Cisco Crosswork Network Controller 7.2 Service Health Monitoring

First Published: 2026-01-28

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

CONTENTS

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 1

Introduction

This section explains these topics:

Audience

This guide is for network administrators who intend to use the Service Health component of Cisco Crosswork Network Controller in a network environment. This guide assumes familiarity with these following topics:

- Platform Infrastructure and installation of Crosswork Network Controller components. For more information, refer to the Crosswork Network Controller 7.2 Installation guide.
- Provisioning Layer 2 Virtual Private Network (L2VPN) and Layer 3 Virtual Private Network (L3VPN) services
- Networking technologies and protocols (Border Gateway Protocol - Link State (BGP-LS), Interior Gateway Protocol (IGP) such as Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS), Path Computation Element Communication Protocol (PCEP), and model-driven telemetry
- Traffic Engineering (TE) tunnels, including
  - Resource Reservation Protocol-Traffic Engineering (RSVP-TE) tunnel provisioning
  - Segment Routing Traffic Engineering (SR-TE) policy provisioning

Overview of Service Health

Service Health is a component of Crosswork Network Controller's Advantage package. For more information on all Crosswork Network Controller solution components, see the Crosswork Network Controller 7.2.0 Installation guide.

Service Health extends the capabilities of Crosswork Network Controller by offering a service-level perspective that goes beyond monitoring devices at a physical level. While Crosswork Network Controller can alert you to a failed link or one that has reached capacity, Service Health assesses the impact of these issues on services traversing the links, including L2 and L3 VPN services.

Service Health provides operators with the ability to pinpoint exactly where and why a service is degraded, offering targeted tools for service-specific monitoring and assurance.

The application offers APIs and samples for developing additional service monitoring tools as needed. Additionally, it includes packages ready for use in monitoring common scenarios, such as:

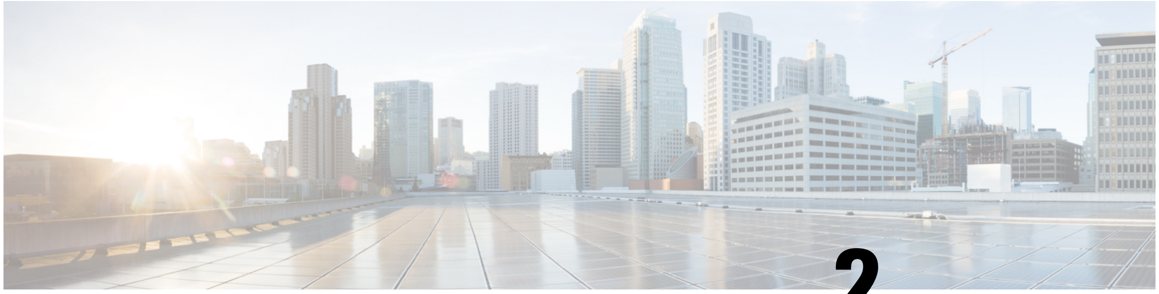
- Health monitoring of point-to-point L2VPN services (for example, T-LDP and EVPN)
- Health monitoring of multipoint L2VPN services (such as EVPN E-LAN and E-Tree L2VPN EVPN)
- Health monitoring of L3VPN services
- Analysis of health of degraded services to provide information to aid troubleshooting
- Visualization of service health status and its logical dependency tree

Service Health is designed to be extensible, allowing you to add new service monitoring capabilities to meet your requirements.

Service Health APIs

Users can integrate other Crosswork Network Controller components and third party applications with Service Health by using application programming interfaces (APIs) to deliver new capabilities into their network operations.

For more information, see the Cisco Crosswork Network Automation API Documentation on Cisco DevNet.



CHAPTER 2

Getting started

This section explains these topics:

Important changes in behavior

The following table describes important changes in behavior in this release.

Topic	Description
Discontinuation of Amazon Web Services (AWS) external storage for Service Health historical data.	Service Health historically allowed configuration of external storage using an AWS cloud account to store monitoring data beyond the internal storage capacity. This external storage acted as a cloud-based archive while the internal storage functioned as a local cache. Service Health no longer supports storing historical data on AWS. With this change, historical data retention is now limited to the 50 GB internal storage capacity. To prevent access to older historical data, regularly observe the health of the monitored services.

Before you begin

We recommend that you familiarize yourself with the following concepts and complete any planning and information-gathering steps:

- From the Crosswork Network Controller home page, the **VPN service health** dashlet delivers an at-a-glance summary of VPN service statuses. This enables enhanced visibility and filtering for monitoring status, such as errors, in Service Health.

  The **VPN Services** page presents detailed health and monitoring data for all VPN services, enhanced by customizable columns and powerful filters that facilitate focused troubleshooting.

  The **Service Health Dashboard** complements these views by aggregating service metrics and highlighting SLA breaches, enabling efficient oversight of network performance and service quality. For additional information on viewing and filtering monitored VPN services, see Analyze Service Health.

- Crosswork Network Controller monitors services at two levels: Basic and Advanced.

  - **Basic Monitoring**: This type of monitoring offers the option of monitoring a higher number of services and provides limited subservice metrics, resulting in lower resource consumption. Additionally, the graphic map renderings are smaller compared to more detailed monitoring.

  - **Advanced Monitoring**: This monitoring approach is supported for a fewer number of services, as it monitors a larger number of component subservices and consumes more compute resources. Additionally, advanced monitoring results in an increased number of subservice metrics and larger graphic map renderings.

    To view only **Basic** or **Advanced** services from the Crosswork Network Controller home page, click the highlighted number within the dashlet. The VPN Services page appears with the filtered service information.

  For more information, see Service Health scale information.

- Crosswork Network Controller's Service Health supports single virtual machine (VM) deployment and monitors devices at two levels - Basic and Advanced. These monitoring level details also apply to Service Health single VM deployment.

  For more information, see Service Health single VM scale information.

- For L2VPN services, Crosswork Network Controller monitors the overall health based on the subservices, while for L3VPN services, the monitoring occurs at the node level.

- Crosswork Network Controller has implemented a rate-limiting process to manage service monitoring requests efficiently. This means that there may be a delay in publishing service monitoring requests if the number of requests raised per minute exceeds a specific threshold. The thresholds are:

  - **L2VPN services**

    - 50 Basic Monitoring requests per minute per service

    - 5 Advanced Monitoring requests per minute per service

  - **L3VPN services**:

    - 500 Basic Monitoring requests per minute per vpn-node

    - 100 Advanced Monitoring requests per minute per vpn-node

  The rate-limiting process also extends to the monitoring data. For example, during a restore process, when all Data Gateways send data to the Tracker component, the rate at which the Tracker processes this data and forwards it to Assurance Graph Manager is regulated. This may lead to a delayed reporting of Events of Significance (EOS) following the restore.

An event is triggered with a severity level of warning and a corresponding description to notify you of the delay. The event is cleared once Crosswork Network Controller resumes normal publishing of monitoring requests.

- Crosswork Network Controller can store up to 50 GB of monitoring data. When storage usage reaches 70 percent of this capacity, it raises an alarm to alert you of potential storage depletion. If more storage is needed, you can configure external storage in the cloud using an Amazon Web Services (AWS) account. See Configure the additional external storage, on page 51.
- Crosswork Network Controller uses a set of rules, expressed in low-code format and saved in packages called heuristics packages to monitor the health of the services.
  - A Heuristic Package contains what to monitor, how to compute the monitored metrics, and symptoms associated with service health degradation. The overall health of the service is determined by applying the rules from the Heuristic Package.
  - The default Heuristic Packages provided with Crosswork Network Controller are referred to as system packages and cannot be altered. Crosswork Network Controller uses these system packages' predefined rules to deploy various testing probes, including Y.1731, TWAMP, SR-PM, and Provider Assurance Connectivity (previously known as Accedian Skylight), to evaluate service health and determine whether the service complies with the Service Level Agreement (SLA) (applicable only to Provider Assurance Connectivity probes).

    If the default system packages do not fully meet your needs, you have the flexibility to customize them to better suit your specific requirements. Export an existing package, modify it, and import it to create a custom Heuristic Package. See Heuristic Packages, on page 55.
- **Monitoring type** filtering for VPN services is available when using the Service Health interface, allowing you to display VPN services based on their monitoring type: Basic or Advanced. This filtering capability enables rapid identification of the Heuristic Package applied to each service and supports efficient monitoring management.

  For example, to quickly review all VPN services configured with Advanced monitoring for compliance checks, apply the Advanced filter to immediately generate a focused list without manually reviewing individual service configurations.
- Extended CLI support using Crosswork Network Controller's system device packages allows for more comprehensive service monitoring capabilities. These packages are capable of deriving exact sensor paths for metric health calculation, and can be installed as a bundle. To add or extend CLI-based KPI collections, you will need support from Cisco Professional Services. Engage with your Cisco account team for more details regarding this.

Getting started

Service Health is available as part of the Crosswork Network Controller Advantage Package (refer to the *Get Started* chapter in the Crosswork Network Controller 7.2 Installation guide).

Summary

You need a functional Crosswork Network Controller environment with devices onboard and services provisioned before you can start monitoring services. This workflow includes links to documents and processes needed to accomplish those tasks, which are beyond the scope of this document.

Workflow

To set up and start monitoring services, complete Steps 1 through 6. Steps 7 to 9 are optional and cover advanced use cases.

1. Install Crosswork Network Controller Advantage package.
    - See the Crosswork Network Controller 7.2 Installation guide.

2. Do the basic reachability checks from the Crosswork Network Controller UI.
    - See Setup workflow in the Crosswork Network Controller 7.2 Administration guide.

3. Create and provision the required Layer 2 VPN (L2VPN) and Layer 3 VPN (L3VPN) services.
    - You can create and provision services using both the Crosswork Network Controller UI or using APIs:
        - *Orchestrated Service Provisioning* chapter in the Crosswork Network Controller 7.2 Solution Workflow Guide.
        - Crosswork Network Controller API Documentation on Devnet.

4. Determine if you would like to configure additional external storage.

Note You can configure external storage at any time.

- If you anticipate monitoring health of many services, Cisco recommends configuring external storage after you install Service Health and before you begin monitoring the services.

  See Workflow: Managing stored data, on page 7.

5. Enable health monitoring for the provisioned services.
    - Start monitoring VPN services.

      See Start the Service Health monitoring, on page 15.

6. Establish your operational processes for responding to degraded services.
    - Deep dive into the impacted services and subservices health, and drill down to the root cause of the service degradation.

      See Workflow: Analyzing the cause of service degradation, on page 8.

7. (Optional) Use SR-PM to probe and monitor links and TE policies in the network.
    - Use SR-PM to measure performance metrics of TE policies and links.

      See Workflow: Using SR-PM for monitoring links and TE policies, on page 9.

8. (Optional) Use Provider Connectivity Assurance to probe Service Health.
    - Using external probes from Provider Connectivity Assurance can provide additional insights into the health of the service.

Note Provider Connectivity Assurance integration is available as a limited-availability feature in this release. Engage with your account team for more information.

See Workflow: Monitoring Service Health using Cisco Provider Connectivity Assurance (formerly Accedian Skylight), on page 9.

9. (Optional) Customize and import Heuristic Packages.
   - Service Health offers a default set of Heuristic Packages for monitoring. If these packages do not fully meet your needs, you have the option to customize these packages to align with your specific requirements..

     See Workflow: Customizing Heuristic Packages, on page 10.

Monitor service health using these workflows

Use this section to perform procedures in different scenarios and functionalities described in the Getting started section.

- Workflow: Managing stored data, on page 7
- Workflow: Analyzing the cause of service degradation, on page 8
- Workflow: Using SR-PM for monitoring links and TE policies, on page 9
- Workflow: Monitoring Service Health using Cisco Provider Connectivity Assurance (formerly Accedian Skylight), on page 9
- Workflow: Customizing Heuristic Packages, on page 10

Workflow: Managing stored data

Crosswork Network Controller provides 50 gigabytes (GB) of storage for monitoring data. If storage reaches its limit, the system deletes the least recently used monitoring data first.

Summary

When the storage exceeds 70% capacity, Crosswork Network Controller generates an alarm prompting you to configure external storage in order to save older monitoring data. The actions detailed in the section describe how to monitor storage usage, reduce the amount of data being stored and how to add additional external storage.

Workflow

To manage stored date, use this workflow:

1. Reduce the number of services being monitored by stopping the monitoring for a few services. Review the monitoring data already stored on your system. Delete any data you do not need to free up storage space.

- Stop Service Health monitoring, on page 20

2. Switch services that are using Advanced Monitoring to Basic Monitoring to monitor the services in lesser detail.

    - Edit the monitoring settings, on page 17

3. If you still need additional storage, configure additional external storage on Amazon Web Services (AWS) Cloud.

    - Configure the additional external storage, on page 51

Workflow: Analyzing the cause of service degradation

To analyze the cause of service degradation is an operational workflow and it is iterative.

Summary

Explore the impacted services and subservices' health, and examine the root cause of the service degradation using any of these methods.

Workflow

1. View monitored services and identify degraded services.

    - View monitored services, on page 33

2. Identify cause of the service degradation.

    - Identify the root causes using last 24 Hr metrics, on page 45
    - Identify the active symptoms and root causes of a degraded service, on page 38
    - Use the Assurance Graph to identify root causes, on page 43

3. Confirm if the reported degradation is a valid issue. If the issue is not valid, you may need to adjust the monitoring level—Basic Monitoring or Advanced Monitoring—to ensure accurate reporting of a service's health.

    Alternatively, modify the system Heuristic Package to create a custom Heuristic Package, which helps resolve false positive reporting of a service's health.

    If the reported issue is valid, proceed to the next step.

    - Edit the monitoring settings, on page 17
    - Heuristic Packages, on page 55

4. Analyze if the service degradation is on account of an issue with device health.

    - View the devices participating in the service, on page 48
    - View the collection jobs, on page 49

Workflow: Using SR-PM for monitoring links and TE policies

To measure the performance metrics of links and TE policies, Crosswork Network Controller can leverage Segment Routing Performance Measurement (SR-PM).

Summary

When this feature is enabled, Crosswork Network Controller gathers and processes additional metrics such as Delay, Delay Variance, and Liveness to compute the health and determine if any of the metrics have crossed the threshold compliance.

Workflow

The following workflow describes how to enable SR-PM collection and view performance metrics collected using SR-PM.

1. Enable SR-PM metrics collection in Crosswork Network Controller.
   - Enable the collection of SR-PM metrics , on page 21

2. View the performance metrics of the links and TE policies.
   - TE policies: View the performance metrics of the TE policies, on page 22
   - Links: View the performance metrics of links, on page 24

3. Ensure that the health of the policy or link is reported accurately without any false issues. If false reporting of degradation is observed, you can create a custom Heuristic Package by modifying the system Heuristic Package to provide customized and accurate health reporting.
   - Heuristic Packages, on page 55

Workflow: Monitoring Service Health using Cisco Provider Connectivity Assurance (formerly Accedian Skylight)

Crosswork Network Controller can use external probing from Cisco Provider Connectivity Assurance (formerly Accedian Skylight) to measure performance metrics of the L3VPN services.

Note Monitoring L3VPN services using Provider Connectivity Assurance is supported only with Advanced monitoring and requires a Provider Connectivity Assurance Essentials license. See Provider Connectivity Assurance Licensing Tiers for more information.

Summary

The performance metrics of the L3VPN services are compared with the contracted service-level agreement (SLA, defined in the Heuristic Package) with the results accessible on the UI for further analysis.

Workflow

To add Provider Connectivity Assurance as a provider in Crosswork Network Controller, follow steps 1 and 2 in the workflow. Follow the remaining steps iteratively for operational purposes.

1. Install the Provider Connectivity Assurance Solution.
   - Refer to the Provider Connectivity Assurance Solution documentation and the Provider Connectivity Assurance installation guide for information on installing the Provider Connectivity Assurance solution and deploying it with Crosswork Network Controller.

Note Sign up and create an account with the self sign-up tool to access the Provider Connectivity Assurance documentation.

2. Add Provider Connectivity Assurance as a provider in Crosswork Network Controller.
   - Add the Provider Connectivity Assurance as a provider, on page 25
3. Set up probe sessions for the L3VPN service.
   - Monitor Service Health using Cisco Provider Connectivity Assurance (formerly Accedian Skylight), on page 24
4. View the metrics in the Crosswork Network Controller UI.
   - View the probe session details, on page 27
5. Analyze the cause of the service degradation.
   - Identify the active symptoms and root causes of a degraded service, on page 38
6. Confirm whether the reported degradation is valid. If it is not valid, modify the system Heuristic Package to create a custom Heuristic Package for a customized report on service health.
   - Workflow: Customize Heuristic Packages

Workflow: Customizing Heuristic Packages

Crosswork Network Controller uses Heuristic Packages as the core logic to monitor and report the health of services. Heuristic Packages define a list of rules, configuration profiles, supported subservices and associated metrics for every service type. Heuristic Packages provided by the system are read-only and cannot be modified.

Summary

If you find that the Heuristic Packages provided by the system do not meet your monitoring requirements, in terms of monitoring metrics or monitoring thresholds, you can create a customized Heuristic Package that caters to your specific monitoring requirements using the procedures in this workflow.

Customizing Heuristic Packages is not included in the standard Day 2 support responsibilities. For assistance, please reach out to the Cisco account team or contact Cisco Professional Services.

Workflow

To customize Heuristic Packages, follow the workflow:

1. Analyze your network services to identify monitoring requirements.

   Check the system Heuristic Packages for rules, subservices, and metrics to ensure that the system packages do not already include the necessary metrics, services, or thresholds.

   Determine the package that most closely matches the conditions you wish to identify in your network.

   - Basic and advanced monitoring rules, subservices, and metrics
   - Service and subservice dependency example

2. Export the package or packages that include the functions you want to use.

   - Heuristic Packages, on page 55

3. Using the supplied packages as your template, build a new package that gathers the data you need to make determinations about the health of the service you want to monitor. In the simplest use case, you may simply need to edit the threshold points based on the service level agreements (SLAs) used in your network. In more complicated use cases, you might need to build a Heuristic Package from scratch.

   - Create the custom Heuristic Package, on page 57

4. Import the customized Heuristic Package in Crosswork Network Controller.

   - Import the custom Heuristic Packages, on page 59

5. Apply the custom package to each service that requires it.

   - Start the Service Health monitoring, on page 15
   - Edit the monitoring settings, on page 17

6. Verify that the custom package is providing the monitoring data that you need to meet your requirements.

   - View monitored services, on page 33

Service Health audit logging

Crosswork Network Controller provides enhanced audit logging capabilities for Service Health operations. The system includes the Source IP address for specific changes, allowing you to track the originating IP address of users making modifications using either the UI or API. This provides a comprehensive audit trail.

This increased visibility is critical for maintaining accountability and strengthening operational security across various Service Health configurations.

Service Health audit logs record the Source IP for these actions:

- **Importing custom Heuristic Packages**: identifies the source of new or updated custom packages.
- **Enabling or disabling service monitoring**: records who initiated changes to the monitoring status of services.

- **Changing monitoring levels**: tracks modifications to the granularity or type of service monitoring.

This logging capability provides a clear record of administrative actions and serves purely for auditing purposes without impacting system operation.

Viewing source IP in Service Health audit logs

The inclusion of source IP in Service Health audit logs does not require explicit configuration. You can access and review these enhanced audit logs through the Crosswork Network Controller UI.

Before you begin

To view source IP in Service Health audit logs:

Procedure

Step 1 From the main menu, choose **Administration** > **Audit Logs**.

Step 2 Filter or search for entries related to Service Health, Heuristic Packages, or service monitoring actions.

Step 3 The audit log entries for these actions now include the source IP address. This information helps identify the originating IP of the user who performed the action.

Service Health Monitoring scale information

You can monitor a maximum of 52,000 services in total. This means you may monitor either 52,000 services using only Basic Monitoring, or a combination of Basic and Advanced Monitoring up to 52,000 services total, with no more than 2,000 using Advanced Monitoring.

