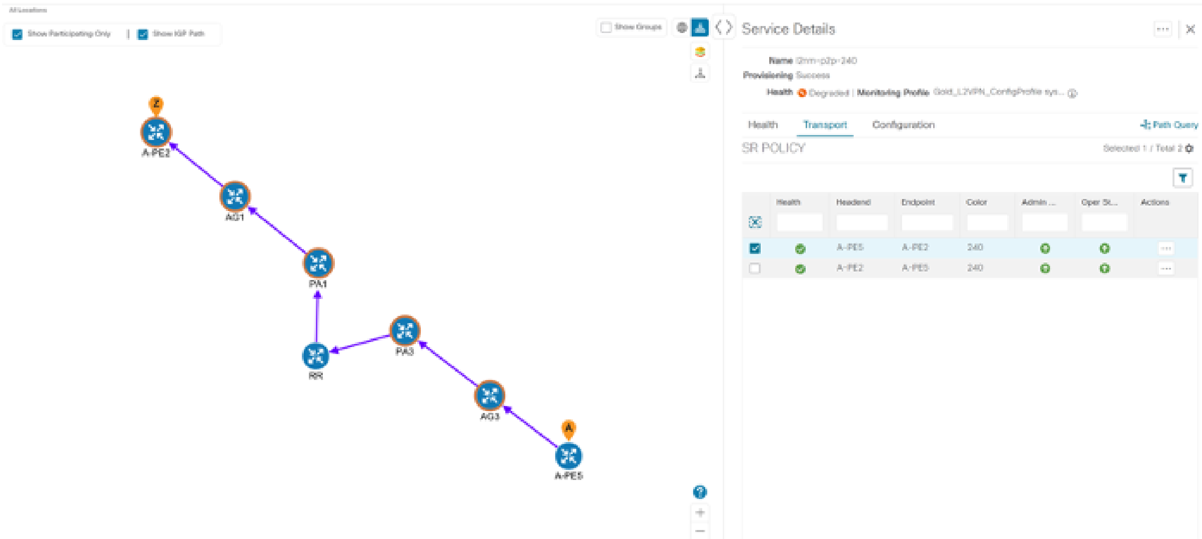
b) In the map, you will see the VPN as an overlay on the topology. It shows a representation of the endpoints and a solid line that indicates that it is a virtual path.

c) Select the **Show Participating Only** check box if you do not want to see the devices that are not involved in the selected VPN.



Step 2 Under the Actions column, click and choose **View Details** to drill down to a detailed view of the VPN service, including the device configurations, the computed transport paths, and the health status for transport paths.

Step 3 In the Transport tab, select one or more SR-TE policies to see the path from endpoint to endpoint on the map. The image below shows the path for PE-C to PE-A. The **Show IGP Path** check box in the top left corner of the map is selected so the physical path is shown. The dashed line indicates that this link is being used to transport multiple services.



Step 8 Inspect a degraded service using Service Health and Last 24Hr Metrics to identify issues

In this step, we will review the Service Health assurance graph and utilize the Last 24Hr Metrics to identify issues within a specific time range. By isolating the issues within a specific time range, you can drill down on the details that may have caused the degraded (or down) service that can lead to troubleshooting the service or the node to address detailed symptoms. For this example, we will inspect a degraded service.

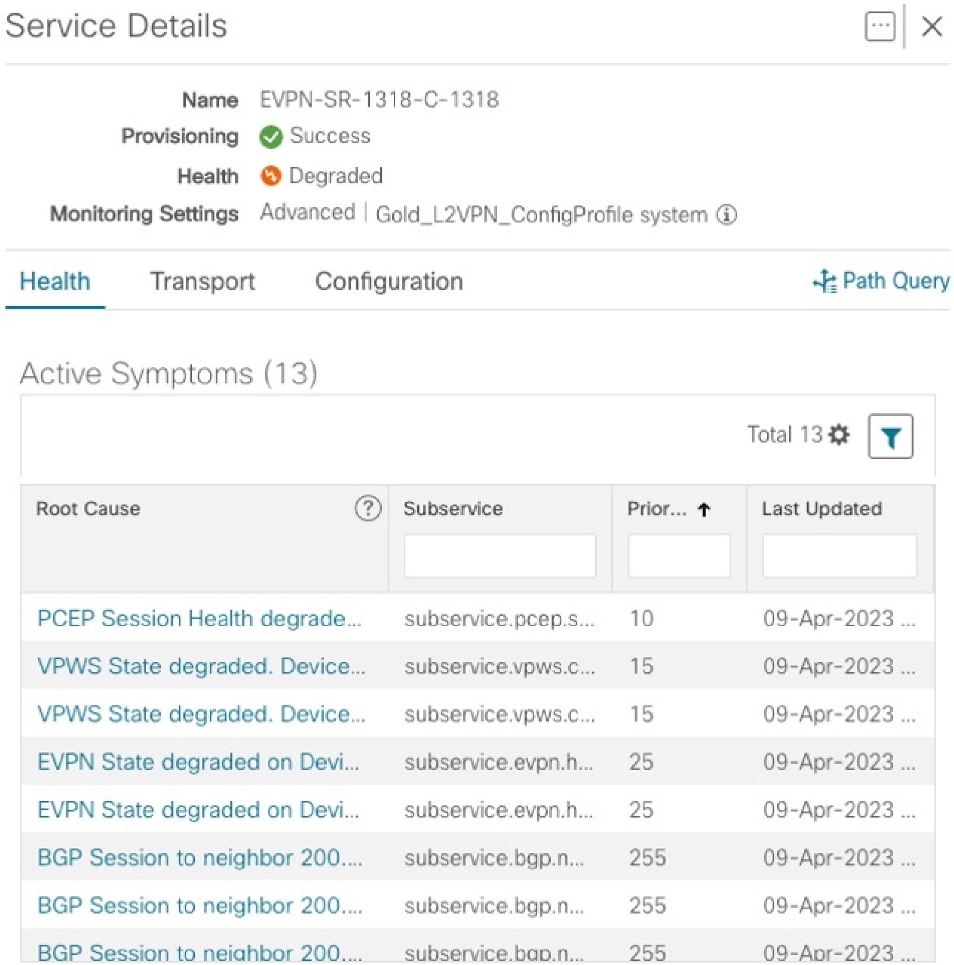
Step 1 Click **X** in the top-right corner to return to the VPN Services list.

Step 2 Click on the name of a service that shows as being degraded. The map will update to highlight the service you selected.

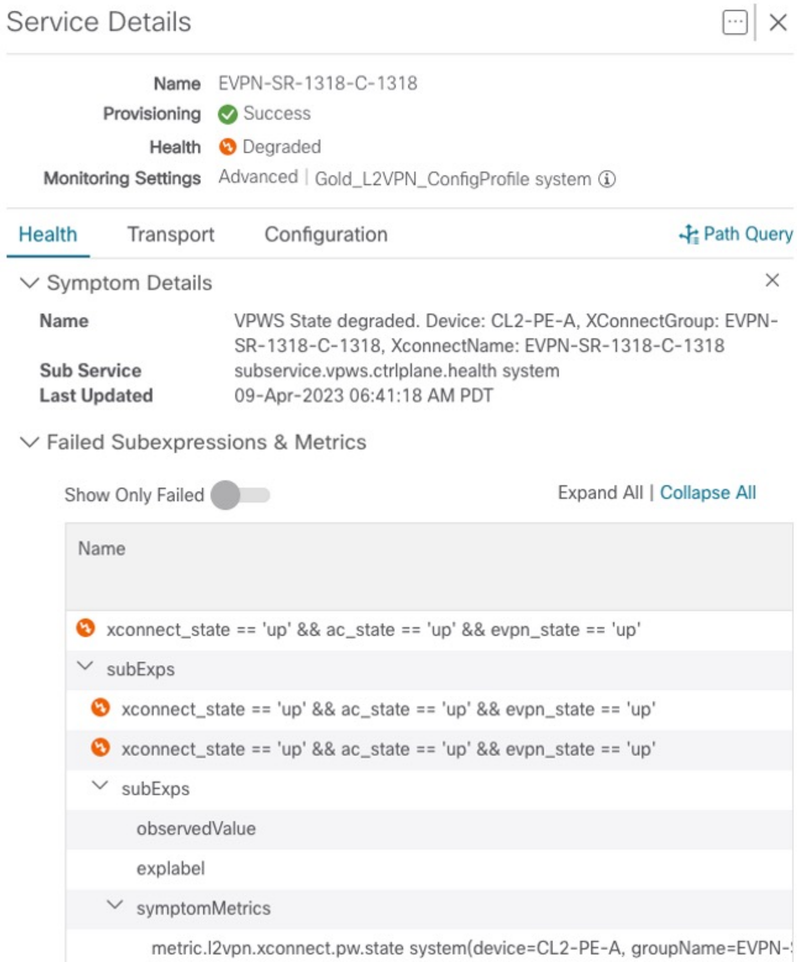
Degraded services show an orange icon in the Health column. You can filter by health state (Down, Degraded, Good) by clicking in the space at the top of the column and selecting the appropriate filter. To clear the filter, click the **X** next to the designated filter appearing in the space at the top of the column and it will remove all filtering and default to showing all VPN Services in the list.

Note If a service is not yet being monitored, the icon in the Health column will show as the color grey. To enable monitoring for such a service, click ⋯ and select **Start Monitoring.**

Step 3 In the Actions column, click ⋯ and click **View Details**. The Service Details panel appears on the right side where you can review Active Symptoms for the service (including the Root Cause, Subservice, Priority, and Last Updated details) present in the Health tab if the service is being currently monitored. Review the details provided.



Step 4 Click on a Root Cause and view both the Symptom Details and the Failed Subexpressions & Metrics information.



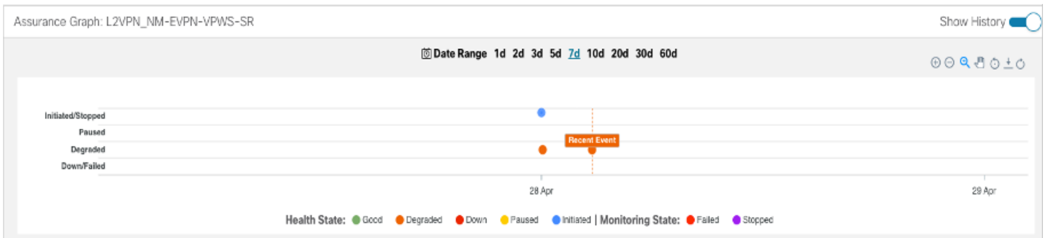
Step 5 To further isolate the degraded service details, click **X** in the top-right corner to return to the VPN Services list.

Step 6 Again, click on the name of the degraded service in the list. The Service Details panel appears and the map updates, isolating the corresponding devices participating with that service.

Step 7 In the Actions column, click  for the degraded service in the list and click **Assurance Graph**. The topology map of services and subservices appear with the Service Details panel showing Service Key, Status, and Sub Services details. Metrics also appear, such as Jitter-RT (Jitter Round Trip), Latency-RT (Latency Round Trip), PktLoss-DS (Packet Loss from Destination to Source), and PktLoss-SD (Packet Loss from Source to Destination). Additionally, a table of Active Symptoms listing Root Cause, Subservice, Priority, and Last Updated details is populated.

Note This will take time to update after a service has been enabled for monitoring, and may take up to 5-10 minutes.

Step 8 At the top-right of the screen, select the **Show History** mode toggle. The historical Date Range graph appears. This graph shows different ranges of historical health service monitoring details from one day (1d) up to sixty days (60d). You can select the (+) icon at the top-right to zoom in on the event or use your mouse to draw a rectangle over events to further zoom. Events that are consecutive may appear as a line of white space.

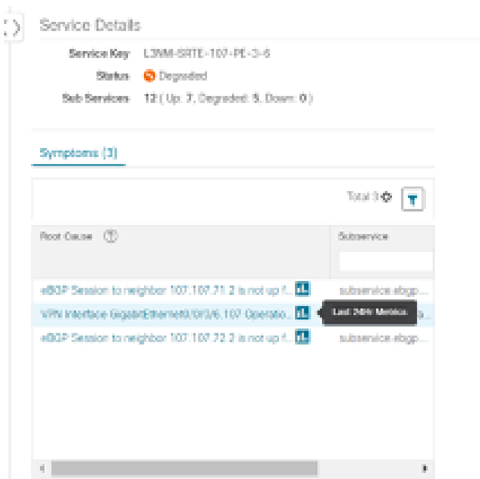


Note When you select an event on the Date Range graph, a tool tip with information about that event appears (such as date and time of the event, and severity level and number of symptoms). Click anywhere within the chart to hide the tool tip.

Step 9 Review the Root Cause information by either hovering your mouse over a particular row or click the arrow to expand the Service Details panel to full screen mode. Columns can be resized using your mouse or you can select the gear icon to deselect or select columns you want to appear.

Note Once you enable **Show History** mode, Root Cause information in the Active Symptoms table will start to show the blue Last 24Hr Metrics icon. Data from the device will be initially delayed, however, and may take some time before **Last**

24Hr Metrics begins to populate with data. Until then, the value of zero is reported.



Step 10 You can also use the map and click on the degraded node to bring up Service Details information on both Active Symptoms and Impacted Services.

- **Active Symptoms**: Provides symptom details for nodes actively being monitored.
- **Impacted Services**: Provides information for services that are impacted by issues based on historical monitoring of health status.

Note If you view the Subservice Details panel, each subservice metric (Jitter-RT, Latency-RT, PktLoss-DS, PktLoss-SD) will initially report a value of zero. Based on a device's configuration, it may take up to 10 minutes for the metric values to begin reporting.

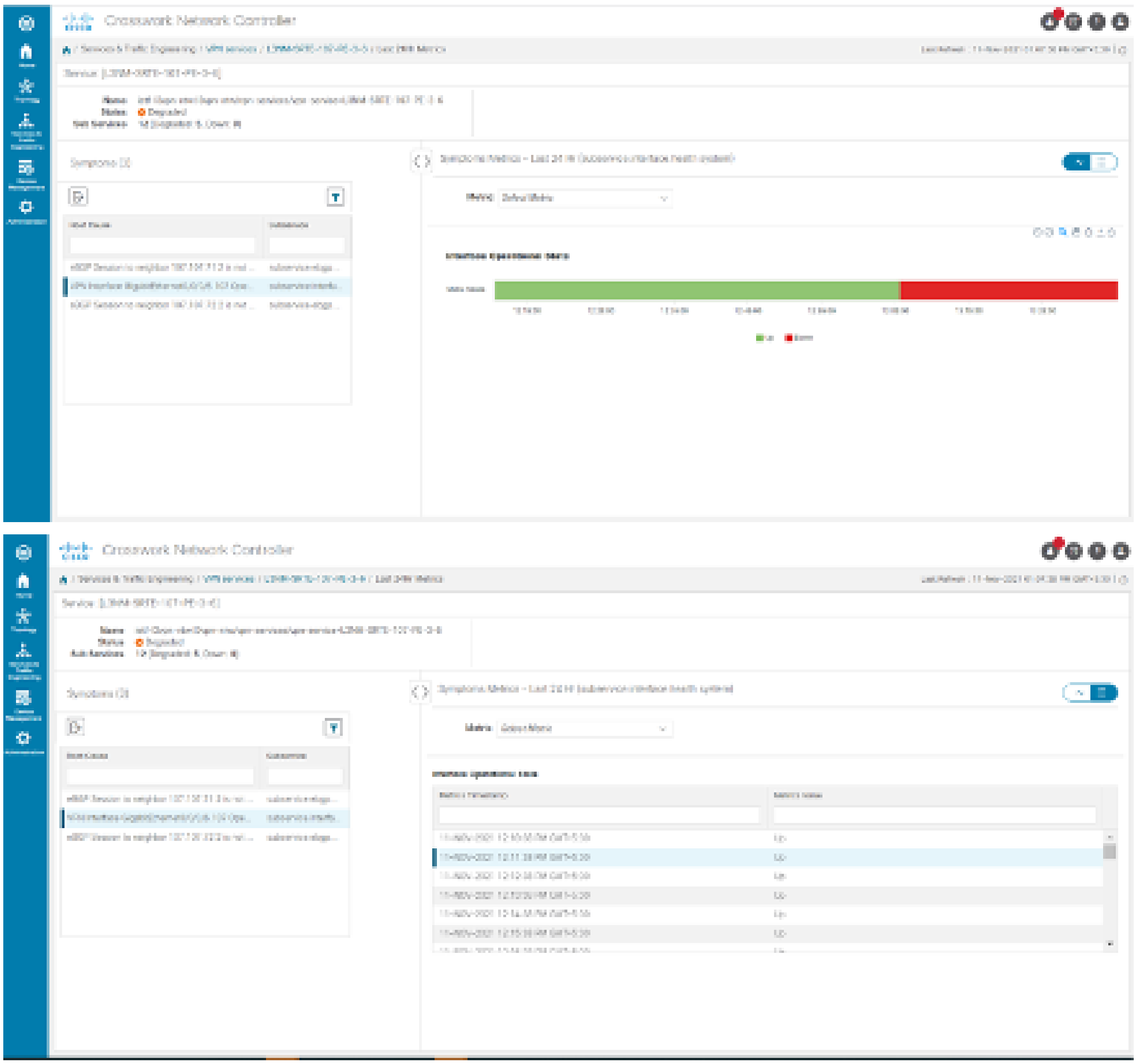
Step 11 Use the active and impacted information toinspect the degraded service details to determine the issues that led to the degraded service

Step 12 To further isolate the possible issues and to utilize the **Last24Hr Metrics**, perform the following steps:

a) In the Date Range graph, use your mouse to select the range of historical health service monitoring details from one day (1d) up to sixty days (60d).

Note At the top-right of the Date Range graph, select the appropriate icons to either zoom in or out, horizontally scroll through the date ranges, or refresh the graph to go back to the most recent event, for example. You can also use your mouse to draw a rectangle over events to further zoom in on degraded devices. Events that are consecutive may appear as a line of white space.

b) Click on a degraded service in the graph. The Service Details panel reloads, showing any active symptoms and the root causes to be inspected. Expand the table and information as necessary for further details.



Step 13 Next, select the **Show: Down & Degraded Only** check box in the top-left corner of the map so only Subservices which are degraded, along with other dependent but healthy subservices, appear. Inspect the Service Details panel showing the active symptoms and their root cause.

Step 14 Deselect the **Show: Down & Degraded Only** check box and select the **Soft Dependencies** check box in the top-left corner of the map. Soft Dependencies implies that a child subservice's health has a weak correlation to its parent's health. As a result, the degraded health of the child will not result in the parent's health degradation. Use the **+** or **–**

symbols in the bottom-right corner of the map to zoom in or out on services mapped. Select the **?** to view the Link Color Legend that explains all of the icons, symbols, badges, and colors and their definitions

Note You can also select the **Subservices** icon in the top-right corner of the map to show service appearance options.

Step 15 Select the degraded service in the map to show the subservice details .

Step 16 Select the **Active Symptoms** tab to show any root causes for the service health details that are displayed and then select the **Impacted Services** tab to show services where their health is degraded.

Step 17 Click **X** in the top-right corner to return to the VPN Services list and in the Actions column, click for the degraded service in the list and click **Assurance Graph** to show the Service Details panel.

Step 18 Again, select the **Show History** toggle in the top-right corner of the Service Details panel before selecting the blue metrics icon in one of the Root Cause rows. The Symptoms Metrics – Last 24 Hr bar chart appears. This chart provides details of the metric patterns for different sessions states (such as active, idle, failed) for individual root cause symptoms with Status, Session, Start Time, and Duration information to assist in troubleshooting prevailing issues. Use your mouse to hover over the chart to view the different details.

Continue to troubleshoot a service health issue using Parameterized Jobs

To further troubleshoot a service health issue (such as a device that is degraded due to not properly fetching data), continue with the following steps to examine if the issue is associated with a collection job.

Step 19 Select **Administration > Collection Jobs**.The Collection Jobs screen appears.

Step 20 Select the Parameterized Jobs tab.

Step 21 Review the Parameterized Jobs list to pinpoint devices that may have service health degradation issues. By reviewing Parameterized Jobs, you can identify and focus on GMNI, SNMP, and CLI-based jobs by their Context ID (protocol) for further troubleshooting purposes.

Step 22 In the Job Details panel, select the collection job you want to export and click the **export** button to download the status of collection jobs for further examination. The information provided is collected at the time the export is initiated in a .csv file. The Export Collection Status pop up appears.

Note When exporting the collection status, you must fill in the information each time an export is executed. In addition, make sure to review the Steps to Decrypt Exported File content available on the Export Collection Status pop up to ensure you can access and view the exported information.

Step 23 Click **Export**.

Step 24 To check the status of the exported collection job data, click **View Export Status** at the top right of the Job Details panel. The Export Status Jobs panel appears providing the status of the export request.

Step 25 Review the exported .csv file for collection job details and the possible cause of the degraded device.

Summary and Conclusion

In this scenario, we observed how simple it is to create explicit SR-TE policies and attach them to a L2VPN service in order mandate a static path for the mission-critical traffic. We saw how editing a pre-defined template and then importing it into the system enables quick and easy provisioning of services and SR-TE policies. We were then able to visualize the actual traffic paths on the map. Lastly, we used Service Health to monitor the health of the new service using the Assurance Graph, Last 24hr Metrics, and SubExpressions metrics to view when service may have been up, degraded, or down, and what the root causes were identified.

Scenario 4 – Provision an L2VPN service over an RSVP-TE tunnel with reserved bandwidth

For the continuous stream transmission required for rich data media types, such as video and audio, bandwidth reservation is often required to provide higher quality of service. Cisco Crosswork Network Controller supports the creation and management of RSVP-TE tunnels to reserve guaranteed bandwidth for an individual flow. RSVP is a per-flow protocol that requests a bandwidth reservation from every node in the path of the flow. The endpoints, or other network devices on behalf of the endpoints, send unicast signaling messages to establish the reservation before the flow is allowed. If the total bandwidth reservation exceeds the available bandwidth for a particular LSP segment, the LSP is rerouted through another LSR. If no segments can support the bandwidth reservation, LSP setup fails and the RSVP session is not established.

In this scenario we will:

- Create RSVP-TE tunnels with reserved bandwidth.
- Enable Bandwidth on Demand functionality.
- Provision a VPN service from PE-A to PE-B and attach the RSVP-TE tunnels as underlay configuration.
- Visualize the path of the traffic when link utilization is below the bandwidth threshold. This path would change if the bandwidth utilization on the link crossed the specified threshold.

Assumptions and Prerequisites

Scenario 4 to provision an L2VPN service over an RSVP TE Tunnel with reserved bandwidth the following are the assumptions and prerequisites.

- For transport mapping to L2VPN service, devices must be configured with the **l2vpn all** command.
- For Service Health enablement and usage to monitor a services health, Service Health must be installed.
- For steps to enable Service Health during this scenario, see Scenario 3, Step 6 Enable Service Health monitoring. For additional Service Health related details, see Scenario 1 – Implement and Maintain SLA for an L3VPN Service for SR-MPLS (using ODN), Scenario 3 – Mandate a Static Path for an EVPN-VPWS Service using an Explicit MPLS SR-TE Policy, and the Initializing Heuristic Packages to Monitor the Health of a Service.
- (Optional) Service Health provides **Internal Storage** of monitoring data up to a maximum limit of 50 GB. This data is stored on your system. If you exceed the limit of the internal storage, the least recently used historical data will be lost. If you choose to extend Service Health storage capacity, you can configure **External Storage** in the cloud using an Amazon Web Services (AWS) cloud account. By leveraging External Storage, all existing internal storage data will be automatically moved to the external cloud storage (see **Configuring Service Health External Storage Settings** appendix for more details) and your internal storage will act locally as cache storage. Configuring External Storage for Service Health ensures you will not lose historical data for services that continue to monitor a service's health, and will retain service health data for any service you choose to stop monitoring when you select the option to retain historical monitoring service for the data. For more information on Internal and External Storage, and how to retain historical monitoring service data when stopped, see the Appendix sections **Configuring Service Health External Storage Settings** and **Stopping Service Health monitoring**.

• (Optional) For initializing a Heuristic Package to monitor health of a services, see the Appendix section, **Initializing Heuristic Packages to monitor the health of a service**, for detailed steps to be performed prior to starting monitoring.

Step 1 Create an RSVP-TE tunnel for both directions of the L2VPN

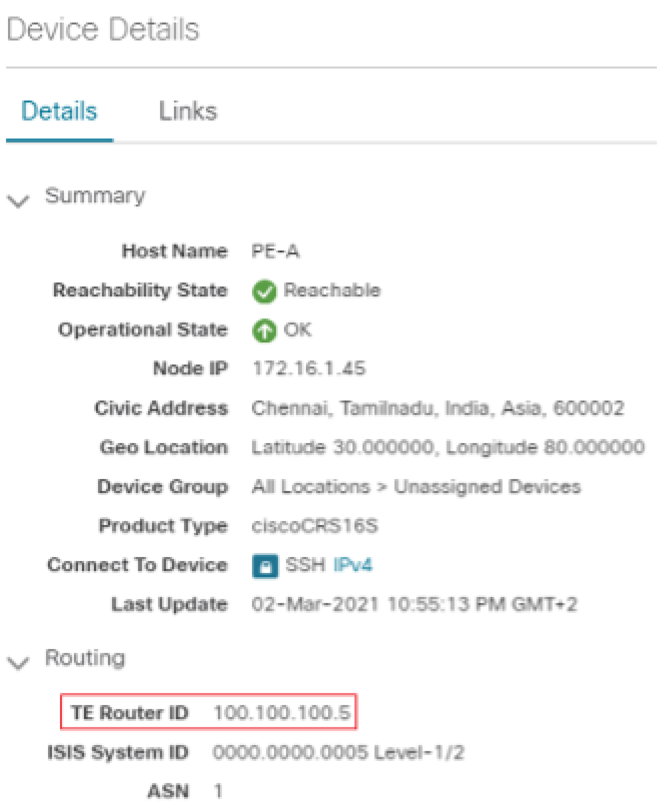
In this step, we will create an RSVP-TE tunnel from PE-A to PE-B and from PE-B to PE-A, and we'll reserve bandwidth of 1200 on the link.

Step 1 Go to Services & Traffic Engineering > Provisioning**(NSO)** > **RSVP-TE** > **Tunnel**.

Step 2 Click + to create a new RSVP-TE tunnel and give it a unique name. Click **Continue**.

Step 3 In the Identifier field, enter a numeric identifier for the tunnel. You will use this identifier later when you associate this RSVP-TE tunnel with the L2VPN service. For this example, the identifier is **2220**.

Step 4 In the source and destination fields, enter the loopback0 IP address of the source (PE-A) and the destination (PE-B) devices. This is the TE router ID. To find the TE router ID, go to Topology and click on a device in the map or in the list of devices. The Device Details pane opens and the TE router ID is shown under the Routing section.



Step 5 Define the endpoints:

a) Under head-end, select the headend device from the dropdown list.

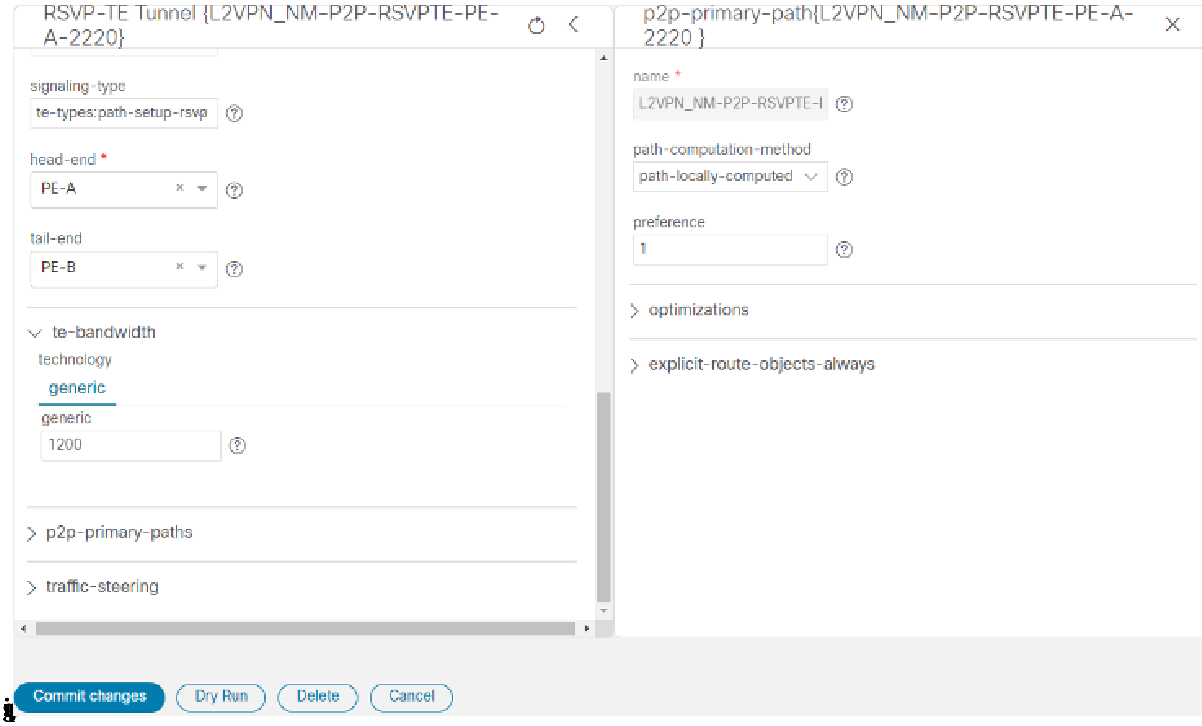
b) Under tail-end, select the tailend device from the dropdown list.

Step 6 Reserve bandwidth on the link. Under te-bandwidth > generic, enter the bandwidth threshold for the link.

Step 7 Define the path of the RSVP-TE tunnel.

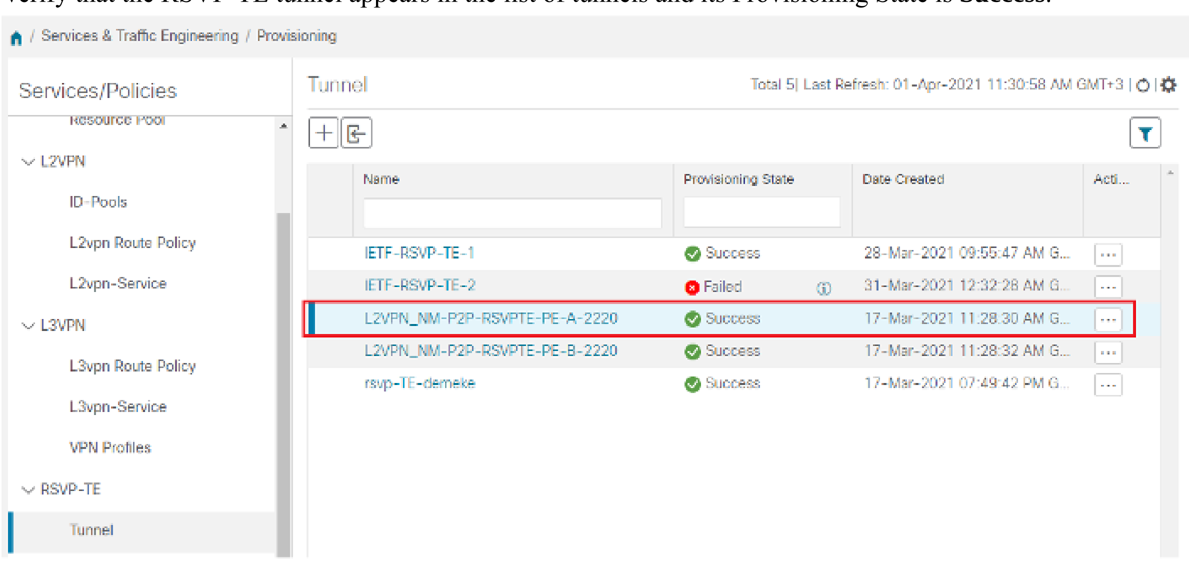
You have the option to define an explicit path or to have the path locally computed by the participating devices. Alternatively, you can have the SR-PCE compute a path dynamically. For this scenario we will have the path locally computed.

a) Under p2p-primary-paths, click + to create a new path.
b) In the pane that opens on the right, give the path a name.
c) Select the path computation method – **path-locally-computed**.
d) Specify a numeric preference for the path. The lower the number, the higher the preference.
e) Define the optimization metric, in this case,

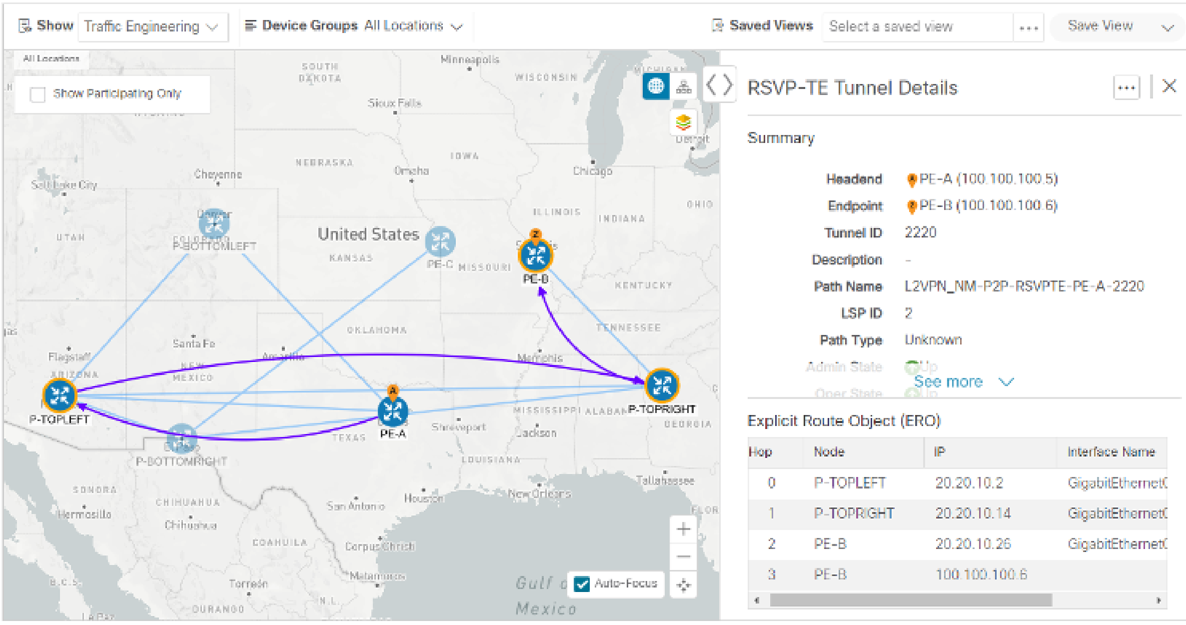


Step 8 Click **Commit Changes**.

Step 9 Verify that the RSVP-TE tunnel appears in the list of tunnels and its Provisioning State is **Success**.



Step 10 Click on the tunnel name to visualize the tunnel on the map and to see the tunnel details.



Step 2 Create the L2VPN service and attach the RSVP tunnel to the service

In this step, we will create a P2P L2VPN service using the provisioning GUI. If you want to create the service by importing a template, refer to Scenario 3—Mandate a static path for an EVPN-VPWS service using an explicit SR-TE policy

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **L2VPN** > **L2vpn Service**.

Step 2 Click + to create a new service and give it a unique name. Click **Continue**.

Step 3 Choose vpn-common:t-ldp in the vpn-svc-type field.

Step 4 Define each VPN endpoint individually – PE-A and PE-B.

a) Under vpn-nodes, click +.

b) Select the relevant device from the vpn-node-id and ned-id dropdown lists and click **Continue**.

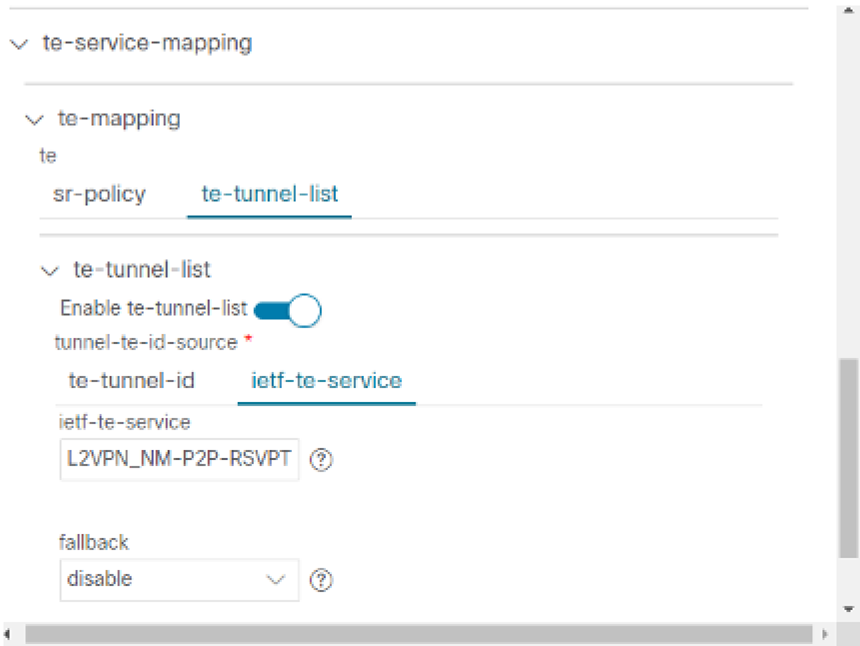
Step 5 Define the LDP signaling options by creating one or more pseudowires. In this case, specify the TE router ID of the peer device (PE-B), and provide a unique numeric label to identify the pseudowire.

Step 6 Attach the RSVP tunnel to the service:

a) Under te-service-mapping > te-mapping, click the te-tunnel-list tab.

b) Click the **ietf-te-service** tab.

c) Enter the name of the RSVP-TE tunnel you want to attach to this L2VPN service. The tunnel ID will be extracted from the tunnel configuration.

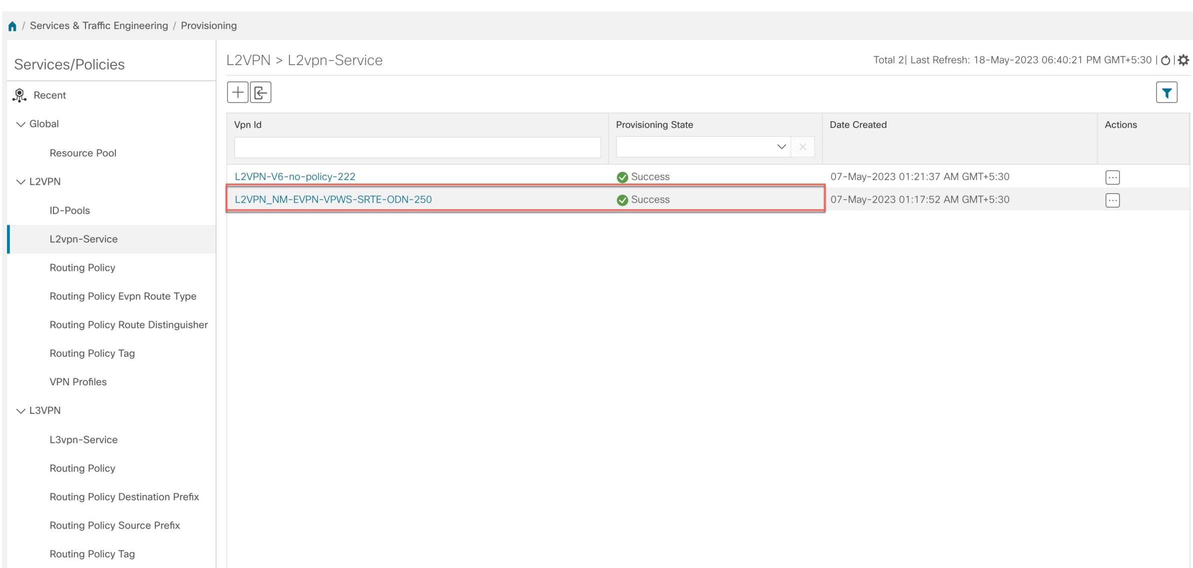


Note If you have an RSVP-TE tunnel on the device that was configured externally to Crosswork Network Controller, you can provide the tunnel ID under the te-tunnel-id tab.

Step 7 Define the VPN network access. In this case, we are using dot1q encapsulation and we have specified the physical interface (GigabitEthernet0/0/0/2) and the VLAN ID (2220).

Step 8 Follow the above steps for PE-B as well.

Step 9 Click **Commit Changes**. Verify that the L2VPN appears in the list of VPN services and that its Provisioning state is **Success**.



Step 3 Visualize the L2VPN service on the map

In this step we'll take a look at the representation of the L2VPN on the map and we'll see the paths the traffic will take from PE-A to PE-B and vice versa, based on the RSVP-TE tunnels we created.

Step 1 In the L2VPN Service table, click on the service name. The map opens and the service details are shown to the right of the map.

or

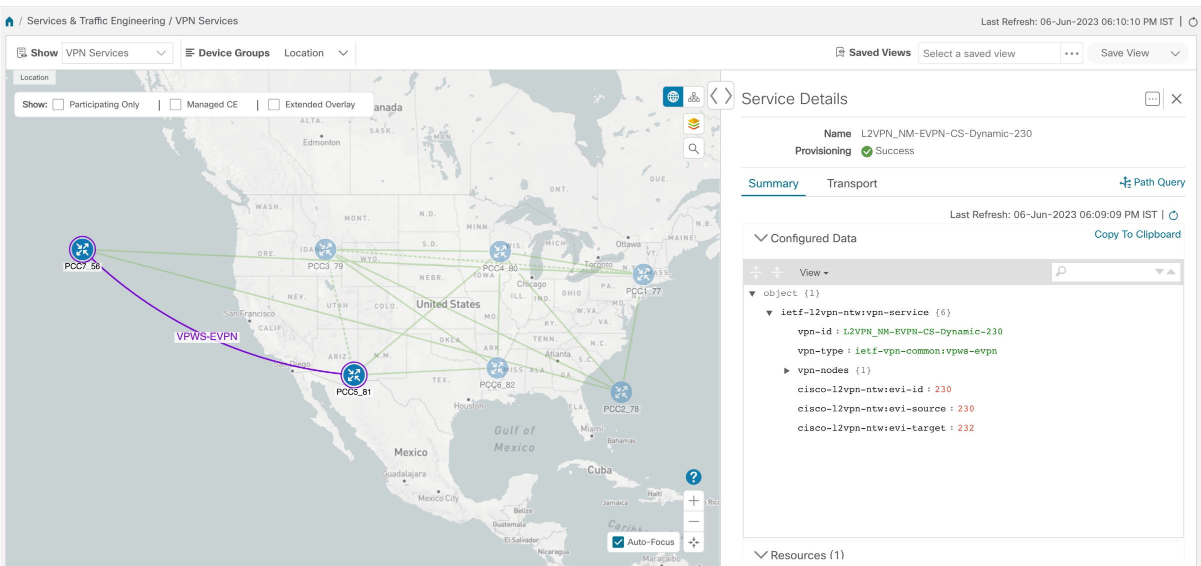
a) Go to **Services & Traffic Engineering** > **VPN Services**.