Table 1: Monitoring support

Type of monitoring	Supports
Basic Monitoring	52,000 services
Advanced Monitoring	2,000 services

Note For large Layer 3 (L3) VPN deployments, we support either Basic or Advanced monitoring for up to three large VPNs, with a maximum of 4,000 VPN nodes and up to 20,000 endpoints per deployment.

Note If you enable large VPN monitoring services while L2 and L3 services are still being discovered, health reporting may be delayed by up to nine hours because of heavy system load and concurrent processing. For best practices, workflow steps, and examples of VPN health reporting that typically completes in one hour, see the Enabling large VPN services in Service Health article.

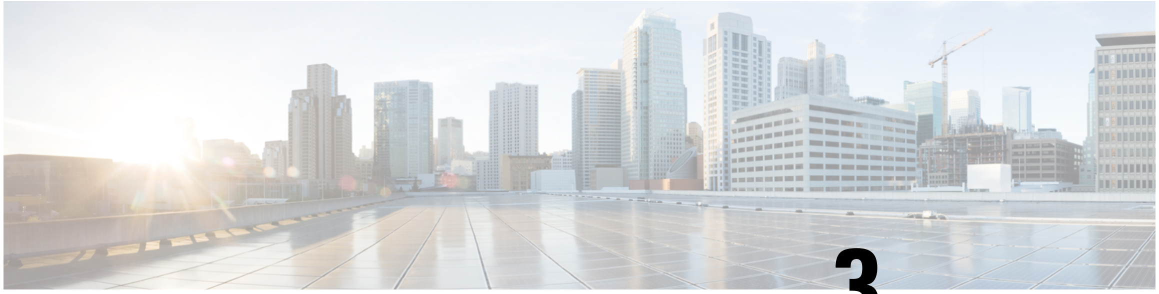
Service Health Monitoring single-VM scale information

You can monitor a maximum of 2,200 services using Basic Monitoring and Advanced Monitoring, with 200 of those services using Advanced Monitoring. In addition, one L3VPN (more than 200 nodes) service and 200 probe sessions for end-to-end monitoring are available.

For more information on Service Health Monitoring single virtual machine (VM) support, see the Crosswork Network Controller 7.2 Administration guide.

Table 2: Single-VM scale information

Type of monitoring	Supports
Basic Monitoring	2,000 services
Advanced Monitoring	200 services
L3VPN (up to 200 nodes)	1 service
Probe sessions for end-to-end monitoring	200 sessions



CHAPTER 3

Monitor Service Health

This chapter covers the following topics:

Start the Service Health monitoring

Start monitoring the health of a service.

Before you begin

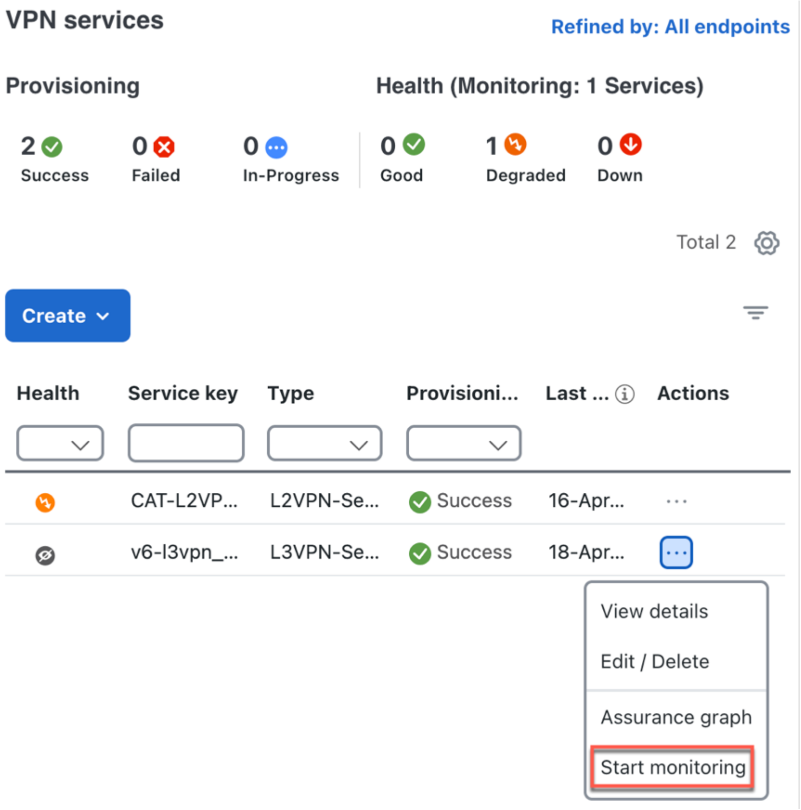
The following procedure assumes that you have already provisioned L2VPN and L3VPN services. To create and provision services, refer to the Orchestrated Service Provisioning chapter in the Cisco Crosswork Network Controller 7.2 Solution Workflow Guide.

Procedure

Step 1 From the main menu, choose **Services & Traffic Engineering** > **VPN Services**. The map opens on the left side of the page and the table opens on the right side.

Step 2 For a service not currently being monitored as indicated by a gray icon in the **Health** column for which you wish to enable monitoring, click ⋯ in the **Actions**.

Step 3 Click **Start monitoring**.



VPN services

Refined by: All endpoints

Provisioning: 2 Success, 0 Failed, 0 In-Progress

Health (Monitoring: 1 Services): 0 Good, 1 Degraded, 0 Down

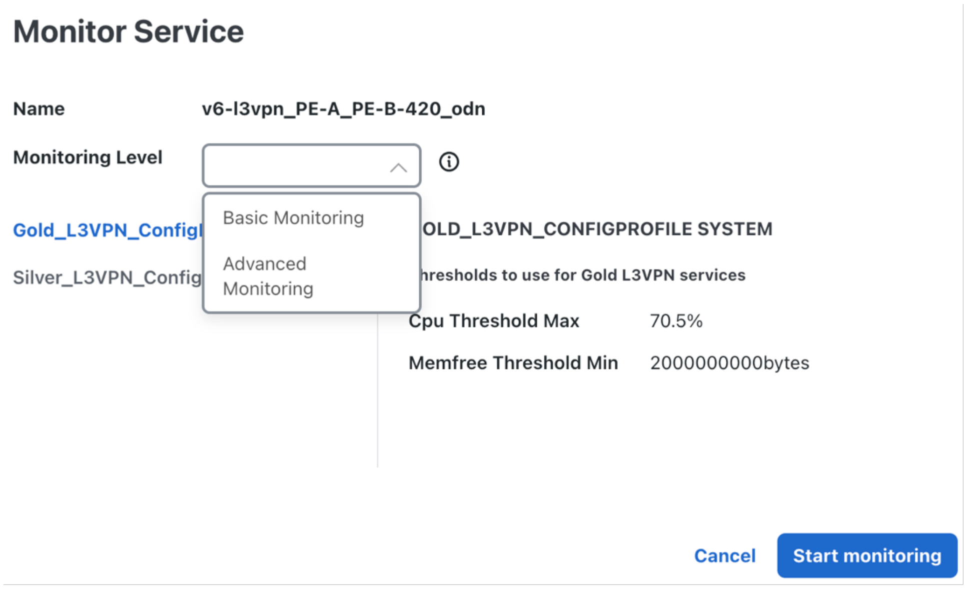
Total 2

Health	Service key	Type	Provisioni...	Last ...	Actions
	CAT-L2VP...	L2VPN-Se...	Success	16-Apr...	···
	v6-l3vpn_...	L3VPN-Se...	Success	18-Apr...	···

Step 4 In the Monitor Service window that appears:

a) Select the **Monitoring level** as **Basic Monitoring** or **Advanced Monitoring**.

b) Select a configuration profile from the list to apply it for monitoring the service.



Monitor Service

Name v6-l3vpn_PE-A_PE-B-420_odn

Monitoring Level

Gold_L3VPN_Config

Silver_L3VPN_Config

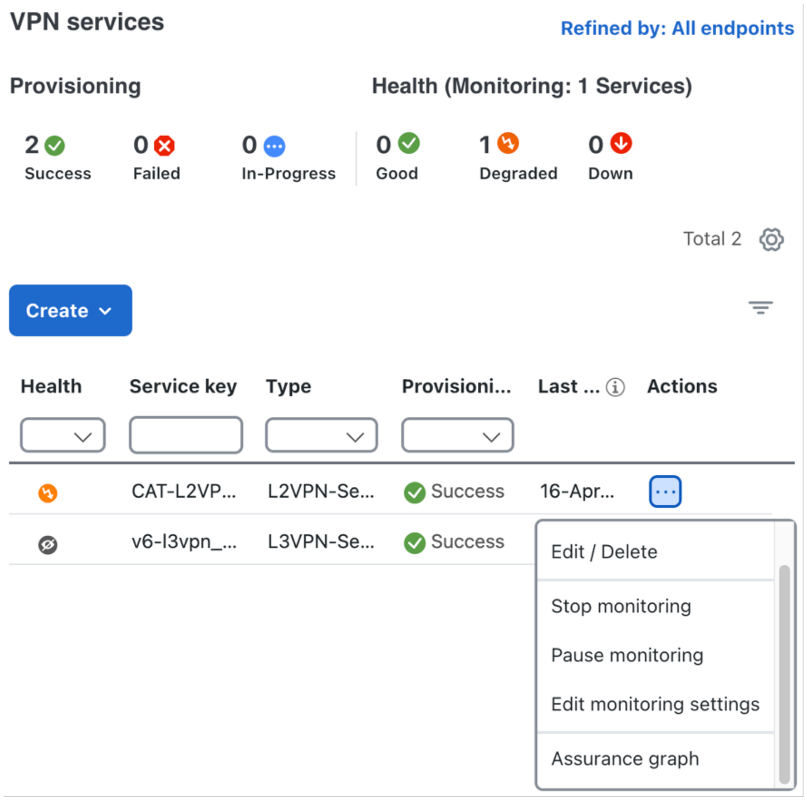
OLD_L3VPN_CONFIGPROFILE SYSTEM

hresholds to use for Gold L3VPN services

Cpu Threshold Max 70.5%

Memfree Threshold Min 2000000000bytes

Step 5 Click **Start monitoring**. The **Health** column reflects the health of the service.



VPN services

Refined by: All endpoints

Provisioning: 2 Success, 0 Failed, 0 In-Progress

Health (Monitoring: 1 Services): 0 Good, 1 Degraded, 0 Down

Total 2

Health	Service key	Type	Provisioni...	Last ...	Actions
Degraded	CAT-L2VP...	L2VPN-Se...	✓ Success	16-Apr...	···
	v6-l3vpn_...	L3VPN-Se...	✓ Success		

What to do next

If the health of the service is degraded, identify the root cause for service degradation and take measures to correct the issue. See Analyze Service Health, on page 33 for more information.

Adjust monitoring settings

The following topics explain the various monitoring settings you can use to adjust Service Health monitoring.

Edit the monitoring settings

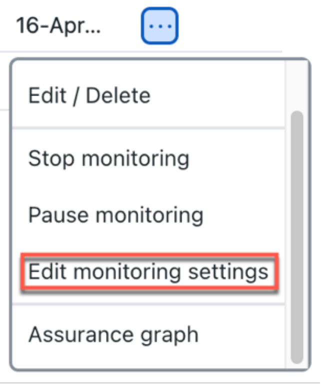
You can update the monitoring level for the service from Basic Monitoring to Advanced Monitoring, or from Advanced Monitoring to Basic Monitoring. You can also update the configuration profile, changing the Gold profile to Silver profile or the Silver profile to Gold profile. See Heuristic Packages, on page 55 for information about configuration profiles.

Procedure

Step 1 From the main menu, choose **Services & Traffic Engineering** > **VPN Services**. The map opens on the left side of the page and the table opens on the right side.

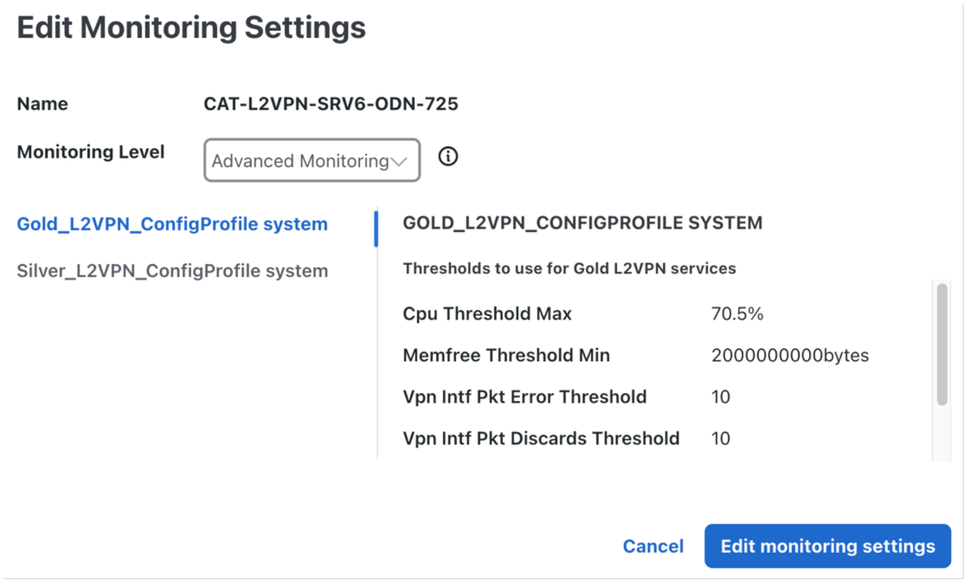
Step 2 In the Actions column, click ⋯ for the service for which you want to edit the monitoring settings.

Step 3 Choose **Edit monitoring settings** from the menu.



The Edit monitoring settings dialog box appears.

Step 4 Choose the **Monitoring level** or the **Configuration profile**, as required.



Edit Monitoring Settings

Name CAT-L2VPN-SRV6-ODN-725

Monitoring Level Advanced Monitoring

Gold_L2VPN_ConfigProfile system

Silver_L2VPN_ConfigProfile system

GOLD_L2VPN_CONFIGPROFILE SYSTEM

Thresholds to use for Gold L2VPN services

Cpu Threshold Max	70.5%
Memfree Threshold Min	2000000000bytes
Vpn Intf Pkt Error Threshold	10
Vpn Intf Pkt Discards Threshold	10

Note
When you switch between Advanced and Basic Monitoring, it can take over 15 minutes for subservice health and active symptoms to become visible.

Step 5 Click **Edit monitoring settings**.

A confirmation dialog box appears.

Step 6 Click **Start *monitoring-type* monitoring**.

Crosswork Network Controller starts monitoring the service's health using the updated values.

What to do next

If Crosswork Network Controller reports that the health of the service is degraded, identify the cause of service degradation and take corrective measures. Refer to Analyze Service Health, on page 33 for more information.

Pause and resume the Service Health monitoring

With this option, you can temporarily pause monitoring the health of services. This approach is useful if a service is down due to a reported outage or scheduled maintenance and you do not want to receive notifications about the degradation. If you pause and then resume monitoring, the system continues using the same Basic or Advanced Monitoring rules and profile options as before the pause. Additionally, historical data and End of Service (EOS) are preserved in the service's history. However, since no data is collected while monitoring is paused, there will be gaps in the historical data for the periods when monitoring was paused.

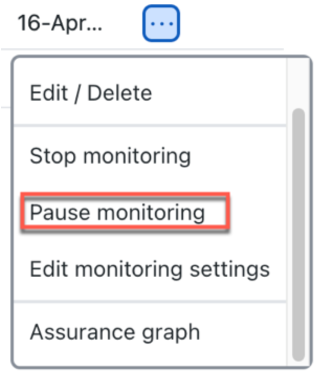
Pause and then resume monitoring a service's health.

Procedure

Step 1 From the main menu, choose **Services & Traffic Engineering** > **VPN Services**. The map opens on the left side of the page and the table opens on the right side.

Step 2 In the Actions column, click ⋯ for the service that you want to pause the monitoring for.

Step 3 Choose **Pause monitoring** from the menu.



A confirmation dialog box appears. Click **Pause monitoring**.

Note
When monitoring is paused, you can still view the Assurance Graph, which will show only the top-level service with a paused icon badge and no child subservices underneath.

Step 4 In the Actions column, if you now click ⋯ for the service that you paused, you will see the **Resume monitoring** option. Click this option to resume monitoring the health of the service.

A confirmation dialog box appears. Click **Resume monitoring**.

When Crosswork Network Controller resumes monitoring a service after a pause, it utilizes the same monitoring rules and profile options that were in place before the pause.

Stop Service Health monitoring

When you choose to stop monitoring a service, the system will prompt you to confirm whether you wish to retain the historical monitoring data. The following options are available:

- **Retain historical data**: If you choose to retain the historical data, all monitoring information collected prior to the stoppage will remain accessible. This data will be preserved and available for analysis when monitoring is resumed. The monitoring settings will also be retained, ensuring a seamless transition back to active monitoring with historical context.
- **Do not retain historical data**: If you decide not to retain the historical data, all monitoring settings and historical data will be purged from the database. This action will also delete the Assurance Graph for the stopped service. Subsequent monitoring of the service will start anew, without any reference to previous data.

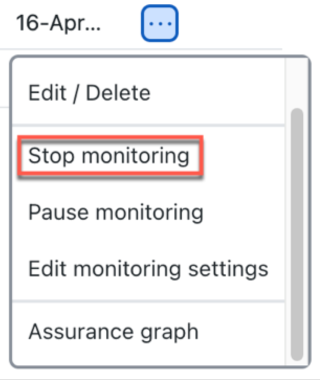
Stop monitoring the health of a service.

Procedure

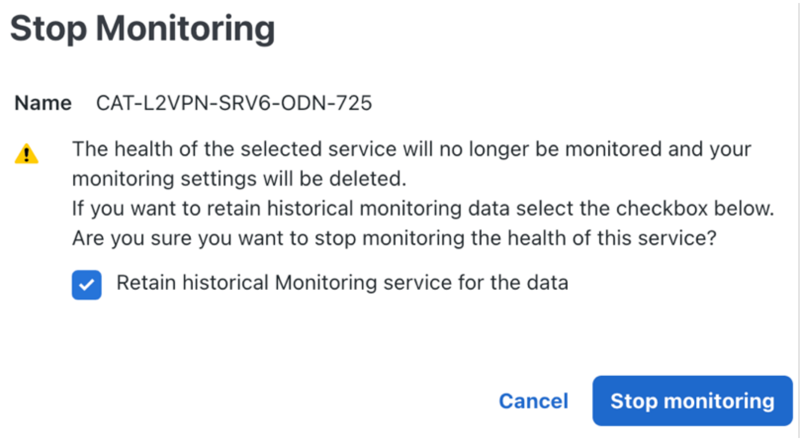
Step 1 From the main menu, choose **Services and Traffic Engineering** > **VPN Services**. The map opens on the left side of the page, and the table opens on the right side.

Step 2 In the **Actions** column, click ⋯ for the service you want to stop monitoring.

Step 3 Choose **Stop monitoring** from the menu.



Step 4 The Stop Monitoring dialog box appears. To retain historical service data, select the **Retain historical Monitoring service for the data** check box.



Stop Monitoring

Name CAT-L2VPN-SRV6-ODN-725

The health of the selected service will no longer be monitored and your monitoring settings will be deleted.
If you want to retain historical monitoring data select the checkbox below.
Are you sure you want to stop monitoring the health of this service?

☑ Retain historical Monitoring service for the data

Step 5 Click **Stop monitoring**.

Step 6 If you stopped monitoring a service and selected **Retain historical monitoring service for the data**, you can later start monitoring that service with historical data still available. From the **Actions** column of the service, click ⋯ and select **Start monitoring**.

Enable SR-PM monitoring for links and TE policies

To measure the performance metrics of links and Traffic Engineering (TE) policies Segment Routing Multiprotocol Label Switching (SR-MPLS), Resource Reservation Protocol-Traffic Engineering (RSVP-TE), Service Health leverages the Segment Routing Performance Measurement (SR-PM) feature. This feature enhances the capabilities for troubleshooting and health analysis by providing detailed, historical, and consolidated views of links in the network and transport path metrics. This enables network and service operators to proactively manage, troubleshoot, and optimize network infrastructure.

Enable the collection of SR-PM metrics

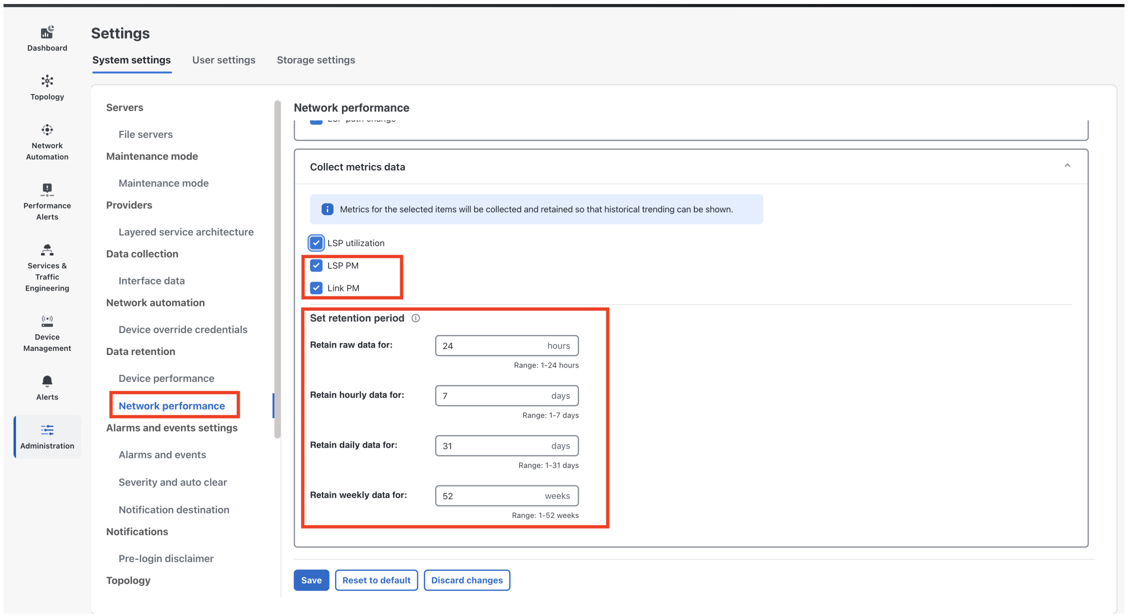
Enable the SR-PM metrics collection.

Procedure

Step 1 From the main menu, select **Administration** > **Settings** > **Data retention** > **Network Performance**. The **Network performance** pane opens on the right.

Step 2 Under **Collect metrics data**, select:

- **LSP PM** - to enable metrics collection for SR policies
- **Link PM** - to enable metrics collection for links



Network performance

Collect metrics data

Metrics for the selected items will be collected and retained so that historical trending can be shown.

☑ LSP utilization

☑ LSP PM

☑ Link PM

Set retention period

Retain raw data for: 24 hours
Range: 1-24 hours

Retain hourly data for: 7 days
Range: 1-7 days

Retain daily data for: 31 days
Range: 1-31 days

Retain weekly data for: 52 weeks
Range: 1-52 weeks

Step 3 To retain historical data and view trends of these metrics, select the duration for which data should be collected and retained.

Note
Metric data is collected and retained only for the options for which you have enabled SR-PM metric collection.

View the performance metrics of the TE policies

SR-PM data collection is supported for SR-MPLS, SR-CS, and RSVP-TE policies. The metric data is used to assess policy health and to indicate if any metrics have violated SLAs defined in the Heuristic package. You can view the KPI metrics and the operational and administration status of the service on the policy tab in the **Service Details** page. If you have enabled data retention, the historical data and trends are available in the **History** tab.

These metrics are collected for TE policies when SR-PM collection is enabled:

- Delay—available only for SR-MPLS and RSVP-TE policies
- Delay Variance (jitter)—available only for SR-MPLS and RSVP-TE policies

Note The RSVP policy under **Transport** > **RSVP** tab in VPN Services and the **Traffic Engineering** > **RSVP** tab represent the same Traffic Engineering policy. Both pages display RSVP Performance Measurement (PM) metrics with identical values. However, the Threshold label appears only in the VPN Services – Transport tab when Service Health monitoring is enabled and the device has delay measurement configured for the policies.

Steps to view KPI metrics for a TE policy.

Before you begin

Ensure that you have taken care of the following to view metrics from SR-PM collection:

- Added devices, TE policies and created device groups.
- Enabled SR-PM collection in Crosswork Network Controller and optionally enabled data retention to view historical data and trends.
- Enabled SR-PM metric collection on devices.

Note Refer to the device-specific documentation for details. These details are beyond the scope of this guide.

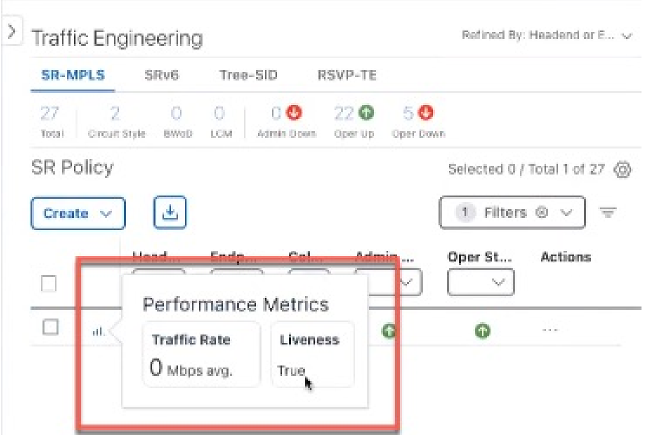
Procedure

Step 1 Navigate to the Traffic Engineering topology map. From the main menu, select **Services and Traffic Engineering** > **Traffic Engineering**.

Step 2 Click the policy tab that you are interested in.

For example, to view policy performance metrics for SR-MPLS policies, click the SR-MPLS tab.

Figure 1: SR-MPLS policy performance metrics in the Traffic Engineering table



Step 3 Hover your mouse over the graph icon to view the KPI metrics in a carousel view. Alternatively, locate the policy that you are interested in from the TE table. In the **Actions** column, click ⋯ > **View Details**. The **Service Details** page opens and displays the KPI metrics for the policy in the **Performance metrics** section.

Step 4 To view historical data, click the **History** tab. A chart showing the trends is displayed for each metric here. Click a time frame in the chart to view the trend of the policy during the selected period.

View the performance metrics of links

Link interface metrics are a set of indicators that measure the performance and quality of the communication between two or more network devices. They include parameters such as bandwidth, delay, jitter, and packet loss. Link interface metrics can help network administrators monitor and troubleshoot network issues, optimize network resources, and plan for future network expansion or upgrade.

This procedure is for viewing the link metrics.

Before you begin

Ensure that you have onboarded devices and created the required device groups.

Procedure

Step 1 From the main menu choose **Topology**.

Step 2 Select a link to view its details in any of the following ways:

a) By clicking a link on the topology map
b) By clicking a link from the **Links** tab in the topology map
c) By clicking a link from the **Links** tab in the **Device Details** page.

The **History** tab provides useful insights into the performance and trends of the network. You can select the time interval to analyze the data.

Monitoring health of services using CS-SR policies

Crosswork Network Controller supports monitoring the health of L2VPN point-to-point services (only IETF:L2VPN:EVPN VPWS) using Circuit-Style Segment Routing (CS-SR) policies.

When the L2VPN service is configured to use circuit-style transport, Crosswork Network Controller automatically initiates monitoring of the service in both directions (A–Z and Z–A) using the `subservice.cssr.policy.health` subservice.

The subservice monitors and reports the **Admin Status**, **Operational Status**, and any flip-flops in the operational status. The operational state of the CS-SR policy is measured using the Liveness metric. This measures if the path is live and capable of carrying traffic, providing a simpler yet effective way to ensure the health of the path.

Monitor Service Health using Cisco Provider Connectivity Assurance (formerly Accedian Skylight)

Crosswork Network Controller can leverage external probing, provided by Cisco Provider Connectivity Assurance (formerly Accedian Skylight), to measure metrics of the L3VPN services in the network. The metrics are compared with the contracted service-level agreement (SLA) (defined in the Heuristic Package), and the results are made available on the user interface (UI) for further analysis.

Monitoring L3VPN services using Provider Connectivity Assurance is only possible with Advanced Monitoring and requires a Provider Connectivity Assurance Essentials license. See Provider Connectivity Assurance Licensing Tiers for more information. Sign up and create an account using the self sign-up tool to access the Provider Connectivity Assurance.

High-level flow

1. When you provision an L3VPN service with probe intent and enable service monitoring, the Orchestrator component of Provider Connectivity Assurance learns the probe intent and the probe topology from the provisioned service.

   The following probe intents are supported:

   - **Agent configurations**: ne-id, VLAN, IP, sub-interface.
   - **Topology**: point-to-point, hub-spoke, full-mesh.

2. Probe sessions with Provider Connectivity Assurance are set up automatically to monitor the service by invoking the relevant RESTConf APIs. The RESTConf APIs that are invoked to provision probe sessions include endpoint, session, service, and session activation. The maximum number of probe sessions per service are capped at 200 (for all connection types).

3. The gateway component of Provider Connectivity Assurance streams the probe metrics to Data Gateway, which collects this data using a parameterized collection job for Service Health over gNMI. The following probe metrics are collected:

   - Forward and Reverse Delay.
   - Forward and Reverse Variance.
   - Forward and Reverse Packet Loss.

4. Metrics collected during probe sessions are analyzed. The system raises symptoms as needed and displays them on the Crosswork Network Controller UI.

Add the Provider Connectivity Assurance as a provider

Set up the necessary credential and certificate profiles for the Provider Connectivity Assurance to facilitate secure communication via HTTPS and gNMI protocols.

Before you begin

Ensure that you have taken care of the following prerequisites before onboarding Provider Connectivity Assurance as a provider:

- Install the Provider Connectivity Assurance software. Refer to the Provider Connectivity Assurance documentation for information about installing Provider Connectivity Assurance and deploying it with Crosswork Network Controller.

Note You need an account with Provider Connectivity Assurance to access the documentation. Sign up and create an account with the self sign-up tool.

- Have the following certificates from Provider Connectivity Assurance downloaded on your local system or in a folder that can be accessed by Crosswork Network Controller:
  - CA certificate
  - Client certificate
  - Client key

Procedure

Step 1 **Create a credential profile.**

a) Navigate to **Administration** > **Device Management** > **Credential Profiles** and click the Add (+) button to create a new profile.
b) Enter a name, add the following credential protocols: **HTTPS** and **gNMI**. Enter the username and password for both connections.
c) Click **Save**.

Step 2 **Create a certificate profile.**

a) Navigate to **Administration** > **Certificate Management** and click the Add (+) button.
b) Enter a name and select the **Certificate Role** as **Accedian Provider Mutual Auth**.
c) Upload the certificates: ca_cert.pem, client_cert.pem, and client_key.key.
d) (Optional) Enter the passphrase for the certificate chain.
e) Click **Save**.

Step 3 **Add Provider Connectivity Assurance as a provider in Crosswork Network Controller.**

a) Navigate to **Administration** > **Manage Provider Access**.
b) Click **+** and enter details in the fields as follows:

- **Provider Name**: Enter a name.
- **Credential profile**: Select the credential profile that you created for Provider Connectivity Assurance.
- **Family**: Select ACCEDIAN_PROXY.
- **Certificate profile** : Select the Provider Connectivity Assurance certificate profile.

  Note
  This field is displayed after you select the **Family** as ACCEDIAN_PROXY.

- **Connection types**: Supported protocols are automatically updated from the Provider Connectivity Assurance credential profile.
- **IP addresses**: Enter the IP address or the Fully Qualified Domain Name (FQDN).

  Important
  If the server certificates present in Provider Connectivity Assurance are generated using a Fully Qualified Domain Name (FQDN), enter the FQDN only in this field. Do not enter an IP address. Entering an IP address when the server certificates are generated with FQDN will cause issues in Provider Connectivity Assurance authentication and reachability.

- **Ports**: Enter 443 for HTTPS and a port value for gNMI.

- **Encoding Type**: Select PROTO.

  Note
  Only encoding of type **PROTO** is supported.

c) Click **Save**.

What to do next

Confirm that the Provider Connectivity Assurance provider is reachable from Crosswork Network Controller.

Check the reachability of the Provider Connectivity Assurance

Resolve all certificate and HTTPS credential issues to ensure Provider Connectivity Assurance is reachable.

Procedure

Step 1 Navigate to **Administration** > **Manage Provider Access** from the main menu.

Step 2 Ensure Provider Connectivity Assurance shows a green reachability status without errors.

If there are certificate errors, the provider will be displayed as **Degraded** and not reachable.

Step 3 Provider Connectivity Assurance might still be displayed as reachable on the Crosswork Network Controller provider's list page in spite of the following issues:

- Invalid HTTPS credentials.
- Incorrect ports, IP addresses, or credentials for the gNMI protocol (since reachability checks for gNMI are not performed).

Step 4 After resolving any certificate or HTTPS credential issues, delete and onboard Provider Connectivity Assurance in Crosswork Network Controller again.

View the probe session details

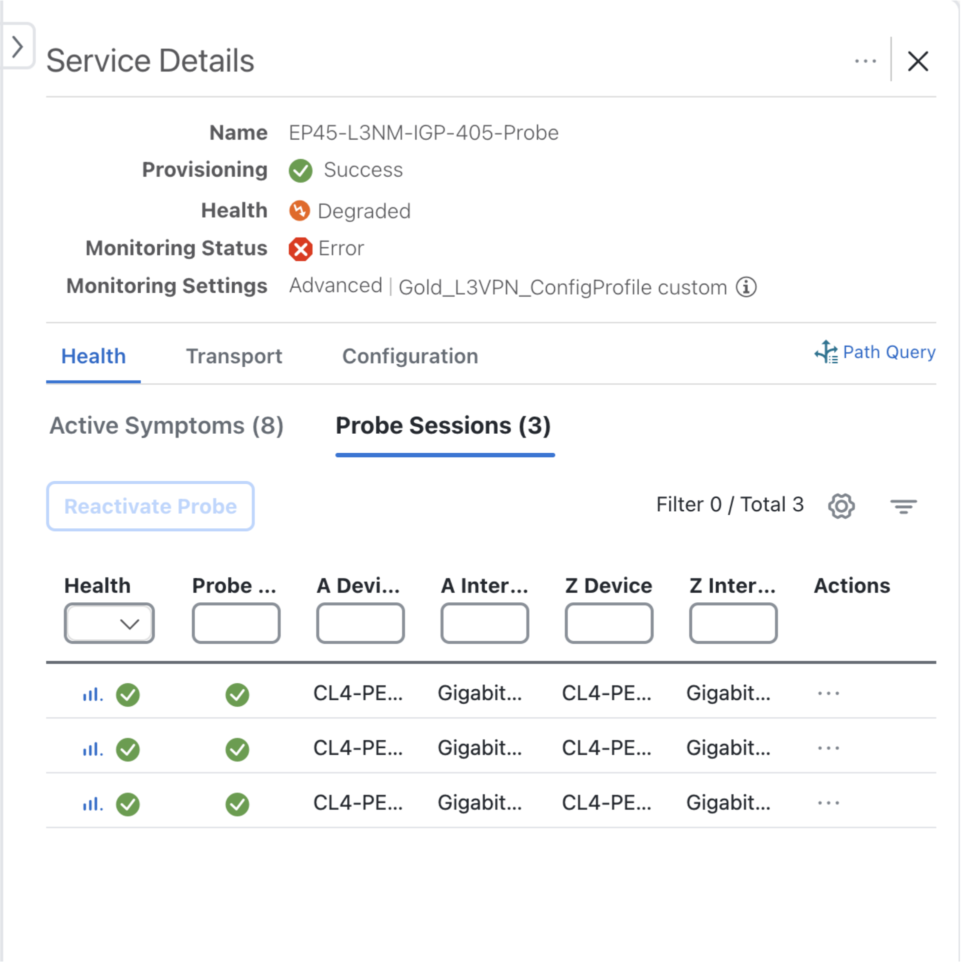
View details from Provider Connectivity Assurance probe sessions for L3VPN services and Y1731 probe sessions for L2VPN services are displayed separately in the **Probe sessions** tab of the service.

Procedure

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 For the service you are interested in, in the **Actions** column, click **View details**.

Step 3 In the **Service details** page that is displayed, click the **Probe sessions** tab.



Service Details

Name EP45-L3NM-IGP-405-Probe

Provisioning ✓ Success

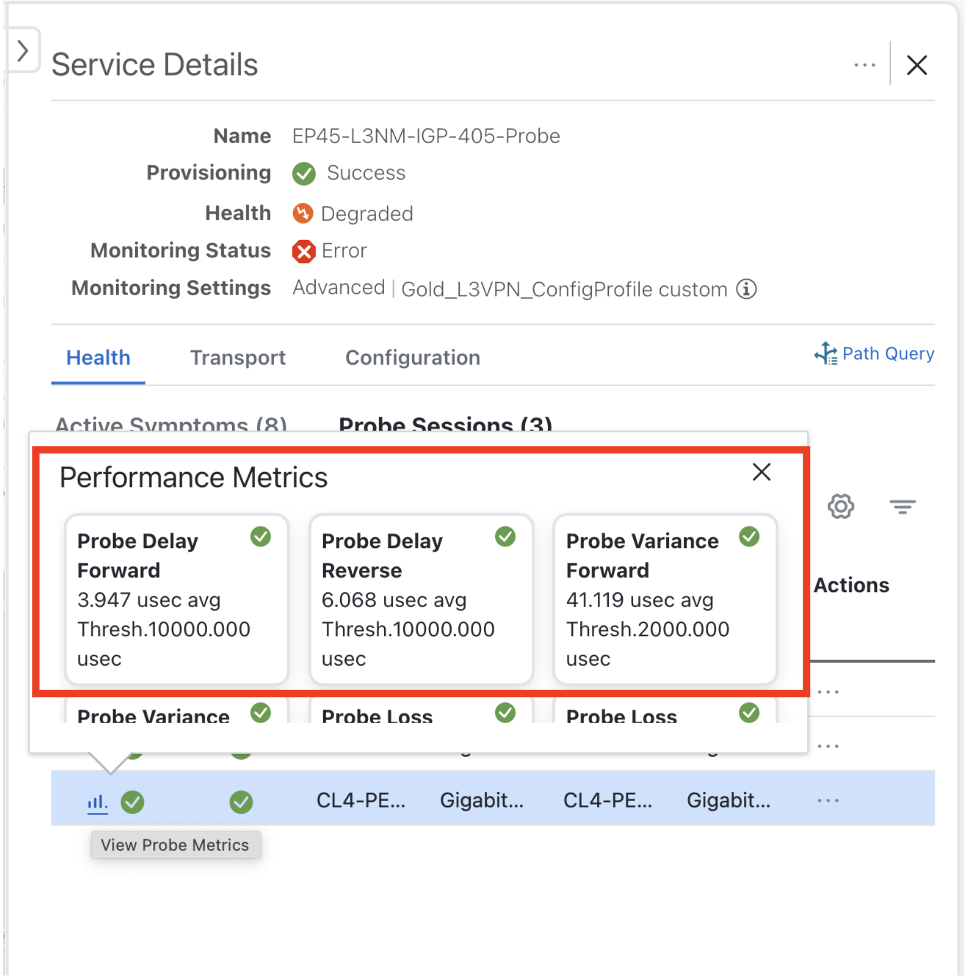
Health Degraded

Monitoring Status ✗ Error

Monitoring Settings Advanced | Gold_L3VPN_ConfigProfile custom ⓘ

Health	Probe ...	A Devi...	A Inter...	Z Device	Z Inter...	Actions
✓	✓	CL4-PE...	Gigabit...	CL4-PE...	Gigabit...	···
✓	✓	CL4-PE...	Gigabit...	CL4-PE...	Gigabit...	···
✓	✓	CL4-PE...	Gigabit...	CL4-PE...	Gigabit...	···

Step 4 Click the graph icon next to a probe session for a detailed view of the performance metrics.