The map opens and a table of VPN services is displayed to the right of the map.

b) Click on the VPN in the Services table. When there are many services in the table, you can filter by name, type, or provisioning state to help locate the VPN.

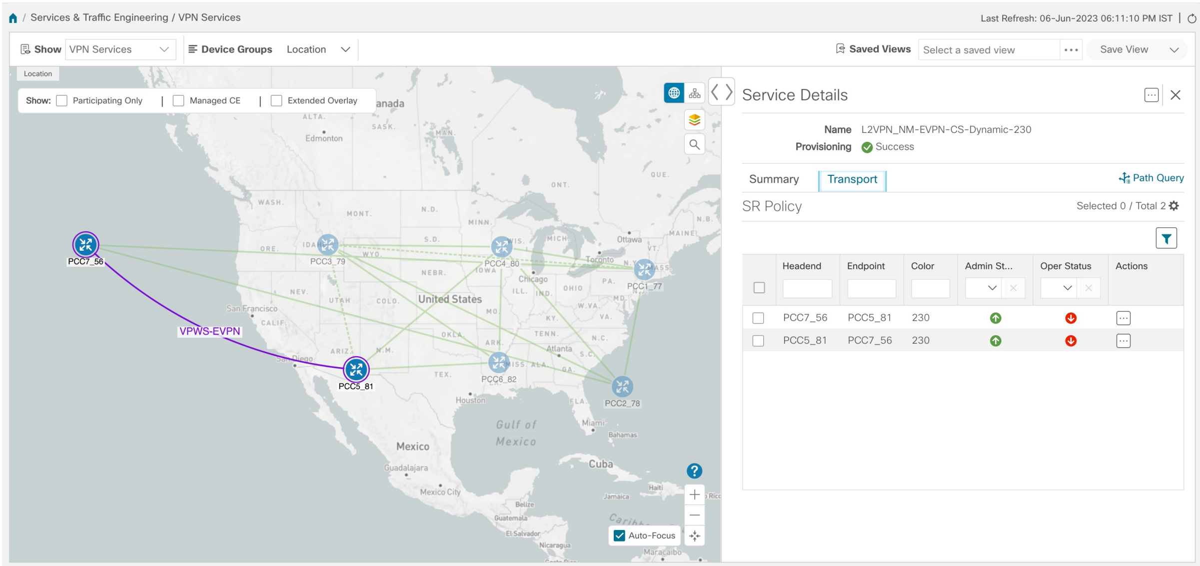
In the map, you will see the VPN as an overlay on the topology. It shows a representation of the three endpoints and a dashed line that indicates that it is a virtual path.

Note The image below shows the VPN overlay in the geographical map. Use the buttons at the top right of the map to toggle between the logical and geographical maps.



Step 2 To see the hops in the route between PCC7_56 and PCC5_81, click the Transport tab and select one or more of the underlying TE tunnels to see the path from endpoint to endpoint on the map. The image below shows both RSVP-TE

tunnels selected in the Transport tab and the route from PCC7_56 to PCC5_81 as shown on the logical map.



Step 3 As the RSVP-TE tunnels are configured with a reserved bandwidth, if the bandwidth utilization across the link exceeds the specified bandwidth, the path would be rerouted.

Summary and Conclusion

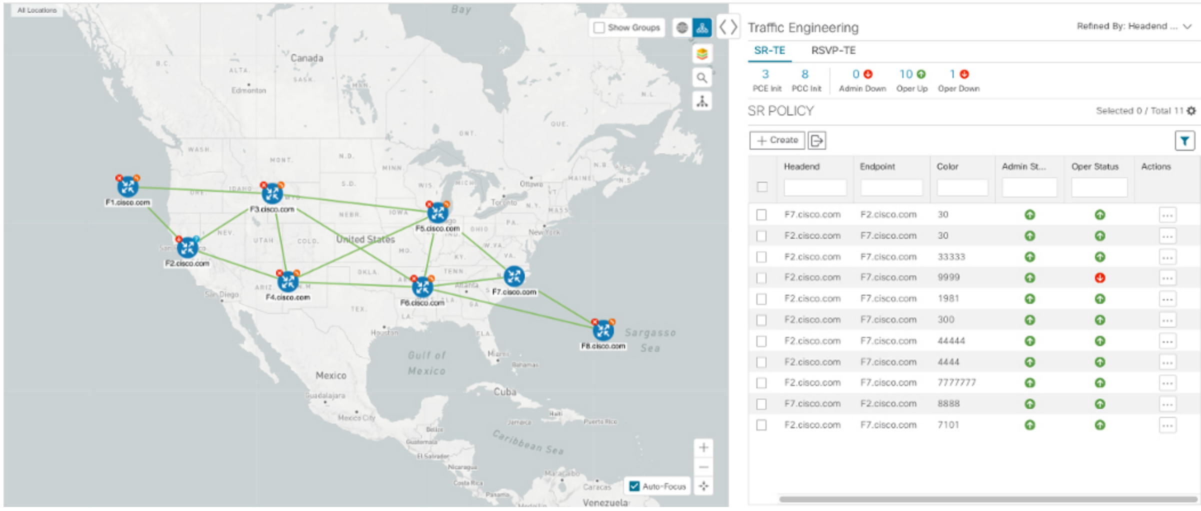
This scenario illustrated how to create RSVP-TE tunnels with reserved bandwidth and attach them to an L2VPN service to meet the high quality of service requirements for continuous streaming of rich data media. We observed the path on the map. This path would be recomputed if the bandwidth utilization on the link crossed the bandwidth reservation threshold.

Scenario 5 – Provision a Soft Bandwidth Guarantee with Optimization Constraints

Service providers must be able to provide fast connections with the lowest latency possible to meet the needs of customers’ peak traffic utilization times and to dynamically optimize services based on the customers’ changing priorities throughout the day. For this purpose, the operator might need to reserve bandwidth on specific links to ensure a dedicated path that can handle a set amount of traffic with a specific optimization intent. The Bandwidth on Demand (BWoD) feature within Crosswork Network Controller enables this functionality. Paths with the requested bandwidth are computed when available. If a path that guarantees the requested bandwidth cannot be found, an attempt will be made to find a *best effort* path.

In this scenario, we will use BWoD to calculate the lowest TE metric path with a specified amount of available bandwidth between two endpoints.

This scenario uses the following topology as a base:



The goal is to create a path from F2.cisco.com to F7.cisco.com that can accommodate 250 Mbps of traffic while keeping the utilization at 80%. BWoD will initially try to find a single path to accommodate the requested bandwidth without exceeding the utilization threshold. If a single path cannot be found, BWoD may recommend splitting the path.

In this scenario we will:

- Orchestrate a new SR-TE policy with bandwidth and TE constraints.
- Configure and enable BWoD.
- Verify the state of the SR-TE policy and view the path on the map.

Step 1 Create a BWoD SR-TE Policy with the Requested Bandwidth and Optimization Intent

To create a BWoD SR-TE Policy with the Requested Bandwidth and Optimization Intent

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)** > **SR-TE** > **Policy**.

Step 2 Click + to create a new SR-TE policy and give it a unique name. Click **Continue**.

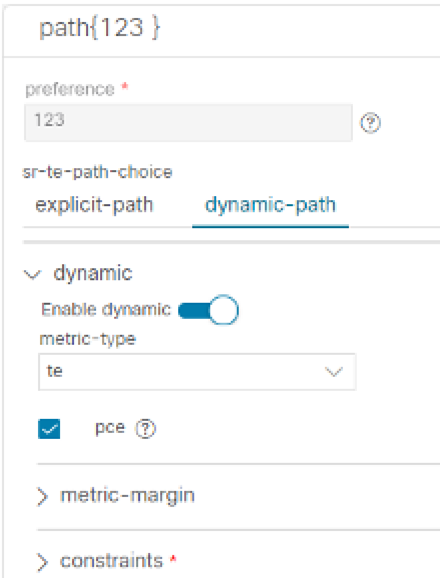
Step 3 Define the endpoints:

a) Under head-end, click **+** and select the headend device from the dropdown list and click **Continue**. Click **X** to close the Headend pane.
b) Enter the IP address of the tail-end device.
c) Enter a color to identify the traffic.

Step 4 Define the parameters on which the path will be computed:

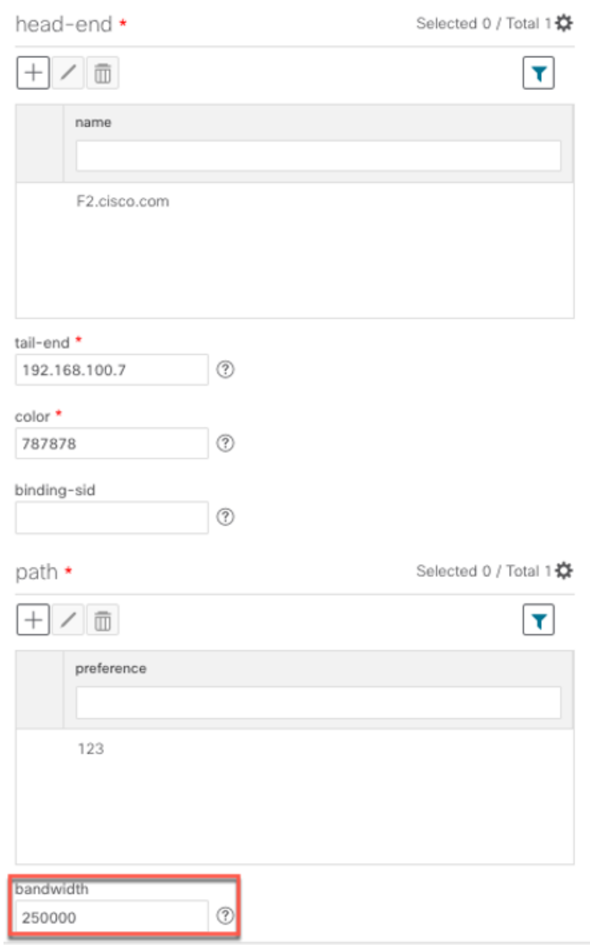
a) Under path, click +.
b) Enter a path preference and click **Continue**.
c) In the dynamic-path tab, select **te** in the metric-type dropdown list as the optimization objective.

d) Select the **pce** check box to have the SR-PCE compute the paths for this policy.

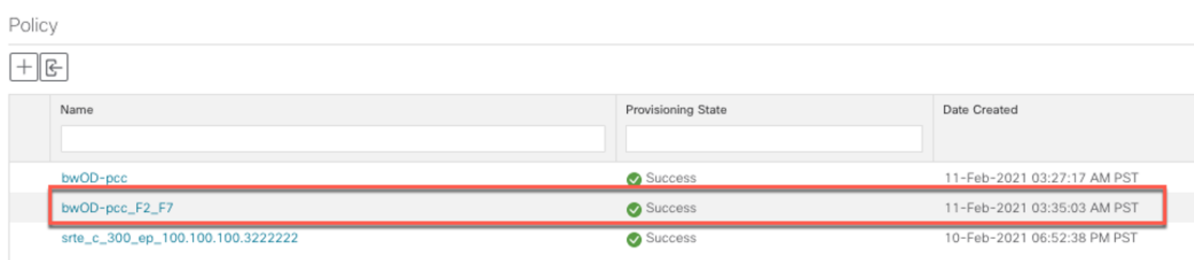


e) Click **X** to close the path pane.

Step 5 In the **Bandwidth** field enter the requested bandwidth in Kbps. In this case, we are requesting **250** Mbps or 250,000 Kbps.



Step 6 Click **Commit Changes**. The new policy is created and appears in the list of SR-TE policies. The provisioning state should be **Success**.

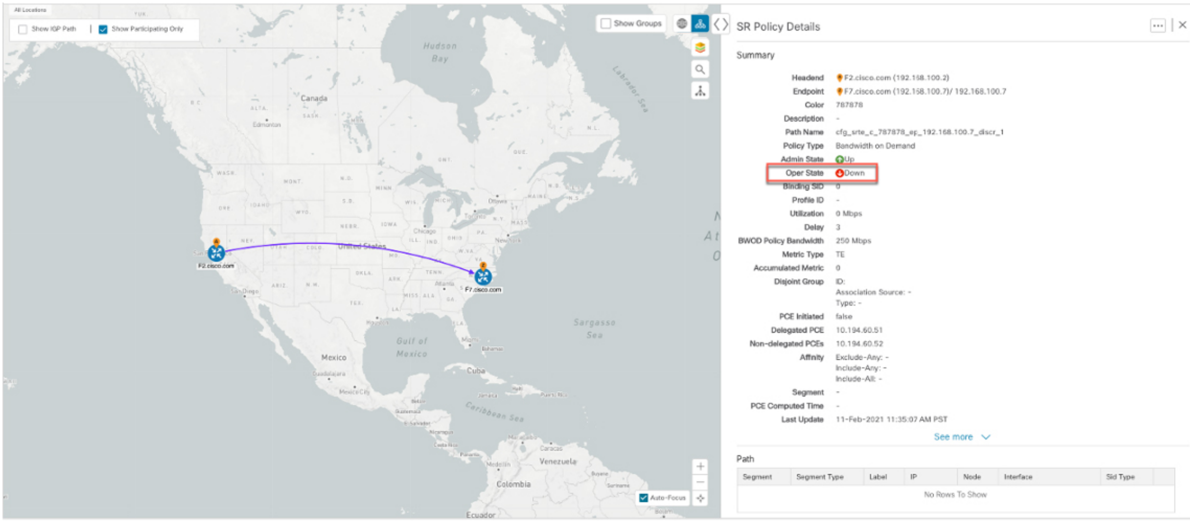


Step 7 Verify the new policy by viewing its details and its representation on the map:

a) Click ⋯ in the Actions column and choose **View**.

b) The map opens with the SR-TE policy details displayed to the right of the map.

> **Note** The operational state of the policy is down because the SR-PCE alone is not able to address bandwidth computations before the BWoD functionality within Crosswork Network Controller is enabled.



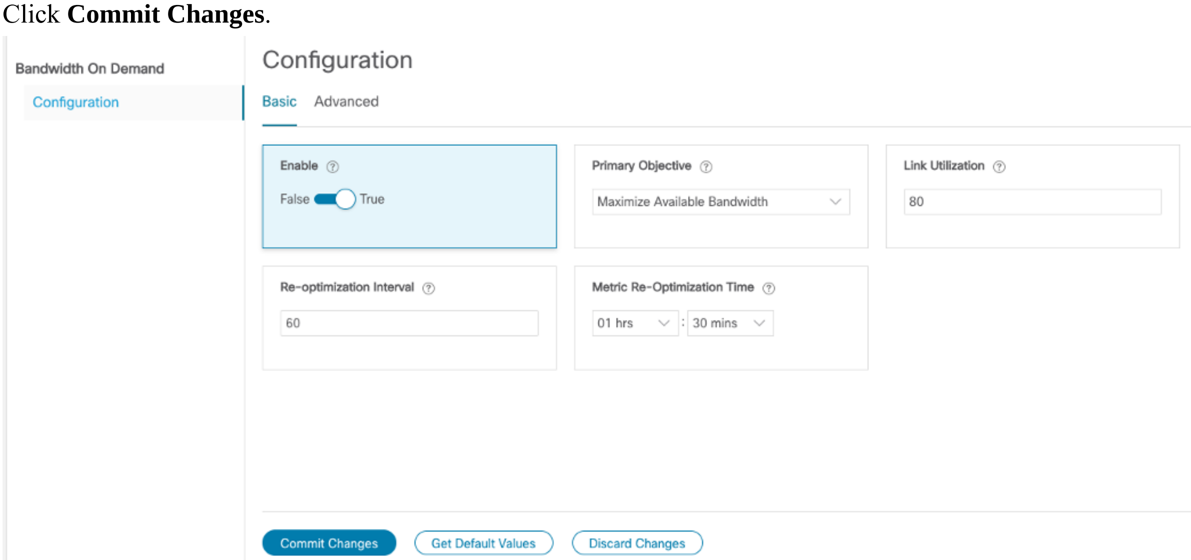
Step 2 Enable and Configure BWoD

Procedure to enable and configure BWoD

Step 1 Go to **Services & Traffic Engineering** > **Bandwidth on Demand**.

Step 2 Toggle the Enable switch to True, and enter 80 to set the utilization threshold percentage. To find descriptions of other options, hover the mouse over.

Step 3 Click **Commit Changes**.



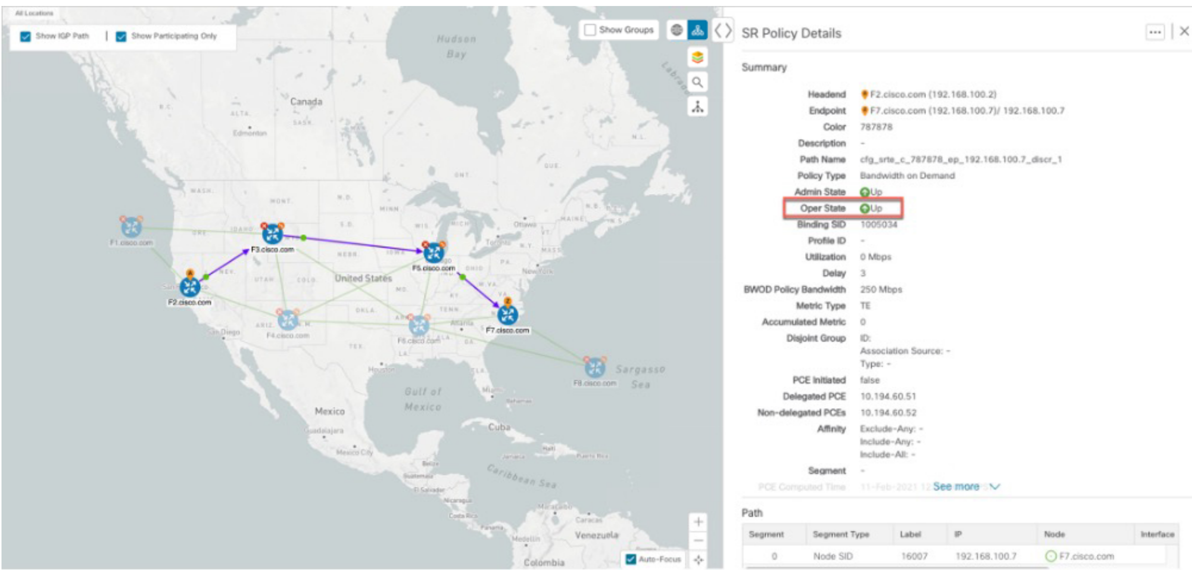
Step 3 Verify that the policy's operational state is now Up and view the path on the map

Procedure to verify that the policy's operational state is now Up and view the path on the map

Step 1 Go to **Services & Traffic Engineering** > **Provisioning**.

Step 2 In the Policy table, locate and select the path computed for the endpoints.

Step 3 The path is shown as an overlay on the map. Check the **Show IGP Path** check box to see the physical path between the endpoints.



Summary and Conclusion

Operators can set and maintain bandwidth requirements based on optimization intent using the BWoD functionality provided in Cisco Crosswork Network Controller. This scenario illustrated how to provision an SR-TE policy with a specific bandwidth request. We saw how to enable BWoD functionality so that traffic is rerouted automatically to maintain bandwidth requirements. This automation alleviates the task of manually tracking and configuring paths to accommodate bandwidth requirements set by SLAs.

CHAPTER 4

Bandwidth and Network Optimization

This section explains the following topics:

Overview

Objective

Tactically optimize the network in real time during periods of congestion, and strategically reserve bandwidth for business-critical links to avoid congestion entirely.

Challenge

Network congestion leads to poor end-customer experiences. Congested links, high latency, and other network impairments lead to a poor perception of the services carried across your network or result in an inability to meet the service level agreements (SLAs) you have with your customers. In the worst-case scenario, your network issues lead to SLA or contract violations and the loss of your brand equity. Network operators need a toolset to help automate bandwidth optimization, steer traffic with little operator intervention, and ensure that critical links always have sufficient bandwidth to avoid congestion.

Solution

Cisco Crosswork Network Controller provides two means for meeting this challenge:

- Local Congestion Mitigation (LCM) is a tactical solution for bandwidth management and congestion mitigation. It is best applied when you are attempting to solve congestion issues directly, on the devices themselves, without a full-scale traffic matrix or advanced planning.
- Circuit-Style Segment Routing (CS-SR) is a strategic traffic engineering solution that permits you to reserve bandwidth in advance for critical links, avoiding congestion issues entirely for these high-priority links.

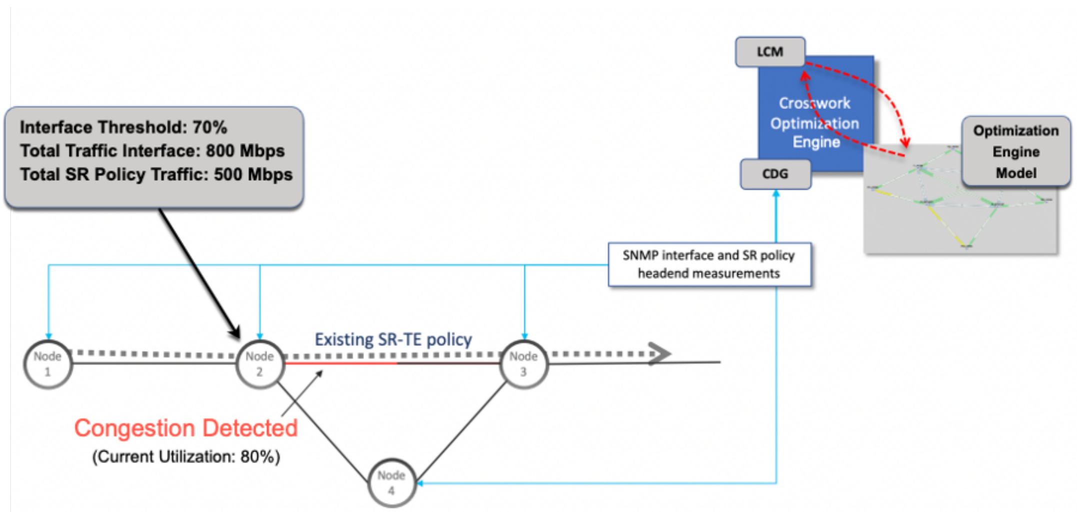
Local Congestion Mitigation (LCM)

Instead of optimizing for bandwidth resource in the network by rerouting traffic in the entire network (end-to-end path optimization), LCM checks the capacity locally, in and around the congested area, at an interface level and reroutes traffic between the endpoints of the congested interface (local interface-level optimization). Focusing on an issue locally eliminates the need for simulating edge-to-edge traffic flows in the network through a full traffic matrix, which is both cumbersome to create and is less scalable as node counts continue to increase.

When congestion is detected in the network, LCM provides recommendations to divert the minimum amount of traffic away from the congested interface. LCM performs the collection of SR-TE policy and interface counters through SNMP. It estimates the amount of traffic that may be diverted and, if the user approves, performs the mitigation through the deployment of Tactical Traffic Engineering (TTE) SR-TE policies. Mitigating congestion locally does not require the use of the full Segment Routing Traffic Matrix (SR-TM). TTE SR-TE policies are created at the device on only either side of the congested link, with the shortest paths possible that do not congest interfaces elsewhere.

How Does LCM Work?

1. LCM first analyzes the Optimization Engine Model (a realtime topology and traffic representation of the physical network) on a regular cadence.
2. In this example, after a congestion check interval, LCM detects congestion when Node 2 utilization goes above the 70% utilization threshold.



3. LCM calculates how much traffic is eligible to divert.

   LCM only diverts traffic that is not already routed by an existing SR policy (for example: unlabeled, IGP-routed, or carried via FlexAlgo-0 SIDs). The traffic within an SR policy will not be included in LCM calculation and will continue to travel over the original programmed path.

   Eligible traffic is computed by taking the interface traffic statistics that account for all traffic on the interface and subtracting the sum of traffic statistics for all SR-TE policies that flow over the interface.

   Total interface traffic – SR policy traffic = Eligible traffic that can be optimized

   This process must account for any ECMP splitting of SR policies to ensure the proper accounting of SR policy traffic. In this example, the total traffic on congested Node 2 is 800 Mbps. The total traffic of all SR policies routed over Node 2 is 500 Mbps.

The total traffic that LCM can divert in this example is 300 Mbps: 800 Mbps – 500 Mbps = 300 Mbps

4. LCM calculates the amount of traffic that must be sent over alternate paths by subtracting the threshold equivalent traffic from the total traffic on the interface. In this example, the amount to be diverted is 100 Mbps:

   800 Mbps – 700 Mbps (70% threshold) = 100 Mbps

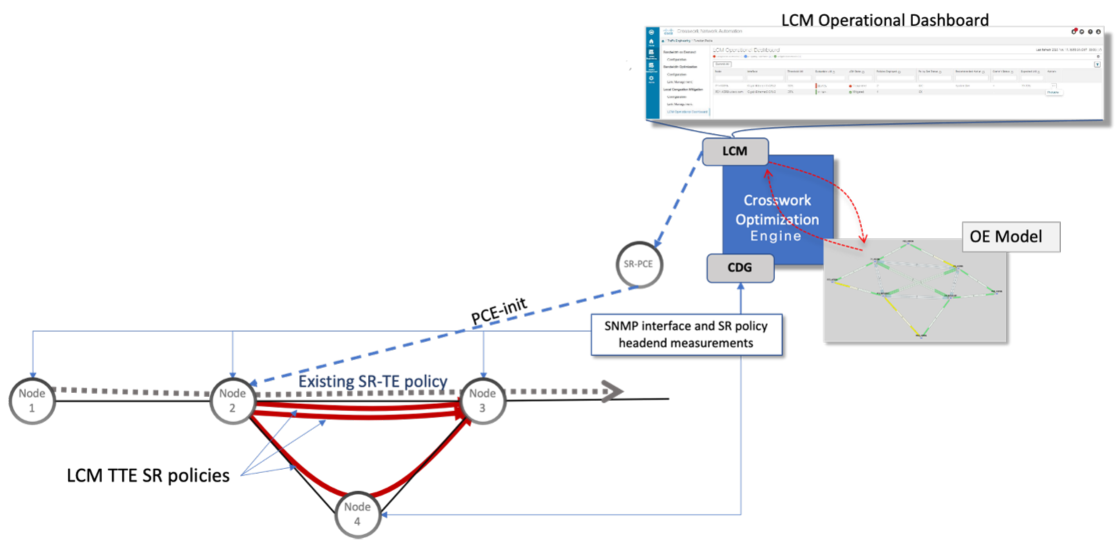
   LCM must route 100 Mbps of 300 Mbps (eligible traffic) to another path.

5. LCM determines how many TTE SR policies are needed and their paths. The ratio of how much LCM eligible traffic can stay on the shortest path to the amount that must be rerouted, will determine the number of TTE SR policies that are needed on the shortest versus alternate paths, respectively.

   In this example, LCM needs to divert one-third of the total eligible traffic (100 Mbps out of 300 Mbps) away from the congested link. Assuming a perfect ECMP, LCM estimates that three tactical SR-TE policies are required to create this traffic split: one tactical SR-TE policy will take the diversion path and two tactical SR-TE policies will take the original path. There is sufficient capacity in the path between Node 2 and Node 4. Therefore, LCM recommends three TTE SR policies (each expected to route approximately 100 Mbps) to be deployed from Node 2 to Node 3 via SR-PCE:

   - 2 TTE SR policies to take a direct path to Node 3 (200 Mbps)
   - 1 TTE SR policy takes a path via Node 4 (100 Mbps)

   These recommendations will be listed in the **LCM Operational Dashboard**.



Assuming you deploy these TTE SR policies, LCM continues to monitor the deployed TTE policies and will recommend modifications or deletions as needed in the LCM **Operational Dashboard**. TTE SR policy removal recommendations will occur if the mitigated interface would not be congested if these policies were removed (minus a hold margin). This helps to avoid unnecessary TTE SR policy churn throughout the LCM operation.

Circuit-Style Policies

Circuit-Style Segment Routing Policies (CS-SR, or CS policies) are connection-oriented transport services that you can use to implement what are sometimes referred to as "circuit emulations" or "private lines".

Combining segment-routing architecture's adjacency SIDs with stateful PCEP path computation, CS policies provide:

- Persistent, dedicated, bi-directional, co-routed transport paths with predictable latencies and other performance metrics in both directions.
- Guaranteed bandwidth commitments for traffic-engineered services using these paths.
- End-to-end path protection to ensure there is no impact on Service Level Agreements.
- Automatic monitoring, maintenance and restoration of path integrity.
- Flexible operations, administration and management of Circuit-Style paths.
- A software-defined replacement for older CEM infrastructure, such as SONET/SDH.

How Do Circuit-Style Policies Work?

Initial configuration of CS policies follows these steps:

1. Crosswork Network Controller and its applications discover and map the network topology.
2. Crosswork users enable CS policy support, specifying the base bandwidth to be allocated to CS policies as a whole, and a threshold percentage of bandwidth usage which, when exceeded on any CS-calculated path, will generate an alarm. So, for example, on a 1 GB link with 20 percent of bandwidth reserved for Circuit Style use, CS policies can use up to 200 Mbps of that link. Note, however, that if the bandwidth minimum threshold is set to the default of 80 percent, alarms will be generated as soon as 160 Mbps of the link is used.
3. Network operators create a CS policy for each set of nodes for which they want to establish a guaranteed path. The policy specifies the two nodes to be linked by the main path, the bandwidth to be reserved, and the backup path. To ensure bandwidth and path failures can be accommodated, the configuration must include bi-directionality, path protection, and performance-management liveness-detection settings.
4. When the operator commits the CS policy, the device-resident Path Computation Client (PCC) will request the Crosswork-resident PCE server to compute candidate Working and Protected paths that conform to the CS policy's bandwidth and other constraints (using a single PCEP request message).
5. The PCC computes both paths and deducts the CS policy-guaranteed bandwidth for them from the total available bandwidth allocated when CS policy support was enabled.
6. Crosswork replies to the PCC with the primary Working and Protected path lists and commits to, or "delegates", them. The topology map displays the current Active and Protected paths between the two nodes, using the colors configured when the CS policy was configured, and labels the two endpoint nodes so they can be identified as CS policy endpoints.

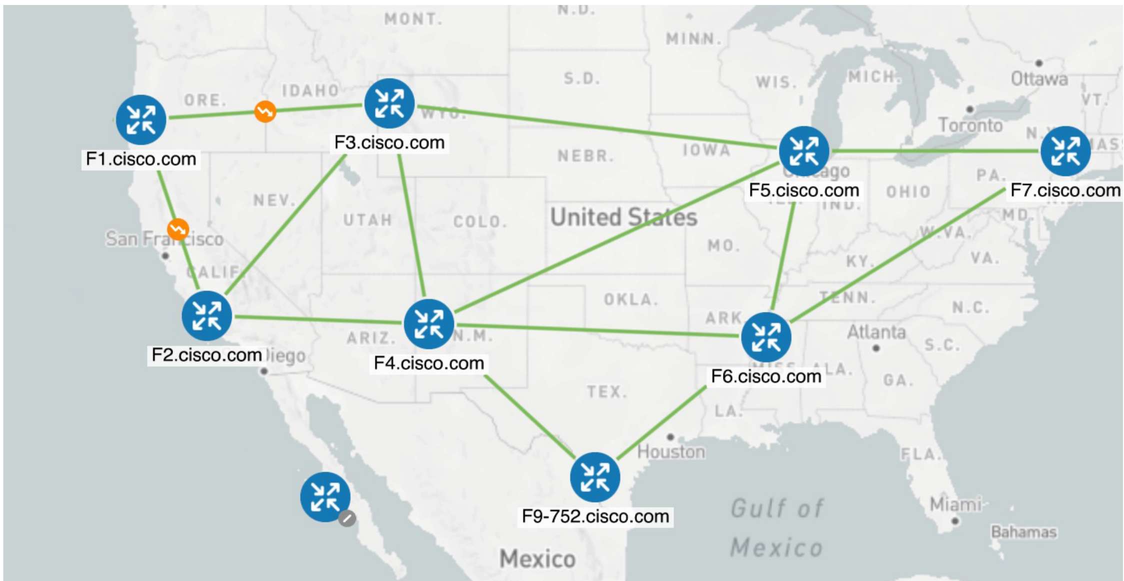
After the initial configuration:

1. Crosswork monitors the delegated path and the active CS policies. It updates the available and reservable bandwidth in the network in near real time.
2. Crosswork generates threshold-crossing alarms when bandwidth usage or additional CS policy requirements exceed the configured reserved bandwidth or bandwidth usage threshold.
3. If delegated paths fail for any reason, Crosswork recomputes paths as needed.

Scenario 6 – Use Local Congestion Mitigation (LCM) to reroute traffic on an over-utilized link

In this scenario, we will enable LCM and observe the congestion mitigation recommendations to deploy Tactical Traffic Engineering Segment Routing (TTE SR) policies when utilization on a device's interfaces exceeds the defined utilization threshold. We will preview the recommended TTE SR policies before committing them to mitigate the congestion.

This example uses the following topology:



We will enable LCM with a configuration that results in the link between **F3.cisco.com** and **F5.cisco.com** becoming over-utilized. We will then review the mitigation solutions Crosswork calculates. In this example, it is left to the operator to choose to apply the solution.

Assumptions and Prerequisites

The following is a non-exhaustive list of high-level requirements for proper LCM operation:

Congestion Evaluation

LCM requires traffic statistics from the following:

- SNMP interface traffic measurements
- SNMP headend SR-TE policy traffic measurements

Congestion Mitigation

The headend device must support PCE-initiated SR-TE policies with autoroute steering.

Devices should be configured with **`force-sr-include`** to enable traffic steering into SR-TE policies with autoroute. For example:

```
segment-routing traffic-eng pcc profile <id> autoroute force-sr-include
```

- The headend device must support Equal Cost Multi-Path (ECMP) across multiple parallel SR-TE policies.

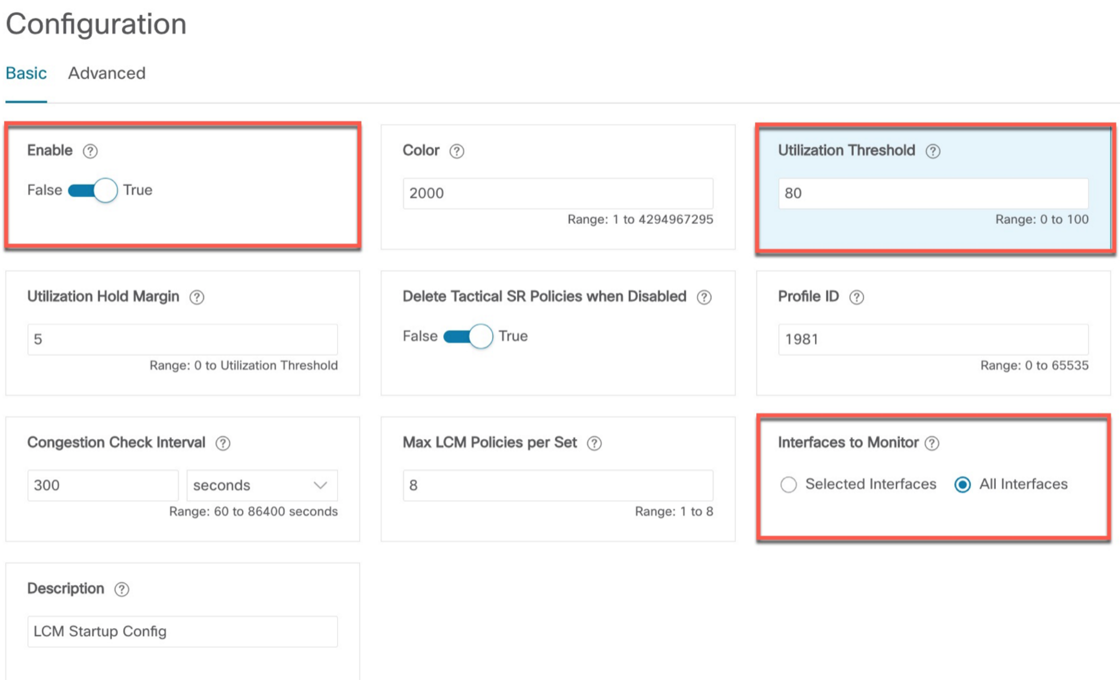
For more information, contact your Cisco Account representative.

Step 1 Enable LCM and configure the global utilization thresholds

To enable LCM and configure the global utilization threshold

Step 1 Go to **Services & Traffic Engineering** > **Local Congestion Mitigation** > **Domain-ID** and click **Configuration**.

Step 2 Toggle the Enable switch to True, and enter the global utilization threshold you want to set. In this case, the threshold is set at 80%, and the **Interfaces to Monitor** > **All Interfaces** option is selected. To see information about other configuration options, hover the mouse over **?** (help icon).



Step 3 Click **Commit Changes**.

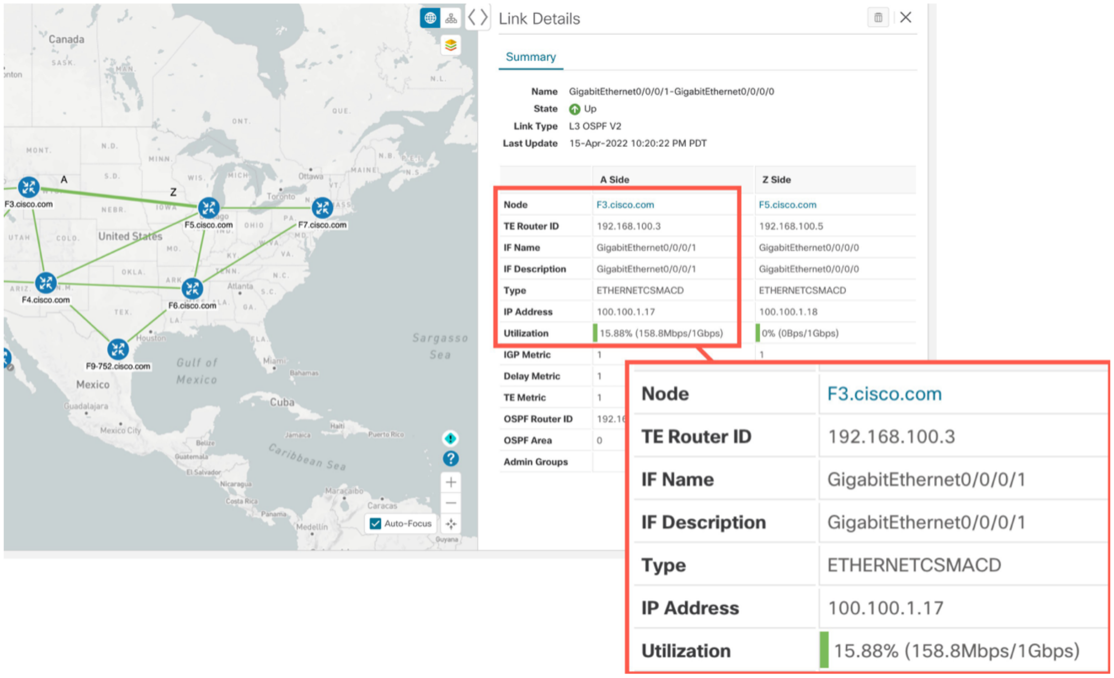
Note After committing the configuration changes, LCM will display *recommendations* on the **LCM Operational Dashboard** if congestion occurs on any monitored interfaces. LCM will *not* commit or deploy new TTE policies automatically. Later, you will be able to preview the recommended TTE policies and decide whether or not to commit and deploy them onto your network.

Step 2 View link congestion on the map

The link between **F3.cisco.com** and **F5.cisco.com** is now congested. Let's see that on the map.

Step 1 Go to **Services & Traffic Engineering** > **Traffic Engineering**.

Step 2 Click on the link to view link details, including utilization information. Note that utilization on the P4-NCS5501 interfaces has surpassed the custom LCM threshold defined at 13%.



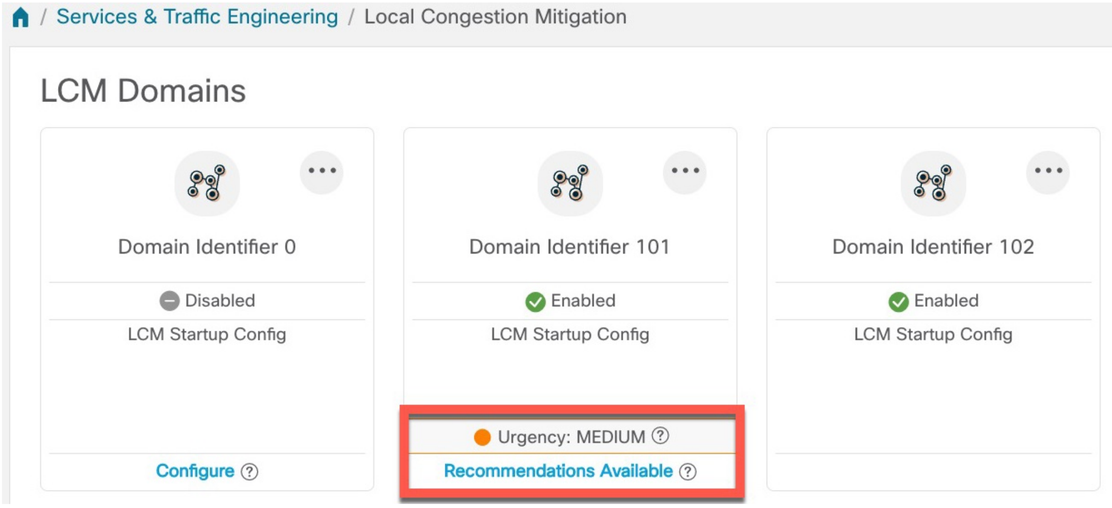
Step 3 Implement LCM recommendations

LCM has detected the congestion and computed tactical policies to mitigate the congestion, which we can preview and then decide whether or not to commit them.