Service Details

Name: EP45-L3NM-IGP-405-Probe
Provisioning: ✓ Success
Health: Degraded
Monitoring Status: ✗ Error
Monitoring Settings: Advanced | Gold_L3VPN_ConfigProfile custom

Performance Metrics

Probe Delay Forward ✓	Probe Delay Reverse ✓	Probe Variance Forward ✓
3.947 usec avg Thresh.10000.000 usec	6.068 usec avg Thresh.10000.000 usec	41.119 usec avg Thresh.2000.000 usec

Probe Variance ✓ | Probe Loss ✓ | Probe Loss ✓

✓	✓	CL4-PE...	Gigabit...	CL4-PE...	Gigabit...	···

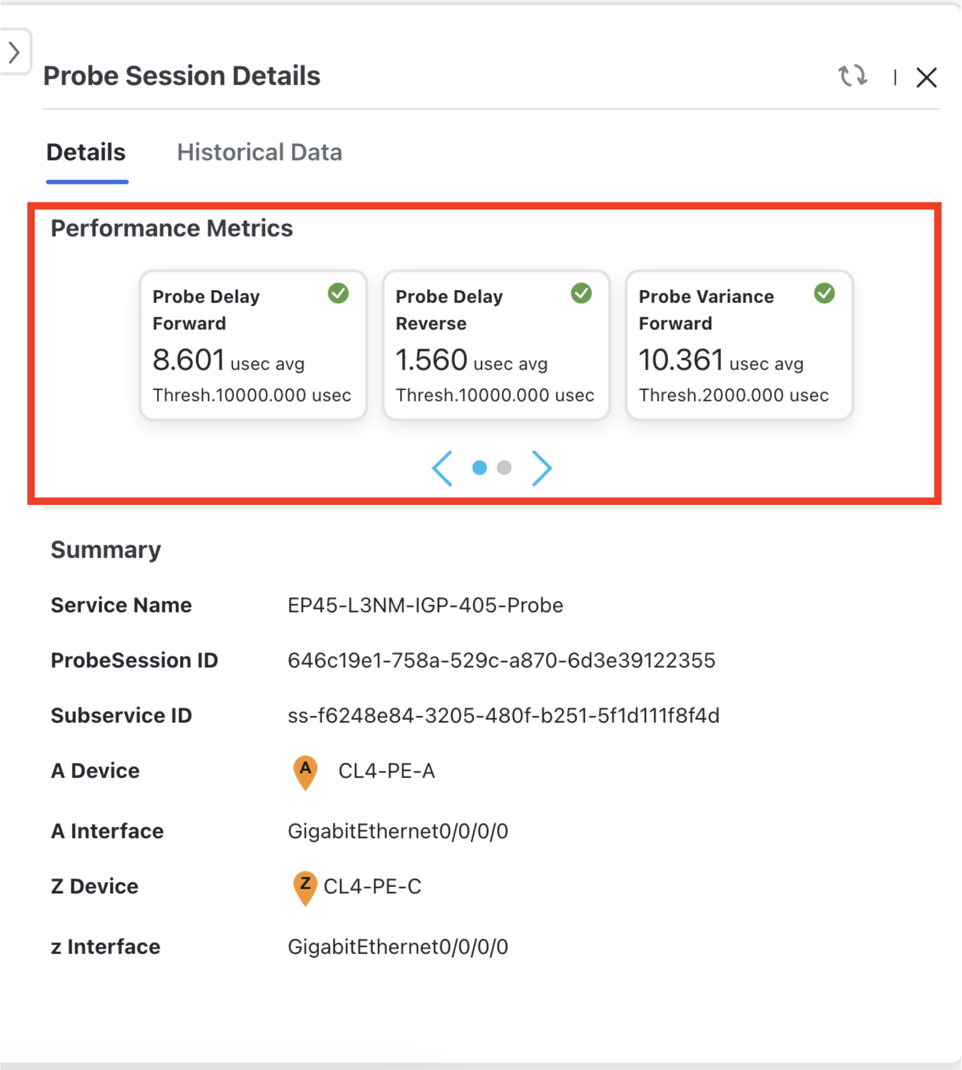
If a metric has crossed the defined threshold, a red icon is displayed in the corresponding performance metrics dashlet.

Step 5 To view the performance metrics for a service in a carousel view, click the ··· icon in the **Actions** column.

The **Probe session details** window opens displaying the metrics in a carousel view.

Note

If there are any probe provisioning errors, the monitoring status of the service is **Monitoring error**. Click the **Reactivate probe** to restart the probe session for the service. If the probe session reactivates successfully, the **Probe sessions** page automatically updates with the new metrics.



Performance Metrics

Probe Delay Forward	Probe Delay Reverse	Probe Variance Forward
8.601 usec avg	1.560 usec avg	10.361 usec avg
Thresh.10000.000 usec	Thresh.10000.000 usec	Thresh.2000.000 usec

Summary

Service Name	EP45-L3NM-IGP-405-Probe
ProbeSession ID	646c19e1-758a-529c-a870-6d3e39122355
Subservice ID	ss-f6248e84-3205-480f-b251-5f1d111f8f4d
A Device	A CL4-PE-A
A Interface	GigabitEthernet0/0/0/0
Z Device	Z CL4-PE-C
z Interface	GigabitEthernet0/0/0/0

The **History data** tab provides probe metrics data ranging from the past 90 days up to the most recent 24 hours. See Historical data from probe sessions, on page 30 for more information.

What to do next

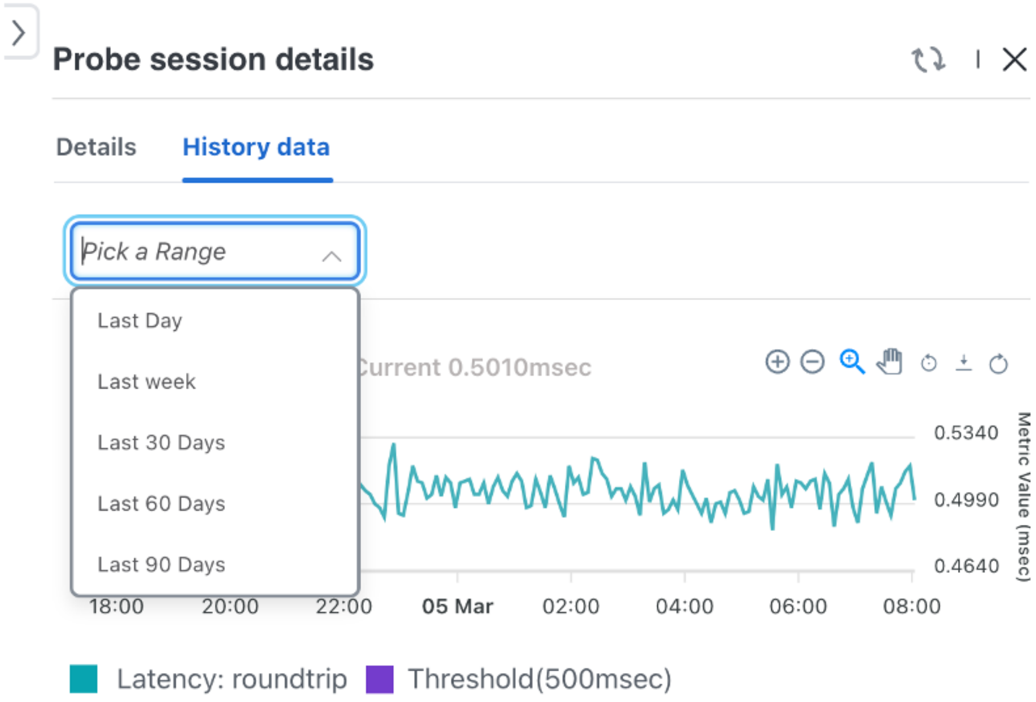
If you find that a service is degraded, analyze the root cause of the degradation to troubleshoot the health of a degraded service. See Analyze Service Health, on page 33 for more information.

Historical data from probe sessions

The historical data from a probe session can be viewed by clicking the **History data** tab in the **Probe session details** page. This tab displays data from the time monitoring was enabled for the service.

If monitoring was stopped and started again later, the **History data** tab will display data only from the time monitoring was restarted. If you chose to retain historical data while monitoring was stopped, that data is preserved and appears in the **Show History** tab of the Assurance Graph of the service and not in this tab.

Charts displaying aggregated average metric data along with their timestamps are shown. To view data from a specific range, select the desired range from the drop-down menu.



Note There may be a difference in the first timestamp displayed in the historical chart metric data. This is because the timestamp displayed is in the operational time zone, while the aggregated timestamp is in UTC format with 00 hours.

Historical probe metrics period	Interval	Aggregate interval
one week (seven days)	two hours	two hours
one month (thirty days)	one day	one day (00 hours, start of the day in UTC)
two months (sixty days)	three days	one day (00 hours, start of the day in UTC)
three months (ninty days)	one week (seven days)	one day (00 hours, start of the day in UTC)

Known issues and limitations with Provider Connectivity Assurance

The following is a list of known issues and limitations when Provider Connectivity Assurance is deployed for probing the health of a service:

1. Entering an IP address instead of the FQDN when server certificates are generated with FQDN will cause issues with provider authentication and reachability. In this case, Provider Connectivity Assurance is shown as **Degraded** in the Crosswork Network Controller Providers list page (**Administration** > **Manage Provider Access**).

2. Provider Connectivity Assurance is always shown as reachable in the Crosswork Network Controller Providers list page despite the following issues in the Provider Connectivity Assurance credentials:

   - Invalid HTTPS credentials

   - Incorrect ports or IP addresses or credentials for gNMI since there are no reachability checks for gNMI

   In these cases, services monitored by Provider Connectivity Assurance probes will have the health as **Degraded** with the symptom as '*Provider Connectivity Assurance provider does not exist in DLM*'. The symptoms are not cleared until you add Provider Connectivity Assurance again, pause and resume the service monitoring.

3. When monitoring is enabled for a service with probe intent but Provider Connectivity Assurance is not added in Crosswork Network Controller, an error about the provider not being available is displayed for each of the probe metrics associated with the subservice.

4. You cannot delete Provider Connectivity Assurance when a probe session is active.

5. The **Active symptoms** tab displays the observed value of the metric at the time the symptom occurred, while the **Probe sessions** tab is constantly updated with the live values of the metrics. Therefore, check the **Probe sessions** tab for the real-time values of the performance metrics.

CHAPTER 4

Analyze Service Health

Use Service Health monitoring to analyze service health with the metrics shown in the UI. Investigate degraded services and subservices to identify the root cause of service degradation.

View monitored services

From the Crosswork Network Controller home page, the **VPN service health** dashlet delivers an at-a-glance summary of VPN service statuses. This enables enhanced visibility and filtering for monitoring status, such as errors, in Service Health.

The **VPN Services** page presents detailed health and monitoring data for all VPN services, enhanced by customizable columns and powerful filters that facilitate focused troubleshooting.

The **Service Health Dashboard** complements these views by aggregating service metrics and highlighting SLA breaches, enabling efficient oversight of network performance and service quality.

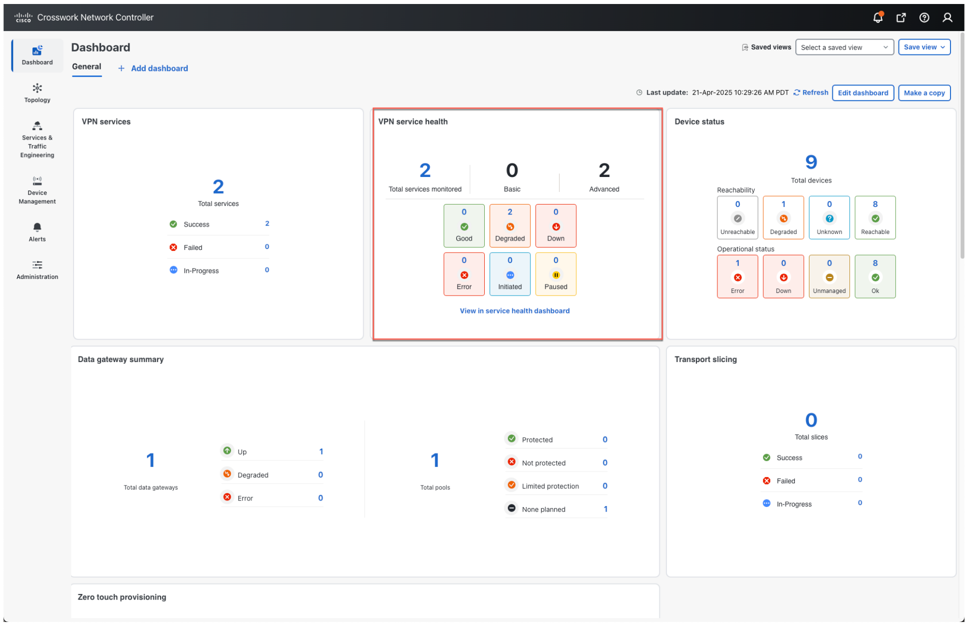
View monitored services from the Crosswork Network Controller home page

You will see the **VPN service health** dashlet on the Crosswork Network Controller home page. This dashlet provides an overview of all monitored VPN services.

Click any status indicators to go to the **VPN Services** page, which displays the selected information with the applied filter.

For example, to isolate VPN services based on their monitoring configuration directly from the Crosswork Network Controller home page, click the number above **Advanced** services in the VPN service health dashlet. The **VPN Services** page opens, filtered by the selected monitoring type.

Figure 2: VPN Service Health dashlet



Only the **Basic** or **Advanced** VPN services are shown on the VPN Services page.

View monitored services from the VPN Services page

The VPN Services page, located under **Services & Traffic Engineering** > **VPN Services**, provides a comprehensive overview of all VPN services, including both L2VPN and L3VPN types.

The VPN services table provides direct indication of monitoring errors, allowing you to quickly identify services experiencing issues without navigating away from the Service Health UI. Services with active monitoring errors are clearly marked in the table, giving immediate insight into their error state.

This page also helps you efficiently monitor service health and status through a variety of filtering and display customization options. The VPN services page includes these key filtering options:

- **Show monitoring error check box**: Quickly filter the list to display only VPN services currently experiencing monitoring errors.

  For example, if you need to focus on services with active issues, selecting this checkbox immediately narrows the view to those requiring attention.

- **Gear icon for column visibility**: Customize which columns appear in the VPN services table. Available columns include:

  - Health
  - Service key
  - Type
  - Monitoring type
  - Provisioning state
  - Monitoring state

- Last updated time
- Date created

Adjusting visible columns helps you focus on the most relevant service attributes.

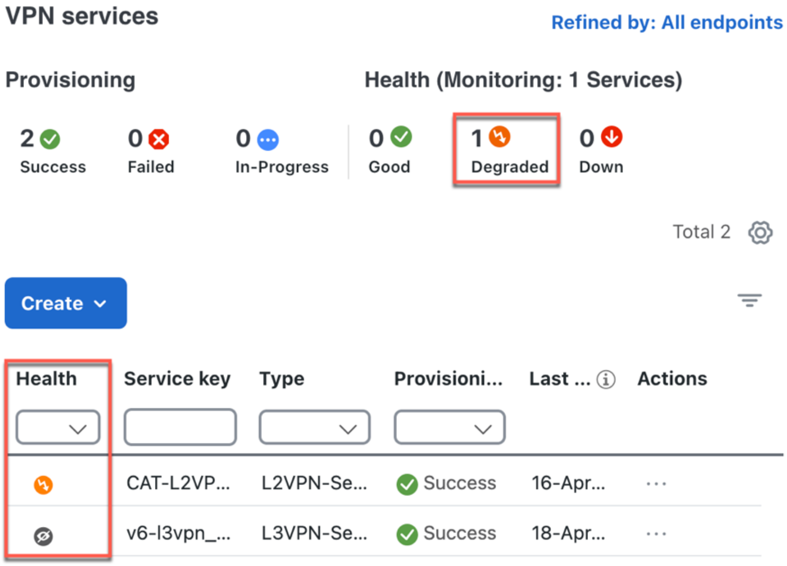
For instance, hiding less critical columns like Date Created can declutter the interface and emphasize health and monitoring details.

- In-column filters: Many columns provide filtering options to refine the displayed services further:
  - **Health**: Down, Degraded, Good, Paused, Initiated, Error, and Unmonitored
  - **Type**: L2VPN-Service, L3VPN-Service
  - **Monitoring type**: Basic, Advanced
  - **Provisioning state**: Success, Failed, In-Progress
  - **Monitoring Status**: Error, Success

    These filters allow you to isolate specific service conditions.

    For example, filtering Health to show only Degraded services helps prioritize troubleshooting efforts.

All the VPN services are listed on this page. The degraded services show an orange icon in the **Health** column.



VPN services

Refined by: All endpoints

Provisioning: 2 Success, 0 Failed, 0 In-Progress

Health (Monitoring: 1 Services): 0 Good, 1 Degraded, 0 Down

Total 2

Create

Health	Service key	Type	Provisioni...	Last ...	Actions
●	CAT-L2VP...	L2VPN-Se...	✓ Success	16-Apr...	···
⊘	v6-l3vpn_...	L3VPN-Se...	✓ Success	18-Apr...	···

To clear the filter, click **X** next to the designated filter at the top of the column. To remove all filters and show all VPN services, click the **X** icon in the Filters field.

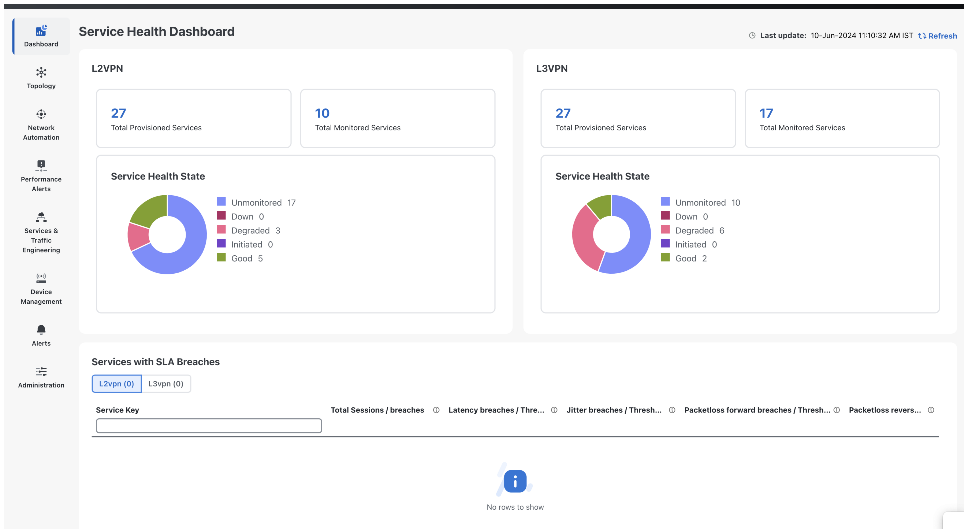
Use these filters to efficiently monitor VPN service health and prioritize troubleshooting, all directly from the VPN Services page.

Note If a service is not yet being monitored, a gray icon is displayed in the Health column. To enable monitoring for such a service, click ⋯ and select **Start monitoring**. For more information, see Start the Service Health monitoring, on page 15.

Use the Service Health Dashboard

To access the Service Health Dashboard from the Crosswork Network Controller home page, click **View in service health dashboard** in the **VPN service health** dashlet. The **Service Health Dashboard** displays a consolidated view of L2VPN and L3VPN services, including the total number of provisioned services and the number of monitored services. The dashboard also displays active monitoring sessions for L2VPN and L3VPN services. It indicates SLA breaches for measured metrics such as latency, jitter, and packet loss in both directions.

Figure 3: Service Health Dashboard



Service Health Dashboard

Last update: 10-Jun-2024 11:10:32 AM IST

L2VPN

27 Total Provisioned Services

10 Total Monitored Services

Service Health State

- Unmonitored 17
- Down 0
- Degraded 3
- Initiated 0
- Good 5

L3VPN

27 Total Provisioned Services

17 Total Monitored Services

Service Health State

- Unmonitored 10
- Down 0
- Degraded 6
- Initiated 0
- Good 2

Services with SLA Breaches

L2vpn (0) | L3vpn (0)

Service Key	Total Sessions / breaches	Latency breaches / Thre...	Jitter breaches / Thresh...	Packetloss forward breaches / Thresh...	Packetloss revers...

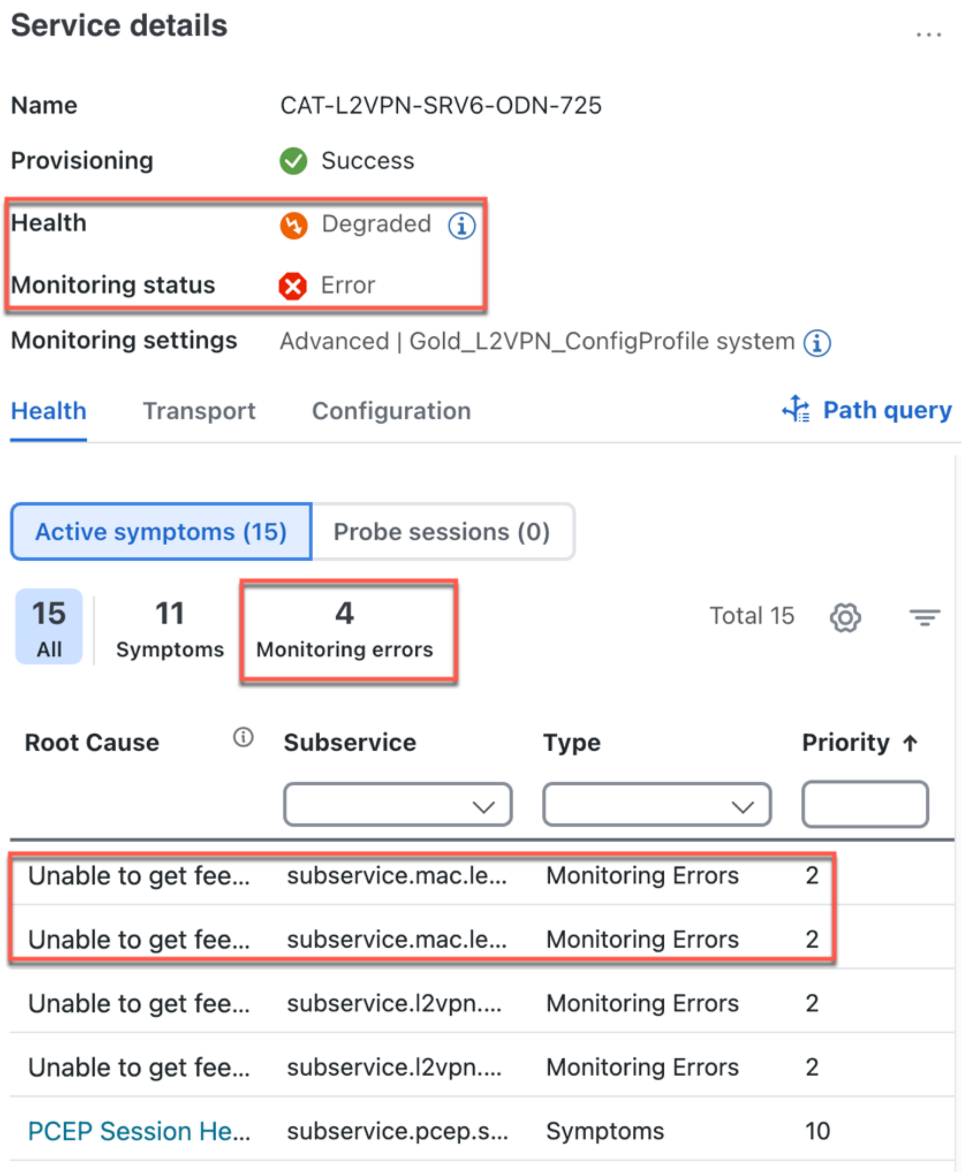
No rows to show

Clicking on any of the status indicators, or different colors in the wheel graph, redirects you to the **VPN Services** page with a filter set for the status you selected.

View monitoring status of services

View the monitoring status of a service from its service details page.

From the main menu, choose **Services & Traffic Engineering** > **VPN Services**. Locate the required. Under the **Actions** column, click **View details**. This page displays the monitoring status and health status of the service.



Service details

Name: CAT-L2VPN-SRV6-ODN-725

Provisioning: ✓ Success

Health: Degraded ⓘ

Monitoring status: ☒ Error

Monitoring settings: Advanced | Gold_L2VPN_ConfigProfile system ⓘ

Root Cause	Subservice	Type	Priority ↑
Unable to get fee...	subservice.mac.le...	Monitoring Errors	2
Unable to get fee...	subservice.mac.le...	Monitoring Errors	2
Unable to get fee...	subservice.l2vpn....	Monitoring Errors	2
Unable to get fee...	subservice.l2vpn....	Monitoring Errors	2
PCEP Session He...	subservice.pcep.s...	Symptoms	10

A monitoring status can be **Healthy** or **Error**.

- **Healthy**: The end-to-end flow of monitoring is working as expected, and Crosswork Network Controller can evaluate service health.
- **Error**: Crosswork Network Controller cannot monitor the current health of the service due to component failures, operational errors, or device errors. The displayed health status is the last known health of the service. Filter monitoring errors using the mini dashboard or filters.

Note Monitoring errors reported on account of device health do not affect the overall health of the service.

In Assurance Graph's historical timeline, EOS is displayed for monitoring errors as well. If the service is healthy and monitoring errors exist, a green warning icon appears. If the service is degraded and monitoring errors exist, an orange warning icon appears. When you click these icons, the symptoms table displays details with type set to **Monitoring errors**.

Note The historical timeline displays monitoring errors only when the monitoring errors setting is enabled via API. There is no option to enable this setting from the UI. Once this setting is enabled, the system starts to log these monitoring errors as EOS and display them in the historical timeline. Refer to the application programming interface API documentation on Cisco Devnet for more information.

Identify the active symptoms and root causes of a degraded service

Analyze the root cause of reported active symptoms and impacted services to determine which issues must be addressed first to maintain a healthy setup and which require further inspection and troubleshooting.

Note L3VPN service monitoring is supported on Cisco IOS XR devices, but not on Cisco IOS XE devices. If a provider and its devices are deleted and then added again for an L3VPN service being monitored, the service remains in the degraded state and displays a monitoring status of Monitoring error. To recover from this error, stop and restart the service monitoring.

To view the active symptoms and root causes for a service degradation:

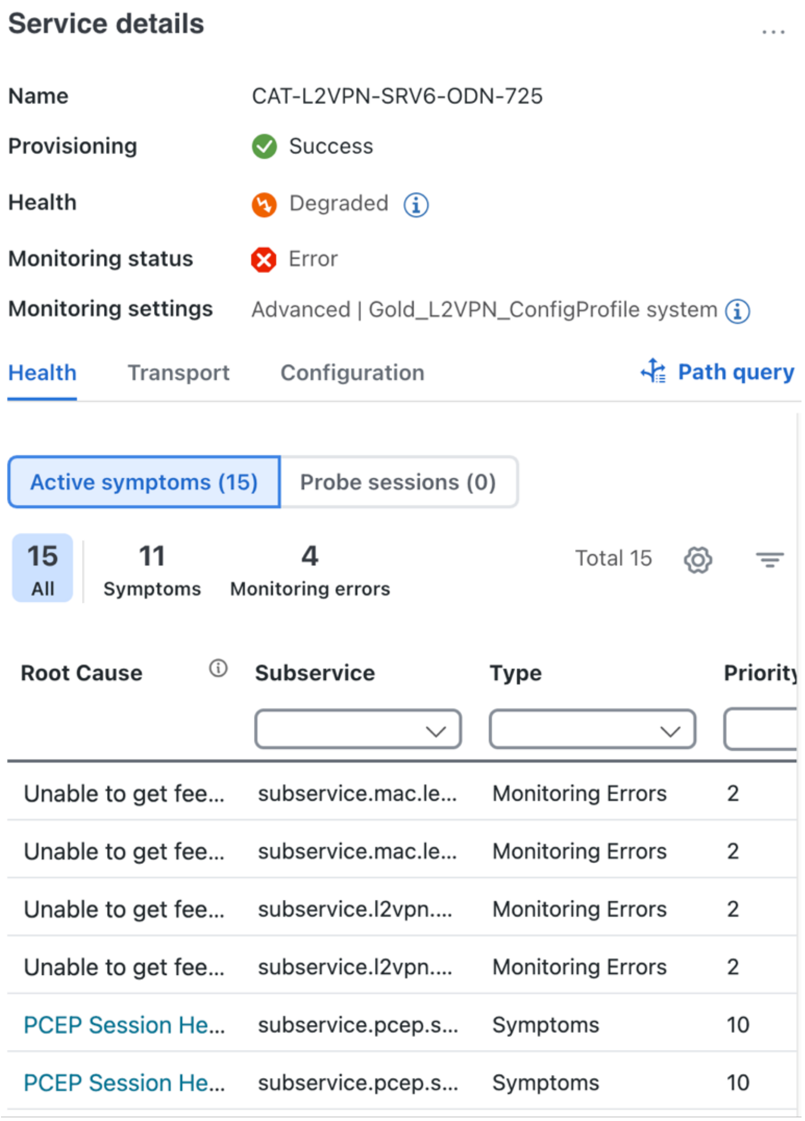
Procedure

Step 1 From the main menu, choose **Services & Traffic Engineering** > **VPN Services**. The VPN Services page opens.

Step 2 In the Actions column for the list of services, click ⋯ and select **View details**. The Service details panel appears on the right side.

Step 3 Select the Health tab and then click the **Active symptoms** tab. By default, the Active Symptoms table displays **Active symptoms** and **Monitoring errors**. To display only Active Symptoms, either click the **Symptoms** tab in the mini dashboard or select **Symptoms** from the filter box under **Type**. The table will then show only Active Symptoms.

Review the active symptoms for the degraded service, which includes the Root Cause, Subservice, Type, Priority, and Last Updated details.

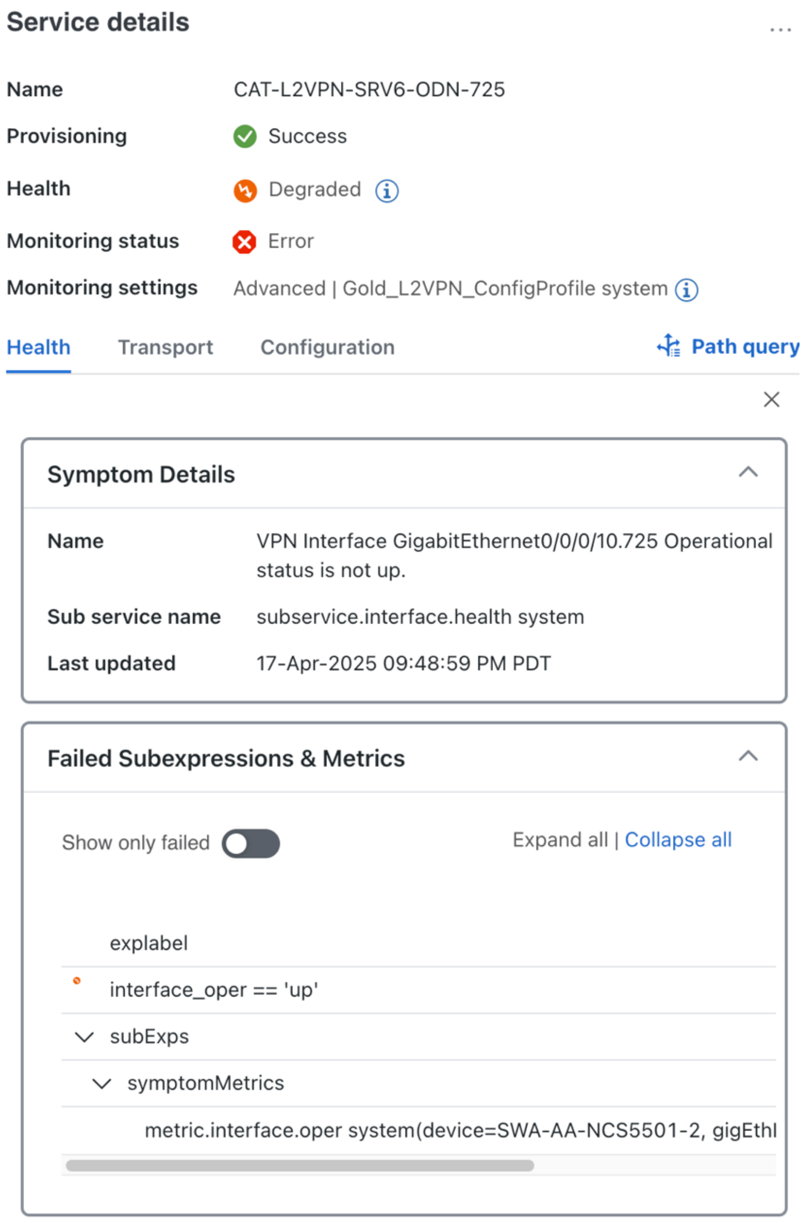


Service details

Name	CAT-L2VPN-SRV6-ODN-725
Provisioning	Success
Health	Degraded
Monitoring status	Error
Monitoring settings	Advanced | Gold_L2VPN_ConfigProfile system

Root Cause	Subservice	Type	Priority
Unable to get fee...	subservice.mac.le...	Monitoring Errors	2
Unable to get fee...	subservice.mac.le...	Monitoring Errors	2
Unable to get fee...	subservice.l2vpn....	Monitoring Errors	2
Unable to get fee...	subservice.l2vpn....	Monitoring Errors	2
PCEP Session He...	subservice.pcep.s...	Symptoms	10
PCEP Session He...	subservice.pcep.s...	Symptoms	10

Step 4 Click on a root cause and view both the **Symptom details** and the **Failed subexpressions & metrics** information. You can expand or collapse all of the symptoms listed in the tree. Use the **Show only failed** toggle to display only the failed expression values.



Service details

Name	CAT-L2VPN-SRV6-ODN-725
Provisioning	✓ Success
Health	Degraded ⓘ
Monitoring status	✕ Error
Monitoring settings	Advanced | Gold_L2VPN_ConfigProfile system ⓘ

Symptom Details

Name	VPN Interface GigabitEthernet0/0/0/10.725 Operational status is not up.
Sub service name	subservice.interface.health system
Last updated	17-Apr-2025 09:48:59 PM PDT

Failed Subexpressions & Metrics

Show only failed

explabel
interface_oper == 'up'
subExps
symptomMetrics
metric.interface.oper system(device=SWA-AA-NCS5501-2, gigEthI

Step 5 Click the **Transport** and the **Configuration** tab, then review the details provided.

Step 6 Click **X** in the top-right corner to return to the VPN Services list.

What to do next

- To monitor the VPN services using Assurance Graph capabilities and inspect any services or related nodes that are degraded, see Use the Assurance Graph to identify root causes, on page 43.
- To identify the issues with the degraded services within a specific time range, use the last 24Hr metrics. For details, see Identify the root causes using last 24 Hr metrics, on page 45.

- To identify a Service Health issue by examining the collection jobs, see View the collection jobs, on page 49.

Assurance graphs

The Service Health status eliminates potential confusion by distinguishing between Service Health and Event Health node status. This screen applies to both L2 and L3VPNs.

In Crosswork Network Controller, a service instance comprises of various subservices, each assured independently. The overall health of the service depends on the health of these subservices.

The Assurance Graph visually represents the service instances and their dependent subservices in a graph format. The topmost node in this logical dependency tree represents the monitored service instance. The child nodes represent its subservices. These subservices may further depend on other subservices.

This graphical representation helps locate problem areas and provides indications of possible symptoms and impacting metrics. It aids in troubleshooting degradation issues. Crosswork Network Controller updates the Assurance Graph automatically when the service instance is modified.

To view a service in the Assurance Graph, from the **Actions** column for the service, select **Assurance graph**. The Assurance Graph displays the graph in the left pane and shows details of the service in the right pane. Toggle **Show History** to view the historical timeline. Each dot on the historical timeline represents one EOS for a service.

For each EOS, you can view the Assurance Graph and symptoms with 24 hours of metrics collected based on the EOS time. For example, if monitoring was stopped for a service, a dot appears indicating that monitoring was stopped.

Clicking and dragging over a selected range on the EOS allows you to zoom in on a range of time. If you hover your mouse over a single event, a pop-up appears showing the service's monitoring status for Time, Event Type, Node Name, Service Health, Symptoms, and Event Health.

The Service Health status eliminates potential confusion by distinguishing between Service Health and Event Health node status. This screen applies to both L2 and L3VPNs.

Assurance Graph for L3VPN services

For L3VPN services, Crosswork Network Controller monitors the service and builds the Assurance Graph's historical timeline at the node level. The historical timeline includes a summary node for each device and feature-level nodes under each summary node. Nodes with dependencies spanning other nodes, such as `path.sla.summary`, have a feature-level summary node in the historical timeline.

Select endpoints

The Assurance Graph builds its view based on the data-sending endpoints (headends) of VPN nodes. If there are more than 50 nodes in the historical timeline, Crosswork Network Controller indicates that it is too large to display. Use the **Select endpoints** option in the historical timeline toolbar to view up to 50 endpoints at a time.

The Assurance Graph filtering is based only on the VPN nodes and does not support filtering by a combination of VPN nodes and endpoints. For example, in a service with 2 VPN nodes, each having 2 endpoints (totaling 4 endpoints), deselecting one endpoint using the **Select endpoints** option will not update the historical timeline. The historical timeline updates only when both endpoints for a VPN node are removed, leading to the entire VPN node being removed from the Assurance Graph.

Note

For a service, the Transport tab displays all discovered transports related to selected VPN nodes, considering both headend and tailend roles based on the import and export policy configured in the service intent. When you use the **Select endpoints** option and deselect a headend endpoint, the **Transport** tab updates to remove the headend endpoint from view but may still show the tailend endpoint if it is relevant to other headends.

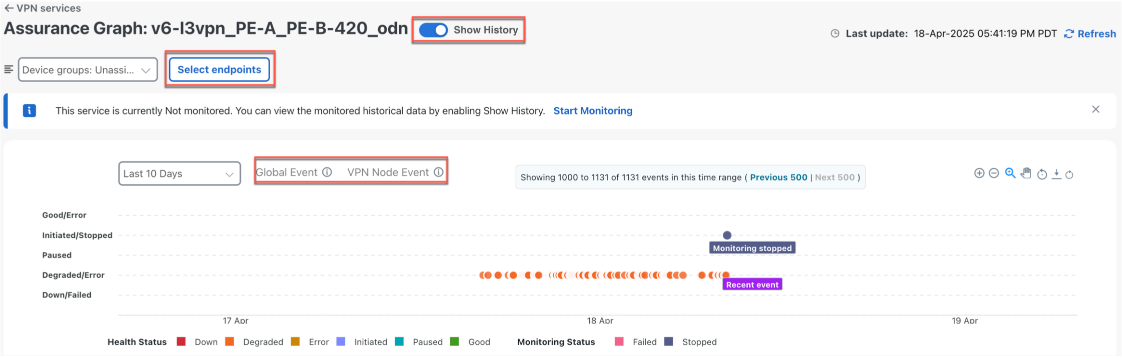
In contrast, the **Assurance Graph** focuses only on headend endpoints. If you deselect an endpoint and no other endpoints are left for that node, the graph removes the entire node from the display.

Show history

When you toggle **Show History** to view the historical data chart, you will see two types of events: **VPN node events** and **Global events**. When you hover over the EOS, the event type appears in its description.

- **Global events**: These events span multiple VPN nodes. For example, an EOS in the probe service (`path.sla.summary`) is classified as a Global event.
- **VPN node events**: These events are specific to a single VPN node.

In **L3VPN services**, symptom counts are shown at the VPN node level, with the Device ID (VPN node name) displayed alongside the symptoms. The timeline series in the Show History view displays these symptoms at the VPN node level (endpoint).