Note that, in this scenario, the congested device is healthy, reachable and in sync with Crosswork. The actions we take and policies we implement will be different if, in addition to congestion, the device is down, unreachable or out of sync.

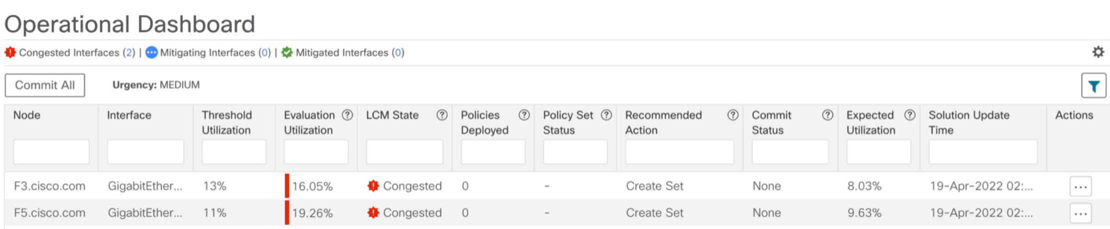
Step 1 Go to **Services & Traffic Engineering** > **Local Congestion Mitigation**.

When congestion is detected, the domain displays the urgency type and recommendations that are available. Click the question mark icons to display more information about the urgency type and when the most recent recommendation was given.

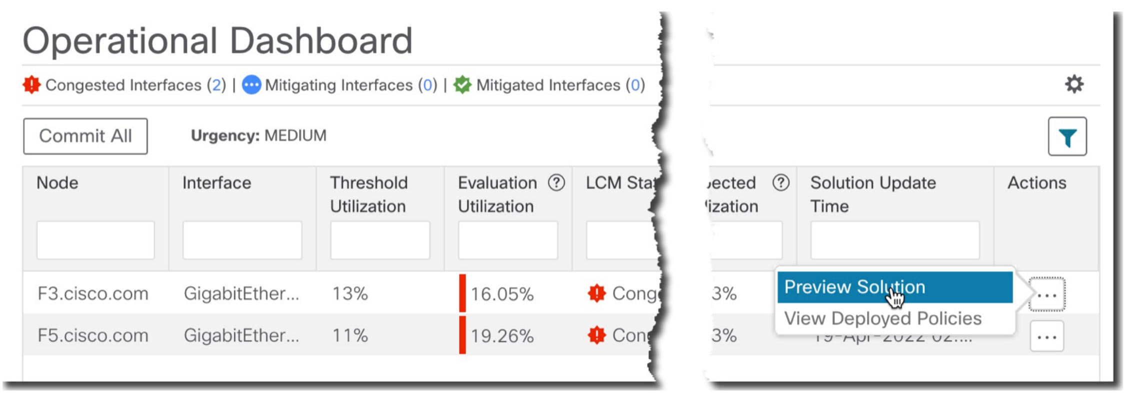


Step 2 Open the Operational Dashboard (**Services & Traffic Engineering** > **Local Congestion Mitigation** > **Domain-ID** >...> **Operational Dashboard**).

The dashboard shows that F3.cisco.com utilization has surpassed 13% and is now at 16.05%. It also shows that F5.cisco.com utilization has also surpassed the 11% threshold and is now 19.26%. In the Recommended Action column, LCM recommends the deployment of TTE policy solution sets (Create Set) to address the congestion on the interface. The Expected Utilization column shows the expected utilization of each of the interfaces after the recommended action is committed.

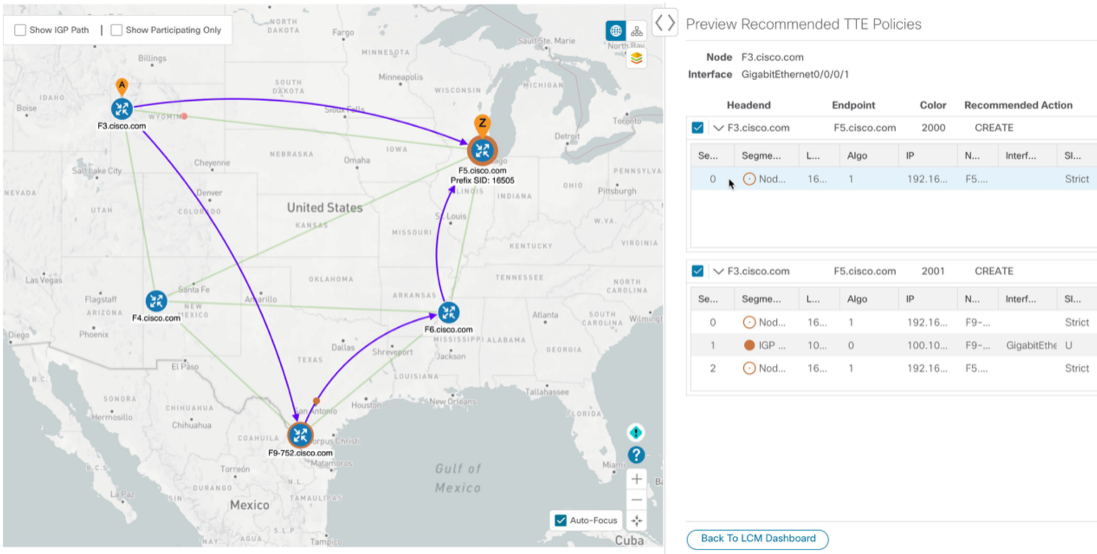


Step 3 Before committing TTE policies, you can preview the deployment of each TTE policy solution set. Click  in the **Actions** column and choose Preview Solution.



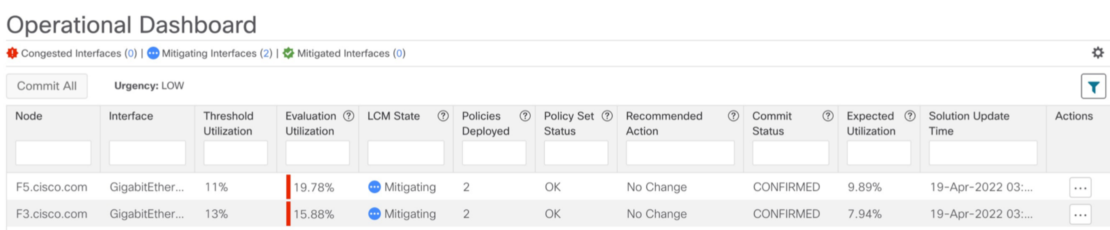
The resulting window displays the node, interface, and the recommended action for each TTE policy. From the Preview window, you can select the individual TTE policies, and view different aspects and information as you would normally do in the topology map. You can expand each policy to view individual segments. After reviewing the potential implications on your network, you can decide whether or not to deploy the bypass policies that LCM recommends.

The following figure shows the recommended TTE policies for node F3.cisco.com and interface GigabitEthernet0/0/0/1. The top path shows the node SID (orange outline), headend and endpoint (A and Z) because the mouse pointer hovers over that segment.



Step 4 After you are done viewing the recommended TTE policies on the map, go back to the **Operational Dashboard** and click **Commit All**. The LCM **State** column changes to **Mitigating**.

All LCM recommendations per domain must be committed in order to mitigate congestion and produce the expected utilization as shown in the **Operational Dashboard**. The mitigating solution is based on *all* LCM recommendations being committed because of dependencies between solution sets.



Step 4 Validate the TTE SR policy deployment

To validate the TTE SR policy deployment, follow the steps given below:

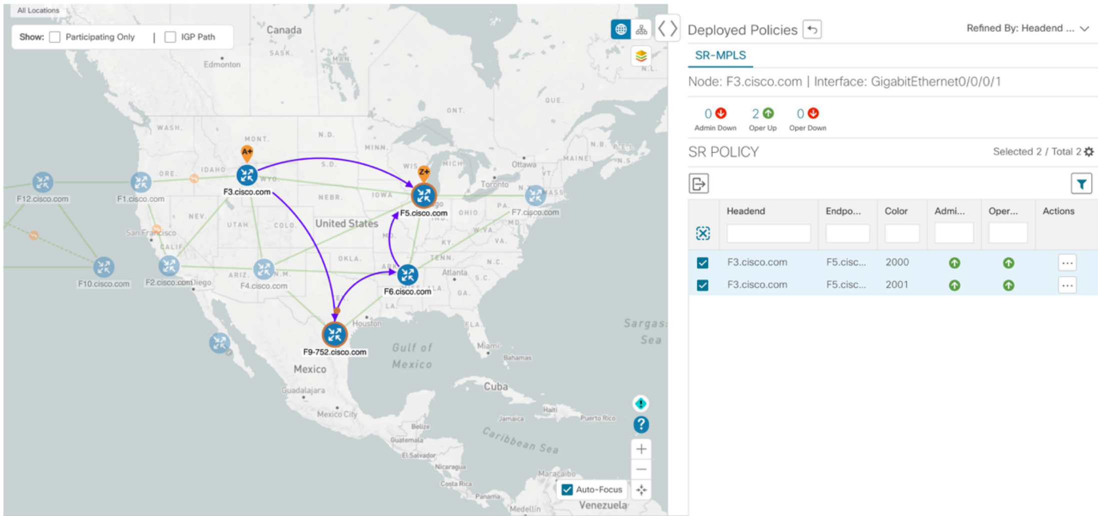
Step 1 Click **bell icon> Events** tab to open the Events window in which you can monitor LCM events. You see events for the LCM recommendations, the commit actions, as well as any exceptions.

Crosswork Optimization Engine will report network events that are detected based on the policies and features you have enabled. For example, if a link drop causes an SR-TE policy to go down, or if LCM detects congestion, an event is displayed in the UI.

Step 2 Return to the **Operational Dashboard** to see that the LCM state changes to **Mitigated** for all TTE policy solution sets.

Note The LCM state change will take up to 2 times longer than the SNMP cadence.

Step 3 Confirm the TTE policy deployment by viewing the topology map. Click in the **Actions** column and choose **View Deployed Policies**. The deployed policies are displayed in focus within the topology map. All other policies are dimmed.



Step 4 View the SR policy details. From the **Actions** column of one of the deployed policies, click and choose **View Details**. Note that the **Policy Type** is **Local Congestion Mitigation**.

Step 5 Remove the TTE SR policies upon LCM recommendation

To remove the TTE SR policies upon LCM recommendation, follow the steps given below:

Step 1 After some time, the deployed TTE SR policies may no longer be needed. This occurs if the utilization will continue to stay under the threshold without the LCM-initiated TTE policies. If this is the case, LCM generates new recommended actions to delete the TTE SR policy sets.

Step 2 Click **Commit All** to remove the previously deployed TTE SR policies.

Step 3 Confirm the removal by viewing the topology map and SR Policy table.

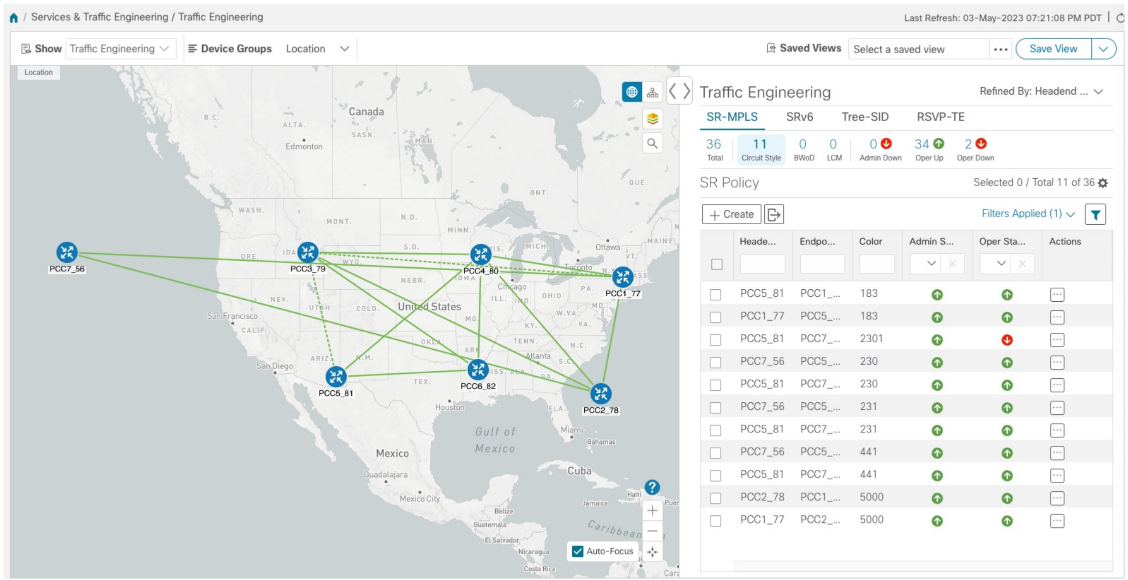
Summary and Conclusion

In this scenario, we observed how to leverage LCM to alleviate traffic congestion in the network. LCM takes the manual tracking and calculation out of your hands but at the same time gives you control as to whether to implement the congestion mitigation recommendations, or not. You can preview the recommendations and see how the potential deployment will take effect in your network before you deploy them. As traffic changes, LCM tracks the deployed TTE SR policies and decides whether or not they are still needed. If not, LCM recommends deleting them.

Scenario 7 – Use Circuit-Style SR Policies to Reserve Bandwidth

In this scenario, we enable CS-SR and set bandwidth-reservation parameters, then configure a CS-SR policy and visualize it on the topology map. We will inspect the policy's details, including its computed Active (working) and Protected (protect) paths.

The examples in this scenario use the following topology:



We will observe what happens when the Active bandwidth-reserved path between the NCS1 and NCS3 nodes fails. We will then re-optimize the failed path.

Assumptions and Prerequisites

The following sections provide a non-exhaustive list of high-level requirements for proper CS-SR operation, including requirements and constraints on the policy attribute values set in each Circuit Style SR-TE policy, and the processing logic followed during path reversions.

In addition to the constraints discussed in the following sections:

- The Crosswork Circuit Style Manager (CSM) feature pack is a feature of the Crosswork Network Automation Essential Suite. All licensed features are available during the 90-day trial period. After the trial period, you must have a license for Crosswork Optimization Engine to continue using CSM.
- Circuit-Style policy configuration was introduced with Crosswork Network Controller (CNC) 5.0. To use it, you must have version 7.9.1 (or later) of the Cisco IOS-XR Path Computation Client (PCC) installed on your devices. If you have been using a previous version of CNC with IOS-XR version 7.7.1 or earlier, please upgrade to version 7.9.1 or later before attempting to configure CS-SR policies.
- When using CSM with Crosswork Network Controller, the UI navigation starts from **Traffic Engineering & Services**. When using CSM with Crosswork Optimization Engine, the navigation starts from **Traffic Engineering**.

CS Policy Attribute Constraints

In this scenario, we will build a CS policy between node NCS1 and node NCS 3. The policy will use the following settings and constraints:

- **PolicyName**: `NCS1-NCS3`
- **Headend Device**: `NCS1`
- **Headend IP Address**: `192.168.20.4`

- **Tailend Device**: `NCS3`
- **Tailend IP Address**: `192.168.20.14`
- **Color-choice**: `1000`
- **Bandwidth**: `10000`
- **path-protection**: Enabled
- **disjoint-path**: Enabled
- **disjoint-path forward-path type**: `Link`
- **disjoint-path forward-path group-id**: `531`
- **disjoint-path reverse-path type**: `Link`
- **disjoint-path reverse-path group-id**: `5311`
- **performance-measurement** : Enabled.
- **performance-measurement profile-type**: `Liveness`
- **performance-measurement liveness-detection**: Enabled
- **performance-measurement profile**: `CS-active`
- **working-path**: Enabled
- **working-path preference**: `100`
- **working-path dynamic-path**: Enabled
- **working-path dynamic-path pce**: Enabled
- **working-path dynamic-path metric type**: `igp`
- **working-path dynamic-path bidirectional-association-choice**: Enabled
- **working-path dynamic-path bidirectional-association-id**: `230`
- **working path dynamic constraints segments**: Enabled
- **working-path constraints segments protection**: `unprotected-only`
- **protect-path**: Enabled
- **protect-path preference**: `100`
- **protect-path dynamic-path**: Enabled
- **protect-path dynamic-path pce**: Enabled
- **protect-path dynamic-path metric type**: `igp`
- **protect-path dynamic-path bidirectional-association-choice**: Enabled
- **protect-path dynamic-path bidirectional-association-id**: `231`
- **protect-path dynamic constraints segments**: Enabled
- **protect-path constraints segments protection**: `unprotected-only`

- **restore-path**: Enabled
- **restore-path preference**: `100`
- **restore-path dynamic-path**: Enabled
- **restore-path dynamic-path pce**: Enabled
- **restore-path dynamic-path metric type**: `igp`
- **restore-path dynamic-path bidirectional-association-choice**: Enabled
- **restore-path dynamic-path bidirectional-association-id**: `232`
- **restore-path dynamic constraints segments**: Enabled
- **restore-path constraints segments protection**: `unprotected-only`

The following table shows all of the options you can choose from when building a policy. It is important to understand that the attributes described in the table act as constraints. Each of them corresponds to elements of the configuration that Cisco Crosswork uses to govern how Circuit-Style path hops are computed. Each value is effectively a path computation or optimization constraint, since they either specify a required property of a path or exclude possible choices for that path.

There are dependencies that must be met as well as combinations that are not allowed. The system will warn you when these sorts of issues arise. We encourage you to experiment to learn how to provision services in your network that match the types of services you want to deliver.

Table 1: Supported Circuit Style SR-TE Policy Attribute Values and Constraints

Attribute	Description
Policy Path Protection	The path protection constraint is required for both sides of a Circuit Style SR-TE policy.

Attribute	Description
Bandwidth Constraint	The bandwidth constraint is required and must be the same on both sides of a Circuit Style SR-TE policy. Bandwidth changes can be made to existing policies, with these effects: • Once you configure the new bandwidth on both sides, Crosswork will evaluate the path. This **will not** result in a recomputed path. • If the new bandwidth is higher, Crosswork checks the existing path to ensure sufficient resources. If all currently delegated paths can accommodate the new bandwidth, Crosswork returns the same path with the new bandwidth value, indicating to the path computation client (PCC) that it was successful. If any of the current paths cannot accommodate the new bandwidth, it returns the old bandwidth value indicating that it was unsuccessful. This evaluation will not be retried unless the bandwidth is changed again. • If the bandwidth is lower, Crosswork returns the same path with the new bandwidth value to indicate to the PCC that it was successful. The user interface shows both the requested and reserved bandwidth under each candidate path when you view the policy details. These values can differ if the requested bandwidth is increased but there is insufficient available CS pool bandwidth along one or more of the paths.
Candidate Paths and Roles	The `Working` path is defined as the highest preference Candidate Path (CP). The `Protect` path is defined as the CP with the second highest preference. The `Restore` path is defined with the lowest preference CP. The headend must have `backup-ineligible` configured. CPs of the same role in each direction must have the same CP preference.
Bi-Directional	All paths must be configured as co-routed. Paths with the same role on both sides must have the same globally unique bi-directional association ID.
Disjointness	Working and Protect paths on the same PCC must be configured with a disjointness constraint using the same disjoint association ID and disjointness type. The disjointness association ID for a Working and Protect path pair in one direction must be unique when compared with the corresponding pair in the opposite direction. Only the `Node` and `Link` disjoint types are supported. The disjoint type used must be the same in both directions of the same policy. The Restore path must not have a disjointness constraint set. Crosswork follows strict fallback behavior for all Working and Protect path disjointness computations. This means that, if node type disjointness is configured but no path is available, Crosswork makes no automatic attempt to compute a less restrictive link type disjoint path.

Attribute	Description
Metric Type	Only the `TE`, `IGP` and `Latency` metric types are supported. The metric type used must match across Working, Protect and Restore paths in both directions.
Segment Constraints	All Working, Protect and Restore paths must have the following segment constraints: • `protection unprotected-only` • `adjacency-sid-only` To ensure persistence through link failures, configure static adjacency SIDs on all interfaces that might be used by Circuit Style policies.
Supported Policy Changes	The following constraints may be changed for an operationally "up" Circuit Style SR-TE policy that has been previously delegated: • Metric type • Disjoint type • MSD • Affinities Once configuration changes are made in a consistent manner across all CPs and both PCCs (for example: the new metric type is the same for all CPs and both sides), Crosswork will initiate a recompute, which can result in new Working, Protect and Restore paths. During any transitory period in which configurations are not in sync between paths on the same PCC or between PCCs, no path updates are sent to the PCCs.
Path Computation	Crosswork computes paths for circuit style policies only after a complete bi-directional, path-protected set of candidate paths has been delegated, including Working and Protect paths on both sides. Crosswork computes the Restore path only after the Working and Protect paths are down. The SR Circuit Style Manager feature pack configuration interface provides a configurable delay timer to control how long after Restore paths are delegated from both sides to wait before the path is computed. This delay allows topology and SR policy state changes to fully propagate to Crosswork, in cases where these changes triggered the Restore path delegation. Path computation is supported for Intra/Inter area/level and Intra/Inter IGP Domain (same AS).
Reversion Behavior	Reversion behavior is controlled by the configuration of the WTR lock timer option under the Protect and Revert paths (it is not relevant for the Working path): • No lock configuration: Revert after a default 5-minute lock • Lock with no duration specified: No reversion • Lock duration <value>: Revert after the specified number of seconds

Unsupported CS Policy Options

The following table lists the CS policy options, attributes and constraints that are not supported in this version of CSM.

Table 2: Unsupported Circuit Style SR-TE Policy Options

Attribute	Description
Unsupported Configurations	The following configurations are not supported: • Metric-bounds • SID-Algo constraints • Partial recovery is not supported with 7.8.x. • State-sync configuration between PCEs of a high-availability pair. These are not required with Circuit Style SR-TE policies. Use of this feature may result in degraded performance. • Multiple Circuit Style SR-TE policies between the same nodes with the same color but different endpoint IPs.
Unsupported Policy Changes	The following configuration changes to a previously delegated and operationally "up" Circuit Style SR-TE policy are not supported: • CP preference • Disjoint Association ID • Bi-directional Association ID To change these configurations for an existing policy, you must first shut down the policy on both sides, make the change (complying with restrictions as detailed above in terms of consistency) and then "no shut" the policy.
Unsupported Path Computation	Automatic re-optimization is not supported for any paths based on changes in topology, LSP state, or any periodic event. Path computation is not supported for Inter-AS.

Path Reversion Logic

Path reversion depends on the initial state of the Working, Protect and Revert paths and the events affecting each path. The scenarios in the following table provide examples of typical reversion behavior.

Table 3: Path Reversion Scenarios

Initial State	Events	Behavior
Working path is down, Protect path is up/active	Working path comes back up	1. Working path recovers to up/standby state. 2. Each PCC moves the Working path to active after the WTR timer expires. 3. Protect path moves to up/standby.

<table>
<tr><th>Initial State</th><th>Events</th><th>Behavior</th></tr>
<tr><td>Working path is down, Protect path is down, Revert path is up/active</td><td>Working path comes back up, then Protect path comes back up</td><td>1. Working path recovers and goes to up/active state
2. Revert path is removed
3. Protect path recovers and goes to up/standby</td></tr>
<tr><td>Working path is down, Protect path is down, Revert path is up/active</td><td>Protect path comes back up, then Working path comes back up</td><td>On side A: The Working path failure is local (the first Adj SID in the SegList is invalid):
1. Protect path recovers and goes to up/active.
2. Recover path is removed.
3. Working path recovers and goes to up/standby.
4. Each PCC moves the Working path to active after the WTR timer expires, Protect path goes to up/standby.

On side Z: Working path failure is remote (first Adj SID in SegList is valid):
1. Protect path recovers but is not brought up, Revert path remains up/active.
2. Working path recovers and goes up/active.
3. Revert path is removed.
4. Protect path goes to up/standby.</td></tr>
</table>

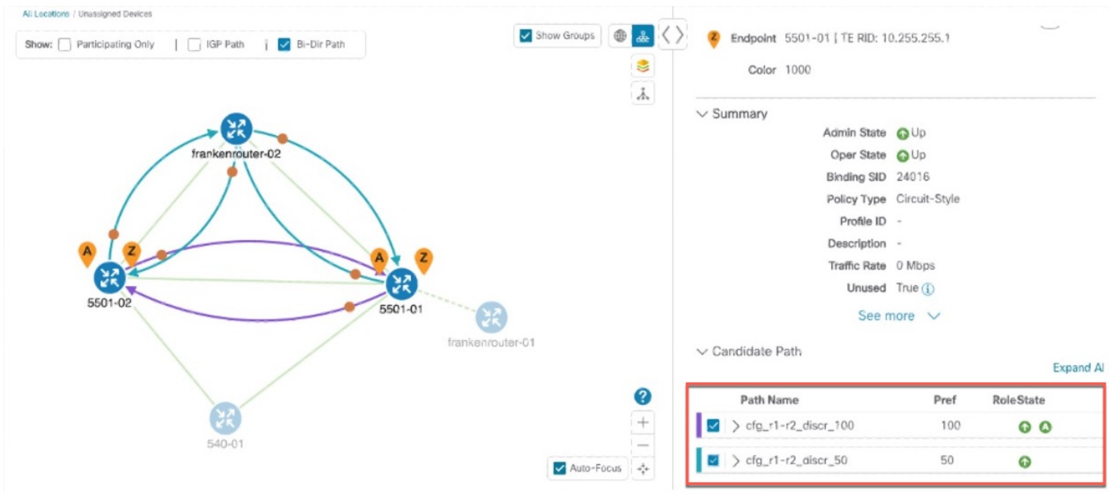
What Happens When Path Failures Occur?

Cisco Crosswork computes paths for CS policies only after a complete bidirectional, path-protected set of candidate paths has been delegated. A path can be considered to have "failed" due to a variety of reasons, including transient changes in workloads caused by congestion elsewhere in the network, or any condition that causes the path not to meet bandwidth expectations. Irrespective of the cause, there are three types of paths used during these kinds of failures. Crosswork activates them as needed, according to their preference settings:

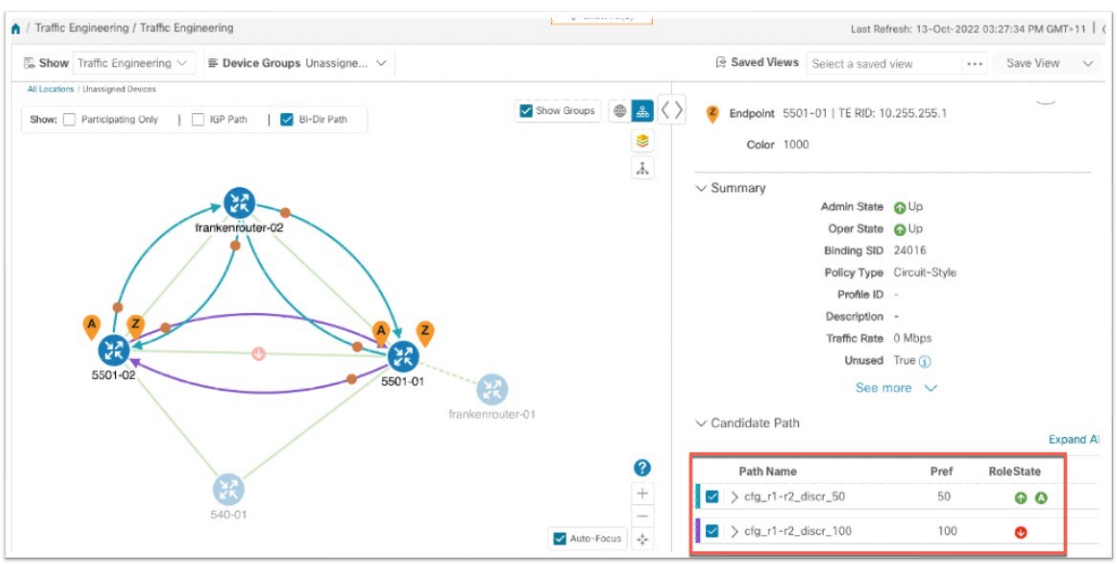
- **Working**—This is the path with the highest preference value. Crosswork always tries to keep the operational (Oper Up) path with the *highest* preference as the *Active* path.
- **Protected**—This is the path with the second highest preference. If the Working path goes down, the Protected path (with the lower preference value) is activated. After the Working path recovers, the Protected path remains active until the default lock duration expires, then the Working path is activated.
- **Restore**—This is the path with the lowest preference path. Crosswork computes the Restore path only when the Working and Protected paths are both down. You can control how long after Restore paths are delegated to wait before the path is computed. This delay allows topology and policy state changes to fully propagate fully propagate through the network and gives Crosswork a chance to gather and analyze telemetry to determine network health.

To address failures effectively and switch from the Working to the Protected path, be sure to configure Performance Measurement (PM) as part of your CS policy. For more information, see Step 4: Configure Circuit Style SR-TE Policies Using Import, on page 114.

The following image shows that the Working and Protected paths of an example CS policy are operational. The *active* path is indicated by the "A" icon shown next to that path in the **State** column in the **Candidate Path** list.

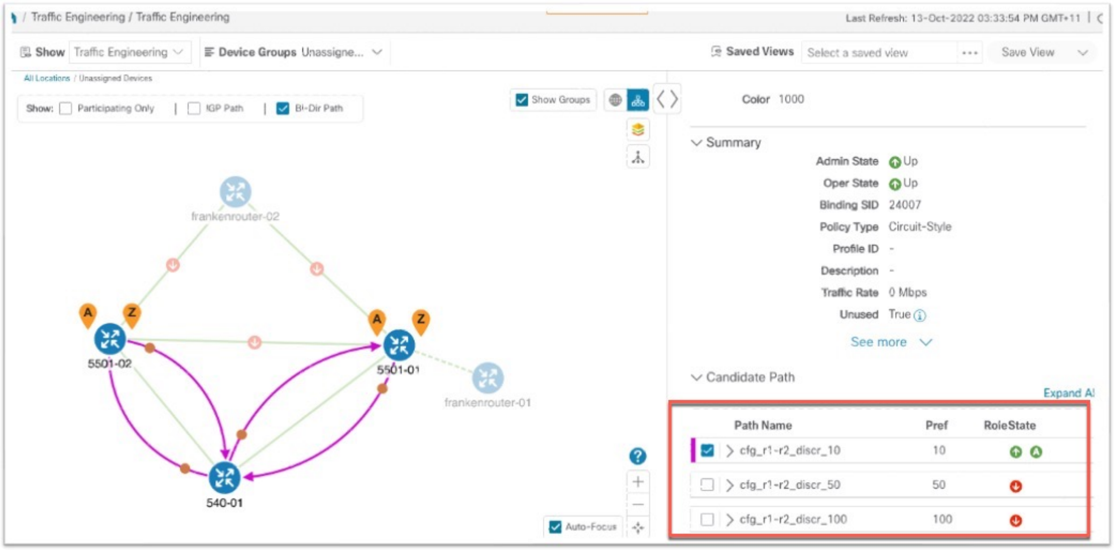


If the Working path performance falls below expectations, the Protected path becomes Active immediately (usually, under 50 milliseconds).



When the Working path comes back up, the Protected path resumes the Protected role again and the Working path (with preference 100) becomes Active again.

If both the Working and Protected paths go down, Crosswork calculates a Restore path and makes it the active path. Note that the Restore path has the lowest preference value of 10. The Restore path only appears in this particular case. If either the Working or Protected paths become operational again, Crosswork will activate them, and the Restore path will disappear from the topology map and from the Candidate Path list.



Workflow

Workflow steps	Detailed procedure links
1. Enable the SR Circuit Style Manager (CSM) feature pack.	Step 1: Enable SR Circuit Style Manager, on page 105.
2-4. Configure Circuit Style SR-TE policies on the devices. **Note** If you haven't enabled the feature pack, the Circuit Style SR-TE policies you configure will appear operationally down.	You can configure Circuit Style SR-TEpolicies using any of the following methods: • On the device, using the CLI. See Step 2: Configure Circuit Style SR-TE Policies Using Device CLI, on page 109 • Using the user interface. See Step 3: Configure Circuit Style SR-TE Policies Using Add, on page 112 • Import a JSON or XML file. See Step 4: Configure Circuit Style SR-TE Policies Using Import, on page 114
5. Verify that the Circuit Style SR-TE policy appears in the SR Policy table and on the topology map.	See Step 5: View Circuit Style SR-TE Policies on the Topology Map, on page 117
6. Verify that the reserved bandwidth pool settings you defined in Step 1 are configured properly.	See Step 6: Verify Circuit Style SR-TE Policy Bandwidth Utilization , on page 124.
7. Trigger path re-computation after path failures.	See Step 7: Trigger Circuit Style SR-TE Path Recomputation, on page 125.

Step 1: Enable SR Circuit Style Manager

In order to manage and visualize Circuit Style SR-TE policies on the topology map, we must first enable SR Circuit Style Manager (CSM) and set bandwidth reservation settings. As soon as you define these settings, CSM computes the best bidirectional failover paths between the two nodes, while observing the requested CSM bandwidth and threshold settings, and the constraints defined in the Circuit Style SR-TE policy. The following steps show how to do this.

CSM tries to ensure that the total reserved bandwidth on all interfaces remains at or below the network-wide resource pool. When the total usage on all interfaces exceeds the threshold value you set, CSM generates a threshold-crossing alarm.

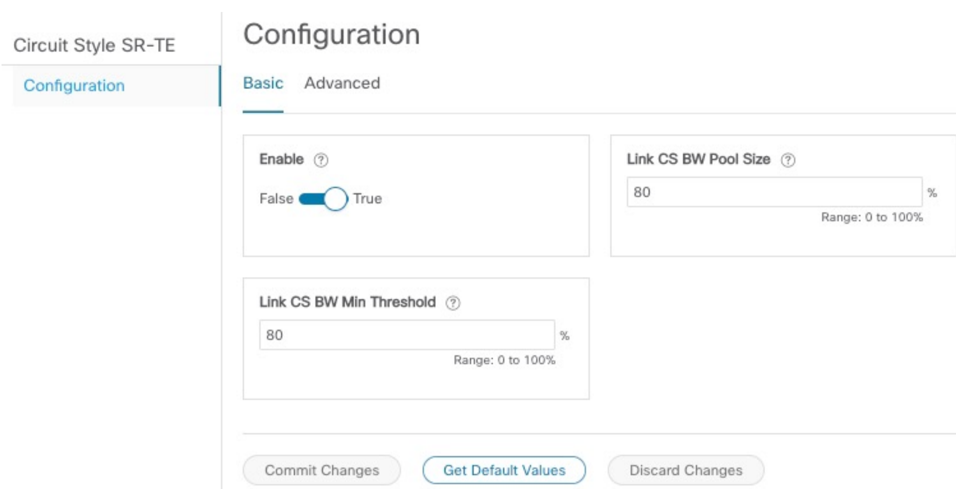
To help you estimate Circuit Style SR-TE bandwidth pool and threshold settings that are reasonable for your organization's implementation, this topic provides two examples showing how CSM handles policies that exceed either the bandwidth pool size or both the pool size and alarm threshold. For the purposes of this scenario, you can enter either one of these examples, or choose settings less likely to be exceeded in a practical implementation.

After enabling CSM, you will need to create Circuit Style SR-TE policy configurations. You can use any of the following methods to create Circuit Style SR-TE policies. In this scenario, we will create the same policy each time, but we will go through each method in order, so that you can decide which methods will best meet your needs:

- Step 2: Configure Circuit Style SR-TE Policies Using Device CLI, on page 109
- Step 3: Configure Circuit Style SR-TE Policies Using Add, on page 112
- Step 4: Configure Circuit Style SR-TE Policies Using Import, on page 114

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

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



Step 3 Enter the required bandwidth pool size and threshold information. The following list describes additional field information. See also the examples below, and choose one of them to enter.

Field	Description
Basic	
Link CS BW Pool Size	The percentage of each link's bandwidth reservable for Circuit Style SR-TE policies.
Link CS BW Min Threshold	The Link CS BW Pool utilization percentage beyond which Crosswork will generate a threshold-crossing event notification.
Advanced	
Validation Interval	This is the interval that CSM will wait before the bandwidth that is reserved for an un-delegated policy is returned to the Circuit Style SR-TE policy bandwidth Pool.
Timeout	The duration CSM will wait for the delegation request, before generating a threshold-crossing alarm.
Restore Delegation Delay	The duration CSM will pause before processing a restore path delegation.

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

Example

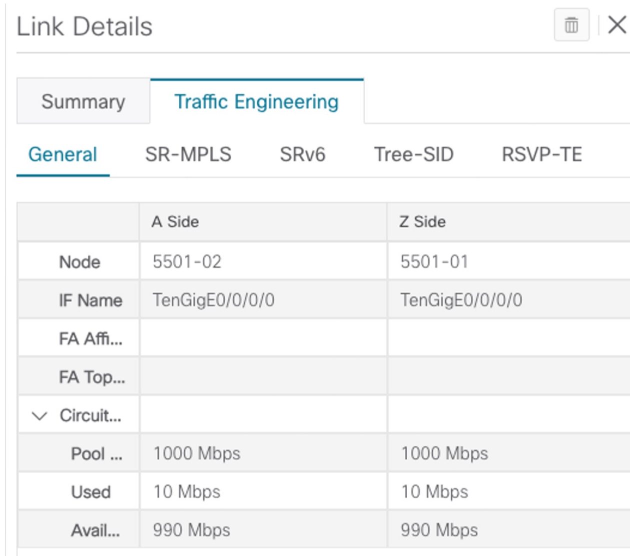
Example: Bandwidth Utilization Surpasses Defined Threshold

In this example, we assume the reserved bandwidth settings are as follows:

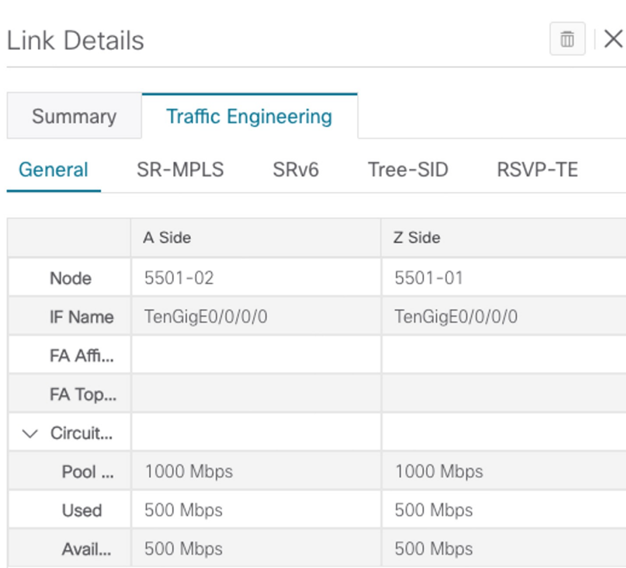
- Bandwidth Pool Size: 10%
- Bandwidth Pool Threshold: 1%

Our two nodes have 10 Gbps Ethernet interfaces, so the bandwidth pool size with these settings is 1Gbs and the alarm threshold is 100 Mbps.

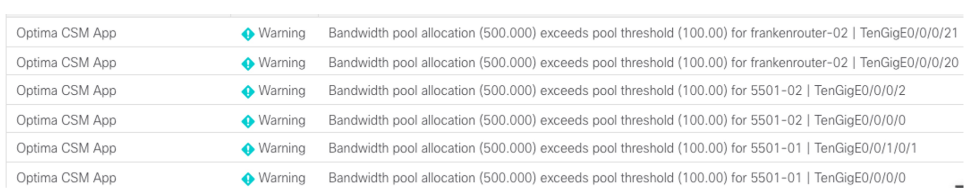
1. We create a CS policy connecting node 5501-02 to node 5501-01 (r02 - r01), with a bandwidth of 100 Mbps.



2. Later, the requested bandwidth for the policy increases to 500 Mbps. The updated CS policy is created and operational (Oper State Up).



3. Since the bandwidth utilization of 500 Mbps with the updated policy is greater than the configured bandwidth threshold (100 Mbps), Crosswork triggers alerts.



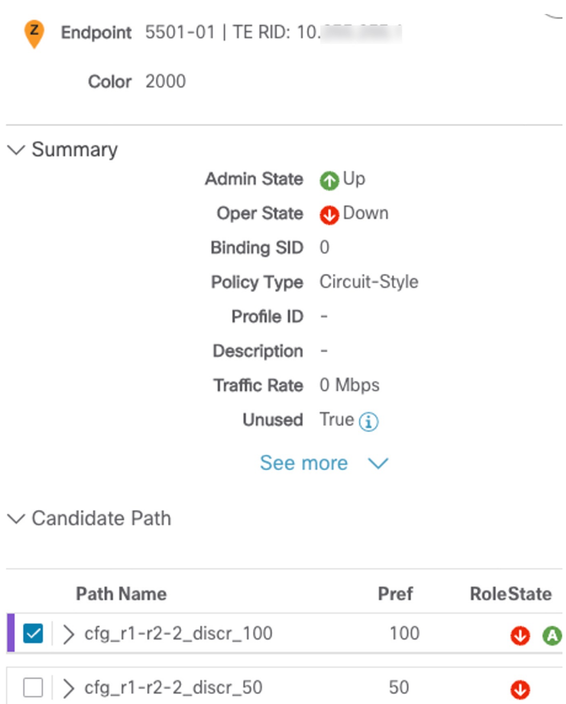
Example: Bandwidth Pool Size and Usage Exceeded

In this example, we assume the reserved bandwidth settings are as follows:

- Bandwidth Pool Size: 10%
- Bandwidth Pool Threshold: 10%

The bandwidth pool size for the 10 Gbps Ethernet interfaces is 1Gbs and the alarm threshold is 100 Mbps.