← VPN services

Assurance Graph: v6-l3vpn_PE-A_PE-B-420_odn

Last update: 18-Apr-2025 05:41:19 PM PDT

This service is currently Not monitored. You can view the monitored historical data by enabling Show History. Start Monitoring

Showing 1000 to 1131 of 1131 events in this time range (Previous 500 | Next 500)

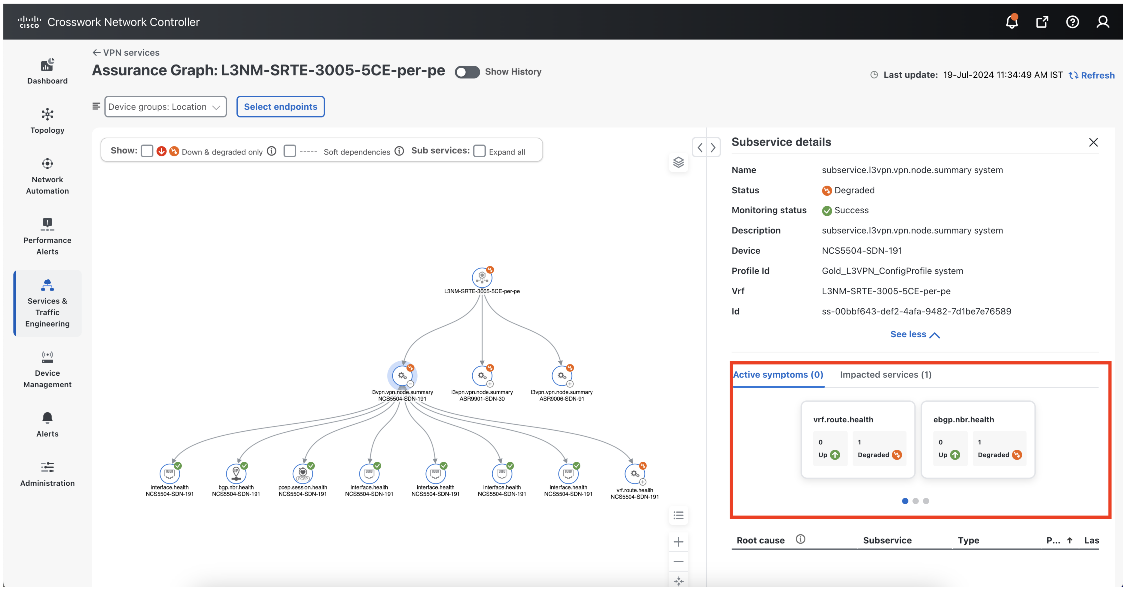
The **Service details** will continue to show the total symptoms count of the service, for the selected EOS time.

Note

If endpoints are selected, the total symptom count shows the combined number of symptoms for the selected endpoints.

Service details

In the **Service details** page, the **Active symptoms** tab shows the health details of feature level nodes, including the number of subservices in the **Up** or **Degraded** state. Clicking on the **Degraded** state in a feature node, filters the table to display symptoms and monitoring errors only for that node.


Crosswork Network Controller
Dashboard
Topology
Network Automation
Performance Alerts
Services & Traffic Engineering
Device Management
Alerts
Administration
← VPN services
Show History
Refresh
Device groups: Location
Select endpoints
Show: □ Down & degraded only □ ----- Soft dependencies Sub services: □ Expand all
L3NM-SRTE-3005-5CE-per-pe
l3vpn.vpn.node.summary NCS5504-SDN-191
l3vpn.vpn.node.summary ASR9901-SDN-30
l3vpn.vpn.node.summary ASR9006-SDN-91
interface.health NCS5504-SDN-191
bgp.nbr.health NCS5504-SDN-191
pcep.session.health NCS5504-SDN-191
interface.health NCS5504-SDN-191
interface.health NCS5504-SDN-191
interface.health NCS5504-SDN-191
interface.health NCS5504-SDN-191
vrf.route.health NCS5504-SDN-191
See less
Active symptoms (0)
Impacted services (1)
Root cause
Subservice
Type
P...
Las

Assurance Graph: L3NM-SRTE-3005-5CE-per-pe

Last update: 19-Jul-2024 11:34:49 AM IST

Subservice details

Name	subservice.l3vpn.vpn.node.summary system
Status	Degraded
Monitoring status	Success
Description	subservice.l3vpn.vpn.node.summary system
Device	NCS5504-SDN-191
Profile Id	Gold_L3VPN_ConfigProfile system
Vrf	L3NM-SRTE-3005-5CE-per-pe
Id	ss-00bbf643-def2-4afa-9482-7d1be7e76589

vrf.route.health: 0 Up, 1 Degraded

ebgp.nbr.health: 0 Up, 1 Degraded

Use the Assurance Graph to identify root causes

Identify, inspect, and drill down to the root cause of a service degradation.

Before you begin

Ensure that Service Health monitoring is enabled for the service you want to inspect. For details, see Start the Service Health monitoring, on page 15.

Note

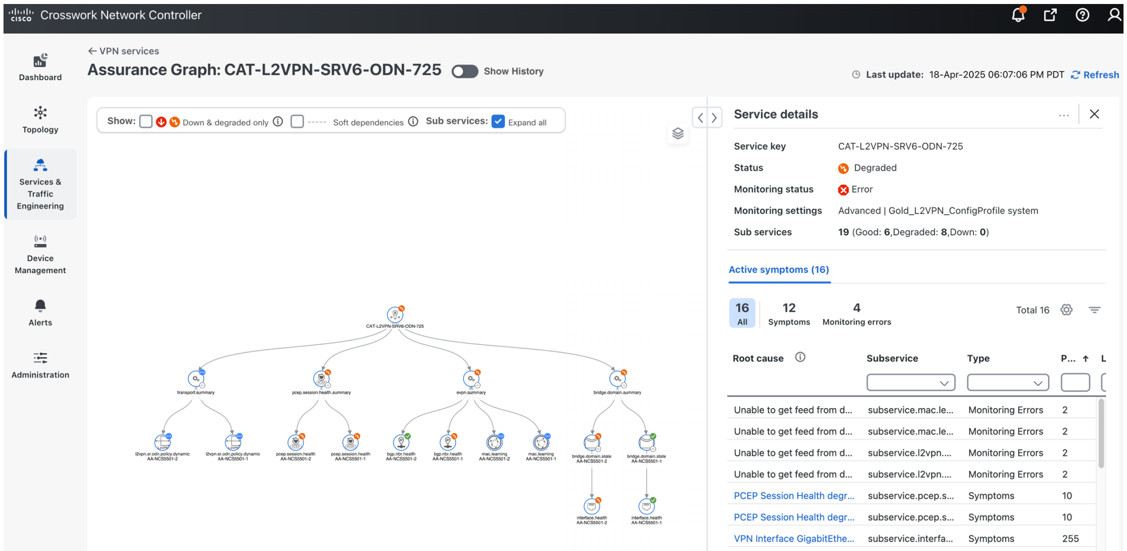
If you delete and then re-add a provider and devices for a monitored L3VPN service, the monitoring status remains degraded and displays as a monitoring error. To resolve this, stop and restart the service monitoring.

Procedure

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

Step 2 In the **Actions** column, click ⋯ for the required degraded service and click **Assurance graph**. The service assurance dependency graph displays services and subservices. The Service Details panel shows the Service Key, Status, Monitoring Status, Monitoring Settings, Subservices, and Active Symptoms details.

This may take up to five to ten minutes to update after a service has been enabled for monitoring.



Assurance Graph: CAT-L2VPN-SRV6-ODN-725

Last update: 18-Apr-2025 06:07:06 PM PDT

Service details

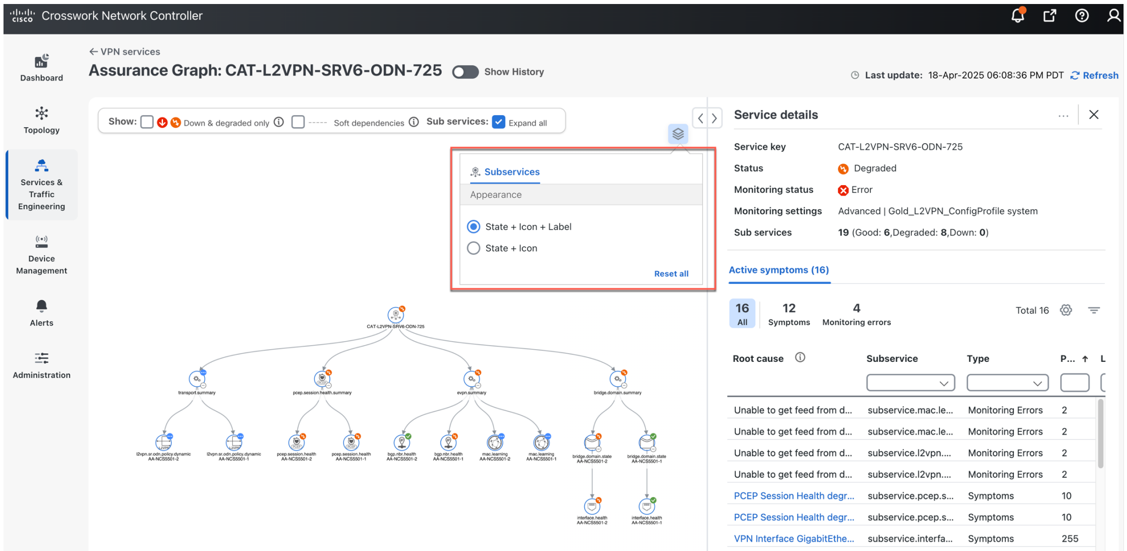
Service key	CAT-L2VPN-SRV6-ODN-725
Status	Degraded
Monitoring status	Error
Monitoring settings	Advanced | Gold_L2VPN_ConfigProfile system
Sub services	19 (Good: 6,Degraded: 8,Down: 0)

Active symptoms (16)

16 All | 12 Symptoms | 4 Monitoring errors | Total 16

Root cause	Subservice	Type	P...
Unable to get feed from d...	subservice.mac.le...	Monitoring Errors	2
Unable to get feed from d...	subservice.mac.le...	Monitoring Errors	2
Unable to get feed from d...	subservice.l2vpn....	Monitoring Errors	2
Unable to get feed from d...	subservice.l2vpn....	Monitoring Errors	2
PCEP Session Health degr...	subservice.pcep.s...	Symptoms	10
PCEP Session Health degr...	subservice.pcep.s...	Symptoms	10
VPN Interface GigabitEthe...	subservice.interfa...	Symptoms	255

At the top-right of the service assurance dependency graph, select the stack icon to select the appearance option for the Subservices: **State + Icon + Label** or **State + Icon**.



Assurance Graph: CAT-L2VPN-SRV6-ODN-725

Last update: 18-Apr-2025 06:08:36 PM PDT

Subservices

Appearance

● State + Icon + Label

○ State + Icon

Reset all

Service details

Service key	CAT-L2VPN-SRV6-ODN-725
Status	Degraded
Monitoring status	Error
Monitoring settings	Advanced | Gold_L2VPN_ConfigProfile system
Sub services	19 (Good: 6,Degraded: 8,Down: 0)

Active symptoms (16)

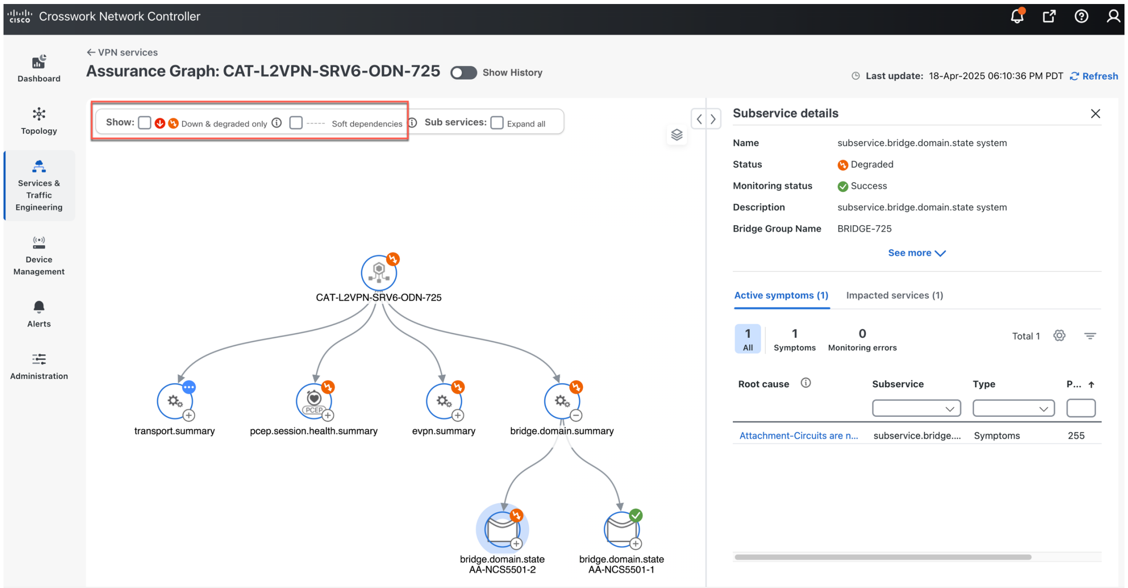
16 All | 12 Symptoms | 4 Monitoring errors | Total 16

Root cause	Subservice	Type	P...
Unable to get feed from d...	subservice.mac.le...	Monitoring Errors	2
Unable to get feed from d...	subservice.mac.le...	Monitoring Errors	2
Unable to get feed from d...	subservice.l2vpn....	Monitoring Errors	2
Unable to get feed from d...	subservice.l2vpn....	Monitoring Errors	2
PCEP Session Health degr...	subservice.pcep.s...	Symptoms	10
PCEP Session Health degr...	subservice.pcep.s...	Symptoms	10
VPN Interface GigabitEthe...	subservice.interfa...	Symptoms	255

Step 3 By default, the Assurance Graph displays a concise view with only the service and the top level dependencies (feature nodes). Select the plus (+) icon in the nodes to expand the graph and to view the dependent details. To expand all the nodes at once, select **Sub services: Expand all** at the top.

Step 4 Select a degraded subservice in the Assurance Graph. The Subservice Details panel shows subservice metrics, 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.



Assurance Graph: CAT-L2VPN-SRV6-ODN-725

Last update: 18-Apr-2025 06:10:36 PM PDT

Subservice details

Name	subservice.bridge.domain.state system
Status	Degraded
Monitoring status	Success
Description	subservice.bridge.domain.state system
Bridge Group Name	BRIDGE-725

Root cause	Subservice	Type	P...
Attachment-Circuits are n...	subservice.bridge....	Symptoms	255

Note

At the top left of the service assurance dependency graph, check the **Down & degraded only** or **Soft dependencies** check boxes to further isolate the subservices. Soft dependencies mean that a child subservice's health is weakly correlated with its parent's health. Therefore, if the child's health is degraded, the parent's health is not affected.

Step 5 Inspect the Active symptoms and Impacted services information, along with the root causes associated with the degraded service, to determine which issues require action to maintain a healthy setup.

What to do next

- To view the active symptoms and root causes, see Identify the active symptoms and root causes of a degraded service, on page 38.
- To identify the issues with the degraded services within a specific time range, use the last 24Hr metrics. For details, see Identify the root causes using last 24 Hr metrics, on page 45.
- To identify a service health issue by examining the collection jobs, see View the collection jobs, on page 49.

Identify the root causes using last 24 Hr metrics

Use the last 24 Hr metrics to identify issues with degraded services during a specific time period. By isolating issues to a specific timeframe, you can examine details that may have caused a service to be degraded or down. This supports troubleshooting for the service or node and addresses symptoms.

Before you begin

Ensure that service health monitoring is enabled for the service you want to analyze. For details, see Start the Service Health monitoring, on page 15.

Note If you delete a provider and devices from a monitored L3VPN service and then add them again, the monitoring status remains degraded, displaying 'Monitoring error'. To resolve this error, stop and restart service monitoring.

Procedure

Step 1 From the main menu, choose **Services & Traffic Engineering** > **VPN Services**. The service assurance dependency graph opens on the left side of the page. The table opens on the right side.

Step 2 In the Actions column, click ⋯ for the degraded-service and click **Assurance graph**. The service assurance dependency graph of services and subservices appears. The Service Details panel shows Service Key, Status, Monitoring Status, Monitoring Settings, Sub Services, and details about Active Symptoms.

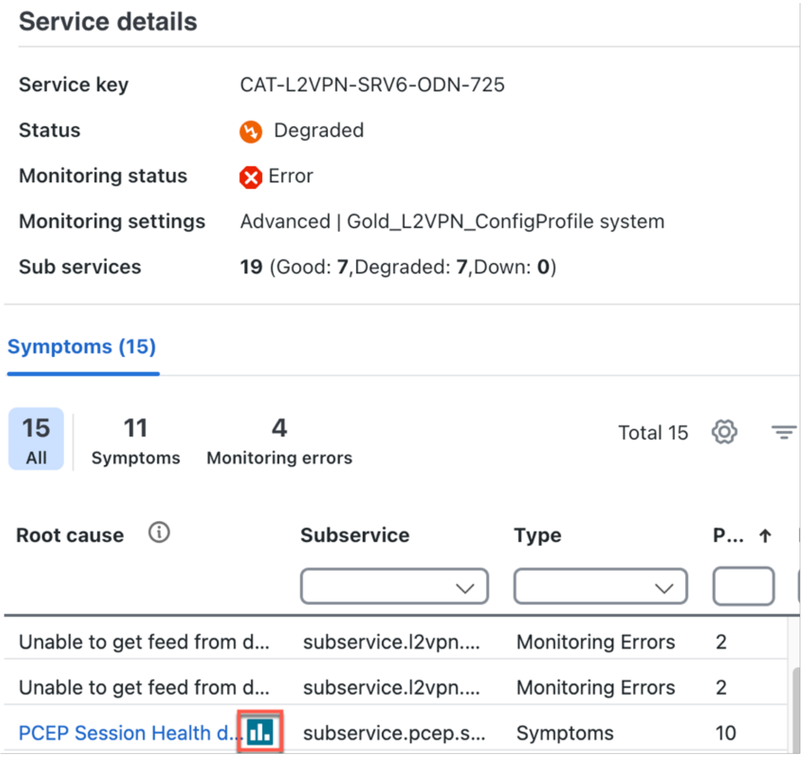
Note
This may take up to five to ten minutes to update after a service has been enabled for monitoring.

Step 3 At the top of the page, click the **Show History** toggle. The historical timeline for the Assurance Graph appears. This timeline shows different ranges of historical health service monitoring details from one day (the **Last Day**) up to the **Last 60 Days**.

a) To view data for a specific time range, select the range from the dropdown menu. Hovering over an event in the historical timeline displays a tooltip with event information.

b) Review the root cause information by clicking a particular event. The Service details panel reloads, showing the active symptoms and the root causes associated with the event. You can resize columns using your mouse, or select the gear icon to choose which columns to display.

Note
Once you enable **Show History** , root cause information in the active symptoms table will start to show the blue last 24 Hr metrics icon. Data from the device will be initially delayed and may take some time before the last 24 Hr metrics begin to populate with data. Until then, zero is reported.



Service details

Service key	CAT-L2VPN-SRV6-ODN-725
Status	Degraded
Monitoring status	Error
Monitoring settings	Advanced | Gold_L2VPN_ConfigProfile system
Sub services	**19** (Good: **7**,Degraded: **7**,Down: **0**)

Symptoms (15)

15 All | 11 Symptoms | 4 Monitoring errors | Total 15

Root cause	Subservice	Type	P... ↑
Unable to get feed from d...	subservice.l2vpn....	Monitoring Errors	2
Unable to get feed from d...	subservice.l2vpn....	Monitoring Errors	2
PCEP Session Health d...	subservice.pcep.s...	Symptoms	10

Step 4 To further isolate the possible issues and to utilize the last 24 Hr metrics, perform the following steps:

a) In the historical timeline, use your mouse to select the range of historical health service monitoring details from one day (the **Last Day**) up to sixty days (**Last 60 Days**).