1. An existing CS-SR policy from node 5501-02 to node 5501-01 (*r02- r01*) uses a bandwidth of 500 Mbps.
2. Later, a new policy requiring a bandwidth of 750 Mbps with a path from node 5501-02 to node 5501-01 to 5501-2 (*r02- r01- r2*) is requested. Since the existing policy and this new policy together exceed the bandwidth pool size and alarm threshold of 1 Gbps (750 Mbps + 500 Mbps = 1250 Mbps), the following behaviors occur:
   - The new CS-SR policy *r02- r01- r2* is created but not operational (Oper State Down).

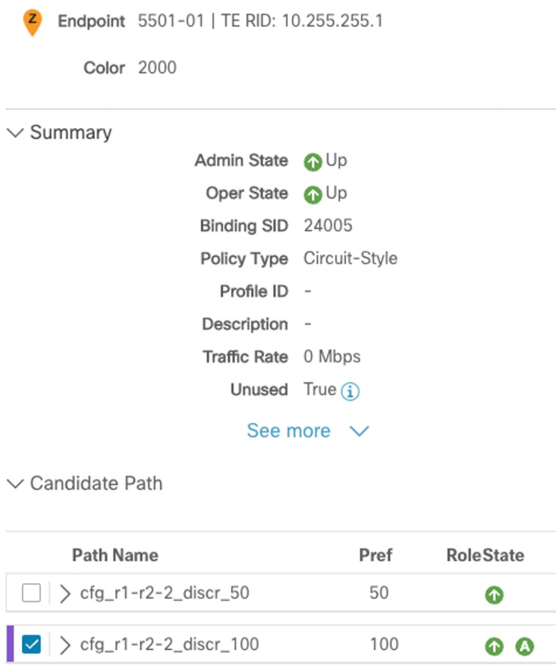


   - Alerts are triggered.

Source	Severity	Description
Optima CSM App	Warning	Unable to compute path for 10. .1 <-> 10.255.255.2 | color 2000 due to CsmUpdateStatus.NO_PATH
SR Policy [10.255.255.2#10.255....	Warning	Policy 'srte_c_2000_ep_10. .1' has operational status as DOWN.
SR Policy [10.255.255.1#10.255....	Warning	Policy 'srte_c_2000_ep_10. .2' has operational status as DOWN.

3. Later, the CS-SR policy *r02- r01- r2* is updated and only requires 10 Mbps. The following behaviors occur:

- Since the total bandwidth required for the two polices (10 Mbps + 500 Mbps = 510 Mbps) now requires less than the bandwidth pool size (1Gbps), CS-SR policy r02- r01- r2 becomes operational (Oper State Up).



- Since the bandwidth utilization (10 Mbps) with the updated policy is below the configured bandwidth threshold (1 Gbps), alerts are cleared.

Step 2: Configure Circuit Style SR-TE Policies Using Device CLI

Prior to Cisco Crosswork, most network engineers created Circuit Style SR-TE policies directly on the devices themselves, using the appropriate network operating system CLI commands. This step of the scenario covers direct CLI policy configuration for a Cisco device. We present it only because this is a legitimate way to create these policies, and so you can see how the configuration implemented using this method matches the configuration for the other, Crosswork-native methods presented in this scenario.

Crosswork Network Controller's topology discovery will automatically recognize CS policy configurations implemented directly on devices, and will help you visualize them on the topology map. However, this method has some important drawbacks. To start with, you will need to be familiar with the CLI commands required to configure Circuit Style SR-TE policies properly. More importantly, Crosswork can *discover* Circuit Style SR-TE policies configured directly on a device, but cannot change or delete them. We encourage you to use instead the **Add** or **Import** methods, which allow you to manage and change your configuration using Crosswork. For help using these methods, skip this step and go on to Step 3: Configure Circuit Style SR-TE

Policies Using Add, on page 112 or to Step 4: Configure Circuit Style SR-TE Policies Using Import, on page 114.

A Circuit Style SR-TE policy configuration must include the destination endpoint, the amount of requested bandwidth, and the bidirectional attribute. See Assumptions and Prerequisites, on page 96 for additional requirements and notable constraints.

When configuring Circuit Style SR-TE policies directly on Cisco devices, make sure the configuration includes a Performance Measurement (PM) Liveness profile. A PM Liveness profile enables proper detection of candidate path liveness and effective path protection. Path Computation Clients (PCCs) do not validate past the first SID, so without PM Liveness, the path protection will not occur if the failure in the Circuit Style SR-TE policy candidate path occurs after the first hop in the segment list. Crosswork supports software-based and hardware-offload PM Liveness configuration methods. For more background on PM Liveness profiles and methods, see Configuring SR Policy Liveness Monitoring.

Step 1 Use your preferred method to access the head-end device console and log in.

Step 2 If applicable, enable the hardware module on the device for PM configuration.

Example:

```
hw-module profile offload 4

reload location all
```

Step 3 Configure the Performance Measurement (PM) Liveness profile on the device. The following example uses a hardware-offload configuration.

Example:

```
performance-measurement
 liveness-profile sr-policy name CS-active-path
  probe
   tx-interval 3300
!
npu-offload enable   !! Required for hardware Offload only
  !
 !
 liveness-profile sr-policy name CS-protected-path
  probe
   tx-interval 3300
  !

npu-offload enable   !! Required for hardware Offload only
  !
 !
!
```

Step 4 Configure the Circuit Style SR-TE policy. All configuration entries shown are required in order for the Crosswork CSM feature pack to manage the policy. Entry values that you must change appropriately for your network (or for your PM Liveness profile) are shown in *italics*. See Assumptions and Prerequisites, on page 96 for additional requirements and notable constraints.

Example:

```
segment-routing
 traffic-eng
  policy NCS1-NCS3

   performance-measurement
```

```
    liveness-detection
     liveness-profile backup name CS-protected      !! Name must match liveness profile defined for
 Protect path
     liveness-profile name CS-active               !! Name must match liveness profile defined for
Active path
    !
   !
   bandwidth 10000
   color 1000 end-point ipv4 192.168.20.4
   path-protection
   ! Path protection is required on both ends of the candidate-paths
  ! Defining the Working path. Must have the highest CP preference
    preference 100
     dynamic
      pcep
      !
      metric
       type igp
      !
     !
     constraints
      segments
       protection unprotected-only
       adjacency-sid-only
      !
      disjoint-path group-id 3 type node
     !
     bidirectional
      co-routed
      association-id 230
 !
    ! Defining the Protect path. Must have second highest CP preference.
    preference 50
     dynamic
      pcep
      !
      metric
       type igp
      !
     !
     constraints
      segments
       protection unprotected-only
       adjacency-sid-only
      !
      disjoint-path group-id 3 type node
     !
     bidirectional
      co-routed
      association-id 231
     ! Defining the restore path. It must have both the lowest CP preference and backup-ineligible
setting
    preference 10
     dynamic
      pcep
      !
      metric
       type igp
      !
     !
     backup-ineligible
     !

     constraints
```

```
      segments
       protection unprotected-only
       adjacency-sid-only
      !
     !
     bidirectional
      co-routed
      association-id 232
     !
    !
    !

     !
    !

   !
  !

 !
!
```

Step 3: Configure Circuit Style SR-TE Policies Using Add

You can create a Circuit Style SR-TE policy between any two nodes using the Crosswork Network Controller **Add** function. This method is the simplest for users who want to be able to use Crosswork to edit or delete the Circuit Style SR-TE policies they create.

This method doesn't completely eliminate the need to be familiar with the CLI command attributes needed to configure Circuit Style SR-TE policies properly. If you prefer a faster method that can also help you to standardize these policies across your network, skip this step and use the method in Step 4: Configure Circuit Style SR-TE Policies Using Import, on page 114.

Step 1 From the main menu, choose **Services & Traffic Engineering** > **Provisioning**.

Step 2 In the **Services/Policies** column on the left, select **SR-TE** > **Circuit-Style Policy**. Crosswork displays the **Create SR-TE > Circuit Style Policy** window.

Step 3 Click [+]. Crosswork displays the **Create SR-TE > Circuit Style Policy** window.

Step 4 In this scenario, we will use the name **`NCS1-NCS3`**. Enter that name in the **Name** field, then click **Continue**.

Step 5 Begin by making the following entries in the respective fields on the **Create SR-TE > Circuit Style Policy**:

- **Name**: **`NCS1-NCS3`**
- **Color-choice**: `1000`
- **Bandwidth**: `10000`
- **path-protection**: Check the checkbox.

Note The color-choice and bandwidth values shown here are examples only. If you decide to follow this example in your network, be sure to use a color-choice value that is not already in use, and a bandwidth value that is available within the percentage you are dedicating to CS policies.

Step 6 Continue the scenario by entering the Circuit Style SR-TE policy constraints and specifications shown in the table below. The user interface for the **Add** function groups policy fields into related categories. Click the **>** icon to expand a category and display its dependent fields.

You will need to change the device names and IP addresses you enter to match actual devices on your network.

Table 4: Example: Circuit Style SR-TE Policy Using Add

Expand this:	To specify this:
head-end	• **Device**: Enter `NCS1`. • **Ip-address**: Enter `192.168.20.4`.
tail-end	• **Device**: Enter `NCS3`. • **Ip-address**: Enter `192.168.20.14`.
disjoint-path	Click **Enable disjoint-path**.
disjoint-path > forward-path	• **Type**: Select `Link`. • **group-id**: Enter `531`.
disjoint-path > reverse-path	• **Type**: Select `Link`. • **group-id**: Enter `5311`.
performance-measurement	Click **Enable performance-measurement**.
performance-measurement > Profile-type	Click **liveness**.
performance-measurement > Profile-type > liveness-detection	Click **Enable liveness-detection**. Then: • **Profile**: Enter `CS-active`. • **Backup**: Enter `CS-protected`.
working-path	Click **Enable working-path**. Then select **dynamic-path**.
working path > dynamic	Click **Enable dynamic-path**. Then: • **pce**: Check the checkbox. • **Metric-type**: Select `igp` • **Bidirectional-association-choice**: Select **bidirectional-association-id** and enter `230`in the field.
working path > dynamic > constraints > segments	Click **Enable segments**. Then in the **Protection** field, select `unprotected-only`.
protect-path	Click **Enable protect-path**. Then select **dynamic-path**.

Expand this:	To specify this:
protect-path > dynamic	Click **Enable dynamic**. Then: • **pce**: Check the checkbox. • **Metric-type**: Select `igp` • **Bidirectional-association-choice**: Select **bidirectional-association-id** and enter `231`in the field.
protect-path > dynamic > constraints > segments	Click **Enable segments**. Then in the **Protection**, field, select `unprotected-only`.
restore-path	Click **Enable restore-path**. Then select **dynamic-path**.
restore-path > dynamic	Click **Enable dynamic-path**. Then: • **pce**: Check the checkbox. • **Metric-type**: Select `igp` • **Bidirectional-association-choice**: Select **bidirectional-association-id** and enter `232`in the field.
restore-path > dynamic > constraints > segments	Click **Enable segments**. Then in the **Protection** field, select `unprotected-only`.

Step 7 When you are finished, click **Dry Run** to validate your changes and save them.Crosswork will display your changes in a popup window.

If you want to configure a service that has requirements that do not match those we describe in this example, contact Cisco Customer Experience.

Step 8 When you are ready to activate the policy, click **Commit Changes**.

Step 4: Configure Circuit Style SR-TE Policies Using Import

If your organization has already implemented Circuit Style SR-TE policies and wants to roll them out on more network devices, the easiest way to do so is using Crosswork Network Controller's **Import** function. You can use **Import** to download a policy template file from Crosswork. The template file will be in JSON or XML format. You can save the template under a new name, insert the policy values of your choice, and then import the modified file.

As well as being fast, using the **Import** function is a good way to standardize Circuit Style SR-TE policies across your network. You can set the same template files to establish CS-SR policies between multiple pairs of devices, varying only the endpoint names and addresses, and any other values as appropriate for each circuit.

Step 1 From the main menu, choose **Services & Traffic Engineering** > **Provisioning**.

Step 2 In the **Services/Policies** column on the left, select **SR-TE** > **Circuit-Style Policy**.

Step 3 Click . Then click the **Download sample JSON and XML files** link. The downloaded ZIP file contains templates for all the Crosswork service types, including Circuit-Style, in JSON and XML formats.

Step 4 Unzip `samplePayload.zip` and locate the `CS-Policy.json` and `CS-Policy.xml` template files.

Step 5 Using the JSON or XML file editor of your choice, open the `CS-Policy` template file and save it under the name **`cs1-cs4`**.

Step 6 If you are using the JSON template file, edit it so that it looks like the example below. If you are using the XML template, go on to the next step.

Example:

CS-SR Policy in JSON

```
{
  "name": "NCS1-NCS3",
  "head-end": {
    "device": "NCS1",
    "ip-address": "192.168.20.4"
  },
  "tail-end": {
    "device": "NCS3",
    "ip-address": "192.168.20.14"
},
  "color": 1000,
  "bandwidth": 10000,
  "disjoint-path": {
    "forward-path": {
      "type": "Link",
      "group-id": 531
    },
    "reverse-path": {
      "type": "Link",
      "group-id": 5311
    }
  },
  "performance-measurement": {
    "profile-type": "liveness",{
      "profile": "CS-active",
      "backup": "CS-protected"
    },
  },
  "path-protection": {},
  "working-path": {
    "dynamic": {
      "constraints": {
        "segments": {
          "protection": "unprotected-only"
        }
      },
      "pce": {},
      "metric-type": "igp",
      "bidirectional-association-id": 230
    }
  },
  "protect-path": {
    "dynamic": {
      "constraints": {
        "segments": {
          "protection": "unprotected-only"
        }
      },
      "pce": {},
      "metric-type": "igp",
```

```
      "bidirectional-association-id": 231
    },
    "revertive": true
  },
  "restore-path": {
    "dynamic": {
      "constraints": {
        "segments": {
          "protection": "unprotected-only"
        }
      },
      "pce": {},
      "metric-type": "igp",
      "bidirectional-association-id": 232
    }
  }
}
```

Step 7 If you are using the XML template file, edit it so that it looks like the example below.

Example:

CS-SR Policy in XML

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cs-sr-te-policy xmlns="http://cisco.com/ns/nso/cfp/cisco-cs-sr-te-cfp">
    <name>NCS1-NCS3</name>
    <head-end>
      <device>cs1</device>
      <ip-address>192.168.20.4</ip-address>
    </head-end>
    <tail-end>
      <device>cs4<device>
      <ip-address>192.168.20.14<ip-address>
    <tail-end>
    <color>1000</color>
    <bandwidth>10000<bandwidth>
    <disjoint-path>
      <forward-path>
        <type>Link</type>
        <group-id>531</group-id>
      </forward-path>
      <reverse-path>
        <type>Link</type>
        <group-id>5311</group-id>
      </reverse-path>
    </disjoint-path>
    <performance-measurement>
      <profile-type>liveness
        <profile>CS-active</profile>
        <backup>CS-protected"</backup>
      </profile-type>
    </performance-measurement>
    <path-protection></path-protection>
    <working-path>
      <dynamic>
        <constraints>
          <segments>{
            <protection>unprotected-only</protection>
          </segments>{
        </constraints>{
          <pce></pce>
          <metric-type>igp</metric-type>
          <bidirectional-association-id>230</bidirectional-association-id>
      </dynamic>
```

```
      </working-path>
      <protect-path>
        <dynamic>
          <constraints>
            <segments>
              <protection>unprotected-only</protection>
            </segments>
          </constraints>
          <pce></pce>
          <metric-type>igp</metric-type>
          <bidirectional-association-id>231</bidirectional-association-id>
        </dynamic>
      </protect-path>
    <restore-path>
      <dynamic>
        <constraints>
          <segments>
            <protection>unprotected-only</protection>
            </segments>
          </constraints>
          <pce></pce>
          <metric-type>igp</metric-type>
          <bidirectional-association-id>232</bidirectional-association-id>
        </dynamic>
      </restore-path>
    </cs-sr-te-policy>
  </config>
```

Step 8 When you have finished editing the file and saved your changes, navigate to **Services & Traffic Engineering** > **Provisioning** > **SR-TE** > **Circuit-Style Policy** again.

Step 9 Click again. In the **File Name** field, enter the path to and file name of your modified template file, or click **Browse** to locate and select it. Then click **Import**.

Step 5: View Circuit Style SR-TE Policies on the Topology Map

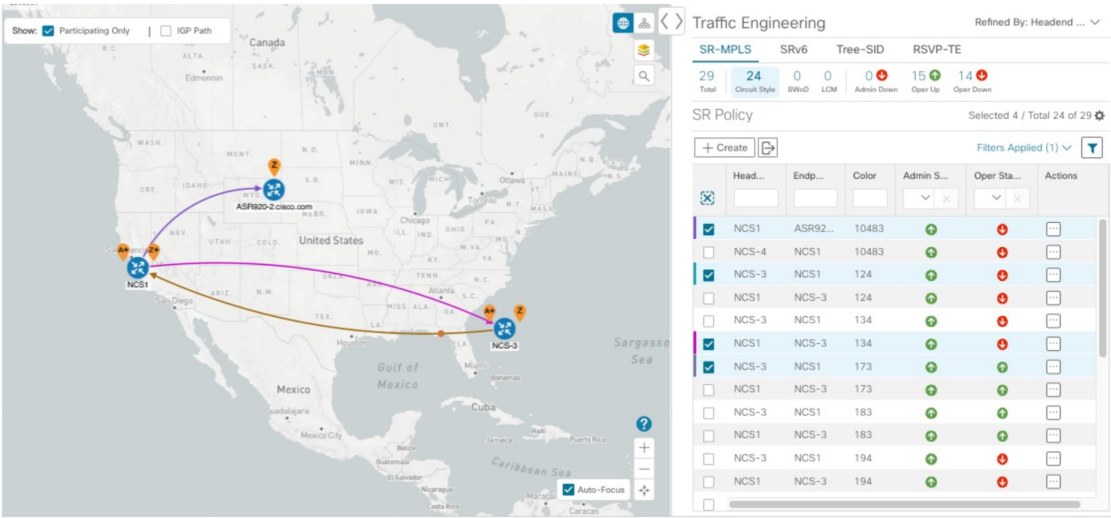
Next, we'll use Crosswork to visualize the NCS1-NCS3 Circuit Style SR-TE policy and isolate it on the map. To make this step more realistic and demonstrate how to focus on just one policy, the scenario assumes that we have multiple active Circuit Style SR-TE policies, not just the policy we created. We'll also view the Circuit Style SR-TE policy details, including endpoints, bandwidth constraints, IGP metrics, and candidate (Active/Working and Protect) paths.

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

The **SR Policy** table lists all policies.

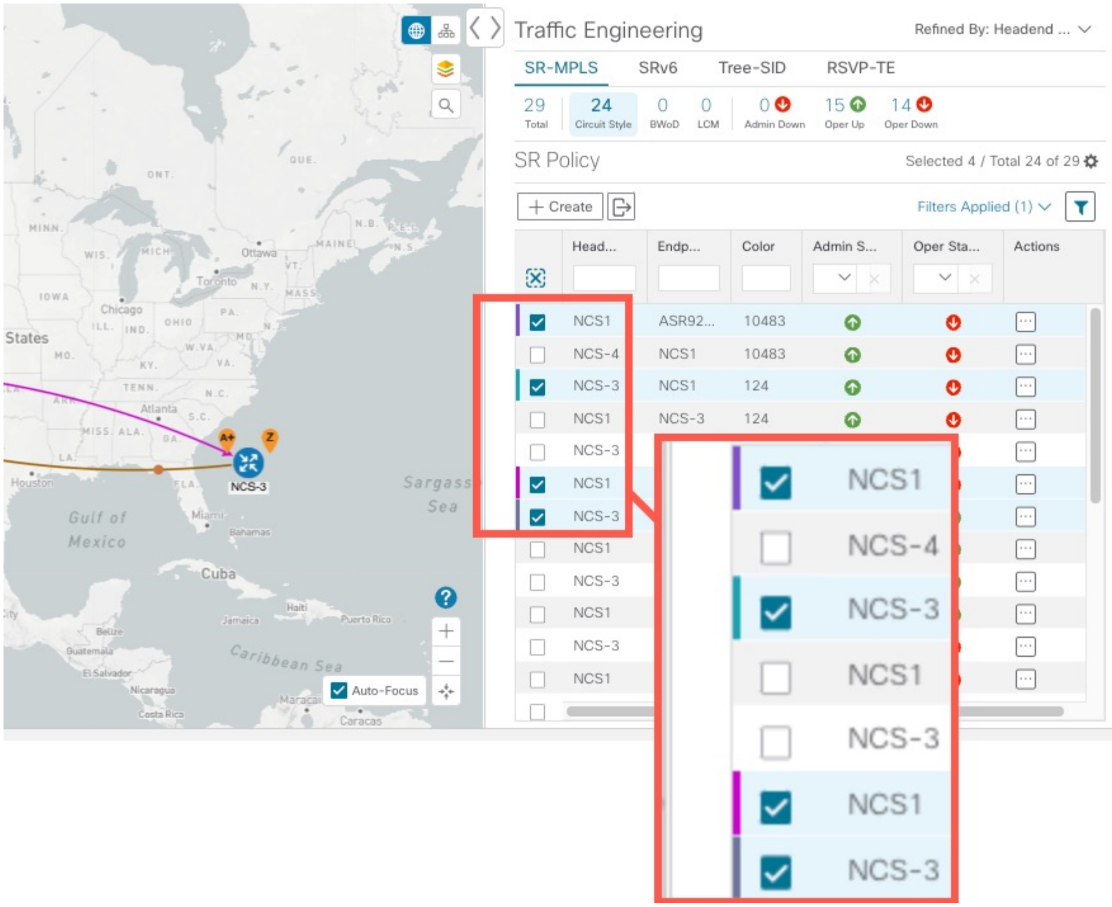
The **SR Policy** table lists all of the Circuit Style SR-TE policies.

Step 2 Check the check box next to **Show Participating Only** so that other links and devices that are not part of the selected Circuit Style SR-TE policies are hidden.

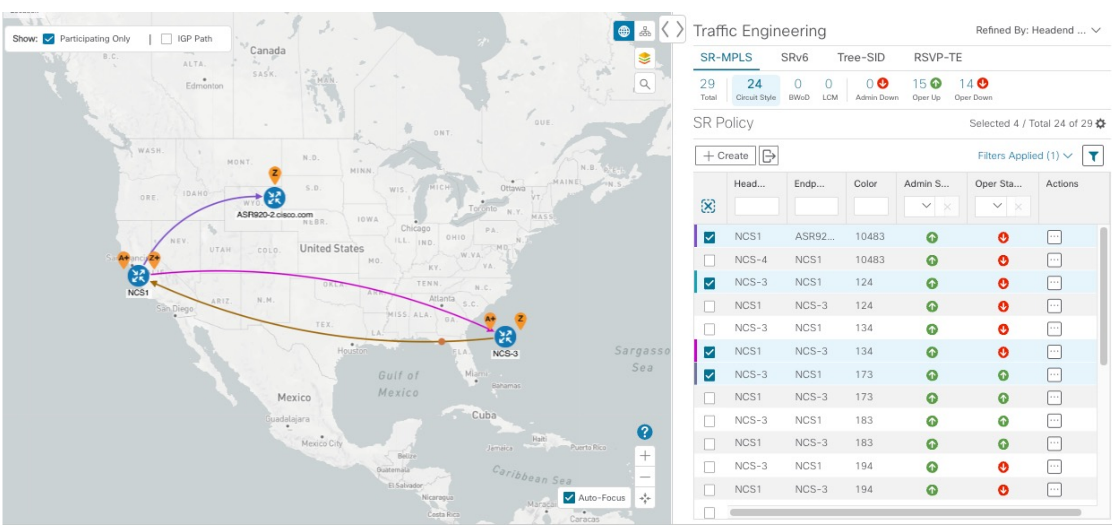


From the topology map above, note the following details:

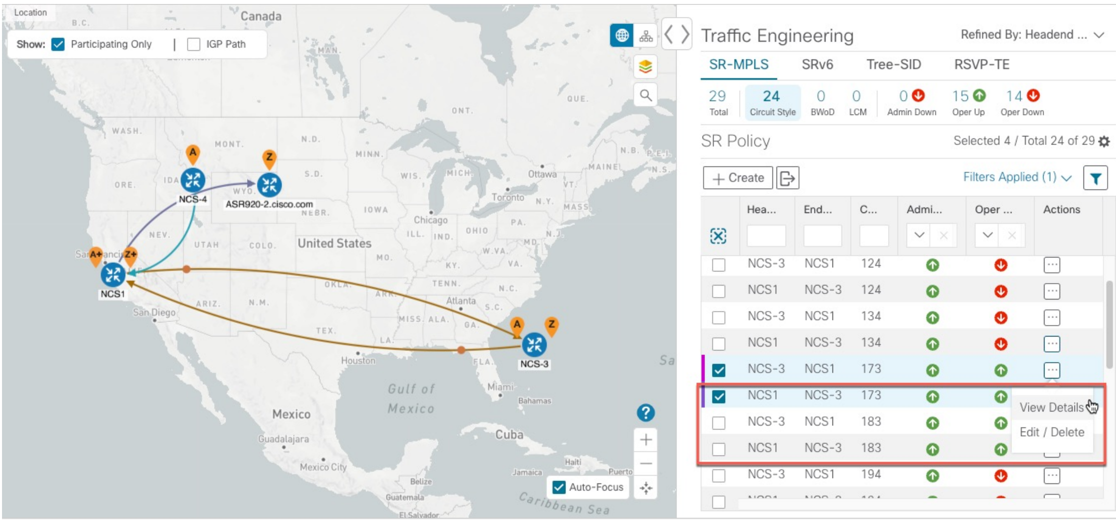
- Four Circuit Style SR-TE policies are checked, but only three paths appear to be visible. The reason for this is that the last two Circuit Style SR-TE policies have the same endpoints (NCS1 and NCS-3) and share the same paths. The brown colored link indicates policies that use the same paths.
- The line representing each path defined in the policy appears on the topology map using a different line color. Vertical color bars next to each checked policy in the **SR Policy** table match the color used for the corresponding path line on the map. To see the color legend, explaining which colors are used and what each color shows, click the ?.



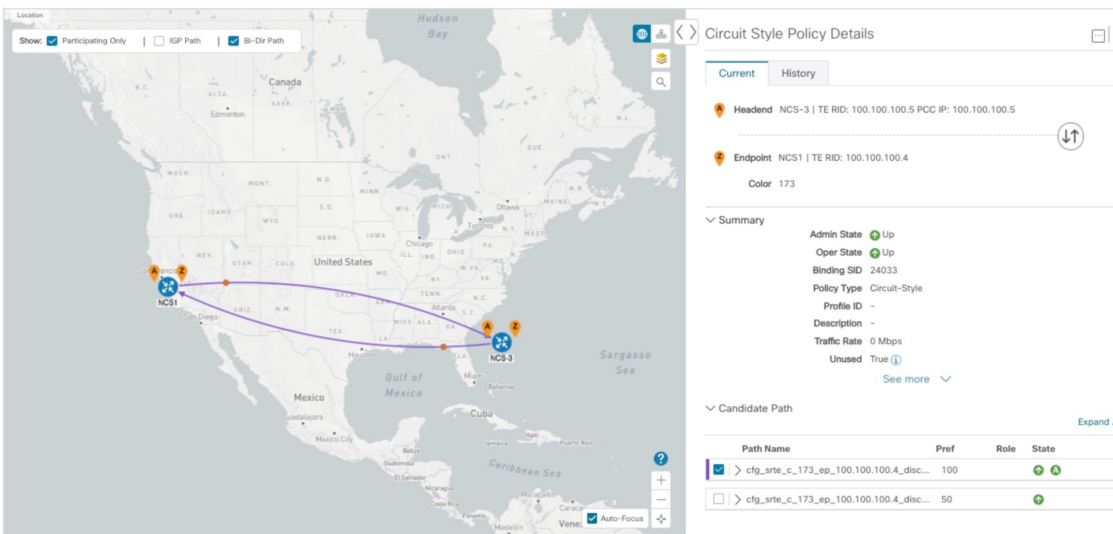
- The **A+** denotes that there is more than one SR-TE policy that originates from a node. The **Z+** denotes that the node is a destination for more than one SR policy.



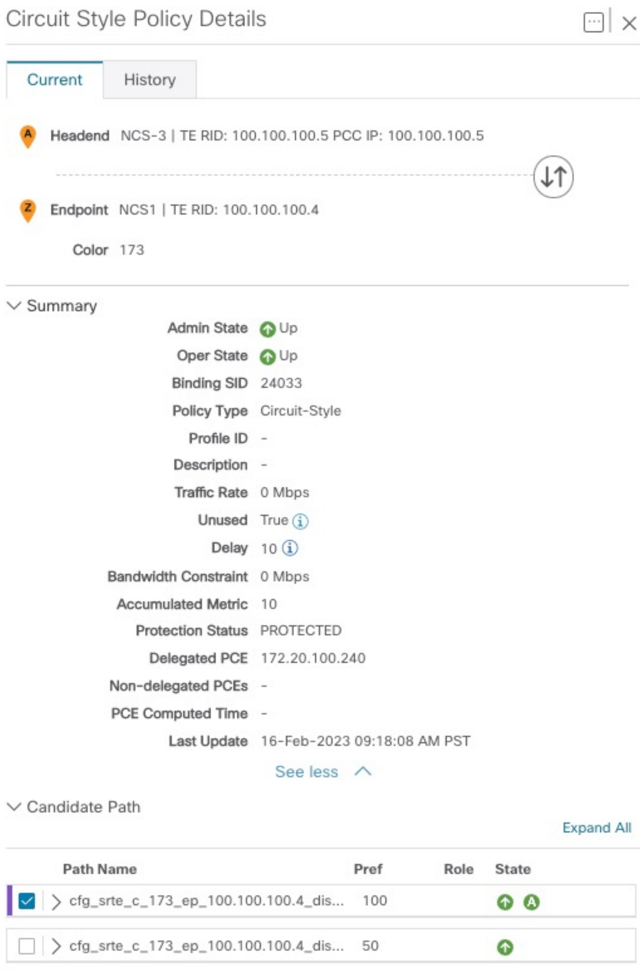
Step 3 From the **Actions** column, click ... > **View Details** for the NCS policies.



Crosswork displays the **Circuit Style Policy Details** window in the side panel. By default, the Active path is displayed on the topology map. The display includes the bidirectional paths between NCS-1 and NCS-3 (i=f the **Bi-Dir Path** checkbox is checked).

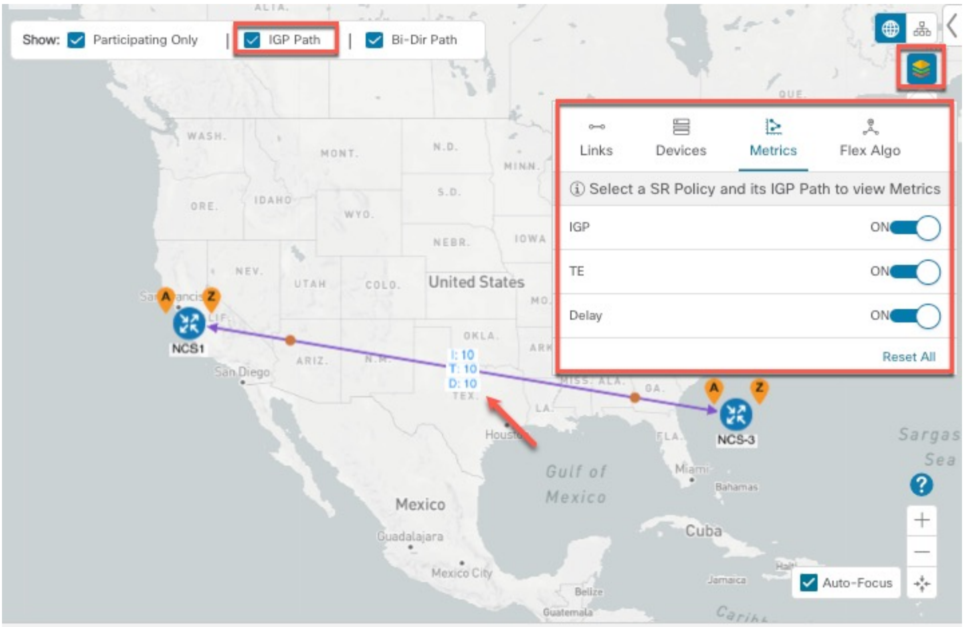


Here is a closer look at the types of Summary details available to you. The **Candidate Path** list at the bottom of the window displays the Active and Protected paths. The Active path is the one that currently takes traffic.



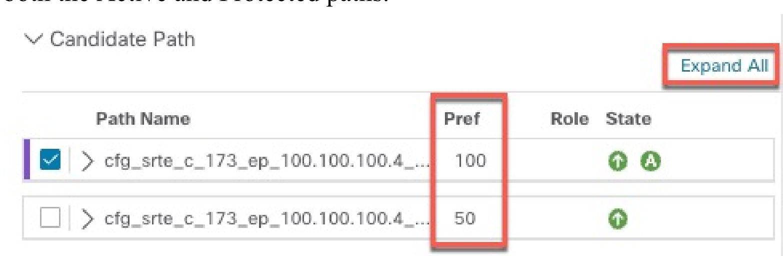
Note The Bandwidth Constraint value may differ from the bandwidth you requested if the value was increased and insufficient resources existed to satisfy demand on all Active and Protected candidate paths.

Step 4 To view the physical path and metrics between endpoints of the selected Circuit Style SR-TE policies, click to turn applicable metrics on and check the **IGP Path** checkbox.



Step 5 View the Active and Protected path configuration details:

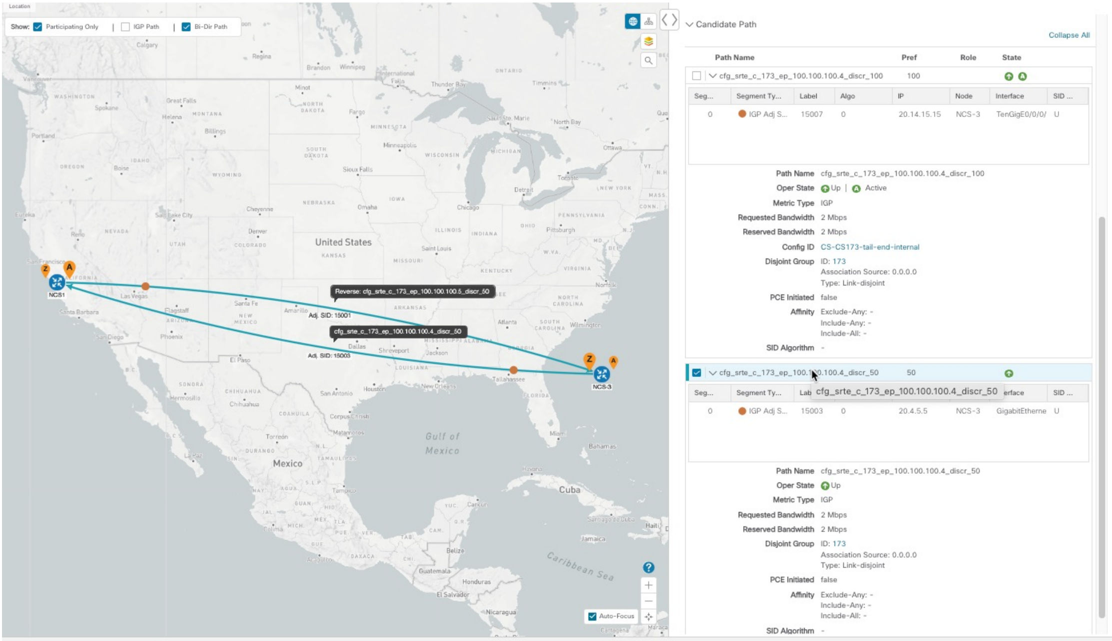
a) Within the **CS-SR Policy Details** window, you can drill down to view more information about the Active and Protected paths. In the following example, the Active path has a preference value of 100 and the Protected path has preference value of 50. The operational (Oper State Up) candidate path with the highest preference will always be the Active path (see What Happens When Path Failures Occur?, on page 102). Click **Expand All** to view more information about both the Active and Protected paths.



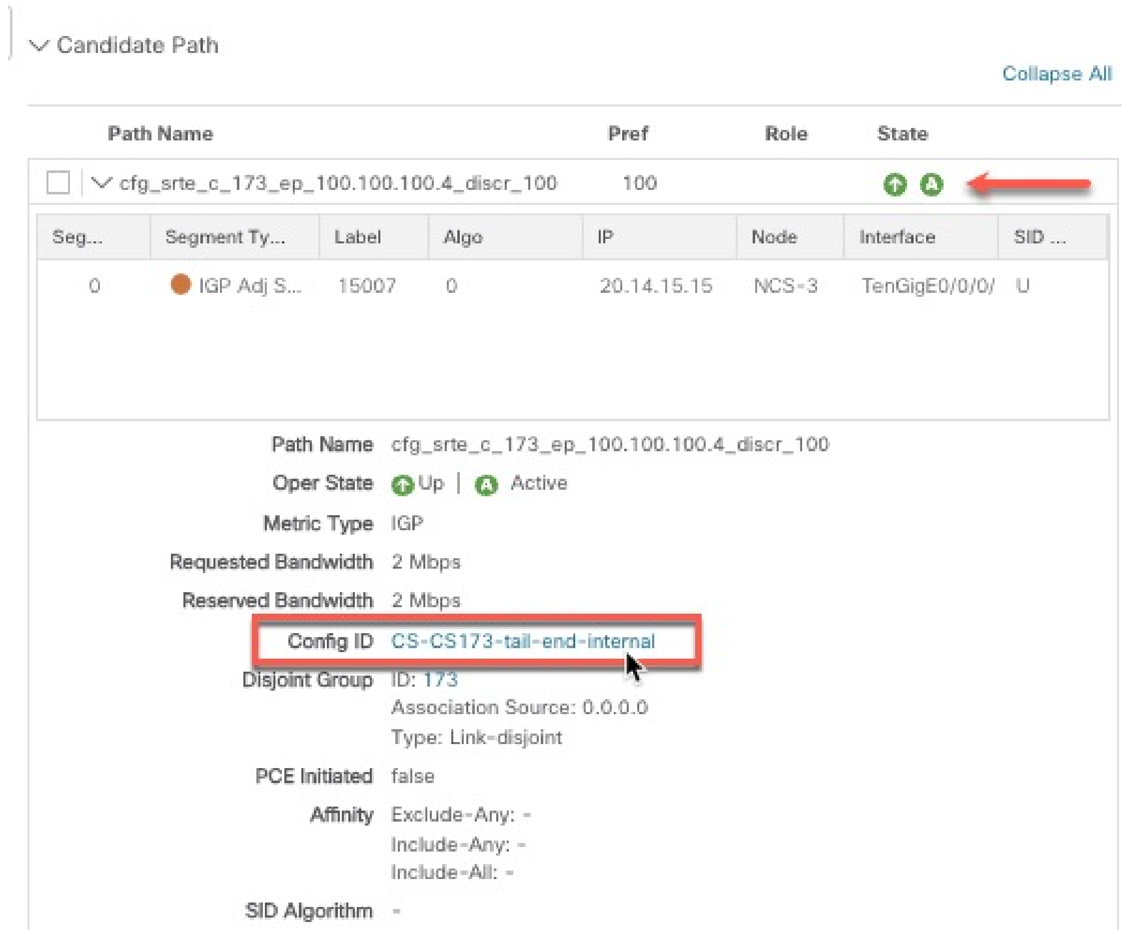
Note

- First preference paths are shown as purple links.
- Second preference paths are shown as blue links.
- Third preference paths are shown as pink links.

b) In the following example, the Protected path is checked and displayed on the topology map. If you hover your mouse over the path name, forward and reverse paths are displayed on the topology map.



c) Here is a closer view of an Active path's configuration details. Notice that it is designated with the "A" icon under **State** to indicate that it is currently the operational Active path. Also, if the policy configuration was done through Cisco Crosswork, you have the option to view the policy configuration. To see the configuration, click the link next to **Config ID**.



Step 6: Verify Circuit Style SR-TE Policy Bandwidth Utilization

Let's verify that the reserved bandwidth pool settings we defined when enabling Circuit Style SR-TE (see Step 1: Enable SR Circuit Style Manager, on page 105) are configured properly. We can also check how much bandwidth is either in use or still available.

Step 1 From the main menu, choose **Services & Traffic Engineering** > **Traffic Engineering** > **SR-MPLS**. Then, under the **SR-MPLS** column, click **Circuit Style**. The **SR Policy** table lists all CS SR policies.

Step 2 In the **SR Policy** table, check the check box next to the participating device whose details you want to see.

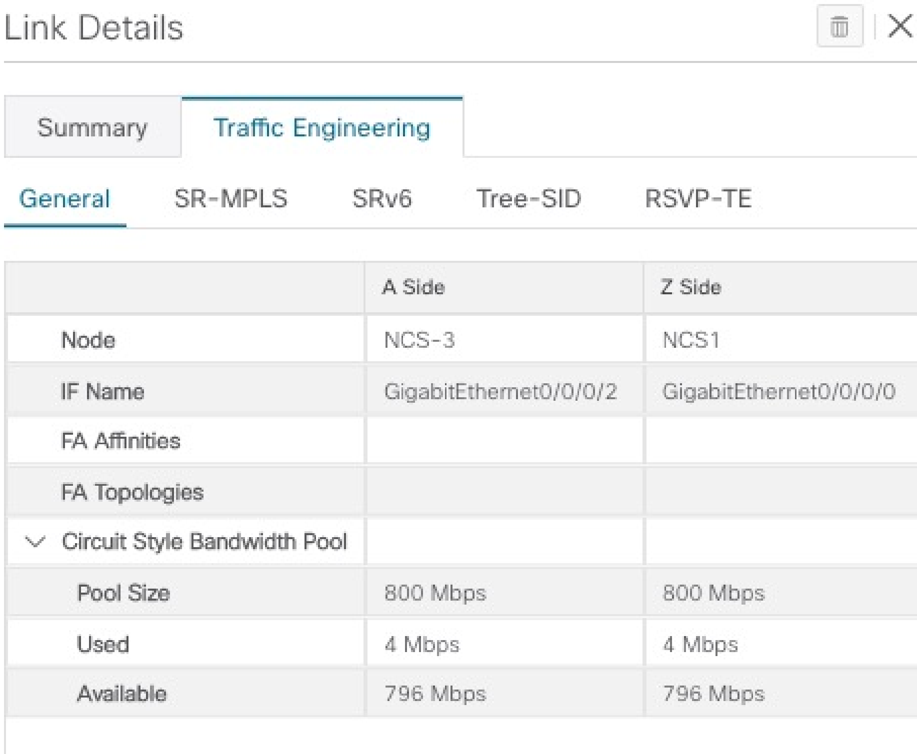
Step 3 On the topology map, click on a participating Circuit Style SR-TE policy node to display the **Device Details** for that node.

Step 4 On the **Device Details** page, click the **Links** tab to display the list of CS-SR and other links on the participating node. Then click on the link whose details you want to see. The **Link Details** list displays a **Summary** of the link information.

Step 5 Click on the **Traffic Engineering** tab, then **General**. The **Link Details** list displays detailed information for the link.

Under **Circuit Style Bandwidth Pool,** you can see the reserved bandwidth pool size, the amount of bandwidth currently being used, and what bandwidth (of the total allocated to Circuit Style SR-TE policies) is still available.

In this example, the reserved bandwidth pool size is displayed as 800 Mbps for NCS-3 and NCS1. The configured settings were earlier defined as 80% for the bandwidth pool size. Since the interfaces on this circuit are both 1 Gbps, we can confirm that Circuit Style SR-TE has correctly allocated 80 percent of bandwidth for these two interfaces.



	A Side	Z Side
Node	NCS-3	NCS1
IF Name	GigabitEthernet0/0/0/2	GigabitEthernet0/0/0/0
FA Affinities		
FA Topologies		
Circuit Style Bandwidth Pool		
Pool Size	800 Mbps	800 Mbps
Used	4 Mbps	4 Mbps
Available	796 Mbps	796 Mbps

Step 7: Trigger Circuit Style SR-TE Path Recomputation

Circuit-Style policies are static in nature, meaning once the paths are computed, Crosswork will not re-compute them automatically. Changes in your network topology or operational status may affect the previously computed Working and Protected paths to the extent that you want Crosswork to re-compute and optimize them for the new situation. In this step, we see a demonstration of how to re-optimize for paths to accommodate these types of changes.

For more details on the logic CSM employs in these cases, see What Happens When Path Failures Occur?, on page 102.

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

Step 2 The SR Policy table displays the status of each of the Active CS-SR policies. One of them is Operationally down.

Step 3 From the **Actions** column next to the CS-SR TE policies whose Operational State is **Down**, click > **View Details**.

Crosswork displays the **Circuit Style Policy Details** window in the side panel. By default, the Active path is displayed and shows the bidirectional paths on the topology map (for these to appear, the **Bi-Dir Path** checkbox in the topology map's **Show** panel must be checked). The **Candidate Path** list at the bottom of the side panel displays the Active (Working) and Protected paths.

Click the **Show more** link to get a closer look at the type of Summary details available. The Candidate Path list displays the Active and Protected paths.

Step 4 To have Crosswork re-optimize these paths: Click  at the top of the **Circuit Style Policy Details** panel and select **Re-optimize**.

Summary and Conclusion

In this scenario, we observed how to use Circuit Style Segment Routing policies to reserve bandwidth for high-priority services and traffic in the network. CS-SR removes the need to manually track and calculate high-priority traffic paths, but still gives you control over how those paths are calculated and optimize bandwidth usage on each path. You can use these policies to ensure that available bandwidth is dedicated for these services. As traffic changes, Circuit Style policies warn you when your dedicated "circuit" paths fail, and allows you to re-optimize them as needed.

CHAPTER 5

Network Maintenance Window

This section explains the following topics:

Overview

Objective

Schedule and automate maintenance workflows with minimal network interruption and most efficient results.

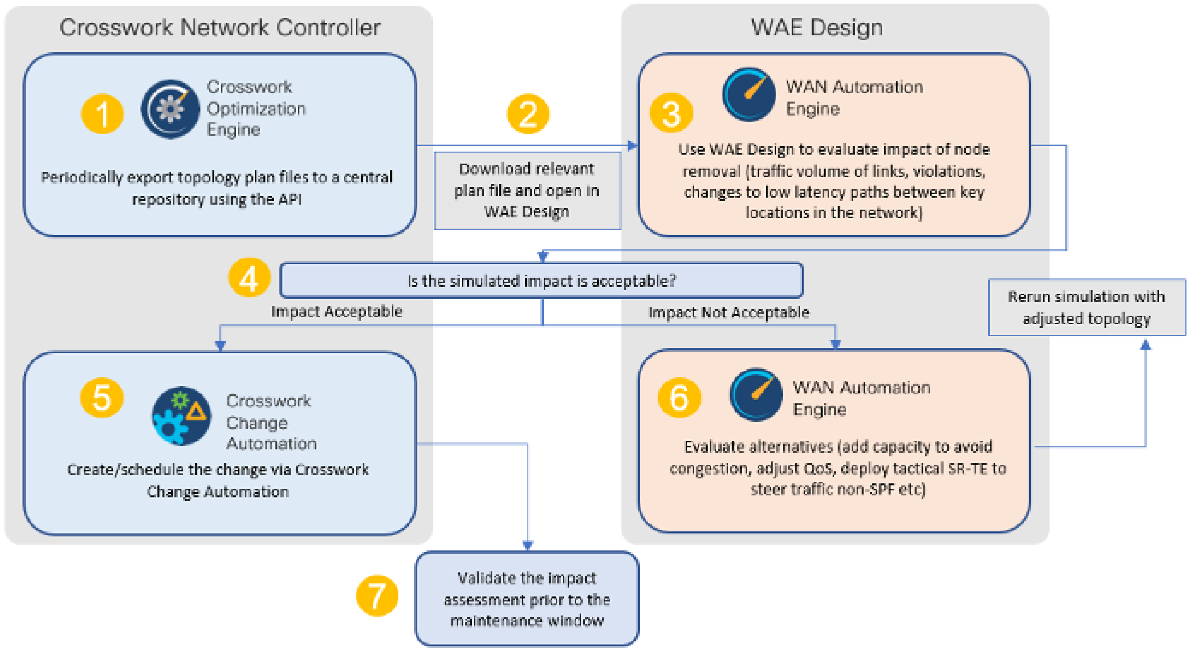
Challenge

Maintenance activities typically require system downtime and temporary disruption of services. Keeping downtime and disruption to a minimum is critical but challenging. Therefore, maintenance activities must take place during a carefully calculated optimal time slot, usually when activity is at its lowest.

Solution

Cisco Crosswork Change Automation and Cisco Crosswork Health Insights are optional add-on applications that provide the functionality needed to automate the scheduling and execution of maintenance tasks. Planning the optimal time for maintenance activities can be done successfully using Cisco WAE Design to simulate "what-if" scenarios based on timed topology snapshots exported from Cisco Crosswork Network Controller using APIs.

How Does it Work?



- Using the Crosswork Network Controller APIs, you can create topology snapshots (plan files) which capture and represent topology state at a given point in time, including the IGP topology as well as interface level statistics (traffic load). For impact analysis purposes, these snapshots should be representative of a time period to be evaluated for an upcoming maintenance activity. For example, if you are planning a router upgrade at midnight on a Monday, you would take snapshots from several Mondays at midnight to evaluate typical traffic loads at this time. You can export these plan files to a central storage repository, where a library of topology plan files can be stored for a specified period of time.
- Cisco WAE Design allows you to explore "what-if" scenarios relevant to the planning of the maintenance window. For example, in the case of upgrading a router, Cisco WAE Design can simulate the resulting traffic load on the remaining devices after traffic is diverted from the device being upgraded. You can also explore the impact of deploying tactical traffic engineering policies to further optimize the topology during the maintenance window. For more information, contact your Cisco Customer Experience representative.

Additional Resources

Cisco Crosswork Change Automation and Health Insights User Guide

Cisco WAE Design documentation

Cisco Crosswork Network Automation API Documentation on Cisco Devnet

Scenario 8 – Perform a software upgrade on a provider device during a scheduled maintenance window

Scenario Context

This scenario assumes that Cisco WAE Design has been used to evaluate the impact of removing a P node from the network to perform a software upgrade during a specific timeframe. In this scenario, we will choose a predefined playbook to automate the execution of the SMU on the device, and we will schedule it to run during the predetermined maintenance window.

Assumptions and Prerequisites

- Cisco Crosswork Change Automation must be installed and running.
- You must have access to Cisco WAE Design.
- The Device Override Credentials must be set for Crosswork Network Change Automation to be functional. Go to **Administration** > **Settings** > **System Settings** > **Network Automation**.

Step 1 Download Topology Plan Files for Impact Analysis

When considering when to take down a device for maintenance so that there will be the least impact to the network, you need information about the traffic trends around that device at the targeted time. Using the Cisco Crosswork Optimization API, you can download plan files that capture a snapshot of the network topology at that time. If you download plan files at the same time over a period of time, you can use Cisco WAE Design to analyze the traffic trends. Based on this analysis, you can decide whether the impact to the network would be acceptable or not.

Refer to Cisco Crosswork Network Automation API Documentation on Cisco Devnetfor more information about the API.

The input for this scenario is as follows:

Step 1 Prepare the input required to download the plan file. You need to specify the version of Cisco WAE design that you will be using for analysis and the format in which you want the plan file, either txt or pln.

Note If you download the plan file as a txt file, you can view it in any text editor. If you download it as a pln file, you can open it only in Cisco WAE design.

The input for this scenario is as follows:

```
'{
  "input": {
    "version": "7.3.1",
    "format": "txt",

    }
  }
}'
```

Step 2 Invoke the API on the Cisco Crosswork Network Controller server using the input prepared in the previous step. For example:

```
curl --location --request POST
'https://10.194.63.198:30603/crosswork/nbi/optima/v1/restconf/operations/cisco-crosswork-
optimization-engine-operations:get-plan \
--header 'Content-Type: application/yang-data+json' \
--header 'Accept: application/yang-data+json' \
--header 'Authorization: Bearer
eyJhbGciOiJIUzUxMiJ9.eyJzdWIiOiJhZG1pbiIsImlzRnJvbU5ld0xvZ2luIjoidHJ1ZSIsInBvbGljeV9pZCI6ImFkb
WluIiwiYXV0aGVudGljYXRpb25EYXRlIjoiMjAyMS0wMy0yMlQxNjozODozNy43NDY2MTZaW0dNVF0iLCJzdWNjZXNzZnV
sQXV0aGVudGljYXRpb25IYW5kbGVycyI6IlF1ZXJ5RGF0YWJhc2VBdXRoZW50aWNhdGlvbkhhbmRsZXIiLCJpc3MiOiJod
```

```
HRwOlwvXC9sb2NhbGhvc3Q6NTQ4OVwvU1NPIiwibGFzdF9uYW1lIjoic21pdGgiLCJjcmVkZW50aWFsVHlwZSI6IlVzZXJ
uYW1lUGFzc3dvcmRDcmVkZW50aWFsIiwiYXVkIjoiaHR0cHM6XC9cLzEwLjE5NC42My4xOTg6MzA2MDNcL2FwcC1kYXNoY
m9hcmQiLCJhdXRoZW50aWNhdGlvbk1ldGhvZCI6IlF1ZXJ5RGF0YWJhc2VBdXRoZW50aWNhdGlvbkhhbmRsZXIiLCJsb25
nVGVybUF1dGhlbnRpY2F0aW9uUmVxdWVzdFRva2VuVXNlZCI6ImZhbHNlIiwiY2hhbmdlX3B3ZCI6ImZhbHNlIiwiZXhwI
joxNjE2NDU5OTIwLCJpYXQiOjE2MTY0MzExMjAsImZpcnN0X25hbWUiOiJqb2huIiwianRpIjoiU1QtODQtOFVlWXMybEt
3R2d1Z3RIYjl4MzVmTFlNTGVVRlp6OURyNGpoeFcxakhsV01VYXdXSWgxbUdTd01aRC1tOEk1S2Z0amI2ZmlWTUhlYnBYY
jBMMFZqRFc2WVppUFVUbHRpNFVpZnNUeG9aQ284WWpPWEc2VlFjS0Mwb29lWjJhc3BWanMzYnA3bHo5VkhySlBCTzl5TDN
GcFRIWXRPeWJtVi1jYXMtMSJ9.Vi4k0w8KsOv5M_O8zBqWochT3k9V9Pn2NjSn5ES9c5Pf-
4ds0o4kk6xuZx5_cggauiEICuUMnzmRzneST-oAuA' \
--data-raw '{
  "input": {
    "version": "7.3.1",
    "format": "txt",
    "
    }
  }
}
```

Step 3 Note the plan file content in the API response. It is encoded for security purposes and must be decoded before you can view the content.

```
{
  "cisco-crosswork-optimization-engine-operations:output": {
    "status": "accepted",
    "plan-file content": "
PE5ldHdvcms+ClByb3BlcnR5CVZhbHVlClRpdGxlCQpWZXJzaW9uCTcuMy4xCgo8TmV0d29ya09wdGlvbnM+Ck9wdGlvbg
lWYWx1ZQpFbmZvcmNlQWRqU0lETG9jYWxpemF0aW9uCVRSVUUKCjxDaXJjdWl0cz4KTmFtZQl<<<>>>Ob2RlQQlJbnRlcm
ZhY2VBCU5vZGVCCUludGVyZmFjZUIJQ2FwYWNpdHkJRGVsYXkJRGlzdMJTmV0SW50U05NUF9FcnJvcglOZXRJbnRTb3VyY
2UJTmV0SW50UkUwQ1BVMW0JTmV0SW50UkUwQ1BVNWZpZXIJQWxnb3JpdGhtCVJmbGFnCU5mbGFnCVBmbGFnCUVmbGFnCVZ
mbGFnCUxmbGFnCg=="
  }
}
```

Step 4 Use a script to decode the plan file or copy the plan file content into a decoder. After decoding the plan file, you can see the content of the model to be used in Cisco WAE Design. It includes a full snapshot of the topology, including the devices, interfaces, links, LSPs, traffic levels, and other information.

Step 5 Open the plan file in Cisco WAE Design, simulate the device going down, and observe the impact on the network. Refer to the Cisco WAE Design documentation for more information.

Step 6 Based on the analysis, decide on an optimal time to execute the SMU.

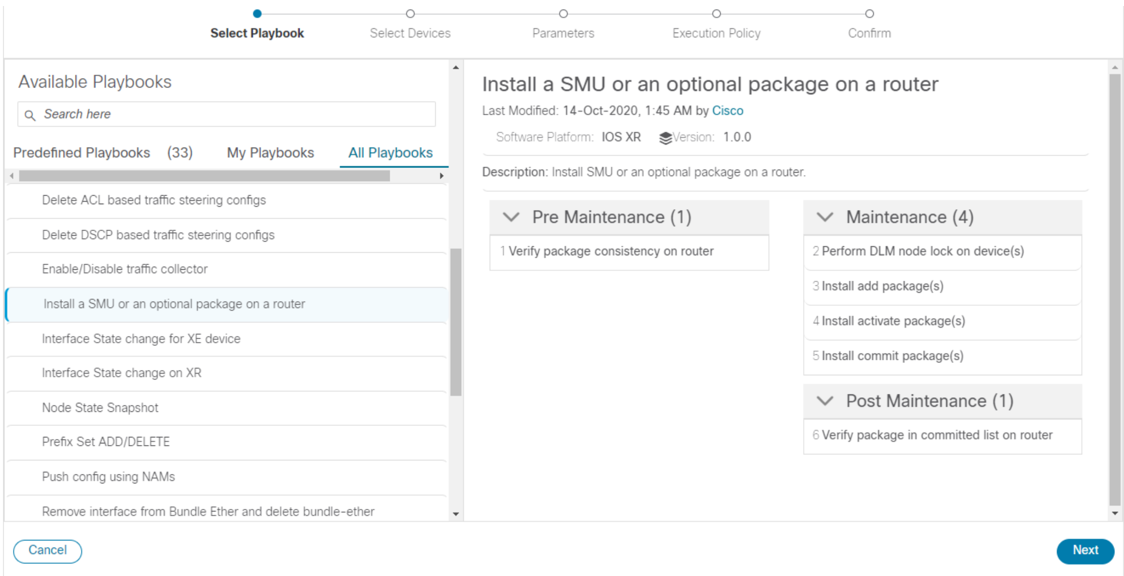
Step 2 Schedule and execute the SMU by running a playbook

If the simulated impact is acceptable, you can create and schedule the change by running a playbook through Cisco Crosswork Change Automation. For this scenario, we will run a predefined playbook to install a Software Maintenance Update (SMU) on devices tagged under a certain geographic location (NY).

Note If the predefined (stock) plays and playbooks do not meet your specific needs, you can create custom plays and playbooks. To create a custom play, go to **Network Automation** > **Play List**, and to create a custom playbook, go to **Network Automation** > **Playbook List**.

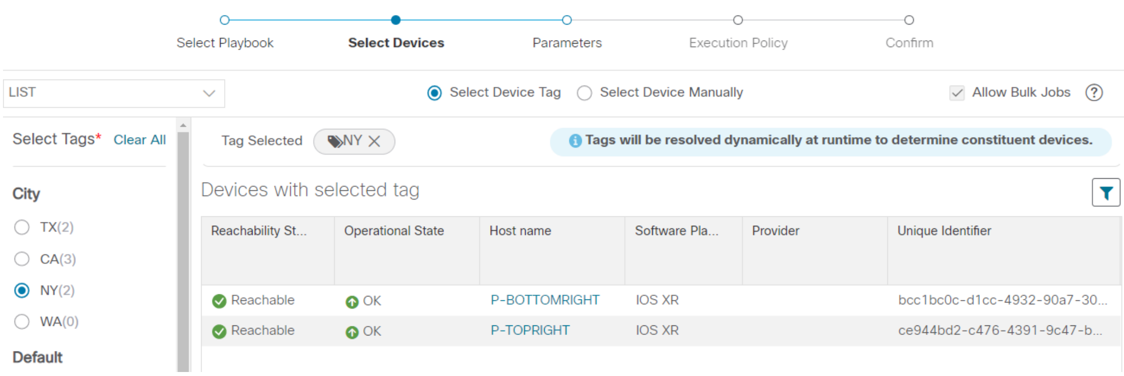
Step 1 Go to **Network Automation** > **Run Playbook**.

Step 2 Browse the Available Playbooks list, and click the Install a SMU playbook. You can also filter using keywords to identify the playbook. Note that the playbook execution stages, supported software platform, software version, and individual play details are displayed on the right side.



Step 3 Click **Next** to go to the next task: Select Devices. All devices tagged with City: NY will be selected for SMU installation.

Step 4 Under the City tag on the left, click **NY**. The devices tagged with NY are listed on the right and are automatically selected.

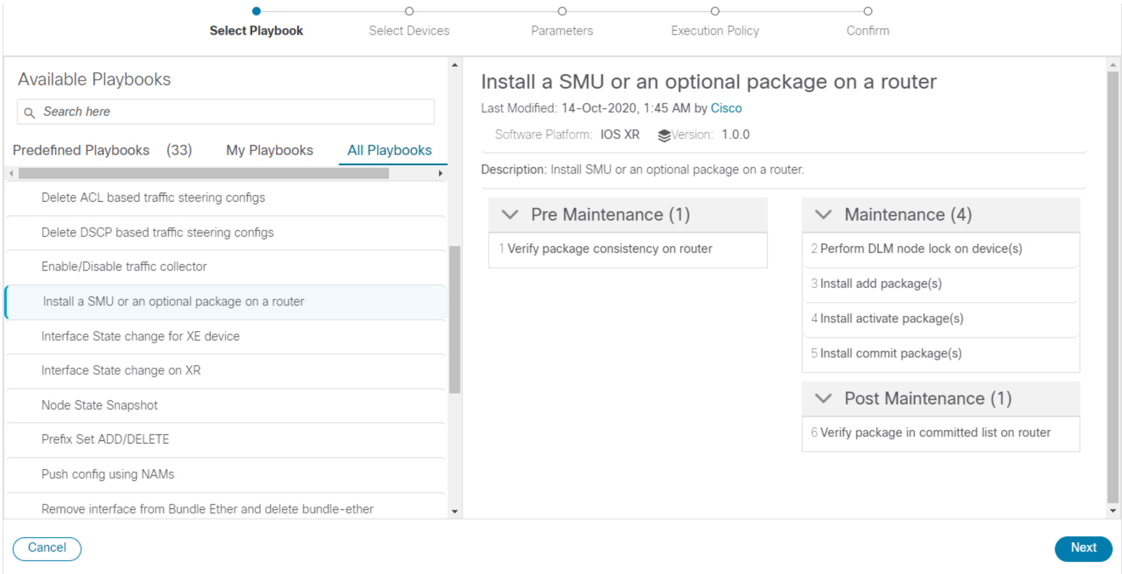


Step 5 Click **Next** to go to the next task: Define Parameters.

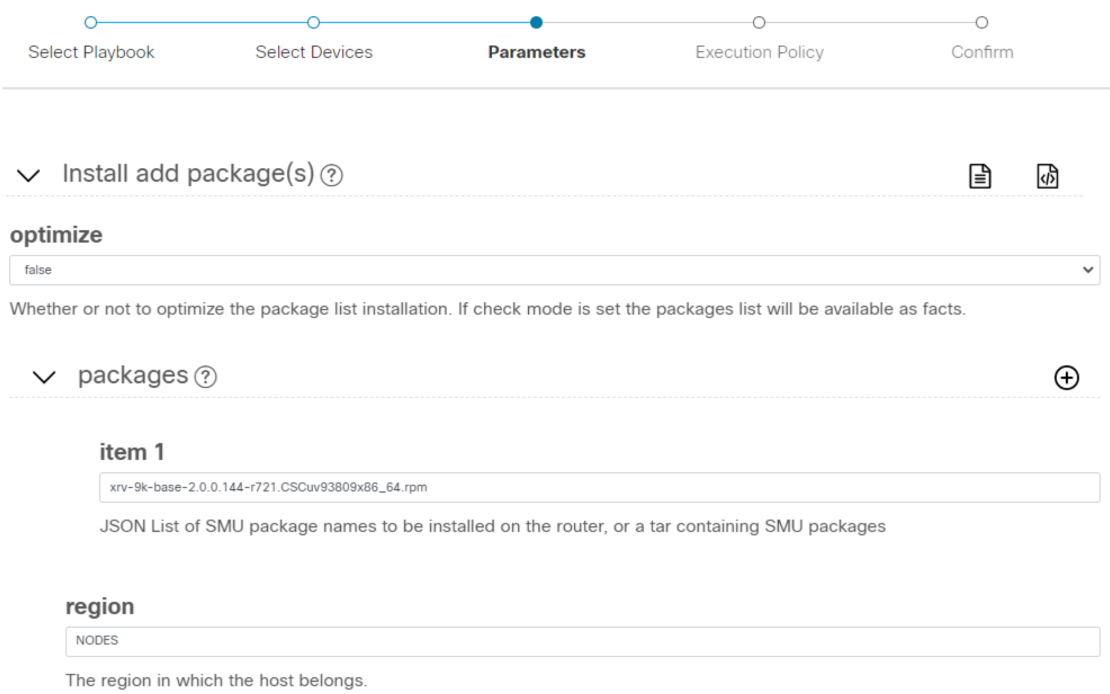
Step 6 Edit the runtime parameters to execute the SMU playbook. Alternatively, you can upload a JSON file that contains the parameter values. The following values are used specifically for this scenario. You can change them as required:

a. Under "verify package consistency on the device" play, set **collection_type** as **mdt**.

b. Under "perform DLM node lock on device" play, set **retry_count** and **retry_interval** as **3** and **5s** respectively.



c. Under "Install add package(s)" play, set **action** as **add**, and **optimize** as **false**. Enter the <SMU package name> in **item 1** and set **region** as **NODES**.



d. Set type as SCP, and enter values for the source, address, destination, and dlm_credential_profile.

e. Under **Install activate package(s)**, click the piece of paper symbol, select action, and set **action** to **Activate**.

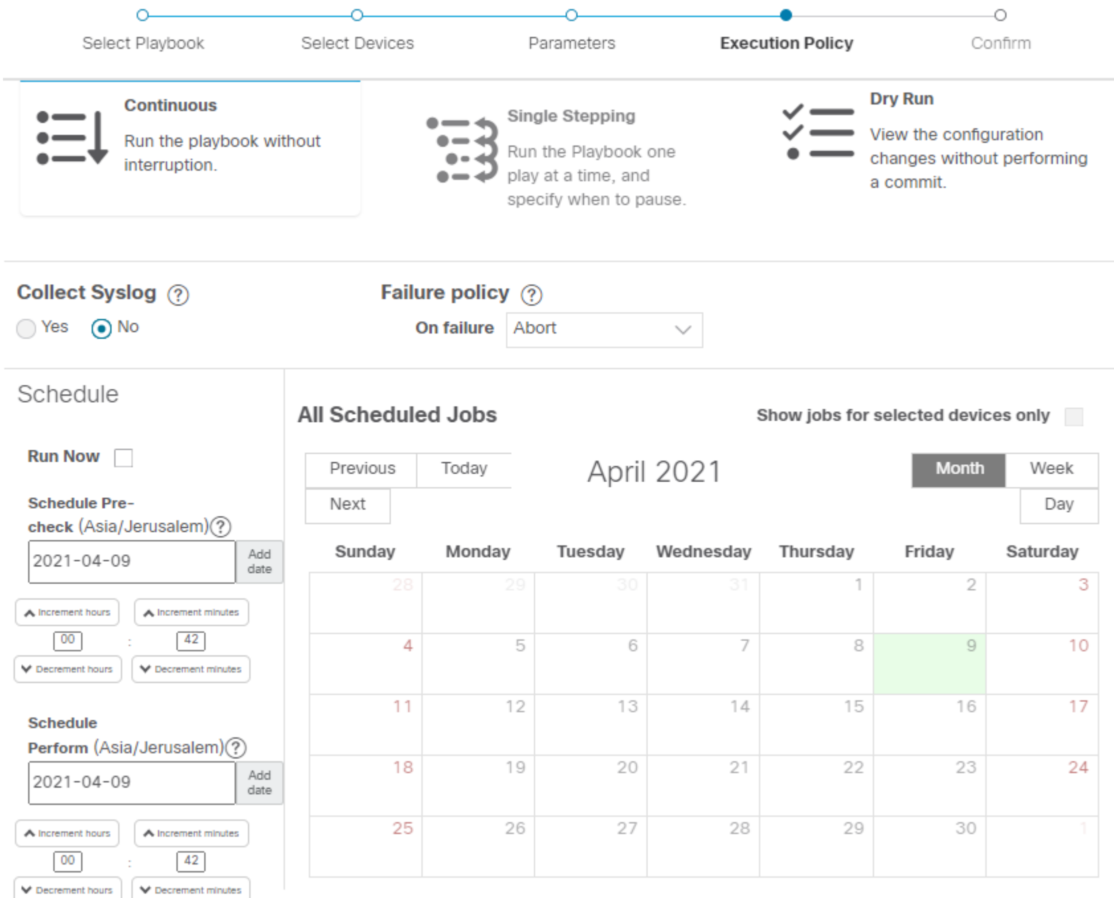
f. Under Install commit package(s), set action to Commit.

g. Under Verify package in committed list on router, set collection_type to mdt, and enter the <SMU package name> in item 1.

Step 7 Click **Next** to go to the next task: Define Execution Policy.

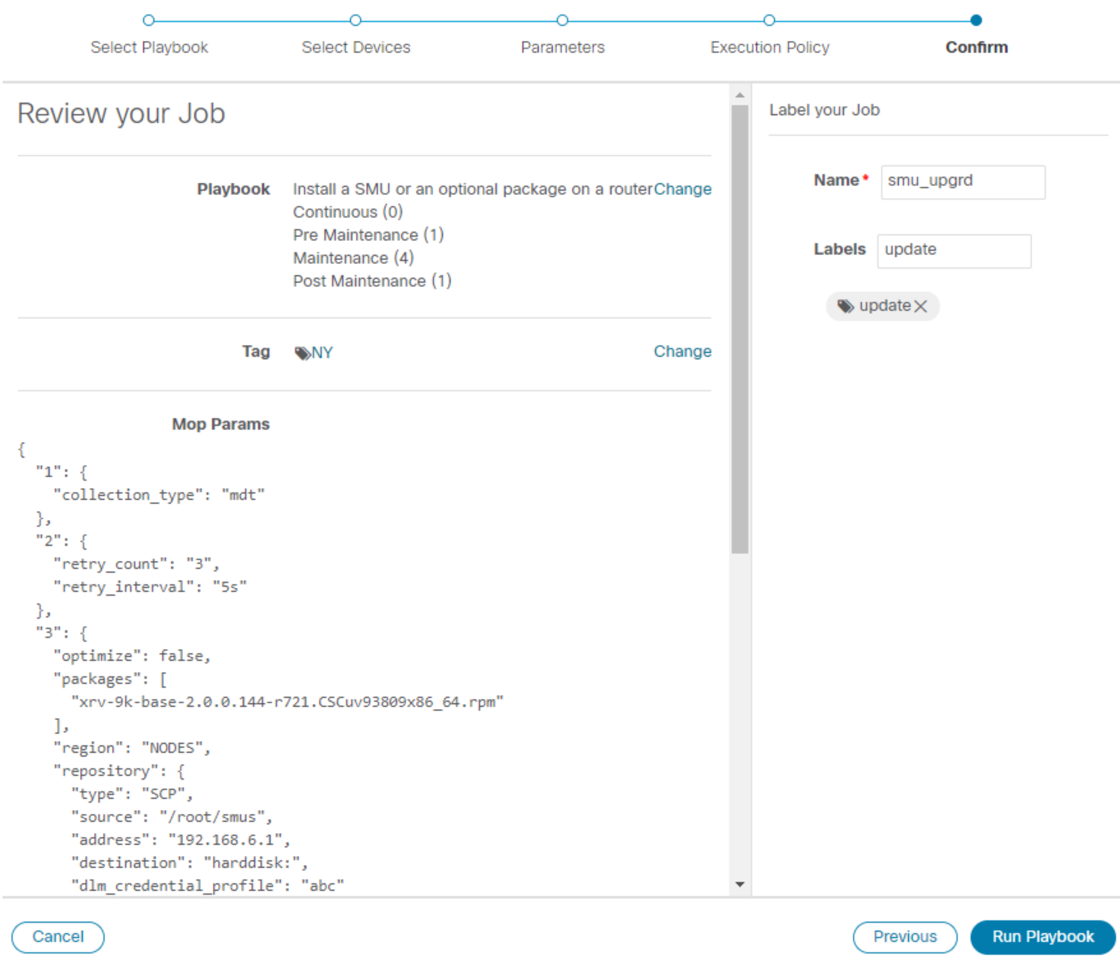
Step 8 Select **Continuous** as the Execution mode so that the playbook will run uninterrupted with no pauses. Under Failure policy, select the action you want taken if the execution fails – abort or rollback.

Step 9 Schedule the execution for the optimal time calculated during the impact analysis stage. Uncheck the **Run Now** option. Note the calendar and timer that are displayed to schedule pre-check and perform plays. Select the date and time for the scheduled maintenance.



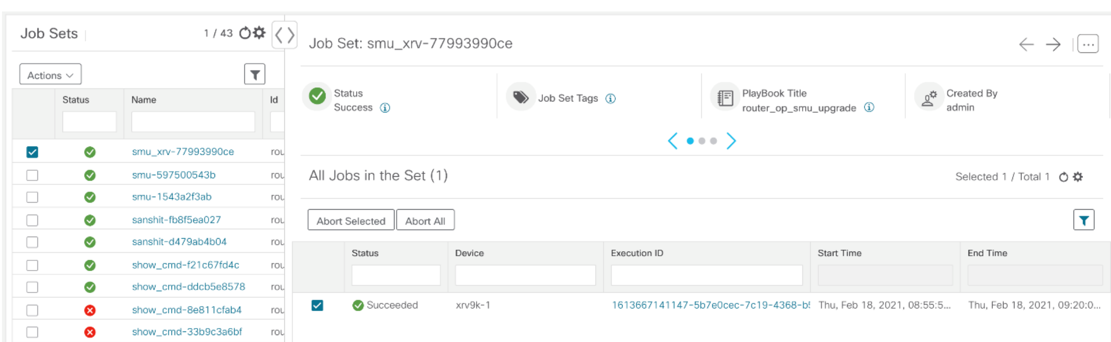
Step 10 Click **Next** to go to the next task: Confirm Job.

Step 11 Review your job details. Label your job with a unique name. Click **Run Playbook**. The SMU installation is now scheduled to run in the planned maintenance window.

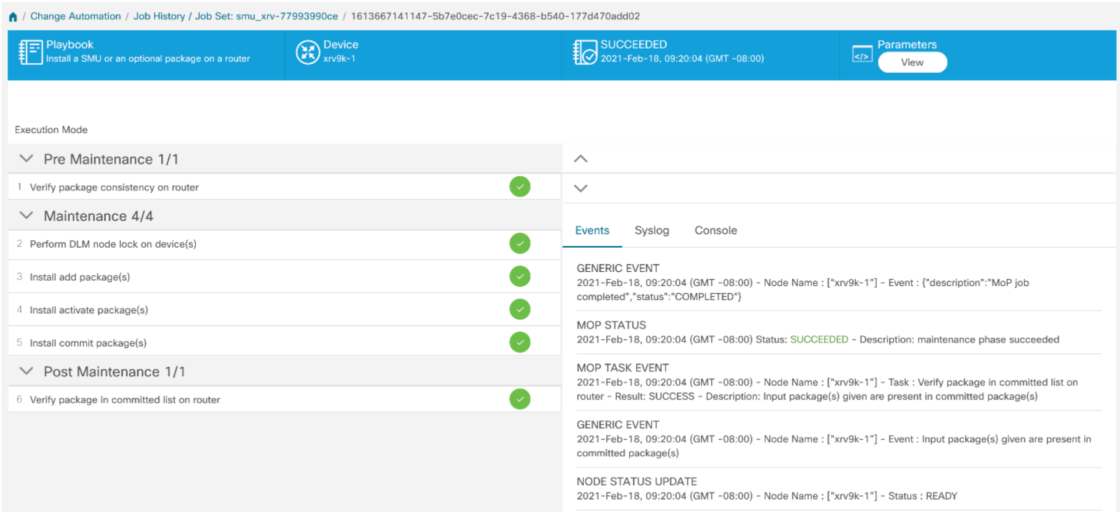


Step 3 Verify the SMU install job completion status

Step 1 After the scheduled maintenance window time, go to **Network Automation** > **Automation Job History**. Under Job Sets, check that the job status icon on the SMU install job is Green, indicating that the scheduled job has run successfully.



Step 2 Select the SMU install job. Note the Job Set details on the right side. Click the **Execution ID** for job details.



Step 3 Double-check that the correct SMU has been installed by executing the "show install active summary" and "show install committed summary" commands on the device and checking that the SMU you installed appears in the list. Some example outputs from these commands are shown below:

```
RP/0/RP0/CPU0:CX-AA-PE4#show install active summary
Mon Apr 12 11:09:20.198 EDT
    Active Packages: 12
        ncs5500-xr-6.6.3 version=6.6.3 [Boot image]
        ncs5500-ospf-2.0.0.0-r663
        ncs5500-mpls-2.1.0.0-r663
        ncs5500-eigrp-1.0.0.0-r663
        ncs5500-isis-2.2.0.0-r663
        ncs5500-li-1.0.0.0-r663
        ncs5500-mpls-te-rsvp-4.1.0.0-r663
        ncs5500-mcast-3.1.0.0-r663
        ncs5500-mgbl-3.0.0.0-r663
        ncs5500-k9sec-3.1.0.0-r663
        ncs5500-routing-4.0.0.17-r663.CSCvr43225
        ncs5500-mpls-te-rsvp-4.1.0.17-r663.CSCvr43225

RP/0/RP0/CPU0:CX-AA-PE4#show install committed summary
Mon Apr 12 11:09:27.092 EDT
    Committed Packages: 12
        ncs5500-xr-6.6.3 version=6.6.3 [Boot image]
        ncs5500-ospf-2.0.0.0-r663
        ncs5500-mpls-2.1.0.0-r663
        ncs5500-eigrp-1.0.0.0-r663
        ncs5500-isis-2.2.0.0-r663
        ncs5500-li-1.0.0.0-r663
        ncs5500-mpls-te-rsvp-4.1.0.0-r663
        ncs5500-mcast-3.1.0.0-r663
        ncs5500-mgbl-3.0.0.0-r663
        ncs5500-k9sec-3.1.0.0-r663
        ncs5500-routing-4.0.0.17-r663.CSCvr43225
        ncs5500-mpls-te-rsvp-4.1.0.17-r663.CSCvr43225