Note
At the top-right of the historical timeline, select the appropriate icons to either zoom in or out, and scroll horizontally scroll through the date ranges. Or, refresh the graph to go back to the most recent event. You can also use your mouse to draw a rectangle over events to further zoom in on the degraded devices. Consecutive events may appear as a line of white space.

b) Click on a degraded event in the historical timeline. The Service details panel reloads, showing any active symptoms and the root causes to inspect. Expand the table as needed to view more information.

Step 5 Check the **Down & degraded only** check box at the top-left corner of the Assurance Graph to display only degraded subservices, along with other dependent but healthy subservices. Inspect the Service details panel, which shows the active symptoms and their root causes. Uncheck the **Down & degraded only** check box and check the **Soft dependencies** check box in the top-left corner of the Assurance Graph. A soft dependency means that a child subservice's health has a weak correlation to its parent's health. As a result, degraded health in the child does not cause the parent's health to degrade.

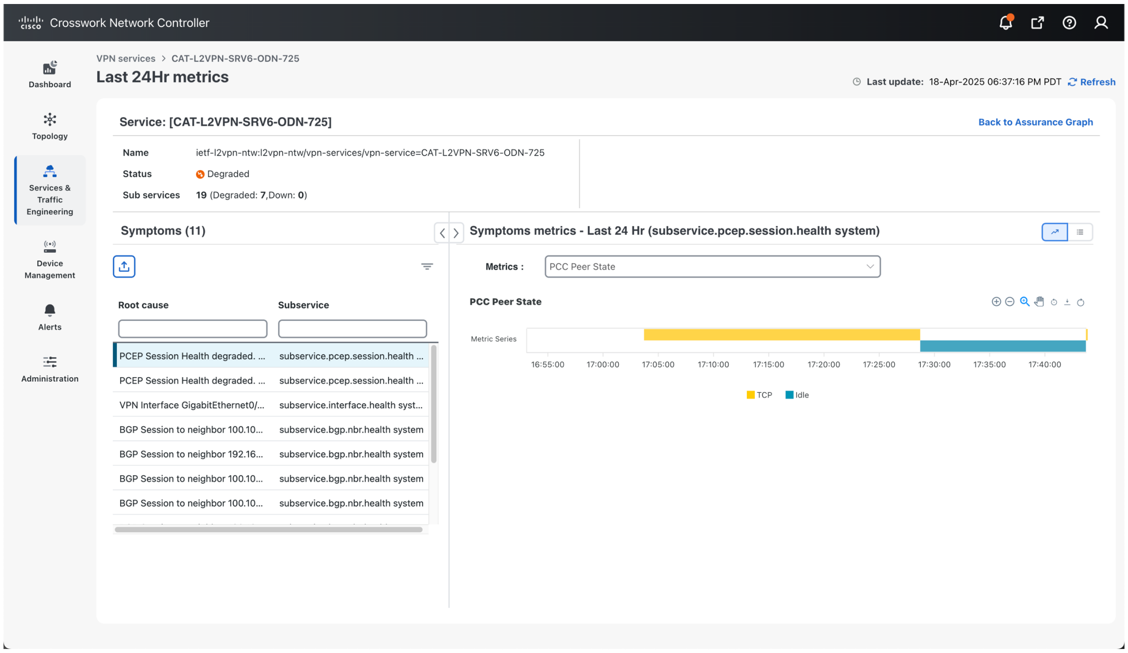
Use the plus or minus symbols at the bottom-right corner of the Assurance Graph to zoom in or out on subservices. Select the **?** icon to view the link color legend, which explains all icons, symbols, badges, and colors.

Step 6 Select the degraded subservice in the Assurance Graph to display its details.

Step 7 Click the **Symptoms** tab to view any root causes for the Service Health details. Then, click the **Impacted services** tab to see the impacted services.

Step 8 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 9 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 shows metric patterns and different session states (such as active, idle, or failed, if applicable) for individual root cause symptoms. It includes Status, Session, Start Time, and Duration information to assist in troubleshooting existing issues. Hover over the chart with your mouse to view details.



Last 24Hr metrics

Last update: 18-Apr-2025 06:37:16 PM PDT

Service: [CAT-L2VPN-SRV6-ODN-725]

Name ietf-l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service=CAT-L2VPN-SRV6-ODN-725

Status Degraded

Sub services **19** (Degraded: **7**,Down: **0**)

Symptoms (11)

Root cause	Subservice
PCEP Session Health degraded. ...	subservice.pcep.session.health ...
PCEP Session Health degraded. ...	subservice.pcep.session.health ...
VPN Interface GigabitEthernet0/...	subservice.interface.health syst...
BGP Session to neighbor 100.10...	subservice.bgp.nbr.health system
BGP Session to neighbor 192.16...	subservice.bgp.nbr.health system
BGP Session to neighbor 100.10...	subservice.bgp.nbr.health system
BGP Session to neighbor 100.10...	subservice.bgp.nbr.health system

Symptoms metrics - Last 24 Hr (subservice.pcep.session.health system)

Metrics : PCC Peer State

PCC Peer State

What to do next

- To view the active symptoms and root causes, see Identify the active symptoms and root causes of a degraded service, on page 38.
- To monitor the VPN services using Assurance Graph capabilities and inspect any services or related nodes that are degraded, see Use the Assurance Graph to identify root causes, on page 43.
- To identify a service health issue by examining the collection jobs, see View the collection jobs, on page 49.

View the devices participating in the service

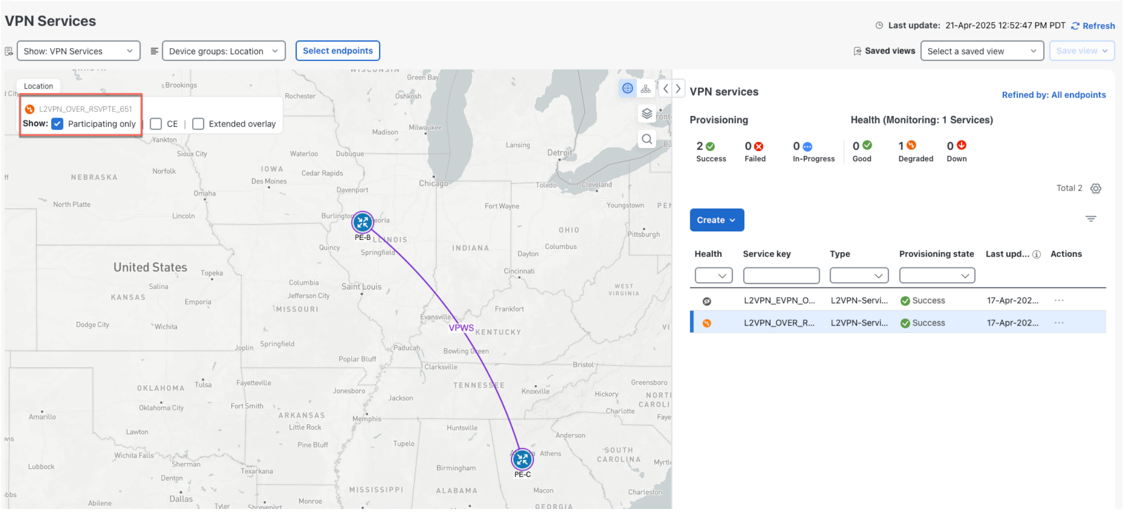
Use the topology view to see which devices participate in the services. You can also identify when a device or interface-related subservice degrades.

Procedure

Step 1 From the main menu, choose **Services and Traffic Engineering** > **VPN Services**.

Step 2 Click a service that shows as degraded. The topology map is updated, isolating the corresponding devices participating in that service.

Step 3 At the upper left of the service assurance dependency graph view, select **Show: Participating only** so that the topology map only shows the devices participating in the service.



VPN Services

Show: VPN Services | Device groups: Location | Select endpoints

Last update: 21-Apr-2025 12:52:47 PM PDT

Saved views: Select a saved view

Location

L2VPN_OVER_RSVPTE_651

Show: ☑ Participating only | □ CE | □ Extended overlay

VPN services

Refined by: All endpoints

Provisioning: 2 Success, 0 Failed, 0 In-Progress

Health (Monitoring: 1 Services): 0 Good, 1 Degraded, 0 Down

Total 2

Health	Service key	Type	Provisioning state	Last upd...	Actions
	L2VPN_EVPN_O...	L2VPN-Servi...	Success	17-Apr-202...	···
	L2VPN_OVER_R...	L2VPN-Servi...	Success	17-Apr-202...	···

Step 4 Hover your mouse over the device icons to view the popup information about Reachability State, Host Name, Node IP, and Type.

A healthy device may display an orange badge when it has an underlying device or interface-related subservice that is not healthy. This highlights unhealthy subservices in the topological view, even when the device itself appears healthy. After examining Service Details for a device, you may find conditions such as low CPU on a subservice node. Identifying these conditions helps you determine steps to address the unhealthy subservice.

View the collection jobs

Use the **Parameterized jobs** tab on the Collection Jobs page to see all active jobs created by Service Health.

If Service Health is not deployed, this page will not contain any data.

Crosswork Network Controller enables you to view parameterized jobs. These jobs are template-based collection jobs that support many tasks, such as CLI collection jobs.

This feature is particularly useful for troubleshooting collection job issues. It allows you to examine the details of individual devices. Devices are identified by their Context ID, which is a protocol indicator. The Context ID specifies whether the jobs are gNMI, SNMP, or CLI-based.

Procedure

Step 1 From the main menu, choose **Administration** > **Collection Jobs**.

Step 2 Select the **Parameterized jobs** tab.

Review the parameterized jobs list to identify devices that may have service health degradation. Use the Context ID (protocol) to focus on gNMI, SNMP, and CLI-based jobs for additional troubleshooting.

Step 3 In the Job details panel, select the collection job you want to export. Select ⤓ to download the status of collection jobs for further examination. The information is saved to a CSV file when you initiate the export.

Note

When exporting the collection status, you must enter the required information each time you execute an export. Make sure to review the **Steps to decrypt exported file** section on the Export Collection Status dialog box to access and view the exported data.

Step 4 Select **Export**.

Step 5 To check the status of the exported collection job data, select **View export status** at the top right of the Job details panel. The Export Status Jobs panel appears and provides the status of the export request.

Step 6 Review the exported .csv file for collection job details and the possible cause of the degraded device.

CHAPTER 5

Configure additional storage

This section explains these topics:

Configure the additional external storage

Set up AWS S3 external storage in Crosswork Network Controller when prompted to ensure continued access to monitored data as internal storage approaches capacity.

Service Health provides internal storage of monitored data up to a maximum limit of 50 GB. The data includes the VPN service status at the time of storage and historical data about the service.

This data is stored by Service Health on your system in the tar.gz archive file format. Each tar.gz file represents an EOS. Service Health uses this data to display it visually in the Crosswork Network Controller UI when you click on an EOS.

When the storage reaches 70% capacity, Crosswork Network Controller generates an alarm prompting you to configure external storage. Service Health automatically deletes the least recently used files when 80% of the 50 GB storage capacity is reached.

When you configure external storage, most internal storage data is automatically moved to external cloud storage, while a portion remains locally in internal storage to serve as a cache for improved performance.

You can use an Amazon Web Services (AWS) cloud account to configure external storage in the cloud. Only AWS S3, an object storage service, is supported.

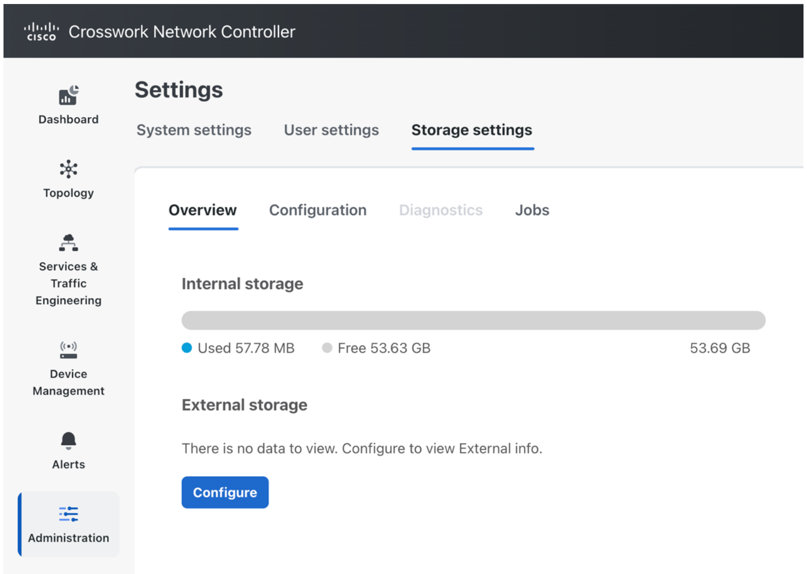
After you configure AWS storage, only 80% of the 50 GB space or 100,000 files are stored locally in Crosswork Network Controller. The remaining files are automatically moved to AWS.

Before you begin

You must have an AWS cloud account set up to configure the external storage.

Procedure

Step 1 From the main menu, choose **Administration** > **Settings** and click the **Storage settings** tab.



Settings

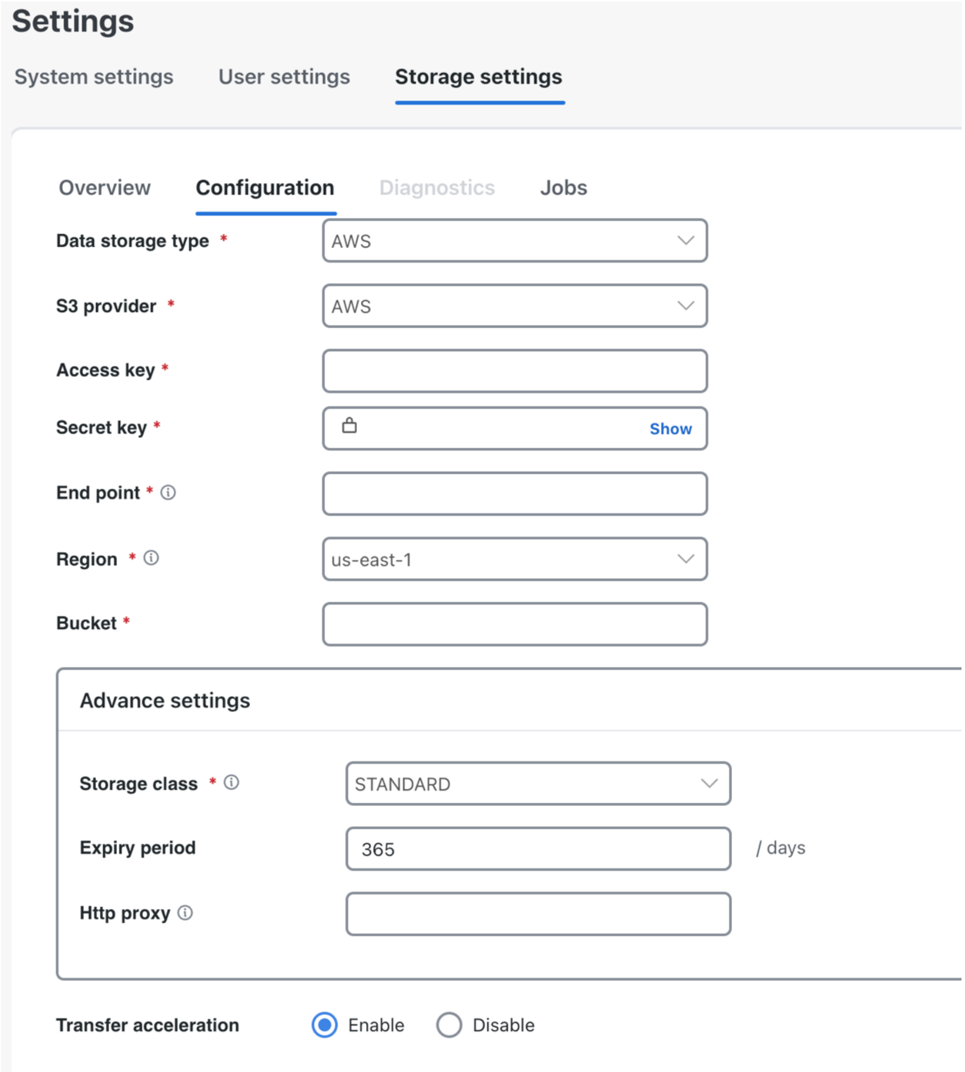
Internal storage

● Used 57.78 MB ● Free 53.63 GB 53.69 GB

External storage

There is no data to view. Configure to view External info.

Step 2 With the Overview tab selected, select **Configure** under the **External storage** section. The Configuration screen appears, and the Data storage type and S3 provider fields are pre-populated with AWS.



Data storage type * AWS

S3 provider * AWS

Access key *

Secret key *

End point *

Region * us-east-1

Bucket *

Advance settings

Storage class * STANDARD

Expiry period 365 / days

Http proxy

Transfer acceleration ● Enable ○ Disable

Step 3 Provide your AWS authentication information for all required fields, such as Access key, Secret key, Endpoint, and others.

Step 4 Select **Copy local data** if you want all files previously stored in the local cache to be copied in bulk to external storage. This action enables incremental upload of new files.

Note
This option is a one-time action when moving from only maintaining local storage and moving to external storage. This action also helps to improve the application performance.

Note
The Expiry period refers to the number of days that historical data files will be stored before being deleted. For example, if the Expiry period is set to one (1), the files will be deleted two days later, at midnight in the operational time zone on the second day.

Step 5 Click **Test and save**.

Step 6 To check the health of your storage setup, click the **Diagnostics** tab and click **Run test**.

By running a test, you can review the external storage diagnostics such as bandwidth, latency, and multiple access test details to help identify possible storage performance issues.

CHAPTER 6

Customize Heuristic Packages

This section explains the following topics:

Heuristic Packages

Service Health uses Heuristic Packages as the core logic to monitor and report the health of services. Heuristic Packages define a list of rules, configuration profiles, supported subservices, and associated metrics for each service type.

To access the Heuristic Packages, select **Administration** > **Heuristic Packages** from the Main Menu.

The **Heuristic Packages** page has two tabs: **System** and **Custom**. The default set of Heuristic Packages provided with Service Health is called system packages. These packages are available in the **System** tab and cannot be modified. To modify a system package, export it, make your changes, and import it as a custom package that matches your preferences.

You can view the custom packages in the **Custom** tab.

Expand each section on this screen to see more details on monitored services and the thresholds used to generate alerts. For finer details and definitions, hover your mouse over the information **i** icon.

- **Rules**: Rules are used to structure services and the dependent subservices and metrics within a specific service type. Dependencies within these rules help define the subservices and the metrics that are required to generate the data used to assess the health of the service. A service can depend on an individual subservice, a list of subservices of the same type, or subservices of different types.

  For a list of rules supported in Service Health, see Basic and advanced monitoring rules, subservices, and metrics, on page 63.

- **Configuration Profiles**: Configuration Profiles define threshold values that act as benchmarks for assessing the health of the service. By setting specific threshold values, Configuration Profiles establish the criteria for determining when a monitored parameter is within an acceptable range or deviates from the norm.

The system heuristics package in Service Health includes two configuration profiles—Silver and Gold—for each service type (L2VPN and L3VPN). You can choose a profile option that aligns with your specific monitoring requirements. A Silver profile has more lenient thresholds than a Gold profile. You can create custom configuration profiles as needed.

- **Sub Services**: Sub services are characterized by a list of metrics to fetch and a list of computations to apply to these metrics in order to produce a health status and associated symptoms for the service.