RP/0/RP0/CPU0:CX-AA-PE4#
```

Summary and Conclusion

In this scenario we saw how to plan for a maintenance window in which to bring down a device in order to install an SMU. The goal is to cause as little impact to the traffic in the network as possible. To analyze the impact on the network, we showed how to download snapshots of the network topology (plan files) at the target time for the maintenance window. The plan files can then be analyzed using Cisco WAE design.

Assuming that the impact was acceptable, we chose a predefined playbook to install the SMU on specific devices and we scheduled it for the planned maintenance window time when there would be the least impact to the network.

CHAPTER 6

Programmable Closed-Loop Remediation

This section explains the following topics:

Overview

Objective

Detect anomalies and generate alerts that can be used for notifying an operator or triggering automation workflows.

Challenge

Discovering and repairing problems in the network usually involves manual network operator intervention and is time-consuming and error prone.

Solution

Incorporating Cisco Crosswork Change Automation and Cisco Crosswork Health Insights into Cisco Crosswork Network Controller gives service providers the ability to automate the process of discovering and remediating problems in the network by allowing an operator to match an alarm to pre-defined remediation tasks. These tasks will be performed after a defined Key Performance Indicator (KPI) threshold has been breached. Remediation can be implemented with or without the network operator's approval, depending on the setting and preferences of the operator.

Using such closed-loop remediation reduces the time taken to discover and repair a problem while minimizing the risk of making a mistake and creating an additional error through high-stakes manual network operator intervention.

How Does it Work?

Smart Monitoring

- The Smart Monitoring feature helps operators collect, filter, and present the data in useable formats, such as graphs and tables. Operators can remain focused on their business goals while the configuration required for the data collection is done by the Cisco Crosswork Network Controller and Cisco Crosswork Change Automation and Cisco Crosswork Health Insights using the feature Zero-touch telemetry.

- By using a common collector to collect network device data over SNMP, CLI, and model-driven telemetry, and making it available as modelled data described in YANG, duplicate data collection is avoided, optimizing the load on both the devices and the network.
- Recommendation Engine analyzes network device hardware and software, configuration, and employs a pre-trained model built from data mining, producing KPI relevant recommendations facilitating per use-case monitoring.
- KPIs cover a wide range of statistics from CPU, memory, disk, layer 1/2/3 network counters, to per protocol, LPTS and ASIC statistics.

Smart Filtering

- Cisco Crosswork Health Insights builds dynamic detection and analytics modules that allow operators to monitor and see alerts on network events based on user-defined logic (KPI).
- Key Performance Indicators (KPIs) Alerting Logic can be:
    - Simple static thresholds (TCA); e.g., CPU load above 90 percent.
    - Moving average, standard-deviation, and percentile based, etc., e.g., CPU load above mean and staying there for five minutes.
    - Streaming jobs which provide real-time alerts or batch jobs which run periodically.
    - Customized for threshold values and visualization dashboards.
    - Customized operator-created KPIs based on business logic.
    - TCAs can be exported or integrated with other systems via HTTP, Slack, and socket interfaces.
- KPIs are associated with dashboards, which provide real-time and historical views of the data and corresponding TCAs.
- KPIs also provide purpose-built dashboards that go beyond raw data and provide valuable information in various infographic style charts and graphs useful for triaging and root-causing complex issues.

Smart Remediation

- Health Insights KPIs can be associated with Cisco Crosswork Change Automation (CCCA) playbooks, which can be either executed manually or via auto-remediation. Remediation workflow could be used to fix the issue or collect more data from the network devices. By proactively remediating the situation, instead of resorting to ad hoc debugging and unscheduled downtime, operators can save time and money, providing better QOE to their customers.
- Health Insights does the correlation of alerts or anomalies on the topology of the network, allowing easy visualization of the impact of events.

Scenario 9 – Achieve Predictive Traffic Load Balancing Using Segment Routing Affinity

Scenario Context

To maintain smooth and optimal traffic flow, operators need to be able to monitor traffic on the interfaces, identify errors such as CRC, watchdog, overrun, and then reroute the traffic so that the SLA is maintained. This process can be automated using Cisco Crosswork Network Controller with Cisco Crosswork Health Insights and Cisco Crosswork Change Automation applications installed.

Assumptions and Prerequisites

Cisco Crosswork Health Insights and Cisco Crosswork Change Automation must be installed and running.

Workflow

The following is a high-level workflow for executing this scenario:

Step 1 Deploy Day0 ODN templates on edge nodes with dynamic path calculation delegated to SR-PCE and the ODN template configured to exclude links that are tagged with a specific affinity; for example, RED affinity. ODN allows a service head-end router to automatically instantiate an SR-TE policy to a BGP next-hop when required (on-demand). The ODN template defines the required SLA using a specific color.

For an example procedure for creating an ODN template, refer to Step 1 Create an ODN template to map color to an SLA objective and constraints, on page 25 in Scenario 1 – Implement and Maintain SLA for an L3VPN Service for SR-MPLS (using ODN).

Step 2 Create an L3VPN route policy to specify the prefixes to which the SLA applies and mark them with the same color used in the ODN template. When traffic from the specified network with a matching color is received, paths are computed based on the SLA defined in the ODN template.

For an example procedure for creating a route policy, refer to Step 1 Create an ODN template to map color to an SLA objective and constraints, on page 25.

Step 3 Provision an L3VPN across the required endpoints and create an association between the VPN and the route policy. This makes the connection between the VPN and the ODN template that defines the SLA.

For an example procedure for provisioning an L3VPN, refer to Step 3 Create and provision the L3VPN service , on page 29.

Step 4 Define and enable the KPIs on the devices. This will continuously monitor the uplink interfaces on the L3VPN endpoints.

For information about defining KPIs, see the Cisco Crosswork Change Automation and Health Insights User Guide.

Step 5 When there is an error on monitored interfaces, mark the dirty link with RED affinity so that it will be excluded, based on the specifications of the ODN template. This is achieved by creating a custom playbook. Cisco Crosswork Network Controller learns the name of the interface generating the alert regarding the error and this is fed into the custom playbook so that the affinity configuration can be pushed to the relevant router, forming a closed-loop automation scenario. In this way, the customer should not experience outages.

For information about defining playbooks, see the Cisco Crosswork Change Automation and Health Insights User Guide.

Step 6 Cisco Crosswork Network Controller continues to monitor the link and when there are no longer alerts, the RED affinity tag can be removed. Define another playbook for this purpose.

CHAPTER 7

Automation of Onboarding and Provisioning of IOS-XR Devices Using ZTP

This section explains the following topics:

Overview

Objective

Allow users to quickly, easily, and automatically onboard new devices and provision them using a Cisco-certified software image and a day-zero software configuration.

Challenge

Deploying and configuring network devices is a tedious task. It requires extensive hands-on provisioning and configuration by knowledgeable personnel, which is time-consuming, expensive, and error-prone.

Solution

Automate onboarding of new devices using Crosswork Zero Touch Provisioning (Cisco Crosswork ZTP). Cisco Crosswork ZTP allows users to provision networking devices remotely, without a trained specialist on site. After establishing an entry for the device in the DHCP server and the ZTP application, all the operator needs to do is connect the device to the network, power on and press reset to activate the devices. A certified image and configuration are downloaded and automatically applied to the device. After it is provisioned in this way, the new device is onboarded to the Crosswork device inventory where it can be monitored and managed like other devices.

How Does it Work?

- Classic ZTP: The DHCP server verifies the device's identity based on the device's serial number, then offers downloads of the boot file and image. After the device is imaged, it downloads the configuration file and executes it.
- Secure ZTP: The device and the Cisco Crosswork ZTP bootstrap server authenticate each other using the device's Secure Unique Device Identifier (SUDI) and Crosswork server certificates over TLS/HTTPS. After a secure HTTPS channel is established, the Crosswork bootstrap server allows the device to request to download and apply a set of signed image and configuration artifacts adhering to the RFC 8572 YANG

schema. After the image (if any) is downloaded and installed, and the device reloads with the new image, the device downloads configuration scripts and executes them.

- Plug and Play (PnP) ZTP: The Cisco PnP agent on the IOS-XE device and the Cisco Crosswork PnP Server authenticate each other over HTTP using a PnP profile supplied on a TFTP server. They then establish a secure connection over HTTPS and the PnP agent downloads and installs image (optional) and configuration artifacts.

Additional Resources

Detailed information is available in the ZTP chapter in the Cisco Crosswork Network Controller Infrastructure 5.0 and Applications Administration Guide.

Scenario 10 - Automatically onboard and provision new devices in the network

Scenario Context

With the exponential growth of service provider networks and their rapid expansion into new customer sites and new locations, there is a need to connect an ever-increasing number of edge devices. At the same time, functional sophistication is increasing, requiring more time to configure those devices and activate new services. Manual processes limit a service provider’s ability to rapidly scale networks and roll out new services in a cost-efficient way.

In this scenario, we will onboard the new IOS-XR devices required to set up a new customer site in a remote location and go live, without the need to send skilled technicians on time-consuming and costly on-site visits to complete the provisioning.

We will leverage the configuration of devices at existing customer sites that are already set up and operating to ensure that the Day0 configuration of the new devices includes whatever is necessary to get the devices up and running quickly and efficiently.

Assumptions and Prerequisites

- Crosswork ZTP must be installed in your Cisco Crosswork Network Controller setup.
- For Classic ZTP, Crosswork and the devices must be deployed in a secure network domain. Secure ZTP does not have this requirement; it is secure across networks.
- The Crosswork server must be reachable from the devices, via an out-of-band management network or an in-band data network.
- If you want to onboard devices to Cisco NSO also, Cisco NSO must be configured as a Crosswork provider. When configuring the NSO provider, be sure to set the provider property key to *forward* and the property value to *true*.

Workflow

This is a high-level workflow for onboarding IOS-XR devices using Cisco Crosswork Classic or Secure ZTP.

To onboard IOS-XE devices, or for more detailed information on these options, see the ZTP chapter in the Cisco Crosswork Network Controller Infrastructure 5.0 and Applications Administration Guide.

- Step 1. Assemble and upload ZTP assets

- Step 2. Create a ZTP profile combining an image file and configuration file
- Step 3. Prepare ZTP device entries for the devices to be onboarded
- Step 4. Set up DHCP for Crosswork ZTP
- Step 5. Initiate ZTP processing to onboard the devices
- Step 6. Monitor the ZTP processing status
- Step 7. Verify your onboarded devices

Workflow

Step 1 Assemble and upload ZTP assets

a) Assemble the following assets before you begin:

- (Optional) Software images. For Classic ZTP, you can use Cisco IOS-XR versions 6.6.3, 7.0.1, 7.0.2, 7.0.12, and 7.3.1 or later. For Secure ZTP, use Cisco IOS-XR 7.3.1 or later (except 7.3.2 and 7.4.1).
- Configuration Files: SH, PY, or TXT files. You can specify up to three different configuration files for Secure ZTP.
- Credentials of the devices to be onboarded
- Serial numbers of the devices to be onboarded

For Secure ZTP only, also assemble:

- Owner certificates - your organization's CA-signed end-entity certificates, installed on your devices and binding a public key to your organization.
- Pinned domain certificate - your organization's CA- or self-signed domain certificate, with its public key pinned to your organization's DNS network domain. The PDC helps your devices verify that images and configurations downloaded and applied during ZTP processing come from within your organization.
- Ownership vouchers - Nonceless audit vouchers that verify that devices being onboarded with ZTP are bootstrapping into a domain owned by your organization. Cisco supplies OVs when a request is submitted with your organization's PDC and device serial numbers.

b) If applying software images: Upload the software images. Go to **Device Management > Software Images**.

c) Upload the configuration files. Go to **Device Management > ZTP Configuration Files**.

d) Upload device serial numbers. Go to **Device Management > Serial Number and Voucher** and click **Add Serial Number**.

e) For Secure ZTP, upload your pinned domain certificate and owner certificates. Go to **Administration** > **Certificate Management** and add your certificates.

f) For Secure ZTP, upload ownership vouchers. Go to **Device Manager > Serial Number and Voucher**.

Step 2 Create a ZTP profile combining an image file and configuration file

Crosswork uses ZTP profiles to automate imaging and configuration processes. While optional, creating ZTP profiles is recommended as the best way to combine a single image file and configuration file based on a product or device family, such as the Cisco ASR 9000 or Cisco NCS5500. We recommend that you create only one day-zero ZTP profile for each device family, use case or role the devices serve in the network.

To create ZTP profiles, go to **Device Management** > **ZTP Profiles**.

Step 3 Prepare ZTP device entries for the devices to be onboarded

Depending on how many devices you are onboarding, you can either prepare and import a CSV file or you can create device entries individually.

a. Go to **Device Management** > **Devices**.

b. Click the **Zero Touch Devices** tab. Then:

- To create a device entry file for many devices, click the **Import** icon and download the CSV template. Edit the template and add entries for each device you want to onboard. See the ZTP chapter for details on the file entries. Then click the **Import** icon again to import your device entry file.
- To create device entries one at a time, click the **Add** icon.

Step 4 Set up DHCP for Crosswork ZTP

Before triggering ZTP processing, you must update your organization's DHCP server configuration file with the IDs for your ZTP device entries and the paths to the image and configuration files stored in the ZTP repository. This allows Crosswork and DHCP to identify these ZTP devices and to respond correctly to each device's requests for connection to the network, and to download image and configuration files. For sample DHCP entries, see the ZTP chapter.

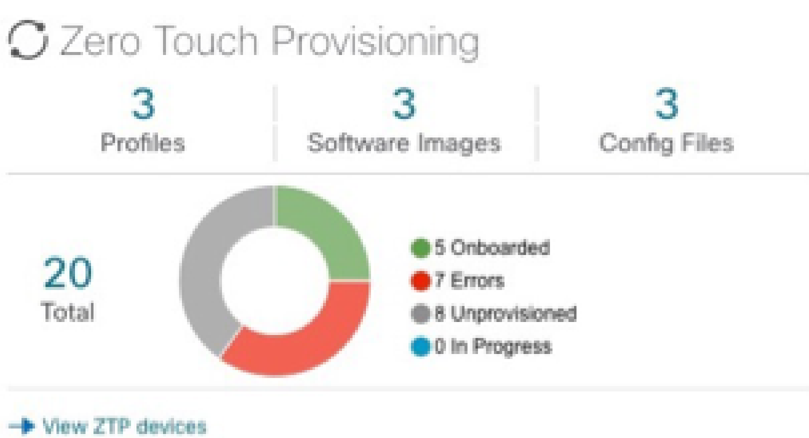
Step 5 Initiate ZTP processing to onboard the devices

Initiate ZTP processing by rebooting each of the devices to be provisioned: Power-cycle it, or press the chassis reset button.

Step 6 Monitor the ZTP processing status

You can monitor the progress of ZTP processing in the dashboard.

a. Click **Home** in the main menu and take a look at the Zero Touch Provisioning dashlet.



b. Click on the **View ZTP devices** link to view the status of individual devices.

Step 7 Verify your onboarded devices

Go to **Device Management > Devices**. Click the **Zero Touch Devices** tab. All of your onboard devices should be listed.

You may need to edit the information for some devices. Some of the information needed for a complete device record either is not needed in order to onboard the device, or not directly available through automation. For example, geographical location data defined using a set of GPS coordinates.

ZTP devices, after being onboarded, are automatically part of the shared Crosswork device inventory. You can edit them like any other device.

CHAPTER 8

Visualization of Native SR Path

This section explains the following topics:

Overview

Objective

Visualize the actual path traffic flows physically through the topology map, even if traffic is on a native SR IGP path (not SR-policy) over inter-AS option C.

Challenge

Visualizing the native SR IGP path is often an operational challenge. Without access to a streamlined and simple to use interface, diagnosing and troubleshooting the native path requires you to repeatedly login to network devices without a solution to improve efficiency.

Solution

With the Path Query option, the objective is to visualize the native path using the traceroute SR-MPLS multipath command to get the actual paths between the source and the destination. With Cisco Crosswork Network Controller, a traceroute command runs on the source device for the destination TE-router ID and helps in retrieving the paths. By using native gRPC calls from the Crosswork server, you are able to get the paths from the device which assist in visualizing the native path through which the traffic flows. Since the traceroute command results in an operation that might take time to converge, Cisco Crosswork Network Controller provides an asynchronous user experience where you can send a request for such an operation and then be notified when the output is ready for inspection.

How Does it Work?

- Create a new path query, defining the headend and endpoint devices to find the available Native SR IGP paths.
- Visualize the available Native SR IGP paths as defined by the query on the topology map.
- Inspect the available paths and review the Output, Nexthop, Source, Destination, and Hop Index information.

- Create additional path queries as needed based on service type and instance and visualize the paths on the topology map.
- Troubleshoot any failed path queries.

Scenario 11 –Troubleshooting paths between native SR paths over inter-AS Option C

Scenario Context

Visualization of the path traffic flows is not readily available without manual tasks from different sources. Once attaining traffic flow paths, the data is often out of date. Cisco Crosswork Network Controller supports the creation of Path Queries, which you define within the GUI. This allows visualization of actual SR IGP paths between the source destination on a topology map. Cisco Crosswork Network Controller provides an asynchronous user experience where the user is notified when results are ready for inspection. This facilitates rapid troubleshooting for issues with native traffic flows.

A**ssumptions and Prerequisites**

- The device should have IOS XR version 7.3.2.
- The device should have gRPC (Remote Procedure Call) enabled. To check, run “show grpc” the in device and follow these steps:
  - For gRPC without a secure connection: If gRPC is showing as not enabled, enable gRPC using the following commands: configure terminal; grpc; no-tls.
  - For gRPC with a secure connection: Upload security certificates to Cisco Crosswork Network Controller in order to connect to the device using the following commands: configure terminal; grpc.
- Cisco Crosswork Optimization Engine server should have the devices imported with gNMI (Network Management Interface) capability and gNMI connectivity for the devices.
  - Make sure the credential profiles include connectivity information for gNMI. Go to **Device Management** > **Credential Profiles**. The Credential Profiles screen appears. Select a profile to edit. On the Edit Profile Devices screen, click **+ Add Another**. For Connectivity Type, select **GNMI**. Add the User Name, Password, and Confirm Password information. Click **Save**.
  - Devices should have gNMI capability enabled in Cisco Crosswork Network Controller while attaching the device. Go to **Device Management** > **Network Devices**. Select the device to edit. The Edit Device Details screen appears. From the required Capability list, select **GNMI**. Click **Save**.
  - Devices should have the gNMI connectivity information enabled. Go to **Device Management > Network Devices**. Select the device to edit. On the Edit Device Details screen, under Connectivity Details, click **+ Add Another**. For Protocol, select **GNMI** and add the IP Address / Subnet Mask information. Type the Port information and for Encoding Type, select **JSON**. Click **Save**.

Workflow

Step 1 Select **Services & Traffic Engineering** > **Path Query**. The Path Query dashboard appears.

Step 2 Click **New Query**. The New Path Query panel appears on the right with the mapped Device Groups panel on the left.

Step 3 Enter the device information in the required fields to find available Native SR IG Paths.

a. Select the Headend device from the list. For this example, select **P-Edge-A1**.

b. Select the Endpoint device from the list. For this example, select **P-Edge-B2**.

Step 4 Click **Get Paths**. The Running Query ID pop-up appears.

Note Path queries may take a moment to complete. When the Running Query ID pop-up appears, you can also select **View Past Queries** to return to the Path Query Dashboard. If you already had path queries in the list, you can view existing details as the new query continues to run in the background, which is indicated by the blue Running icon in the Query State column. When the new query state turn green, completed, it can be viewed.

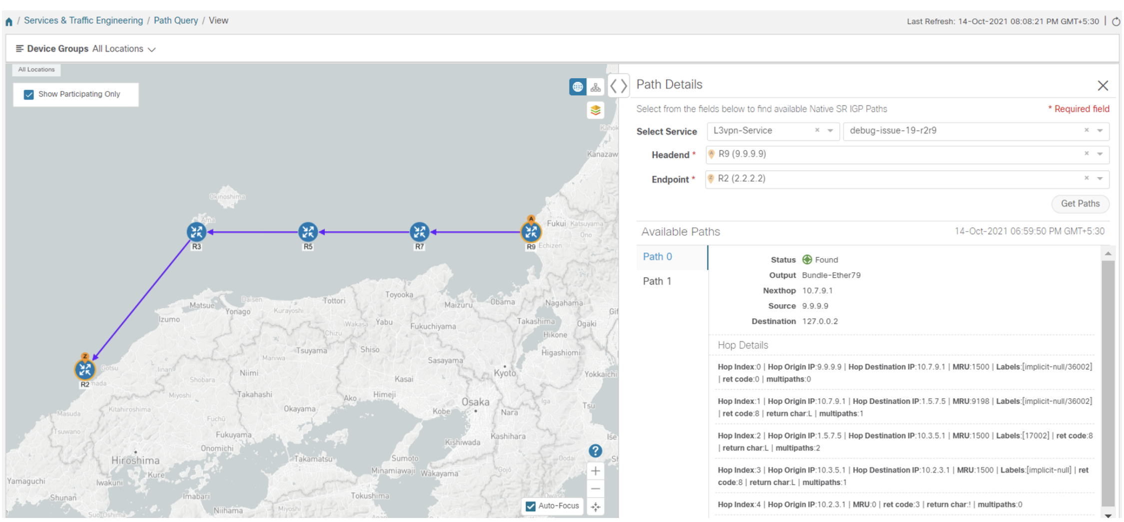
Step 5 Click **View Results** when it becomes available on the Running Query ID pop-up. The Path Details panel appears with corresponding Available Paths details while the defined topology map appears with the available Native SR IG Paths on the left.

Step 6 Click on the Available Paths options (for example, **Path 0** and **Path 1**) to review Status details for Output, Nexthop, Source, Destination, and Hop Index information. When you select one of the available paths, the map will update with the corresponding Device Groups topology mapping of Path 0 and Path 1.

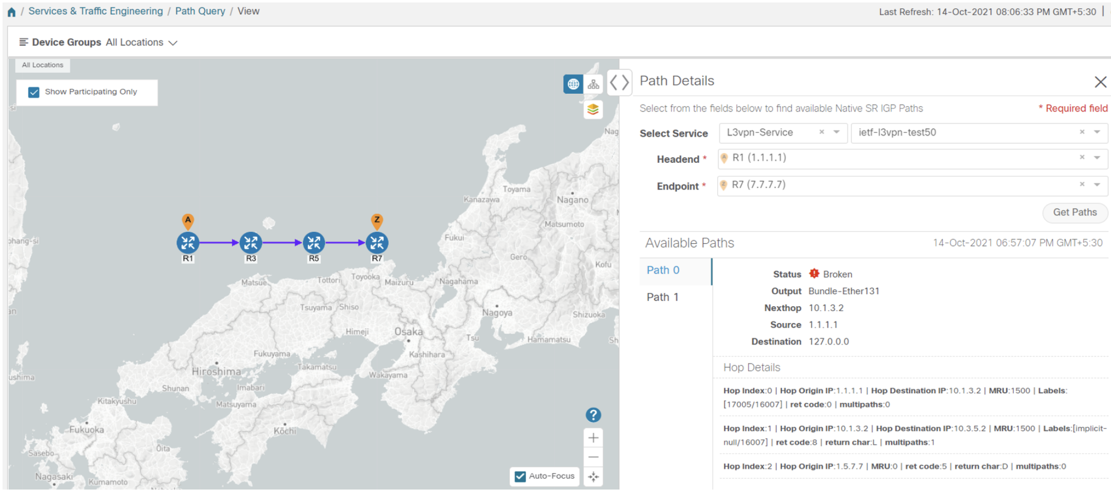
Note Ensure that the **Show Participating Only** check box is selected in the top-right corner of the map.

Note There are three likely status outcomes to a path query. The screen captures below are independent examples not directly associated with the scenario's workflow:

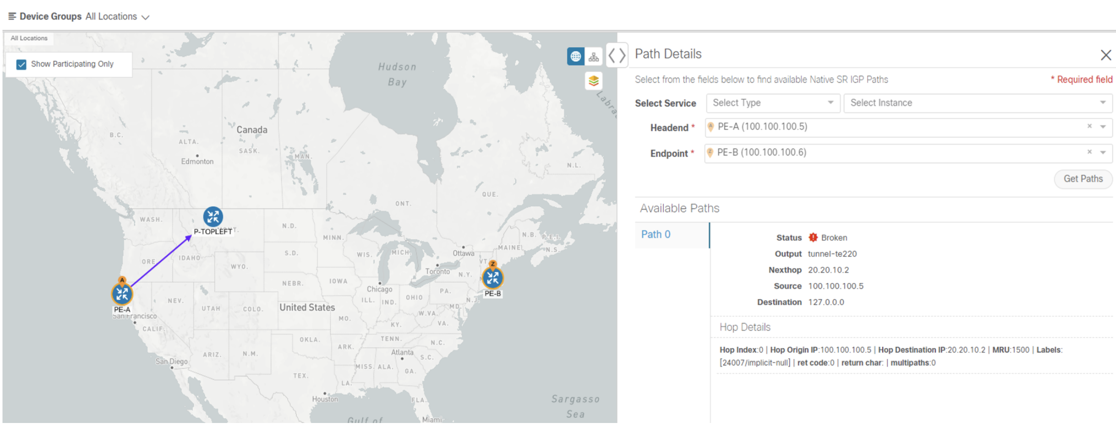
a. **Non-Broken Path (path is complete):** Path Status shows as **Found** with path hop details and overlay shown.

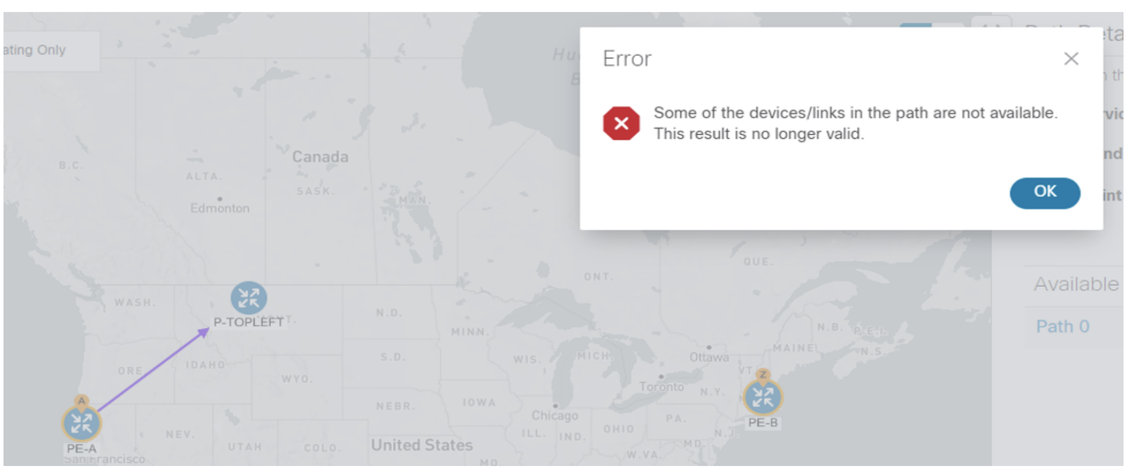


b. **Broken Path (Path is complete):** Path Status shows as Broken with path hop details and overlay shown.



c. **Broken Path (Path is not complete):** Path Status shows as Broken with path hop details partially shown (depending on gNMI output for traceroute – see Step 17 for troubleshooting details) and overlay details partially shown. An Error message will appear indicating that the devices and links are not available.





Step 7 Select **Services & Traffic Engineering** > **Path Query** to return to the Path Query Dashboard.

Step 8 Ensure that the new path Query State column shows as completed with a green icon. The new path in the table will also show a Query ID link, both the corresponding Headend and destination Endpoint, and the Available Paths column will show 2 for both paths.

If a query state is broken, see the last step in the workflow for troubleshooting details.

Step 9 As needed, click on the **Query ID** link or click and select **View Details** to again review the Path Details panel and map.

Step 10 Create additional path queries by selecting **Services & Traffic Engineering > Path Query**. The Path Query Dashboard appears where the previous path queries are listed by Query ID.

> **Note** Make sure to set the **Automatically delete query older than every** < X > option within the number of hours needed from the Path Query Dashboard. The maximum number of hours provided is **24**.

Step 11 Click **New Query**. The New Path Query panel appears on the right with the mapped Device Groups panel on the left.

Step 12 For Select Service, select the Type from the list. In this example, select **L2VPN-SERVICE**.

By utilizing Select Service, when you later select the Headend and Endpoint, the options are conveniently identified according to the relevant VPN service type.

Step 13 For Select Service, select the Instance from the list. In this example, select **L2VPN_NM_P2P-NATIVE-210**.

The topology map will update to show the path between both servers. In this example, **P-Edge-B2** and **P-Edge-C3** are isolated on the map showing the logical path.

Step 14 Select the following from the list:

a. Headend: **P-Edge-B2**.

b. Endpoint: **P-Edge-C3**.

Step 15 Click **Get Paths**.

The Running Query ID pop-up appears.

Step 16 Click **View Results** when it becomes available. The Path Details panel appears with the corresponding Available Paths details, while the defined topology map appears with the available Native SR IG Paths on the left. This view shows the actual, physical hops between B2 and C3 that is carrying the traffic.

Step 17 To troubleshoot any Failed path queries appearing in the Path Query Dashboard's Query State column, select the "I" icon for error details.

In this example, the gNMI protocol is missing from the Connectivity Details for a previous path query with the Headend P-BOTTOMLEFT device and the Endpoint P-BOTTOMRIGHT devices. To troubleshoot the failed path query, do the following:

a. Select Device Management > Network Devices.

b. Find the device by Host Name and select the check box.

c. Click the Edit icon at the top of the table. The Edit Device Details pop-up appears.

d. In this example, the Connectivity Details for Protocol is missing gNMI. Click **+ Add Another** and type GNMI until it appears in the list. Select it.

e. Enter the IP Address / Subnet Mask information and Port field information.

f. Enter the Timeout field as **30**.

g. In the Endcoding Type list, type **JSON** until it appears in the list. Select it and click **Save**.

h. Select Services & Traffic Engineering > Path Query. The Path Query Dashboard appears.

i. Click **New Query**. The New Path Query panel appears.

j. Select the following from the list:

   1. Headend device: **P-BOTTOMLEFT**.
   2. Endpoint device: **P-BOTTOMRIGHT**.

k. Click **Get Paths.**The Running Query ID pop-up appears.

l. Click **View Results** when it becomes available. The Path Details panel appears with corresponding Available Paths details, while the defined topology map appears with the available Native SR IG Paths on the left and is now in a Completed state.

CHAPTER 9

Provision, Visualize, and Analyze Tree Segment Identifier Policies in Multipath Networks

This section explains the following topics:

Overview

Allow users to provision and visualize Tree Segment Identifier (Tree-SID) Segment Routing policies easily and quickly before associating the policies with an L3VPN service model.

Objective

To provision, visualize, and update static Tree-SID policies within your network using Crosswork Network Controller and associate the (mVPN) policies with an L3VPN service model. By provisioning the Tree-SID policies using the Crosswork Network Controller UI and both visualizing and analyzing the multicast paths, root and leaf nodes, transit nodes, and view information about each link among the nodes, provides a holistic view of creating, visualizing, updating, and maintaining point-to-multipoint (P2MP) network configurations. These static Tree-SID policies can now be associated with an L3VPN service model and visualized and edited, as needed, using the Crosswork Network Controller UI.

Challenge

Keeping track of SR PCE and PE paths within networks is a challenge for video broadcasting and streaming service providers, who must use multipath protocols to replicate traffic and send it to different points in the network. Ensuring a high level of service quality forces providers to use difficult manual approaches to visualize, update and maintain their point-to-multipoint (P2MP) network configurations. This slows response to network problems and increases costs.

Solution

Tree-SID is a method of implementing tree-like multicast flows over a segmented routing network. Using Tree-SID, an SDN controller (a device running SR-PCE using PCEP), calculates the tree. Each node (device) in the tree has a specific role in routing the multicast data through the tree. These roles include Ingress for the root or headend node, Transit or Bud for midpoint nodes that are not leaf nodes, and Egress for destination leaf nodes. The tree itself is assigned a single SID label, which represents all of the tree segments and devices in it. The SDN controller is highly flexible, as it understands the segmentation and can construct routing paths using any kind of constraints that network architects can specify.

The most interesting use case for constraint-based Tree-SID use is where routers are configured to deliver two P2MP streams with the same content over different paths. Here, the multicast flow is forwarded twice through the network, each copy following a unique path. The two copies never use the same node or link to reach the destination, reducing packet loss due to network failures on any one of the paths.

By using Crosswork Network Controller, you can now create Static Tree-SID policies using the UI, associate Static mVPN Tree-SID policies with a provisioned L3VPN service, visualize, analyze, and edit or delete your Tree-SID policies to actively manage your multicast network.

Note Static and Dynamic mVPN Tree-SID policies can be associated with a L3VPN service model. In this workflow tutorial, only a Static mVPN Tree-SID policy will be associated with a L3VPN service model, visualized, and analyzed.

How Does it Work?

- Create a Static Tree-SID policy using the Crosswork Network Controller UI
- Visualize and validate the new Static Tree-SID policy
- Associate your Static mVPN Tree-SID policy with an L3VPN service model (or import an existing static or dynamic Tree-SID policy)
- Visualize and analyze the performance details of your Static mVPN Tree-SID paths and nodes within the L3VPN service model
- Edit your existing Static mVPN Tree-SID policy to enhance performance or correct issues with your Tree-SID L3VPN service model

Scenario 12 – Provisioning, Visualizing, and Analyzing Tree Segment Identifier Policies in a Point-to-Multipoint L3VPN Service

Scenario Context

Without Crosswork Network Controller, provisioning and visualizing Tree-SID point-to-multipoint traffic flows is available only using manual tasks from different sources. Restriction to manual tasks means that the creation of Tree-SID policies, associating a policy with an L3VPN service model, visualization, and editing of the policy and/or service is significantly hampered. By using Crosswork Network Controller, you can sidestep the time loss between manual setup and the visualization of the traffic flow paths and avoid data that is often out of date with manual configurations. Crosswork Network Controller supports both creation and discovery of the Tree-SID segmentation directly from network configurations and displays the data flow map immediately. This facilitates rapid troubleshooting for issues with Tree-SID traffic flows.

Crosswork Network Controller’s topology services uses PCE topology and LSP data to discover and visualize pre-configured Tree-SID policies in your network. The PCE controller manages the layer 3 topology, LSP and Tree-SID data using BGP link state, and supports initial discovery and notifications for the Tree-SID trees. Static Tree-SID policies can also be configured and later associated with newly created, or previously configured, L3VPN services directly in the Crosswork Network Controller's UI. Likewise, based on the health of the service and policies, editing capabilities are also performed using the UI to troubleshoot and optimize models operations.

Assumptions and Prerequisites

If your network has PCE and Tree-SID policies already configured on your devices, this workflow assumes, at a minimum, the following basic configuration options:

1. On all nodes involved in the Tree-SID path, irrespective of role:
   a. Enable Path Computation Element Protocol (PCEP)
   b. Enable Computation Client (PCC)
2. Under SR-PCE, on end points: Configure a P2MP SR static or dynamic Policy.
3. On all root and leaf nodes:
   - Enable multicast routing
   - Configure `interface vrf` *<vrf-number>*
   - Configure `router bgp` on topo nodes and PCE. On corresponding neighbors between PCE and PCC nodes, mention the configured `interface vrf` *<vrf-number>*.
   - Configure `route-policy` *<vrf-number>* and `PASS_ALL`
   - Under segment routing traffic engineering: Configure `ODN color <same as vrf-number>`
4. On all leaf nodes only: Configure router PIM, route-policy TREESID_CORE.

Step 1 Create a Static Tree-SID Policy

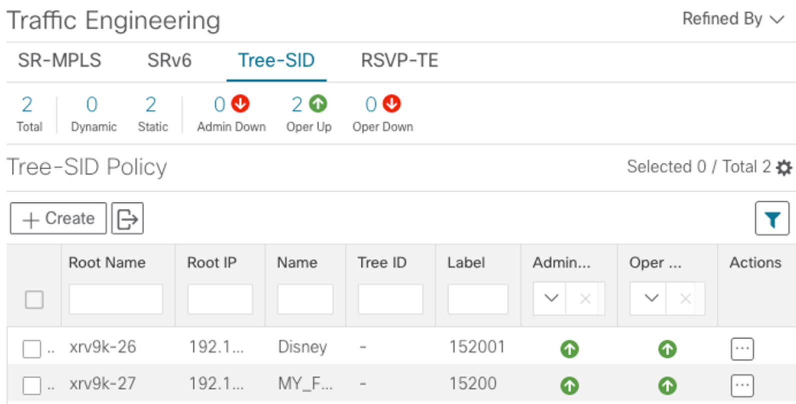
If you are using preconfigured Static or Dynamic Tree-SID policies already configured on your devices, skip to Step 2 in the workflow. If you are configuring Tree-SID policies using the Crosswork Network Controller's UI, this task first creates a Static Tree-SID policy, each representing a leaf or root node, before you have the option to associate the policies with a L3VPN service model that can be visualized and edited as necessary:

Step 1 Go to **Services & Traffic Engineering** > **Traffic Engineering**.

The logical map opens and the Traffic Engineering panel is displayed to the right of the map.

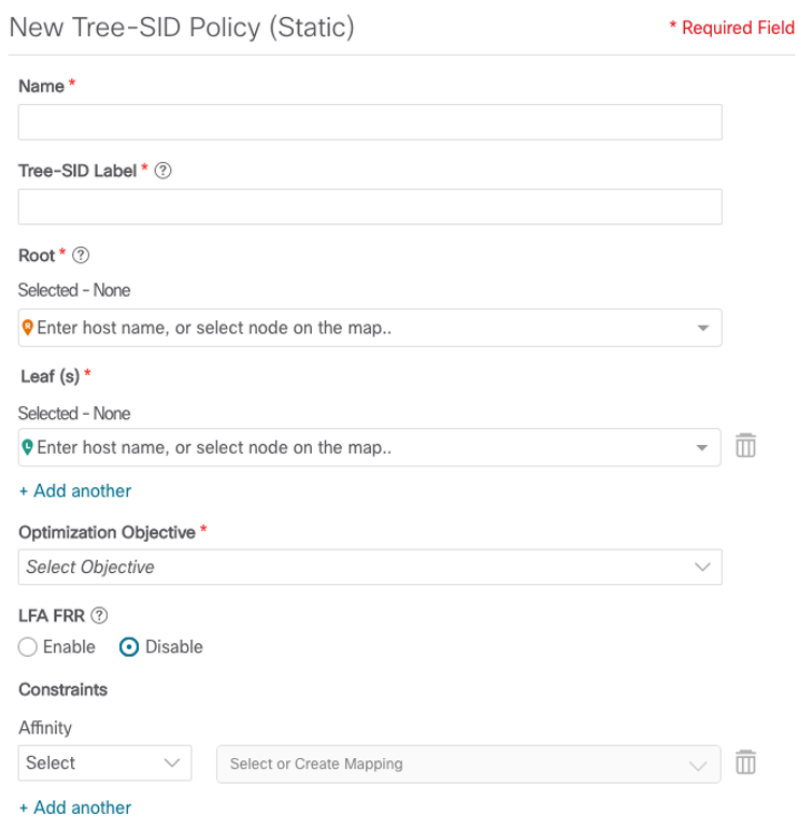
Step 2 In the Traffic Engineering panel, select the **Tree-SID** tab.

The Traffic Engineering Tree-SID Policy screen appears.



Step 3 Click **+ Create**.

The New Tree-SID Policy (Static) screen appears.



Step 4 To enter or select the required Static Tree-SID policy values, do the following:

a) After providing a name for your new Static Tree-SID policy, in the Tree-SID Label field, assign the MPLS label associated with the Tree-SID policy (for example: **152001**).

The Tree-SID Label must be in the range from 16 to 1048575.

b) In the Root field, enter the host name (for example: **xrv9k-26**) or select a node on the map or an existing device in the list. As you type or select the Root information, a Root label for the selected node appears on the map. Only PCC nodes with PCEP session to PCE can be added as a Root node.

c) In the Leaf field, enter the host name (for example: **xrv9k-24**) or select a node on the map. As you type, or select, the Leaf information, Leaf label(s) for the selected nodes appear on the map.

Click **+ Add another** to add additional constraints (for example: **xrv9k-27**) .

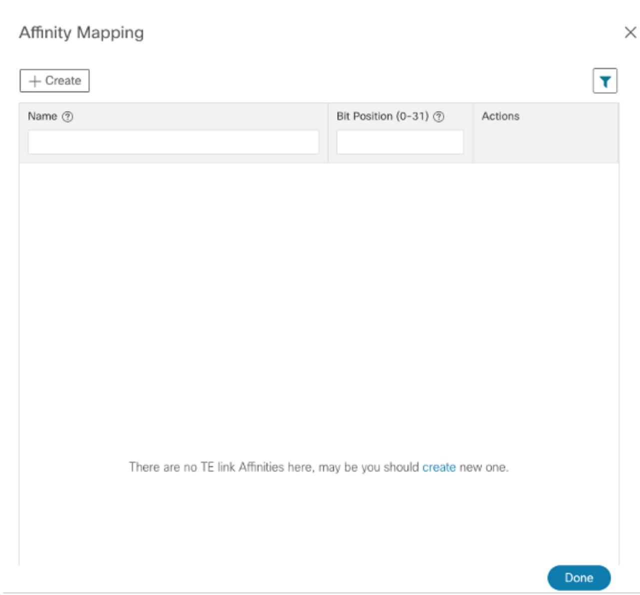
d) For Optimization Objective, select one of the following constraints: Interior Gateway Protocol (IGP) Metric; Traffic Engineering (TE) Metric; Latency (for example: **IGP**).

e) For LFA FRR, either select **Enable** or **Disable** (for example: **Enable**).

By selecting Enable, the Loop Free Alternate Fast Reroute (LFA FRR) is enable on all of the nodes in the distribution tree.

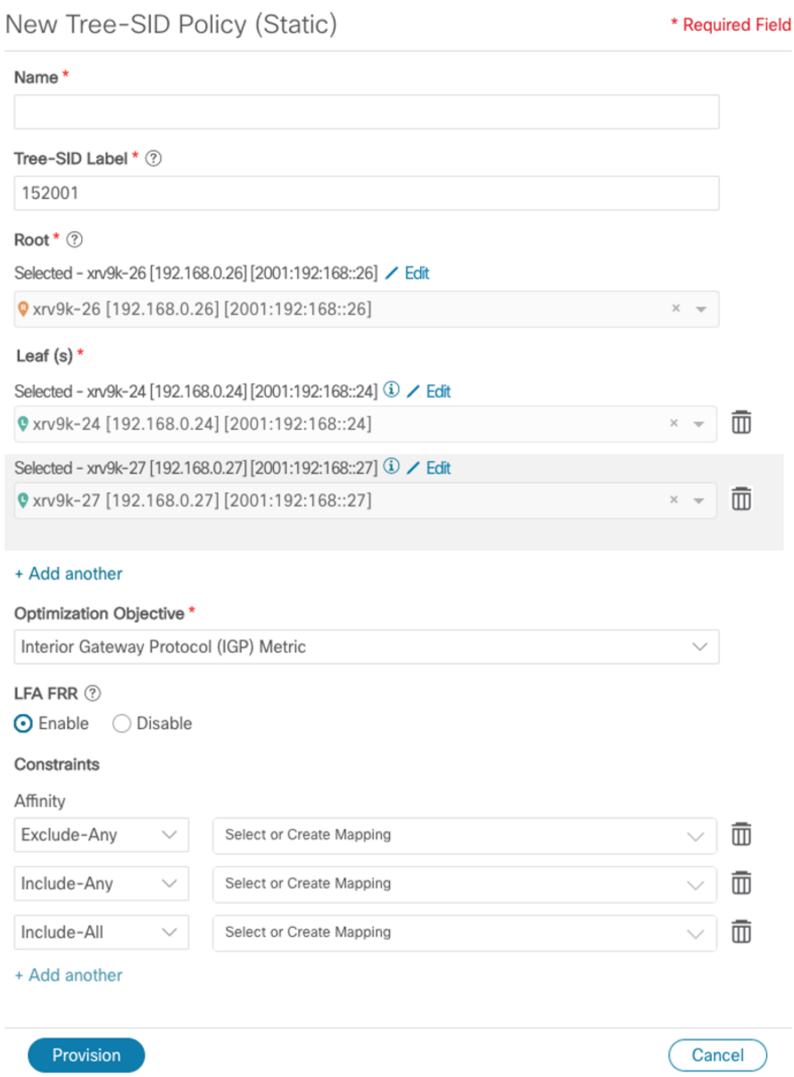
f) For additional Constraints, select one of the following Affinity options: **Exclude-Any**, **Include-Any**, **Include-All**.

In addition, from the Select or Create Mapping drop-down list, click **Manage Mapping**. The Affinity Mapping dialog box opens. For more information on Affinities, see the Configure Link Affinities section in the Crosswork Optimization Engine guide.



g) For Affinity Mapping, type a Name (color) of the mapping and enter the Bit Position (**0 – 31**). Enter the same bit position that is used on the device interface. Click **Done**.

To create additional constraints, click **+ Create**.

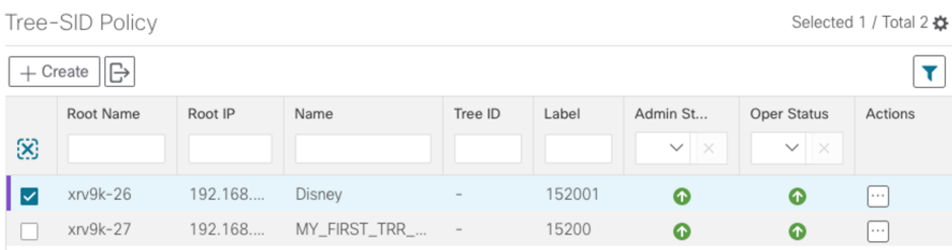


h) To commit the policy, click **Provision** to activate the policy on the network.

The newly provisioned Tree-SID policy may take some time to appear in the Tree-SID table depending on the network size and performance. The Tree-SID table is auto refreshed every 30 seconds. Once the request is successful, either select **View Tree-SID Policy List** or **Create New** to add additional policies. If you select **View Tree-SID Policy List**, the Tree-SID Policy screen appears showing the newly created policy in the table.

Step 2 Visualize and Validate the new Static Tree-SID policy

Step 1 Select the root Tree-SID policy check box from the list. In this example, select **xrv9k-26**.



If there is a large number of policies in the table, filter by Root IP, Name, Label, or other parameter, to help locate the policy you want to visualize.

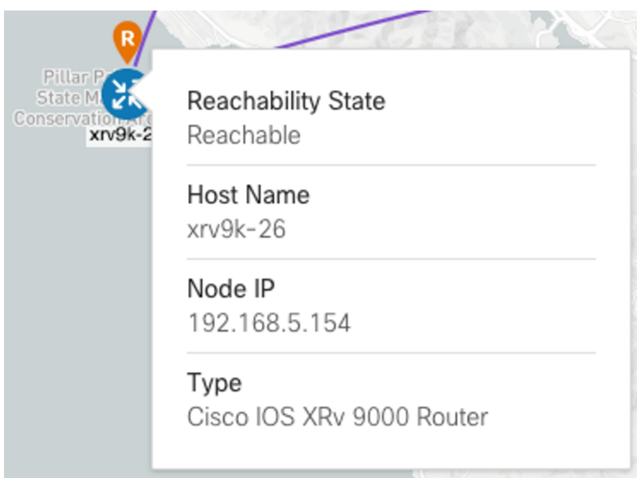
In the map, you will see the selected Tree-SID policy as an overlay on the topology. It shows a representation of the Tree-SID policy routes, with icon flags indicating the root node (**xrv9k-26**, also known as the ingress device) and the two leaf nodes (**xrv9k-24** and **xrv9k-27**, also known as egress devices), with intermediary transit nodes between them. Administrative and operational status for each node is shown in the table.

Note Use the buttons at the top right of the logical map to toggle between the Logical Map and the Geo Map views.

Step 2 Select the **Geo Map** button to view the selected Tree-SID service topology overlaid on a world map.

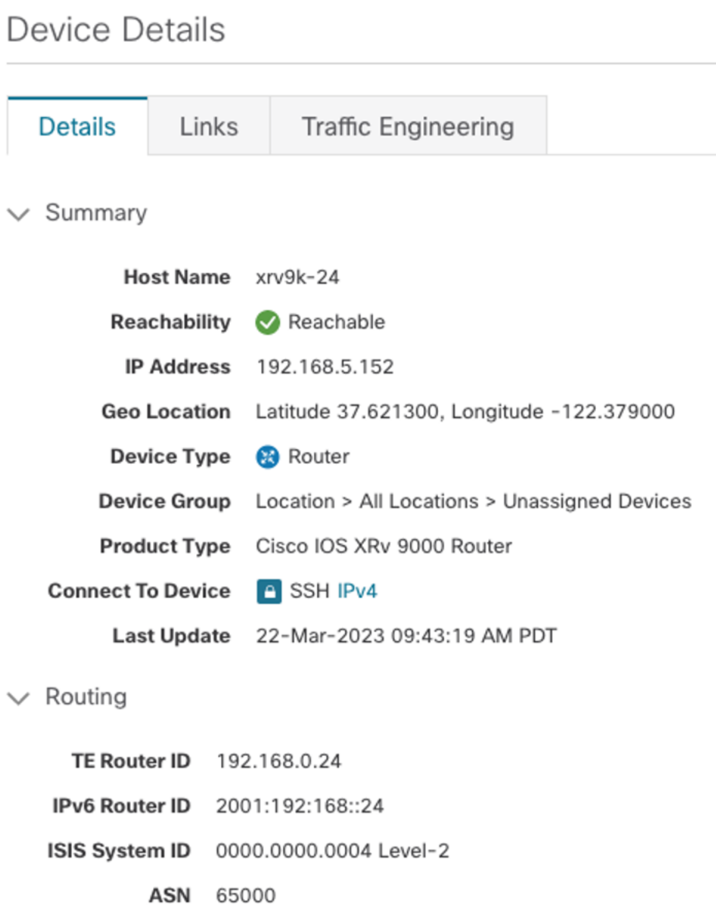
Step 3 In the map, select the **Show: Participating Only** check box to hide underlay devices that are not participating in the selected Tree-SID policy. Then use your mouse to hover over the **xrv9k-26** root device to view its corresponding Reachability State, Host Name, Node IP, and device Type.

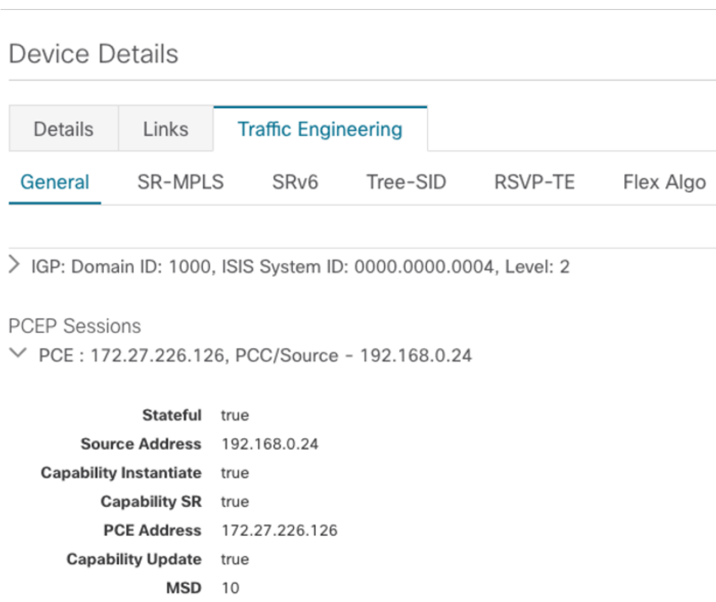
Check any participating Tree-SID device in the same fashion to view their corresponding details.



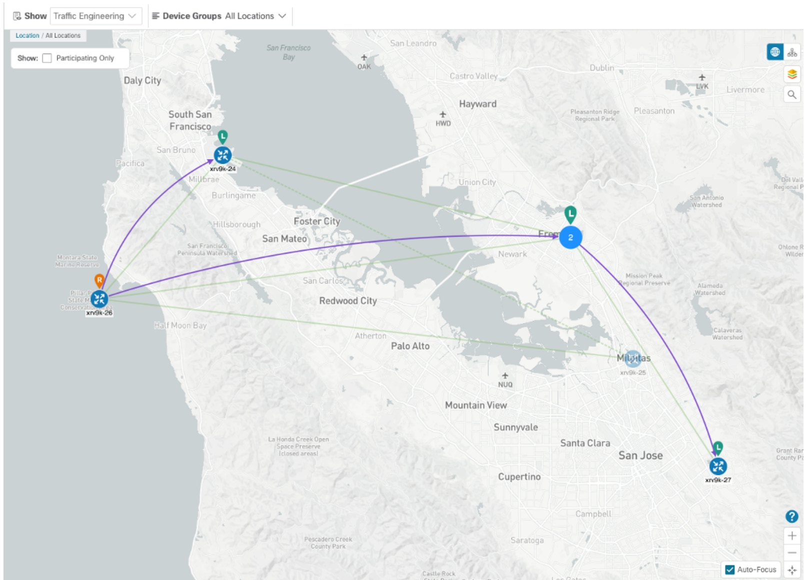
Step 4 In the map, click **xrv9k-24**.

The Device Details screen opens showing **xrv9k-24** information organized by Summary and Routing in the Details tab, and PCEP Sessions in the Traffic Engineering tab.



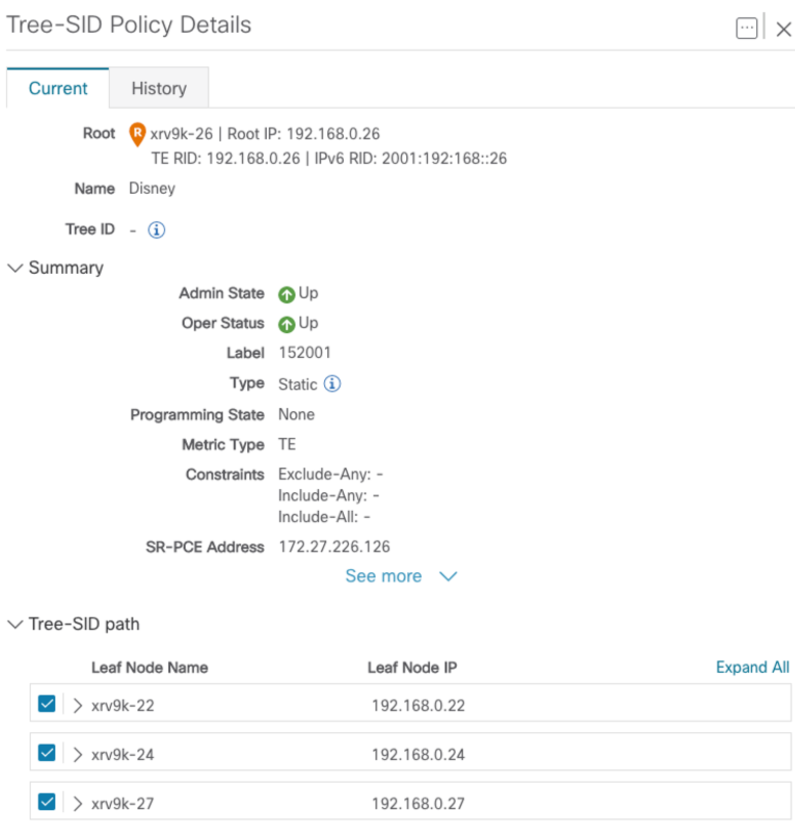


Step 5 Click **X** in the top-right corner to return to the Tree-SID Policy table to close the Device Details screen and then again select the **Tree-SID** tab.



Step 6 In the Tree-SID Policy list for the selected **xrv9k-26** device, click [⋯] in the **Actions** column and select **View Details** to drill down to a current and detailed view of the Tree-SID policy.

The Tree-SID Policy Details screen appears.

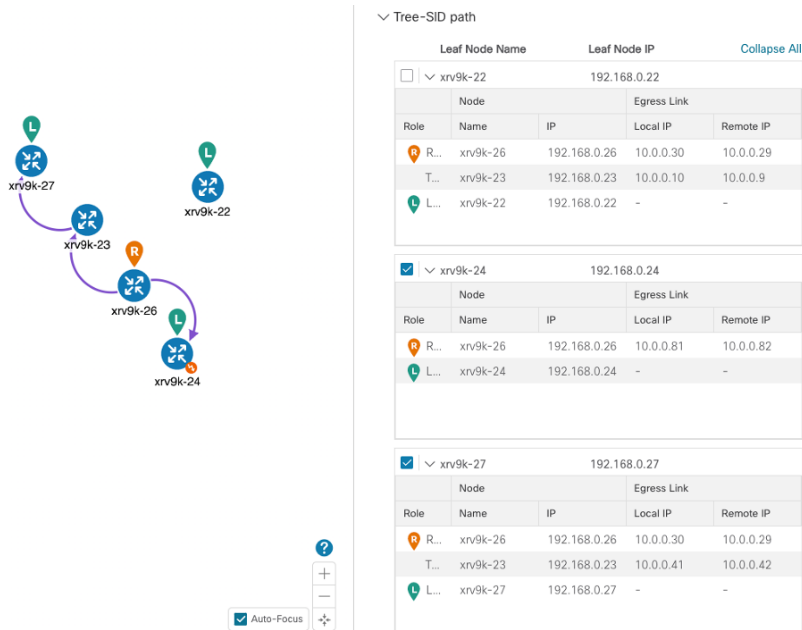


Note To view all of the Tree-SID Policy Details, click **See more**.

Step 7 In the Tree-SID path section, click **Expand All** to view Tree-SID path names and IPs for the **xrv9k-24** and **xrv9k-27** leaf nodes. The list also shows details for the corresponding Root node, all Transit nodes, the two Leaf nodes, and their Egress Link's Local IP and Remote IP information.

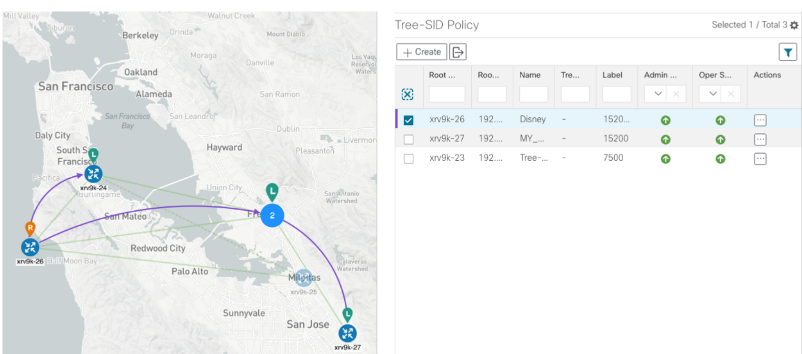
Step 8 Deselect the **xrv9k-22** check box to see Tree-SID path details for **xrv9k-24** and **xrv9k-27** devices only.

The topology updates to show only the selected **xrv9k-24** and **xrv9k-27** Tree-SID routes.



Step 9 Click **X** in the top-right corner to return to the Tree-SID Policy table.

Step 10 Select the Root IP Tree-SID policy **xrv9k-26** check box from the list. Make sure the geographical map option is selected. The geographical map updates to show the policy and its disjunct routes. You can click on individual links and get details on the Tree-SID policies in which each link participates.



Summary and Conclusion

As we observed, you can use the Tree-SID tab and its associated map to visualize Tree-SID defined routes, identify disjunct policy routes, and identify problems with transit nodes, interfaces and links that may affect traffic from the Root to the Leaf nodes.

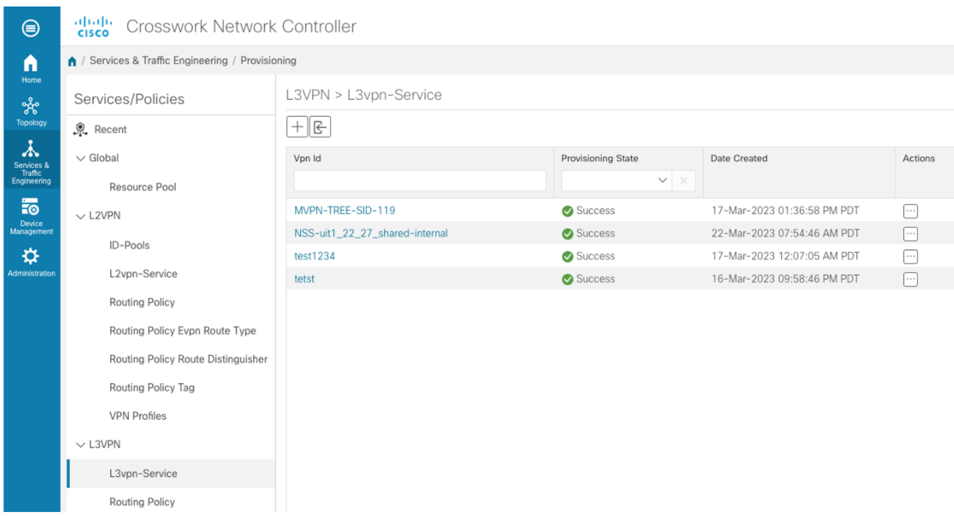
Step 3 Associate the Static Tree-SID Policy with the newly created L3VPN service model

Step 1 Go to **Services & Traffic Engineering** > **Provisioning (NSO)**.

The Provisioning screen appears showing available Services/Policies.

Step 2 Select **L3VPN** > **L3vpn-Service**.

The L3VPN > L3vpn-Service table appears.



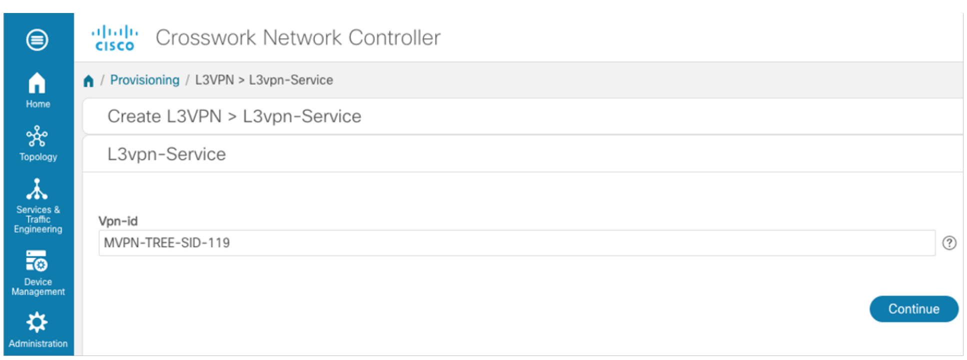
Step 3 To create a new L3vpn-Service, click the [+] symbol.

The Create L3VPN > L3vpn-Service screen appears.

Note You may also click the [import] symbol to import an existing L3vpn-Service.

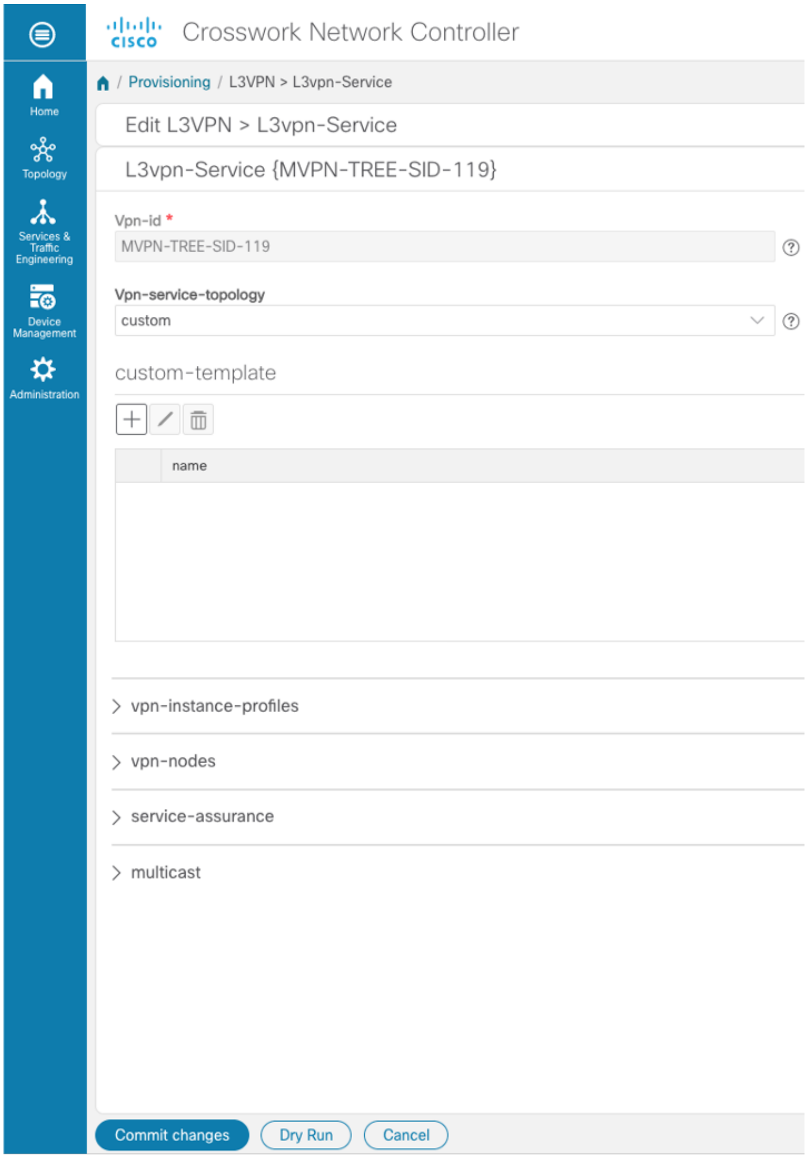
Step 4 In the Vpn-id field, type the unique ID for the service (for example: **MVPN-TREE-SID-119**) and click **Continue**.

Note This identifier has a local meaning (such as within a service provider network).



Step 5 In the Vpn-service-topology drop-down list, select **custom** to define the service topology.

Note Point-to-point VPN service topology is not supported.



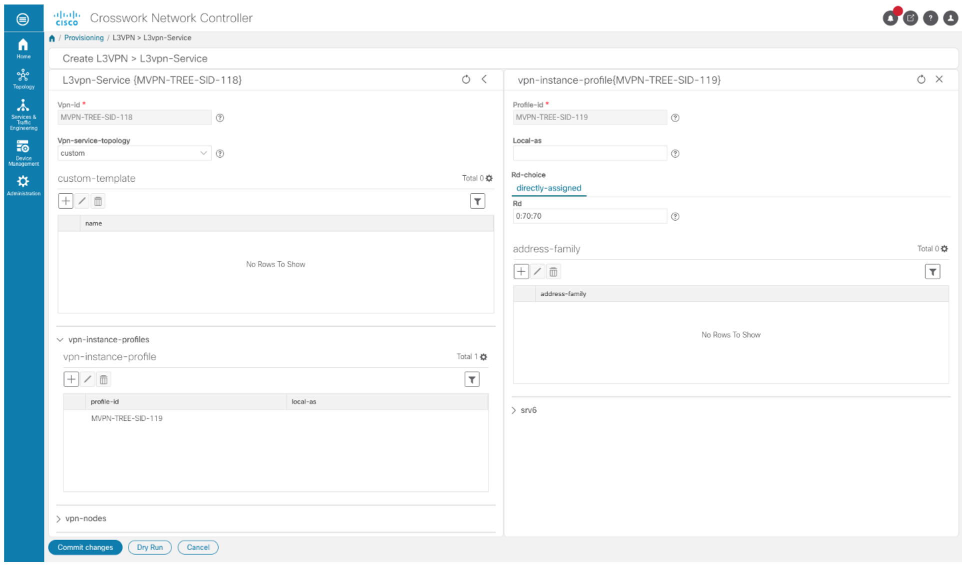
Step 6 Expand the vpn-instance-profile section and click the [+] symbol to add the profile ID.

The vpn-instance-profile panel appears.

Step 7 In the Profile-id field, type the VPN instance profile identifier (for example: **MVPN-TREE-SID-119**) and click **Continue**.

The vpn-instance-profile panel refreshes with additional fields to fill.

Step 8 In the Rd-choice field, enter the directly-assigned Rd that indicates an RD value that is explicitly assigned (for example, **0:70:70**).



Step 9 For address-family, click the + symbol. The address-family panel appears and select **ipv4** from the Address-family drop-down list and click **Continue**.

The address-family{ipv4} panel updates with vpn-targets section included.

Step 10 For vpn-target, click the + symbol so to signify the VPN target id and route-target-type.

The vpn-target panel appears.

Step 11 In the Id field, enter the ID (for example: **91**) and click **Continue**.

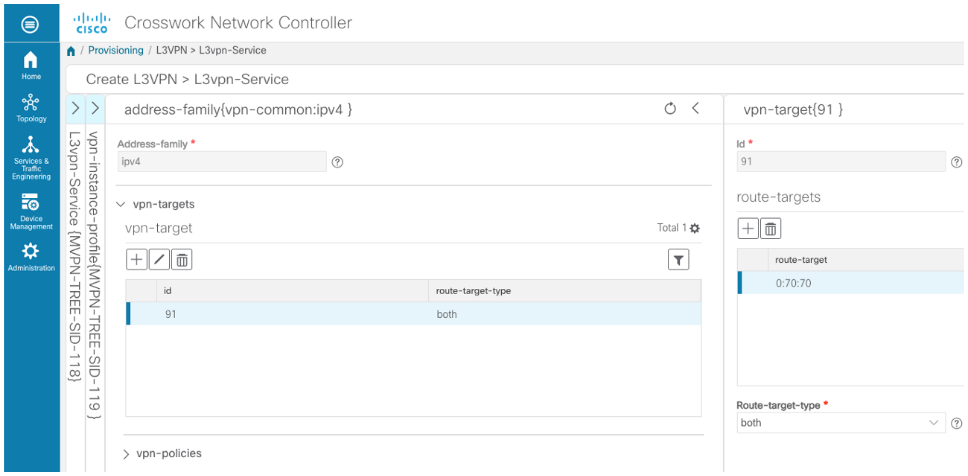
Step 12 In the vpn-target{91} panel, select the Route-target-type drop down list and select **both**.

The address-family{ipv4} panel updates showing the vpn-target id (as **91**) and route-target-type (as **both**).

Step 13 In the vpn-target{91} panel for route-targets, click the + symbol and type the Route-target (for example, **0:70:70**) and click **Continue**. Click **X** to close the panel.

The route-target table updates with the new information. Click **X** in the top right to close all of the remaining panels.

Adding the vpn-instance-profiles is now complete.



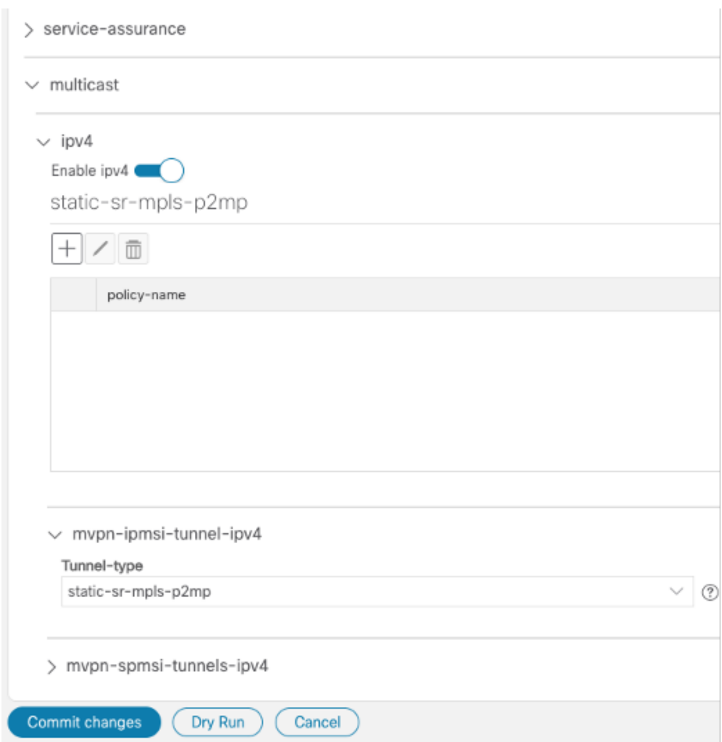
Step 14 Select **multicast** to expand the section.

Subsections, such as ipv4, mvpn-ipmsi-tunnel-ipv4, and mvpn-spmsi-tunnels-ipv4 appear.

Step 15 For the mvpn-ipmsi-tunnel-ipv4 section, from the Tunnel-type drop down list, select **static-sr-mpls-p2mp**.

The Enable ipv4 toggle is now switched on and the static-sr-mpls-p2mp area to define the name of the SR p2mp policy name.

Note The sr-mpls-p2mp available in the drop-down list is for a Dynamic Tree-SID policy.



Step 16 Click the  symbol.

The static-sr-mpls-p2mp panel appears.

Step 17 In the Policy-name field, type the previously created Static Tree-SID policy name (for example: **xrv9k-26**) and click **Continue**.

The static-sr-mpls-p2mp{Static-xrv9k-26} panel updates.

Step 18 In the sr-p2mp-policy area for the group-address, click the symbol to add the address.

The group-address panel appears.

Step 19 In the Address field, type the IPv4 static multicast group address (for example: **1.1.1.1**) and click **Continue**.

The group-address{1.1.1.1} panel refreshes. Click **X** at the top right to close any remaining panels.

Step 20 Click the symbol in the multicast > ipv4 subsection to add the other policy name.

The static-sr-mpls-p2mp panel appears.

Step 21 In the Policy-name field, type the other previously created Static Tree-SID policy name (for example: **xrv9k-24**) and click **Continue**.

The static-sr-mpls-p2mp{xrv9k-24} panel updates.

Step 22 In the sr-p2mp-policy area for the group-address, click the symbol to add the address.

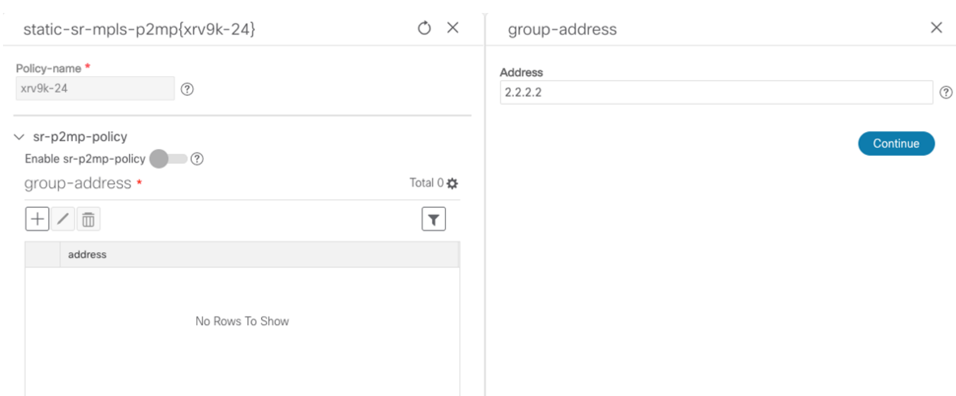
The group-address panel appears.

Step 23 In the Address field, type the IPv4 static multicast group address (for example: **2.2.2.2**) and click **Continue**.

The group-address{2.2.2.2} panel refreshes. Click **X** at the top right to close any remaining panels.

You have now successfully mapped the static Tree-SID policy to the L3VPN multicast service model. Next, you must add the VPN node details.

Note For advanced configurations, you may select mvpn-spmsi-tunnels-ipv4 subsection under the multicast section to define the tunnel-type, switch-wildcard-mode, switch-threshold, per-item-tunnel-limit, group-acl-ipv4 details.



Step 4 Add the VPN nodes

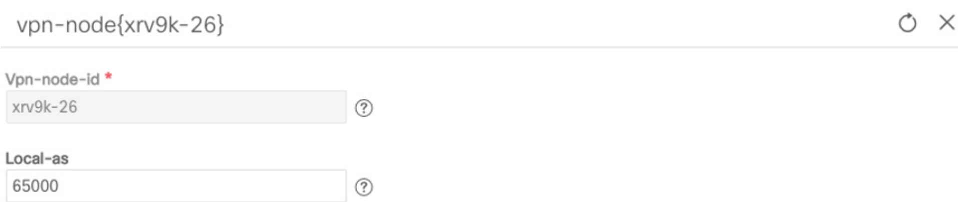
Step 1 In the vpn-nodes section, click the + symbol to add your VPN nodes set up in the Static Tree-SID policy (**xr9k-26**, **xr9k-24**, and **xr9k-27**).

The vpn-node panel appears so to add the VPN node ID.

Step 2 From the Vpn-node-id drop down, select the first of the VPN node (for example: **xr9k-26**) and click **Continue**.

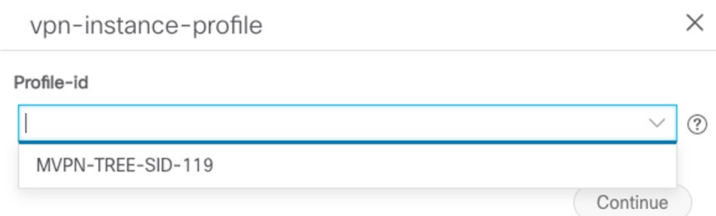
The vpn-node{xr9k-26} panel updates with additional fields.

Step 3 In the Local-as field, type **65000**.



Step 4 In the active-vpn-instance-profiles section, click the + symbol to add the VPN instance profile ID.

Step 5 In the Profile-id drop down list, the previously added profile ID appears. Select it (for example: **MVPN-TREE-SID-119**), click **Continue** and click **X** to close the panel.



Step 6 In the vpn-node{xr9k-26} panel, select the vpn-network-accesses section and click the + symbol to add the vpn-network-access ID. In the Id field, add a number (for example: **1**) and click **Continue**.

The vpn-network-access{1} panel updates with additional fields.

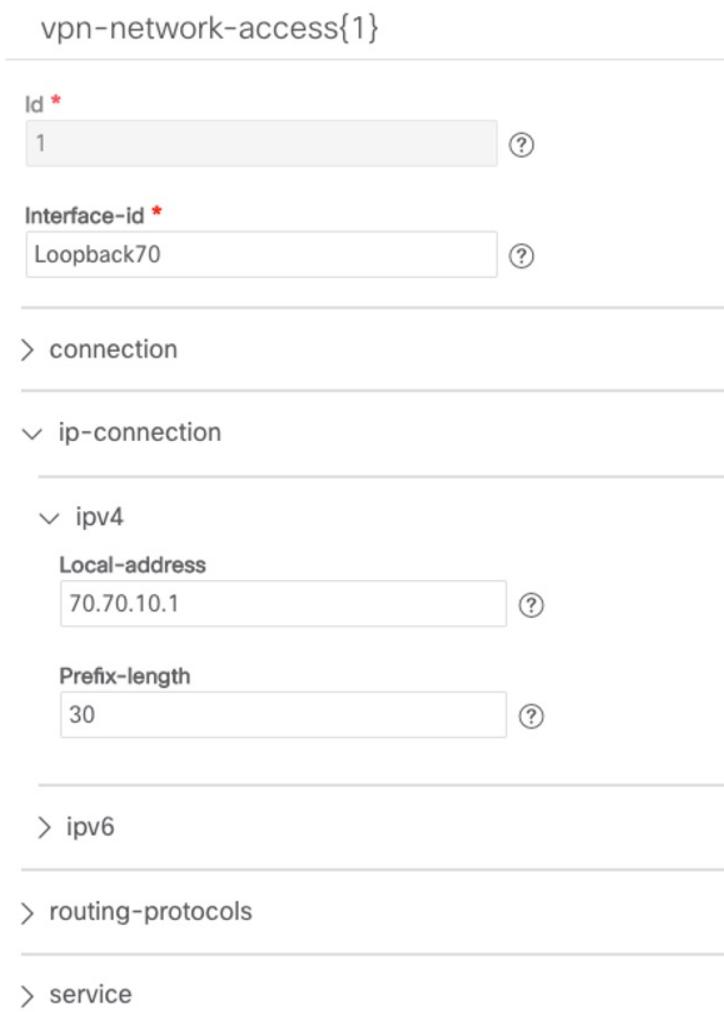
Step 7 In the Interface-id field, type the identifier for the physical or logical interface (for example: **Loopback70**).

The identification of the sub-interface is provided at the connection level and/or the IP connection level.

Step 8 In the ip-connection section, select the ipv4 subsection and in the Local-address field, type the IP address used at the provider's interface (for example: **70.70.10.1**).

Step 9 In the Prefix-length field, type **30**.

The subnet prefix length is expressed in bits. It is applied to both local and customer addresses.



Step 10 In the routing-protocols section, click the + symbol to add the unique identifier for the routing protocol. In the Id field, type **bgp** and click **Continue**.

The routing-protocol{bgp} panel appears.

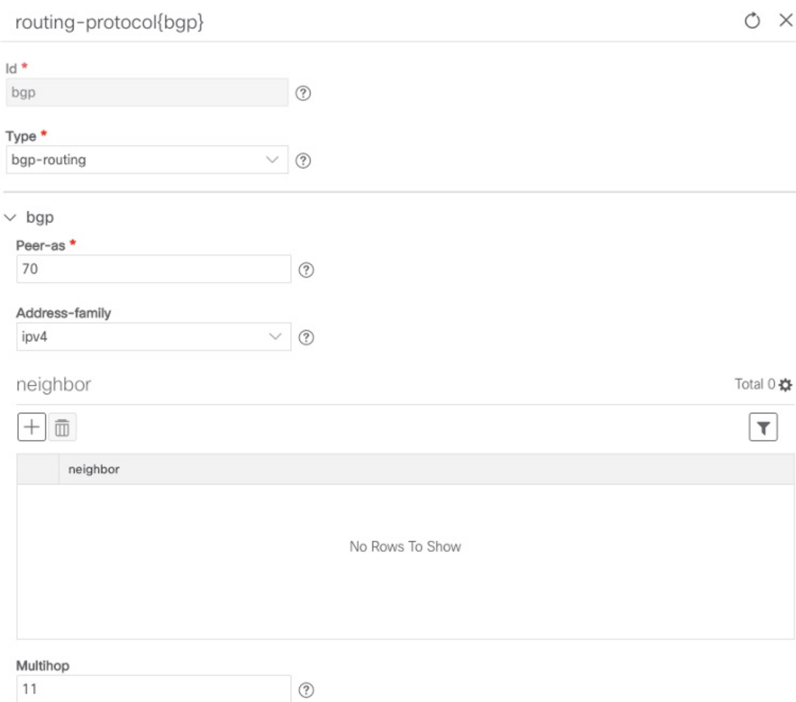
Step 11 In the Type drop down list, select **bgp-routing**.

The routing-protocol{bgp} panel refreshes with additional sections.

Step 12 In the bgp section, for the Peer-as field, type **70** to indicate the customer's ASN when the customer requests BGP routing, and in the Address-family drop down list, select **ipv4**. This node contains the address families to be activated.

Note If you select dual-stack, it means that both ipv4 and ipv6 will be activated.

Step 13 In the Multihop field, type **11** to describe the number of IP hops allowed between a given BGP neighbor and the PE.



Step 14 For neighbor section, click the symbol and in the Neighbor field, type the device address (for example: **70.70.10.2**) and click **Continue**.

Step 15 For redistribute-connected section, click the symbol and from the Address-family drop down list, select **ipv4** and click **Continue**.

The redistribute-connected{ipv4} panel appears.

Step 16 In the Enable field, type **true** to enable the redistribution of connected routes.

Close all panels (click X in the top right corner) until the Create L3VPN > L3vpn-Service screen appears.

Step 17 In the vpn-nodes section, you will see xrv9k-26 listed in the vpn-node table. Select **xrv9k-26** and select the **edit** symbol.

The vpn-node{xrv9k-26} panel appears.

Step 18 Select the multicast section and click the symbol to add the mapping of the policy for each node.

The static-sr-mpls-p2mp panel appears.

Step 19 For the Policy-name drop down list, select the policy you want to add to this node (either the source or the receiver). Select **xrv9k-24** as a receiver and click **Continue**.

The static-sr-mpls-p2mp{xrv9k-24} panel updates with additional fields.

Step 20 For the Role drop down list, select **receiver**.

Close all additional panels (click X in the top right corner) until the Create L3VPN > L3vpn-Service screen appears.

Step 21 Repeat steps 1 – 20 to add the other two VPN nodes set up in the Static Tree-SID policy: **xr9k-24** and **xr9k-26**.

Step 22 After all of the VPN nodes have been added, click **Commit changes**.

Step 5 Visualize and Edit the Static mVPN Tree-SID Policy's L3VPN service model

Step 1 Go to **Services & Traffic Engineering > Provisioning (NSO)**.

The Provisioning screen appears showing available Services/Policies.

Step 2 Select **L3VPN > L3vpn-Service**.

The L3VPN > L3vpn-Service table appears.

Step 3 Locate the newly created L3VPN service ID in the table (**MVPN-TREE-SID-119**) and in the Actions column, click  and select **Config View**.

The Configured Data pop-up screen appears.

Step 4 In the Configured Data pop-up screen, review the data configuration and click **Copy To Clipboard** if you want to save a copy, or click **Cancel** to exit.

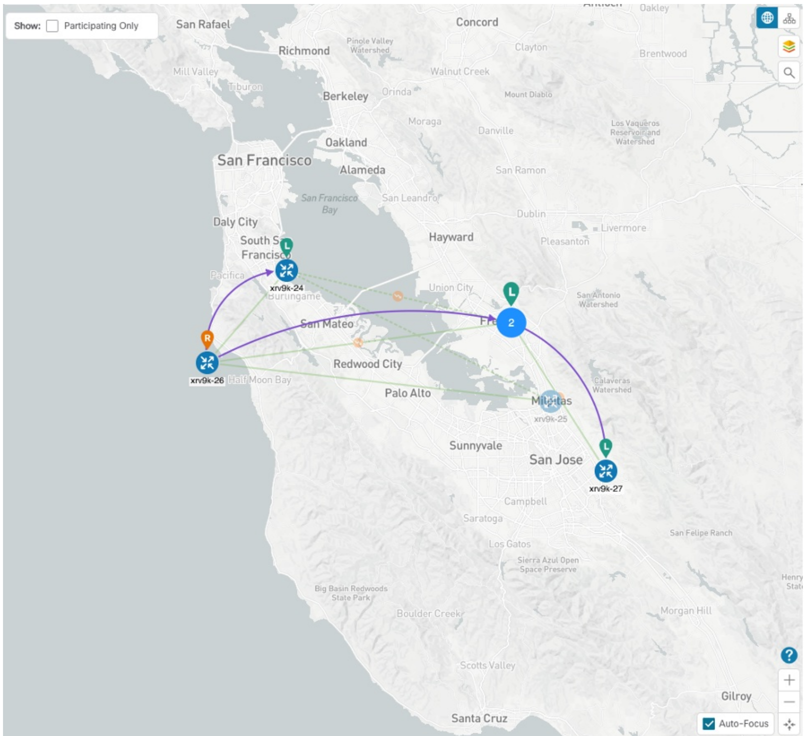
Step 5 To view the new Static mVPN Tree-SID policy associated with the L3VPN service model, click the name of the VPN Id in the table or in the Actions column, click  and select **View**.

The Service Details screen appears with the geographical map showing the newly created L3VPN service and the associated nodes: xrv9k-26, xrv9k-24, xrv9k-27. On the right, the Service Details panel shows the details of the MVPN-TREE-SID-119 service model.

Step 6 In the Service Details panel, select the **Transport** tab to view the Tree-SID Policy information.

Step 7 In the table, select the check box next to xrv9k-26.

In the geographical map, the policy will appear showing the one Root, or source, node (xrv9k-26) and the two Leaf, or receiver, nodes (xrv9k-24 and xrv9k-27).



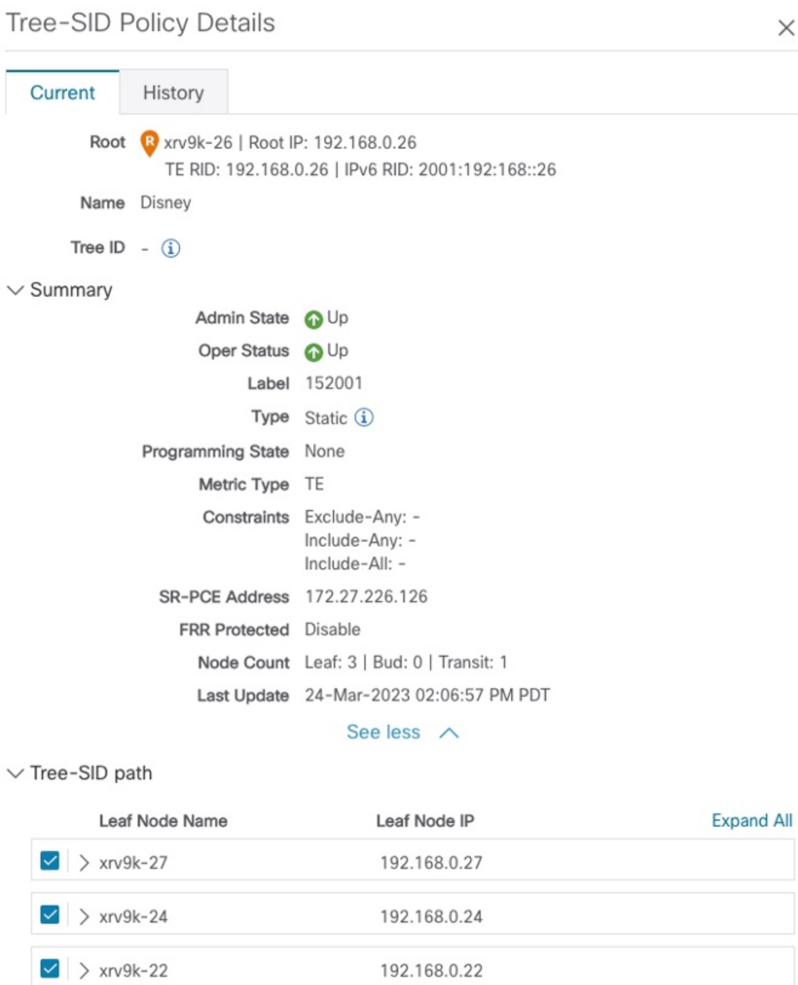
Step 8 Select the second check box next to xrv9k-24.

The geographical map updates.

Step 9 Use your mouse to hover over the Tree-SID policy names in the table. Depending which policy your mouse hovers over, the geographical map will show the designated path(s) between the nodes so to differentiate them from each other.

Step 10 For the first policy in the table, in the Actions column, click  and select **View Details**.

The Tree-SID Policy Details panel appears showing the policy’s details such as the Name, a Summary section, and the Tree-SID path information that can be expanded to show additional detail. You may also select the History tab to view historical information for the policy.



Step 11 To edit, or add additional policies, go to **Service & Traffic Engineering** > **Provisioning (NSO)**, and select **L3VPN** > **L3vpn-Service**.

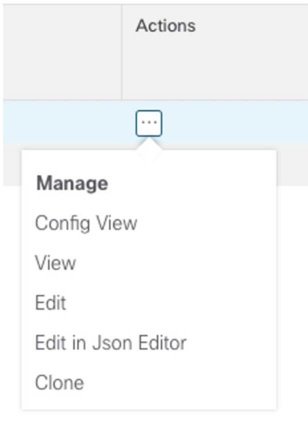
Step 12 For your L3VPN service, in the Actions column, click and select **Edit**.

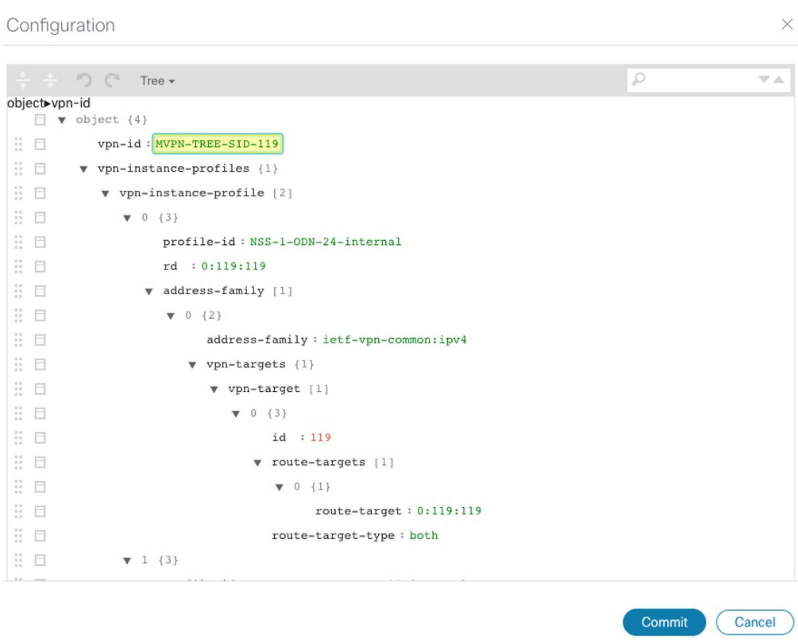
The Edit L3VPN > L3vpn-Service screen appears where you can make additional updates (such as adding VPN nodes so to replace a degraded path so to give it a different route) and modifications to existing details that make up the service.

While editing, to show all or hide the multiple fields that make up the service configuration, select the **Show all fields** toggle at the top right. Click on the toggle for Show all fields to be on. Click the toggle again for the Show all fields to be off, showing just a subset of the fields.

Step 13 In addition, from the **L3VPN** > **L3vpn-Service** screen, click in the Actions column and select **Edit in Json Editor** for your L3VPN service.

The json Configuration editor appears. Using the json Configuration editor, you can highlight different details that make up the service configuration and edit them directly in the json editor.





Step 14 Once completed, either click **Commit** to initiate the changes and update the service's configuration or click **Cancel**.

Summary and Conclusion

As we observed, you can provision new Static Tree-SID policies within the Crosswork Network Controller UI. Once provisioned, you can use the Tree-SID tab and its associated map to visualize Tree-SID defined routes, identify disjunct policy routes, and identify problems with transit nodes, interfaces and links that may affect traffic from the Root to the Leaf nodes. In addition, once the Tree-SID policies are associated with an L3VPN service model, similar capabilities are at hand to visualize and analyze Static Tree-SID policies associated with an L3VPN service model and edit in dynamic ways that improve efficiency, accuracy, and ease of use.

APPENDIX A

Appendix

This section explains the following topics:

Initializing Heuristic Packages to Monitor the Health of a Service

Objective

Enabling Service Health and using system designed Heuristic Packages to monitor the newly created service, or exporting them to your system to make adjustments before importing them back in Cisco Crosswork Network Controller, allows for customization of ongoing, detailed monitoring of your service's health.

Note Three additional Rules have been added to assist in Basic monitoring level rules (Rule-L2VPN-NM- Basic, Rule-L2VPN-NM-P2P-Basic, Rule-L3VPN-NM-Basic) where a rule to generate Assurance Graph information, for example Basic L2VPN NM P2P, services can be used along with two sub services. Heuristic Package Metrics now has the capability for CLI based metrics and GMNI filtering customizations of packages.

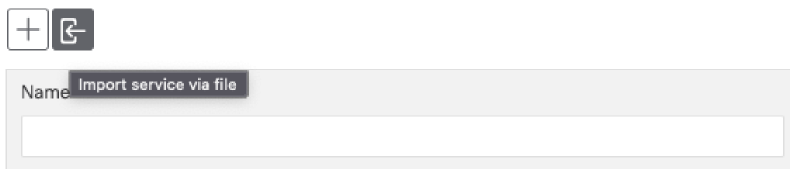
Workflow

Select either a system or custom Heuristic Package for ongoing, specialized Service Health monitoring of your new VPN service.

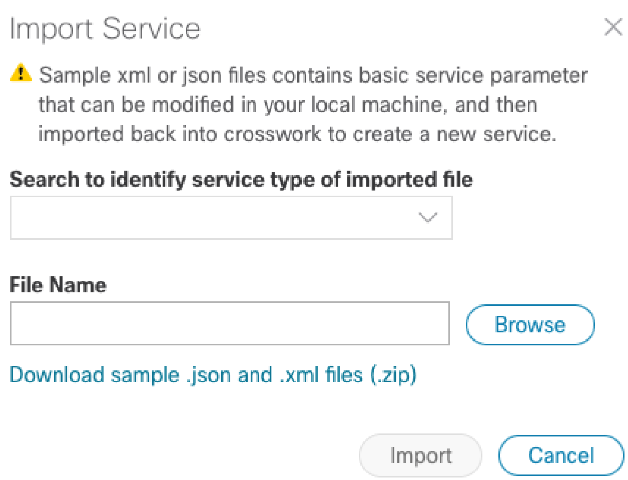
Initialize a Heuristic Package to monitor health of the new service.

1. Go to **Administration > Heuristic Packages**. The Heuristic Packages screen opens with System and Custom tabs. By default, a system defined Heuristic Package is used.

2. From the System tab, you can preview the package detail Rules, Configuration Profiles, Sub-Services, and Metrics by expanding each section for more information and hover your mouse over the information “I” icon for finer details.
3. You can click Export to download a System defined package to your system to make changes to the .json files before importing them to Cisco Crosswork Network Controller as a customized package.
4. If you exported a system file for customization, or you have custom packages on your system you want to import, click Import.



5. The Import Heuristic Packages screen opens and click Browse to find the name of your custom package on your system.



6. Select your custom package and click Import.

Note Your system performance might be impacted during heuristic package import due to high server resource consumption.

7. From the Import Heuristic Packages screen, click Preview to review the details of the package to be imported. Further information on the package’s Rules, Configuration Profiles, Sub-Services, and Metrics appears.
8. Select each option to preview the details of the custom package. Cisco Crosswork Network Controller will provide information on the details and if any details need to be updated before Cisco Crosswork Network Controller will accept the new custom package and allowing it to be imported.
9. After importing the custom package, select it so the new rules and configuration details begin to monitor the ongoing health of your designated services.

Basic and Advanced Monitoring Rules

Service Health monitoring offers two options:

- **Basic Monitoring**: Monitoring using these rules results in fewer compute resources consumed, but more services are monitored in less detail. This monitoring level provides the option of adding up to 52,000 services and results in lower overall CPU consumption, limited sub-service metrics, and smaller map graphic renderings.
- **Advanced Monitoring**: Advanced rules consume more resources, but monitor fewer services in greater detail. This monitoring level lets you add up to 2,000 services and results in higher overall CPU consumption, a greater number of sub-service metrics, and larger map graphic renderings.

Note If you select Edit Monitoring Settings, you may update the Monitoring Level setting from Basic Monitoring to Advanced Monitoring, or from Advanced Monitoring to Basic Monitoring, at any time.

Note In addition to the Service Health monitoring levels of Basic and Advanced, there are two profile options within the system package: Silver and Gold. When you begin monitoring, select either profile. By selecting the Gold profile, more custom configuration options are available compared to Silver. Monitoring profiles may be changed as needed.

For precise details on the services monitored and the thresholds used to generate alerts, view the Heuristic Package Rules and Configuration Profiles you have installed: Select **Administration > Heuristic Packages, t**hen click on the **Rules** or **Configuration Profiles** drop downs.

The following table details the monitoring functions and service metrics applied by each of the Basic and Advanced monitoring rules available with Cisco Network Controller Heuristic Packages.

Rule Name (type)	Monitoring Functionality	Metrics & Subservices
Rule-L2VPN-NM -Basic	• Checks the health of the VPWS xconnect state • Monitors the health of the device: CPU and memory utilization	metric.l2vpn.xconnect.state metric.l2vpn.xconnect.ac.state metric.l2vpn.xconnect.pw.state subservice.device.health subservice.vpws.ctrlplane.health

Rule-L2VPN-NM (Advanced)		

	• Checks the health of the VPWS or EVPN xconnect state
	• Monitors the health of the device: CPU and memory utilization
	• Monitors the delta between received and transmitted packets between VPN interfaces and Pseudo-wire
	• Monitors Y.1731 probe stats for jitter, loss and delay metrics and compares against SLA thresholds
	• Monitors the health status of RSVP tunnel. Subservice health will be marked as ‘degraded’ in either of the below scenarios:
	    • FRR is configured but backup is not ready
	    • FRR backup is active (primary failed and traffic is flowing over FRR backup)
	• Health check for interface metrics: Oper status, interface in/out error packets, interface in/out packet discard
	• Checks BGP Neighbor session health
	• Checks whether all BGP EVPN next hops for a given L2VPN service are reachable over LSP
	• Monitors PCEP session state to all the peers configured on this device.
	• Checks Path Reachability between two endpoints.
	• SR Policy (PCC initiated) health status. Admin should be up. Oper should be up. Oper
		metric.bgp.router.id
		metric.cef.route.labeled.lsp
		metric.l2vpn.xconnect.ac.state
		metric.l2vpn.xconnect.pw.state
		metric.l2vpn.xconnect.state
		metric.device.xconnect.ac.in.packets
		metric.device.xconnect.pw.out.packet
		metric.l2vpn.y1731.connect.cross.check.status
		metric.interface.oper
		metric.interface.in.errors
		metric.device.cpu.load
		metric.device.memory.free
		subservice.bgp.nbr.health
		subservice.bgp.evpn.nexthop.health
		subservice.device.health
		subservice.evpn.health (one for each endpoint)
		subservice.fallback.path.health
		subservice.interface.health (one for each interface)
		subservice.l2vpn.y1731.health
		subservice.path.reachability.to.peer (local to remote and remote to local)
		subservice.path.sla
		subservice.pcep.session.health (one for each endpoint device)
		subservice.plain.lsp.path.health
		subservice.sr.policy.pce.health (one for each endpoint)
		subservice.vpws.ctrlplane.health (local, remote)
		subservice.path.reachability.to.peer
		subservice.fallback.path.health
		subservice.mpls.rsvpte.tunnel.pm.health
		subservice.l2vpn.y1731.health
		subservice.vpws.ctrlplane.health

	should have stayed up since last polling. • Checks whether LSP path exists (in default VRF) towards the given destination device.	subservice.interface.health subservice.device.health subservice.interface.health.summary subservice.path.sla.summary
Rule-L2VPN-NM -P2P-Basic	• Checks the health of the VPWS xconnect state • Monitors the health of the device: CPU and memory utilization	subservice.device.health subservice.vpws.ctrlplane.health
Rule-L2VPN -NM-P2P (Advanced)	• Checks the health of the VPWS xconnect state • Monitors the health of the device: CPU and memory utilization • Health check for interface metrics: Oper status, interface in/out error packets, interface in/out packet discard • Monitors Y.1731 probe stats for jitter, loss and delay metrics and compares against SLA thresholds • Monitors the LSP path to the peer VPN node • Monitors path reachability between two endpoints • Monitors LSP path (in default VRF) towards the given destination IP address • Monitors PCEP session state to all the peers configured on this device • SR Policy (PCC initiated) health status. Admin should be up. Oper should be up. Oper should have stayed up since last polling.	metric.cef.route.labeled.lsp metric.l2vpn.xconnect.ac.state metric.l2vpn.xconnect.pw.state metric.l2vpn.xconnect.state subservice.device.health subservice.interface.health (one for each interface) subservice.l2vpn.y1731.health subservice.p2p.fallback.path.health subservice.p2p.path.reachability.to.peer (path reachability between endpoints) subservice.p2p.plain.lsp.path.health subservice.path.sla subservice.pcep.session.health (one for each endpoint device) subservice.sr.policy.pcc.health subservice.sr.policy.pce.health (one for each endpoint) subservice.vpws.ctrlplane.health (local, remote)

Rule-L2VPN-MP-Basic	For all .summary subservices: Groups together all the device subservices as an aggregator node. It does not have its own health/metric. Its health depends on its child subservice health. Monitors the health of the device Monitors bridge domain state on a given endpoint	subservice.device.summary subservice.bridge.domain.summary subservice.device.health subservice.bridge.domain.state

Rule-L2VPN-MP (Advanced)		

	For all .summary subservices: Groups together all the device subservices as an aggregator node. It does not have its own health/metric. Its health depends on its child subservice health. Monitors the health of the device Groups together all the PCEP session health subservices Monitors PCEP session state to all the peers configured on this device Groups together all the device subservices BGP Neighbor health Monitors whether any routes are present for the given Bridge Domain Groups together all the bridge domain subservices Monitors bridge domain state on a given endpoint Subservice to reflect interface health Groups together all the transport subservices SR Policy health status reflecting SR-PM SLA (if configured). Admin & Oper should be up. Oper should have stayed up since last polling. Delay & Variance should meet SLA if SR-PM is configured to measure delay. Liveness should be up if SR-PM is configured for Liveness. SR Policy health status that include SR-PM. Admin & Oper should be up. And Oper should have stayed up since last polling. Delay & Variance should meet SLA if SR-PM is configured to measure delay. Liveness should be up if SR-PM is configured for Liveness. Monitors MPLS RSVP TE Tunnel Health. Admin, Oper should both be up and if fast reroute is configured, then backup path should be ready to pickup traffic when primary fails. If failover already	metric.device.memory.free (supports XR only) metric.device.cpu.load (supports XR only) metric.sr.te.pcc.peer.state (supports XR only) metric.sr.te.pcc.peer.addrs (supports XR only) metric.bgp.session.state (supports XR only) metric.bgp.neighbors.ipaddr.list (supports XR only) metric.mac.learning.nexthops (supports XR only) metric.l2vpn.bridge.ac.state (supports XR only) metric.l2vpn.bridge.ac.list (supports XR only) metric.l2vpn.bridge.domain.state (supports XR only) metric.interface.oper (supports both XR and XE) metric.interface.in.errors (supports both XR and XE) metric.interface.out.errors (supports both XR and XE) metric.interface.in.discards (supports both XR and XE) metric.interface.out.discards (supports both XR and XE) metric.sr.policy.pcc.admin.state (supports XR only) metric.sr.policy.pcc.oper.state (supports XR only) metric.sr.policy.pcc.oper.up.time (supports XR only) metric.sr.policy.pm.delay.measurement (supports XR only) metric.sr.pm.delay (supports XR only) metric.sr.pm.variance (supports XR only)

	happened to backup then health will be shown as degraded as there is no more redundancy in play. Delay should be considered if SR PM is enabled. If delay is enabled, then variance will be considered. Monitors the policies deployed by the ODN	metric.sr.policy.pm.liveness.detection (supports XR only) metric.sr.pm.liveness.state (supports XR only) metric.sr.policy.pce.admin.state (supports XR only) metric.sr.policy.pce.oper.state (supports XR only) metric.sr.policy.pce.oper.up.time (supports XR only) metric.sr.policy.pce.ietf.policy.name (supports XR only) metric.sr.policy.pm.delay.measurement (supports XR only) metric.sr.pm.delay (supports XR only) metric.sr.pm.variance (supports XR only) metric.sr.policy.pm.liveness.detection (supports XR only) metric.sr.pm.liveness.state (supports XR only) metric.mpls.rsvpte.tunnel.oper.state (supports XR only) metric.mpls.rsvpte.tunnel.admin.state (supports XR only) metric.mpls.rsvpte.tunnel.frr.configured (supports XR only) metric.mpls.rsvpte.tunnel.frr.status (supports XR only) metric.mpls.te.pm.delay.measurement (supports XR only) metric.mpls.rsvp.te.delay (supports XR only) metric.mpls.rsvp.te.variance (supports XR only) metric.l2vpn.odn.sr.policies.list (supports XR only) metric.bgp.router.id (supports both XR and XE) subservice.device.summary subservice.device.health

		subservice.pcep.session.health.summary subservice.pcep.session.health subservice.evpn.summary subservice.bgp.nbr.health subservice.mac.learning subservice.bridge.domain.summary subservice.bridge.domain.state subservice.interface.health subservice.transport.summary subservice.sr.policy.pcc.pm.health subservice.sr.policy.pce.pm.health subservice.mpls.rsvpte.tunnel.pm.health subservice.l2vpn.sr.odn.policy.dynamic
Rule-L3VPN-NM-Basic	• Reports the overall route connectivity health between the current PE device and its connecting CE device • Monitors the health of the device: CPU and memory utilization	subservice.ce.pe.route.health subservice.device.health

Rule-L3VPN-NM (Advanced)	• For all .summary subservices: Groups together all the device subservices as an aggregator node. It does not have its own health/metric. Its health depends on its child subservice health. • Subservice, together with child subservices in L3VPN Rule, report the overall route health between current PE device and its connecting CE device • eBGP Session health • Subservice to reflect interface health • Monitors the health of the device • L3VPN Aggregator Subservice that reflects path reachability from given device, for a given vrf, to peer VPN sites • Monitors both static and dynamically initiated policy • Checks whether plain lsp route exists within given VRF towards given vpn ip-addresses • Monitors PCEP session state to all the peers configured on this device • BGP Neighbor health	

			metric.route.vrf.connected (supports XR and XR IPv6) metric.route.vrf.local (supports XR and XR IPv6) metric.bgp.vrf.session.state (supports XR only) metric.interface.oper (supports both XR and XE) metric.interface.in.errors (supports both XR and XE) metric.interface.out.errors (supports both XR and XE) metric.interface.in.discards (supports both XR and XE) metric.interface.out.discards (supports both XR and XE) metric.device.memory.free (supports XR only) metric.device.cpu.load (supports XR only) metric.l3vpn.sr.policies.list (supports XR and XR IPv6) metric.cef.vrf.route.prefix (supports XR and XR IPv6) metric.sr.te.pcc.peer.state (supports XR only) metric.sr.te.pcc.peer.addrs (supports XR only) metric.bgp.session.state (supports XR only) metric.bgp.neighbors.ipaddr.list (supports XR only) metric.bgp.route.l2vpn.evpn.nexthops metric.bgp.router.id metric.cef.route.labeled.lsp metric.bgp.session.state metric.bgp.neighbors.ipaddr.list metric.route.vrf.connected metric.route.vrf.local metric.device.memory.free

			metric.device.cpu.load
			metric.bgp.vrf.session.state
			metric.l2vpn.xconnect.pw.state
			metric.cef.route.labeled.lsp
			metric.bgp.router.id
			metric.interface.oper
			metric.interface.in.errors
			metric.interface.out.errors
			metric.interface.in.discards
			metric.interface.out.discards
			metric.l2vpn.y1731.connect.cross.check.status
			metric.l2vpn.y1731.connect.peer.mep.status
			metric.l2vpn.y1731.latency.rt
			metric.l2vpn.y1731.jitter.rt
			metric.l2vpn.y1731.pktloss.1way.sd
			metric.l2vpn.y1731.pktloss.1way.ds
			metric.cef.route.labeled.lsp
			metric.cef.route.labeled.lsp
			metric.device.xconnect.ac.in.packets
			metric.device.xconnect.pw.out.packets
			metric.device.xconnect.pw.in.packets
			metric.device.xconnect.ac.out.packets
			metric.sr.te.pcc.ipv4.peer.state
			metric.sr.te.pcc.ipv4.peer.addrs
			metric.cef.route.labeled.lsp
			metric.bgp.router.id
			metric.sr.policy.pcc.oper.state
			metric.sr.policy.pcc.oper.up.time
			metric.sr.policy.pcc.admin.state
			metric.sr.policy.pm.delay.measurement
			metric.sr.pm.delay
			metric.sr.pm.variance
			metric.sr.policy.pm.liveness.detection
			metric.sr.pm.liveness.state

			metric.sr.policy.pce.oper.up.time metric.sr.policy.pce.oper.state metric.sr.policy.pce.admin.state metric.l2vpn.xconnect.state metric.l2vpn.xconnect.ac.state metric.l2vpn.xconnect.pw.state metric.cef.vrf.route.prefix metric.l3vpn.odn.sr.policies.dynamic.list metric.l2vpn.odn.sr.policies.list metric.bgp.router.id metric.mac.learning.nexthops metric.mpls.rsvpte.tunnel.oper.state metric.mpls.rsvpte.tunnel.admin.state metric.mpls.rsvpte.tunnel.frr.configured metric.mpls.rsvpte.tunnel.frr.status metric.mpls.te.pm.delay.measurement metric.mpls.rsvp.te.delay metric.l2vpn.bridge.ac.state metric.l2vpn.bridge.ac.list metric.l2vpn.bridge.domain.state subservice.ce.pe.route.health.summary subservice.ce.pe.route.health subservice.ebgp.nbr.health subservice.interface.health.summary subservice.interface.health subservice.device.summary subservice.device.health subservice.vrf.path.reachability.to.peer.summary subservice.vrf.path.reachability.to.peers subservice.transport.summary subservice.dynamic.l3vpn.sr.policy subservice.vrf.plain.lsp.reachability subservice.pcep.session.health.summary subservice.pcep.session.health

			subservice.bgp.nbr.health.summary
			subservice.bgp.nbr.health
			subservice.bgp.evpn.nexthop.health
			subservice.bgp.nbr.health
			subservice.ce.pe.route.health
			subservice.device.health
			subservice.ebgp.nbr.health
			subservice.evpn.health
			subservice.fallback.path.health
			subservice.interface.health
			subservice.l2vpn.y1731.health
			subservice.p2p.fallback.path.health
			subservice.p2p.path.reachability.to.peer
			subservice.p2p.plain.lsp.path.health
			subservice.path.reachability.to.peer
			subservice.path.sla
			subservice.pcep.session.health
			subservice.plain.lsp.path.health
			subservice.sr.policy.pcc.health
			subservice.sr.policy.pce.health
			subservice.vpws.ctrlplane.health
			subservice.vrf.path.reachability.to.peers
			subservice.vrf.plain.lsp.reachability
			subservice.bridge.domain.summary
			subservice.l3vpn.sr.odn.policy.dynamic
			subservice.l2vpn.sr.odn.policy.dynamic
			subservice.mac.learning
			subservice.mpls.rsvpte.tunnel.pm.health
			subservice.vrf.path.reachability.to.peer.summary
			subservice.path.sla.summary
			subservice.pcep.session.health.summary
			subservice.transport.summary
			subservice.interface.health.summary
			subservice.vpws.ctrlplane.health.summary

		subservice.bridge.domain.state

Service Health Supported Subservices

The following tables provide details of supported Service Health L2VPN/L2VPN flavors and associated subservices (for IOS XE and XR devices). The subservices listed are available out of the box from Crosswork Automated Assurance.

Supported VPN services with associated subservices (for IOS XE devices):

Supported VPN Services	Associated Subservices	Details
L2VPN Point to Point with SR underlay	Path Reachability Y.1731 Health VPN Interface Health Device Health Summary (aggregator) nodes	XE does not support SNMP/gNMI collection type for this subservice (CEF route; PCEP Session State; SRPolicy State; XConnect).
L2VPN Point to Point over MPLS LDP	Path Reachability Y.1731 Health VPWS Control Plane health VPN Interface Health Device Health Summary (aggregator) nodes	XE does not support SNMP/gNMI collection type for this subservice (CEF route; XConnect).
L2VPN P2P Plain	Path Reachability Y.1731 Health VPN Interface Health Device Health Summary (aggregator) nodes	XE does not support SNMP/gNMI collection type for this subservice (CEF route; XConnect). **Note**: The reference to ‘Plain’ implies that L2VPN/L3VPN traffic takes the IGP path and does not use any transports, like SR.

L3VPN SR	Path Reachability CE-PE Route Health eBGP Neighbor Health VPN Interface Health BGP Neighbor Health (DynExp) Summary (aggregator) nodes	XE does not support SNMP/gNMI collection type for this subservice (CEF route; PCEP Session State). SR-ODN is also not supported.

Supported VPN services with associated subservices (for IOS XR devices):

Supported VPN Services	Associated Subservices
L2VPN EVPN SR	Path Reachability Fallback Enabled/Disabled (DynExp) SR Policy – PCC Path SLA Y.1731 Health VPWS Control Plane Health VPN Interface Health Device Health EVPN Health BGP Neighbor Health (DynExp) BGP Nexthop Health (DynExp) PCEP Session Health (DynExp) SR Policy – PCE Summary (aggregator) nodes

L2VPN EVPN Plain	Path Reachability Path SLA Plain LSP Path Health (DynExp) VPWS Control Plane health VPN Interface Health Device Health EVPN Health BGP Neighbor Health (DynExp) BGP Nexthop Health (DynExp) Summary (aggregator) nodes **Note**: The reference to 'Plain' implies that L2VPN/L3VPN traffic takes the IGP path and does not use any transports, like SR.
L2VPN Point to Point over RSVP	Path Reachability Fallback Enabled/Disabled RSVP-TE Health Path SLA Y.1731 Health VPWS Control Plane Health/Xconnect Health VPN Interface Health Device Health
L2VPN Point to Point with SR underlay	Path Reachability Fallback Enabled/Disabled SR Policy – PCC Path SLA Y.1731 Health VPWS Control Plane Health VPN Interface Health Device Health PCEP Session Health (DynExp) SR Policy – PCE Summary (aggregator) nodes

L2VPN Point to Point over MPLS LDP	Path Reachability Fallback Enabled/Disabled Path SLA Y.1731 Health VPWS Control Plane Health VPN Interface Health Device Health Summary (aggregator) nodes
L2VPN P2P Plain	Path Reachability Plain LSP Path Health Path SLA Y.1731 Health VPWS Control Plane Health VPN Interface Health Device Health Summary (aggregator) nodes **Note**: The reference to ‘Plain’ implies that L2VPN/L3VPN traffic takes the IGP path and does not use any transports, like SR.
L3VPN SR	CE-PE Route Health eBGP Neighbor Health VPN Interface Health Device Health Path Reachability Vrf Plain LSP Path Health PCEP Session Health (DynExp) BGP Neighbor Health (DynExp) Summary (aggregator) nodes SR and SRv6 polices

Configuring Service Health External Storage Settings

Objective

Service Health provides Internal Storage of monitoring data up to a maximum limit of 50 GB. This data is stored on your system. If you exceed the limit of the internal storage, historical data will be lost.

Note If you anticipate monitoring a large amount of Service Health services, Cisco recommends you configure External Storage after you install Service Health and before you begin monitoring services so to avoid exceeding the Internal Storage and losing historical data.

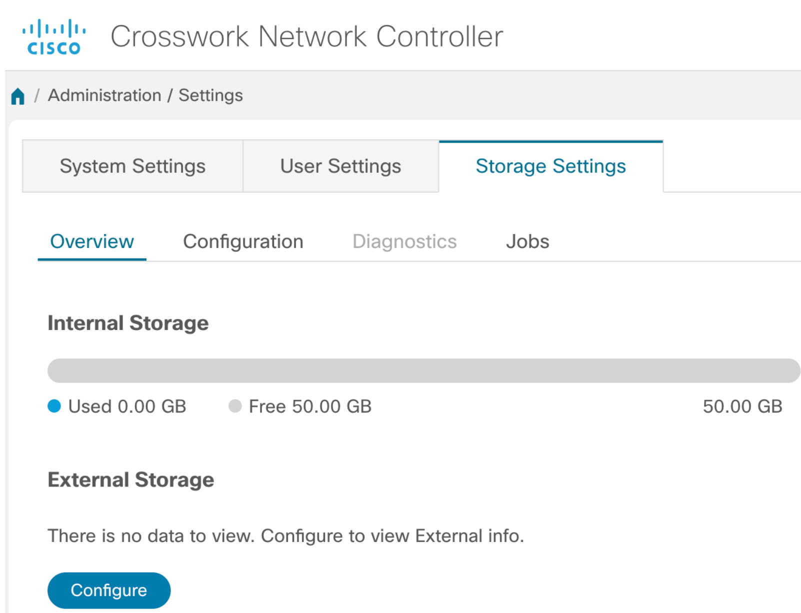
If you choose to extend Service Health storage capacity, you can configure External Storage in the cloud using an Amazon Web Services (AWS) cloud account. By leveraging External Storage, all existing internal storage data will be automatically moved to the external cloud storage and your internal storage will act locally as cache storage. Configuring External Storage for Service Health ensures you will not lose historical data for services that continue to monitor a service's health, and will retain service health data for any service you choose to stop monitoring when you select the option to retain historical monitoring service for the data.

Workflow

To expand storage capacity beyond Internal Storage, configure External Storage using your AWS account to ensure you will not lose historical data for services that continue to monitor a service's health, and will retain service health data for any service you choose to stop monitoring when you select the option to retain historical monitoring service for the data.

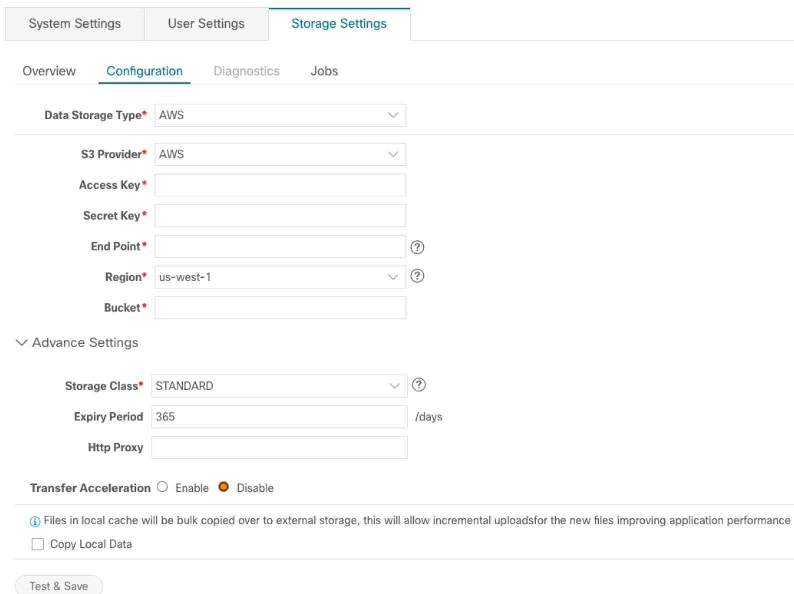
To configure External Storage, do the following:

1. Go to **Administration** > **Settings** and select the **Storage Settings** tab. The Overview screen appears.



2. Under External Storage, click **Configure**. The Configuration screen appears with the Data Storage Type and S3 Provider fields pre-populated with Amazon Web Services (AWS).

Note You must have an AWS cloud account set up so to configure the external storage settings. Refer to the AWS site for more information.



3. Provide your AWS authentication information for all of the required fields (such as Access Key, Secret Key, End Point, etc).

4. Select the **Copy Local Data** check box if you want all files, previously stored in the local cache, to be bulk copied to the external storage. This action will allow for incremental upload of the new files.

Note This option is a one-time action when moving from only maintaining local storage and moving to external storage. This action will also help improve application performance.

Note 'Expiry Period' is the number of days of life for historical data files. If 'Expiry Period' is set to 1, the historical data files will be deleted two days later and the deletion will take place at midnight of the second day.

5. Click **Test & Save**.

6. To check on the health of your storage setup, select the Diagnostics tab and click **Run Test.**

   By running a test, you can review external storage diagnostics such as bandwidth, latency, and multiple Access test details to help identify possible storage performance issues.

Stopping Service Health Monitoring

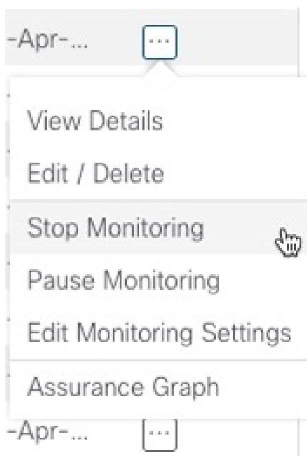
Objective

Service Health provides specific options when you stop monitoring a service. When you stop monitoring a service, Service Health asks if you want to retain the historical monitoring service data. If you retain the historical data, and you later restart monitoring the service, the data collected, when the service was previously monitoring, will be available. If you choose to stop monitoring the service without retaining the historical service data, the monitoring settings are deleted and the historical service data will expire or be purged if you later choose to start monitoring the service later. In addition, the Assurance Graph for that stopped service will no longer be available.

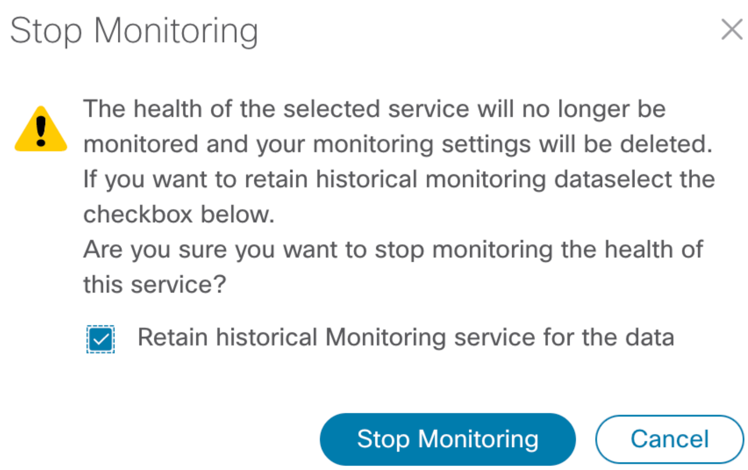
Workflow

To stop monitoring a Service Health service and retain historical monitoring service data, do the following:

1. Click in the Actions column for that service and select **Stop Monitoring** from the menu.



2. The Stop Monitoring service pop up appears. To retain the historical service data for that service, select the **Retain historical Monitoring service for the data** check box.

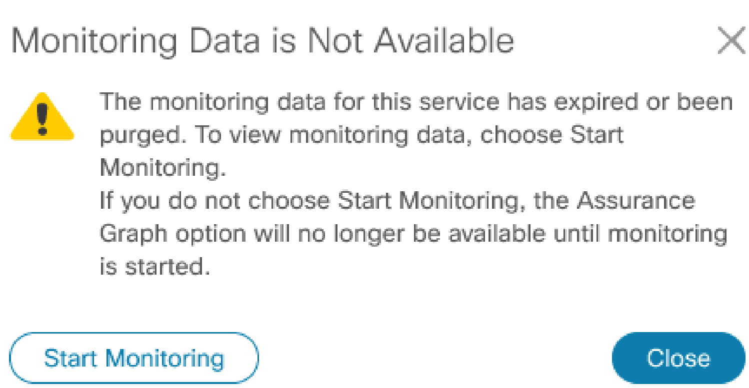


3. Click **Stop Monitoring**.

   The service's historical monitoring data is preserved.

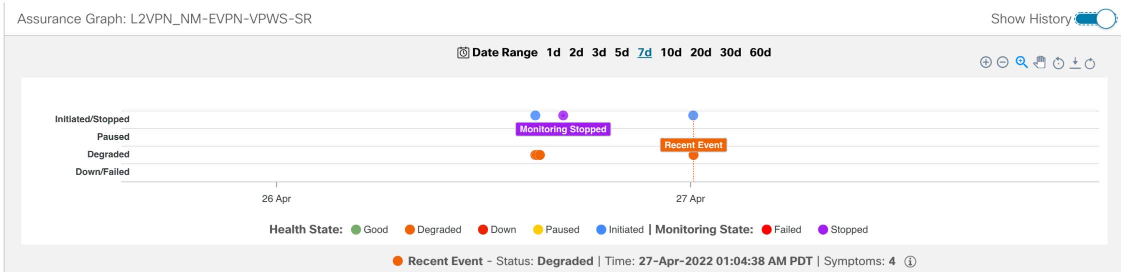
Note If you stop monitoring a service and do not select the **Retain historical Monitoring service for the data** check box, the **Assurance Graph** option will no longer be available because the monitoring settings will have been deleted and the historical service data will have expired or been purged. You may again start to monitor the health of that service and begin service data collection anew.

Note As an alternative to stopping Service Health monitoring is to use the Pause and Resume option. If you pause, and the resume, monitoring a service, it will resume monitoring using the same Basic or Advanced monitoring rule and profile options that were used before the pause action. In addition, historical data and Events of Significance (EOS) will be preserved in the history of the service. However, when the service is paused, previous, and new active symptoms, will not appear or be collected.



4. To view the stopped service in the Assurance Graph, click in the Actions column for that service and select **Assurance Graph** from the menu.

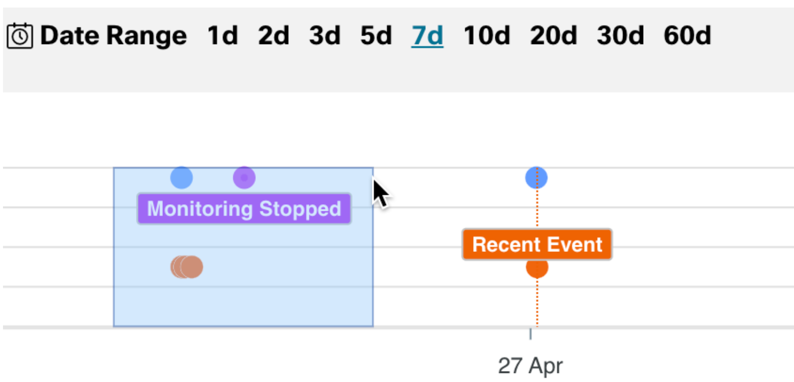
5. Click the **Show History** toggle.

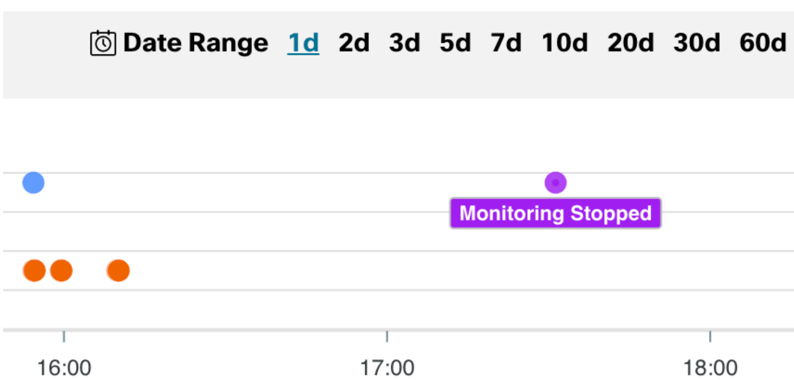


Each dot on the history chart represents one Event Of Significance (EOS) for a service. For each EOS, you can view the Assurance Graph and symptoms with 24 hours metrics collected based on the time of the EOS.

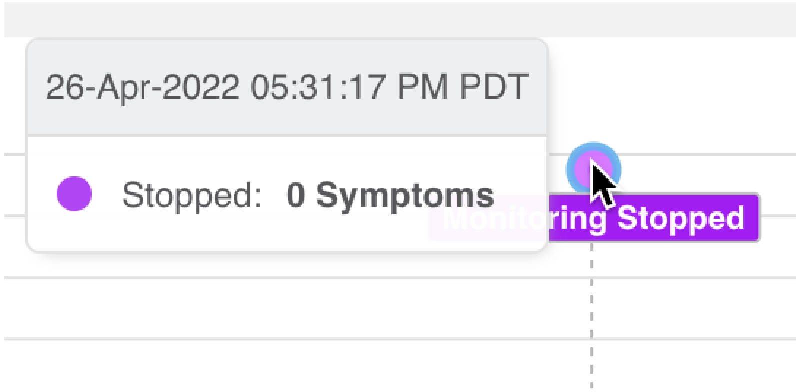
6. In the graph, the service that was stopped will appear indicating Monitoring Stopped.

7. Using your mouse, click and drag over a selected range over the Monitoring Stopped service to zoom in on the time range.





8. Hover your mouse over the Monitoring Stopped service to view the date stamp when the service was stopped and if there were any symptoms associated with the stopped service.



9. If you stopped monitoring a service and selected the **Retain historical Monitoring service for the data** check box, you can later start monitoring that same service with historical data still available. Click in the Actions column for that service and select **Start Monitoring** from the menu.

Note If External Storage has been configured, and you are monitoring a large amount of services, you can ensure that the historical data of the stopped, and restarted, service is preserved for continued monitoring and inspection. See the **Configuring Service Health Storage Settings** section for details.