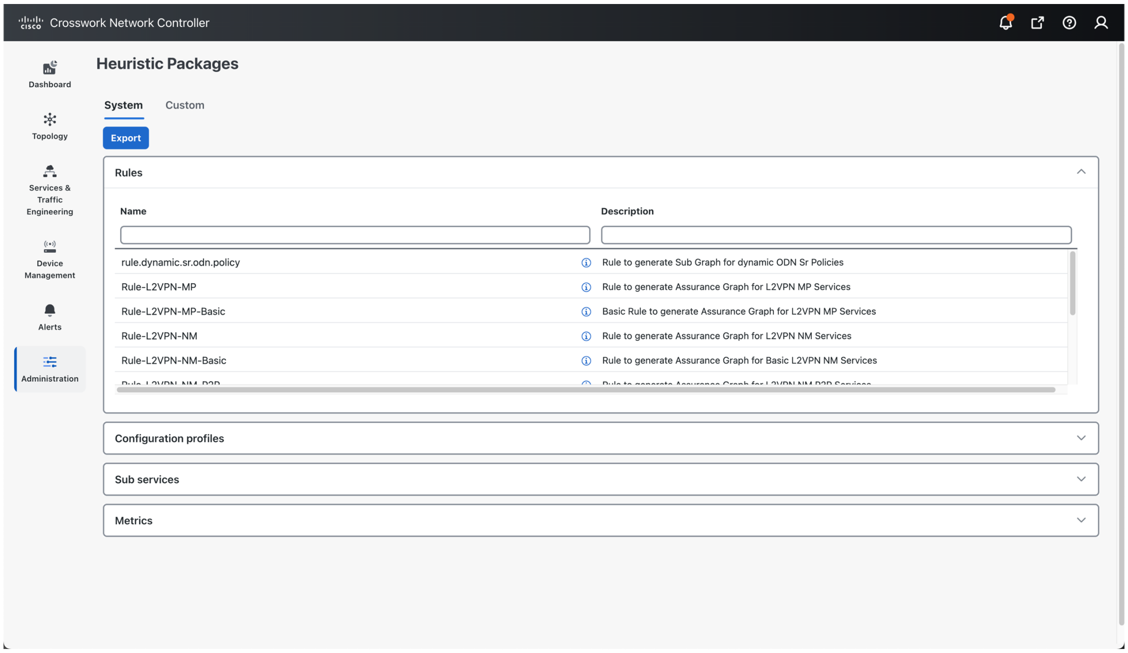
  For example, the subservice *subservice.evpn.health* monitors EVPN health. It is dependent on the metric *metric.l2vpn.xconnect.pw.state*. It evaluates an expression to check if *evpn_state* is **Up**. If the state is degraded, it raises a symptom.

  For list of subservices supported in Service Health, see Supported VPN services with associated subservices, on page 89.

- **Metrics** : Metrics define the operational data that should be fetched from different device types. Service Health uses a metric engine to map device-independent metrics to device-specific implementations, supporting multiple combinations of platforms and operating systems.

  For example, fetching the metric*resource.cpu* depends on the device type. For Cisco IOS XR devices, it uses Model-Driven Telemetry (MDT), while for Cisco IOS XE devices, it relies on CLI scraping using the command `show platform resources`.

Rules, Configuration Profiles, Sub services, and Metrics have a hierarchical relationship. Each Rule is mapped to a service type and depends on several subservices to compute service health. Subservices use metrics, and configuration profiles set threshold values for the metrics. Service Health assesses the health of the service using these values and then builds the Assurance Graph.



Heuristic Packages

Name	Description
rule.dynamic.sr.odn.policy	Rule to generate Sub Graph for dynamic ODN Sr Policies
Rule-L2VPN-MP	Rule to generate Assurance Graph for L2VPN MP Services
Rule-L2VPN-MP-Basic	Basic Rule to generate Assurance Graph for L2VPN MP Services
Rule-L2VPN-NM	Rule to generate Assurance Graph for L2VPN NM Services
Rule-L2VPN-NM-Basic	Rule to generate Assurance Graph for Basic L2VPN NM Services
Rule-L2VPN-NM-P2P	Rule to generate Assurance Graph for L2VPN NM P2P Services

The service and subservice dependency example illustrates the hierarchical relationship between Rules, Sub services, and Metrics.

Customize the Heuristic Packages

The Heuristic Package bundled with Service Health functions as an assurance model for monitoring L2VPN and L3VPN services. However, the configurations of underlay and overlay networking services may vary across deployments. The Heuristic Package can adapt automatically to certain configuration variations, but other variations cannot be seamlessly absorbed. These variations include changes in the service function pack model, introducing a new device type in VPN service deployment, or adding new network features that require monitoring. In these scenarios, you may need to customize configuration profiles, rules, metrics, or subservice class definitions.

Refer to the section Create the custom Heuristic Package, on page 57 for a basic example of how to create a custom package by customizing the configuration profile for a service. For details or help building a custom package based on rules, metrics, or subservices, contact the Cisco Customer Experience (CX) team or your Cisco account team.

Create the custom Heuristic Package

Create a custom Heuristic Package and modify the `CPU_THRESHOLD_MAX` parameter in the `Gold_L2VPN_ConfigProfile` to set the maximum acceptable CPU usage for L2VPN services.

Procedure

Step 1 From the main menu, choose **Administration** > **Heuristic Packages**. The Heuristic Packages page opens with **System** and **Custom** tabs.

Step 2 Click the **System** tab and then click **Export**.

The `exportAPI.tar.gz` package is downloaded to your system.

Step 3 Extract the `exportAPI.tar.gz` file contents. This action provides the `system` folder.

Step 4 In the `ConfigProfile` folder, open the `Gold_L2VPN_ConfigProfile-system.json` file.

Step 5 Edit the file as described in these steps:

a) Change the `namespace` attribute for the profile to `custom`.

b) Increase the `version` attribute number to `2.0`.

c) Search for `CPU_THRESHOLD_MAX` and update the threshold value to eighty (`80`).

```
{
"id": "Gold_L2VPN_ConfigProfile system”,
"name": "Gold_L2VPN_ConfigProfile",
  "namespace": "custom",
  "version": "2.0",
  "description":  "Thresholds to use for Gold L2VPN services",
  "rules": [
    {
      "name": "Rule-L2VPN-NM",
      "namespace": "system"
    },
    {
      "name": "Rule-L2VPN-NM-P2P",
      "namespace": "system"
    },
    {
      "name": "Rule-L2VPN-NM-Basic",
      "namespace": "system"
    },
    {
      "name": "Rule-L2VPN-NM-P2P-Basic",
      "namespace": "system"
    },
    {
      "name": "Rule-L2VPN-MP-Basic",
      "namespace": "system"
    },
    {
      "name": "Rule-L2VPN-MP",
      "namespace": "system"
    }
  ],
  "values": {
    "MAX_ACCEPTABLE_IN_OUT_PKT_DELTA": {
      "description": "Max allowed difference between packets received and packets transmitted",
      "type": "VAL_INT",
      "intVal": {
        "unit": "NA",
        "val":  100
      }
    },
    "VPN_INTF_PKT_ERROR_THRESHOLD": {
     "description": "Acceptable delta of in(or out) packet errors expected between polling intervals",

      "type": "VAL_INT",
      "intVal": {
        "unit": "NA",
        "val":  10
      }
    },
    "VPN_INTF_PKT_DISCARDS_THRESHOLD": {
      "description": "Acceptable delta of in(or out) packet discards expected between polling
intervals",
      "type": "VAL_INT",
      "intVal": {
        "unit": "NA",
        "val":  10
      }
    },
    "LATENCY_RT_THRESHOLD": {
      "description": "High Threshold for latency health checks",
      "type": "VAL_INT",
      "intVal": {
        "unit": "MSEC",
        "val":  500
      }
    },
    "JITTER_RT_THRESHOLD": {
      "description": "Threshold for acceptable jitter",
      "type": "VAL_FLOAT",
      "floatVal": {
        "unit": "MSEC",
        "val":  80
      }
    },
    "PACKET_LOSS_THRESHOLD": {
      "description": "Threshold for acceptable packet loss rate",
      "type": "VAL_FLOAT",
      "floatVal": {
        "unit": "PERCENT",
        "val":  1
      }
  },
  "SRPM_DELAY_THRESHOLD": {
    "description": "High Threshold for SR-PM latency health checks",
    "type": "VAL_INT",
    "intVal": {
      "unit": "MSEC",
      "val":  200

    }
  },
  "CPU_THRESHOLD_MAX": {
    "description": "Threshold for acceptable CPU usage on the device.",
    "type": "VAL_FLOAT",
    "floatVal": {
      "unit": "PERCENT",
      "val":  80
    }
  },
  "MEMFREE_THRESHOLD_MIN": {
    "description": "Threshold for minimum free memory to be available on the device.",
    "type": "VAL_FLOAT",
    "floatVal": {
      "unit": "BYTES",
      "val":  2000000000
    }
  }
 }
}
```

Step 6 Save the file after making the changes.

Step 7 Create a compressed tar.gz file from the `system` folder.

What to do next

Import the custom Heuristic Package into Crosswork Network Controller. See Import the custom Heuristic Packages, on page 59.

Import the custom Heuristic Packages

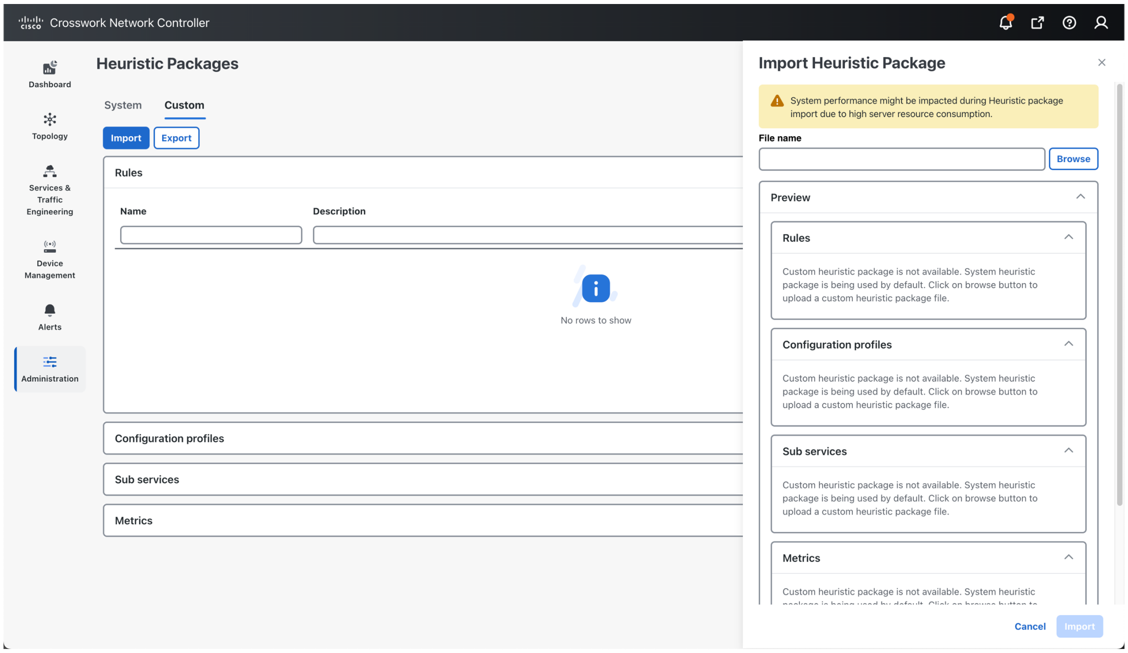
Import the custom Heuristic Package into Crosswork Network Controller.

Procedure

Step 1 From the main menu, choose **Administration** > **Heuristic Packages**. The Heuristic Packages page opens with **System** and **Custom** tabs.

Step 2 Click the **Custom** tab and then click **Import**. The **Import Heuristic Packages** panel appears.

Step 3 Click **Browse** to locate the custom package archive file (*.tar.gz file) on your system.



System performance might be impacted during Heuristic package import due to high server resource consumption.

Custom heuristic package is not available. System heuristic package is being used by default. Click on browse button to upload a custom heuristic package file.

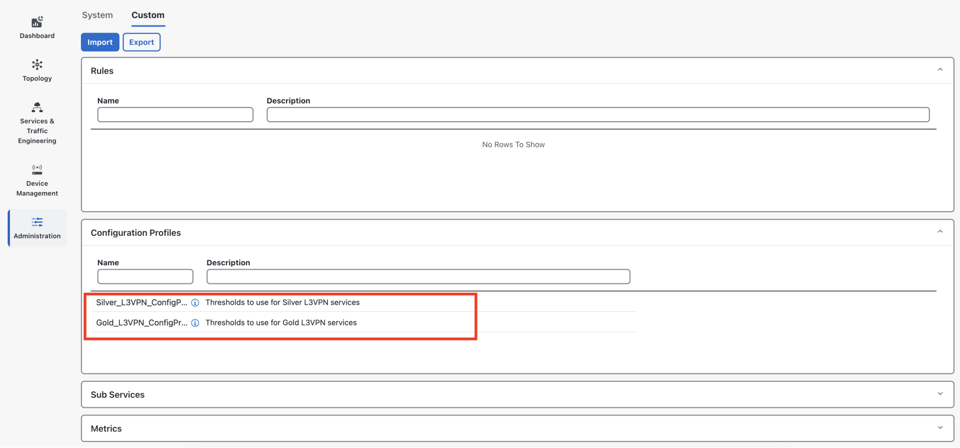
Step 4 Select your custom package and click **Preview** to review the details of the package to be imported. Information for the package's Rules, Configuration Profiles, Sub Services, and Metrics appears.

Note

High server resource consumption during Heuristic Package import might impact your system performance.

Select each option to preview the details of the custom package. Crosswork Network Controller will validate the package and display an error message if it finds issues or display a success message if validation is successful.

Step 5 Select the check box to acknowledge the warning and click **Import**. The package is imported into Crosswork Network Controller and appears in the **Custom** tab in the **Configuration profiles** section.



Name	Description
Silver_L3VPN_ConfigP...	Thresholds to use for Silver L3VPN services
Gold_L3VPN_ConfigPr...	Thresholds to use for Gold L3VPN services

What to do next

Stop monitoring the service before using custom heuristic packages. To resume monitoring, select the custom package and click **Start monitoring**. See Step 3 in the procedure Start the Service Health monitoring, on page 15 for more information.

Custom Heuristic Packages and rule mismatches

When you upload a custom Heuristic Package, it immediately becomes the standard for all new monitoring sessions, replacing the default system Heuristic Package. To ensure monitoring consistency and prevent rule mismatches, you must manage your existing and new monitoring sessions after the upload.

Key actions and considerations:

- **Activation of custom Heuristic Package**:
  - All monitoring sessions started after the upload will automatically use the new custom Heuristic Package.
- **Handling existing monitoring sessions**:
  - Any monitoring session that began before the custom Heuristic Package upload will continue using the previous rules.
  - You must stop and restart these sessions to ensure they align with the new custom Heuristic Package.
- **Editing Restriction**:
  - The Edit Monitoring Settings option does not function for sessions created with previous system Heuristic Packages.
  - Attempting to edit such sessions will result in a monitoring rule mismatch error and prompt you to restart the session.

After uploading, immediately identify and manually stop all monitoring sessions that began before the custom Heuristic Package upload.

Then, start new monitoring sessions to utilize the updated custom Heuristic Package. If you receive a monitoring rule mismatch error when editing, restart the session.

Example user workflow

1. Upload the custom Heuristic Package.
2. Stop any monitoring sessions that started before the upload.
3. Start new monitoring sessions (these will use the custom package).
4. If prompted by a mismatch error, restart the affected session.

Note Always manage monitoring sessions promptly after uploading a custom Heuristic Package to maintain rule alignment and prevent errors.

Custom Heuristic Packages after Crosswork Network Controller upgrades

Crosswork Network Controller supports both system and user-defined (custom) Heuristic Packages.

When you upgrade Crosswork Network Controller, your custom Heuristic Packages are not automatically updated with new system package changes. Manual updates are required to access the latest metrics and improvements.

In this release, these system Heuristic Package changes impact SR-PM (Segment Routing Performance Monitoring) and RSVP-TE (Resource Reservation Protocol-Traffic Engineering) subservices:

New metric classes

- heuristic_packages/MetricClass/metricSrPmRxPackets.json
- heuristic_packages/MetricClass/metricSrPmTxPackets.json
- heuristic_packages/MetricClass/metricMplsRsvpTeRxPackets.json

Updated subservices

- heuristic_packages/SubserviceClass/subserviceSrPolicyPccPmHealth.json
- heuristic_packages/SubserviceClass/subserviceSrPolicyPmHealth.json
- heuristic_packages/SubserviceClass/subserviceSrPolicyPcePmHealth.json

If you have imported or modified a custom Heuristic Package, review and manually update your package to include these new metric classes and subservice changes.

This ensures your customizations remain compatible and you continue to receive complete service health data for SR-PM and RSVP-TE features. For more information on creating custom Heuristic Packages, see Create the custom Heuristic Package.

APPENDIX A

Reference - basic monitoring and advanced monitoring Rules

This section explains the following topics:

- Basic and advanced monitoring rules, subservices, and metrics, on page 63

Basic and advanced monitoring rules, subservices, and metrics

Service Health monitoring in Cisco Crosswork Network Controller provides two distinct monitoring levels—Basic and Advanced—that enable scalable and detailed service performance assessment. These monitoring rules leverage Heuristic Packages to evaluate service components and metrics, empowering network operators to maintain service integrity and quickly identify issues impacting network services.

Table 3: Basic and advanced monitoring rules for L2VPNs

Rule name (type)	Monitoring functionality	Metrics and subservices
Rule-L2VPN-NM- Basic	• Checks the health of the VPWS xconnect state. • Monitors the policies deployed by the ODN. • Monitors the health of the device: CPU and memory utilization.	subservice.l2vpn.sr.odn.policy.dynamic subservice.device.health subservice.vpws.ctrlplane.health metric.l2vpn.xconnect.state metric.l2vpn.xconnect.ac.state metric.l2vpn.xconnect.pw.state

Rule name (type)	Monitoring functionality	Metrics and subservices
Rule-L2VPN-NM (Advanced)		

Rule name (type)	Monitoring functionality	Metrics and subservices
	• Checks • the health of the VPWS or EVPN xconnect state. • BGP Neighbor session health. • whether all BGP EVPN next hops for a given L2VPN service are reachable over LSP. • path reachability between two endpoints. • whether LSP path exists (in default VRF) towards the given destination device. • Health check for interface metrics: Oper status, interface in/out error packets, interface in/out packet discard. • Monitors • the policies deployed by the ODN. • the health of the device: CPU and memory utilization. • the delta between received and transmitted packets between VPN interfaces and Pseudo-wire. • Y.1731 probe stats for jitter, loss, and delay metrics, and compares against SLA thresholds. • 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	subservice.l2vpn.sr.odn.policy.dynamic 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 subservice.interface.health subservice.device.health subservice.interface.health.summary subservice.path.sla.summary metric.bgp.router.id metric.cef.route.labeled.lsp metric.l2vpn.xconnect.ac.state metric.l2vpn.xconnect.pw.state metric.l2vpn.xconnect.state

Rule name (type)	Monitoring functionality	Metrics and subservices
	ready. • FRR backup is active (primary failed and traffic is flowing over FRR backup). • 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.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
Rule-L2VPN-NM- P2P-Basic	• Checks the health of the VPWS xconnect state. • Monitors the policies deployed by the ODN. • Monitors the health of the device: CPU and memory utilization.	subservice.l2vpn.sr.odn.policy.dynamic subservice.device.health subservice.vpws.ctrlplane.health

Rule name (type)	Monitoring functionality	Metrics and subservices
Rule-L2VPN-NM- P2P (Advanced)	• Checks • the health of the VPWS xconnect state. • the health for interface metrics: Oper status, interface in/out error packets, interface in/out packet discard. • the SR Policy (PCC initiated) health status. Admin should be up. Oper should be up. Oper should have stayed up since last polling. • Monitors • the policies deployed by the ODN. • the health of the device: CPU and memory utilization. • LSP path (in default VRF) towards the given destination IP address. • the LSP path to the peer VPN node. • path reachability between two endpoints. • PCEP session state to all the peers configured on this device. • Y.1731 probe stats for jitter, loss, and delay metrics, and compares against SLA thresholds.	subservice.l2vpn.sr.odn.policy.dynamic 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) metric.cef.route.labeled.lsp metric.l2vpn.xconnect.ac.state metric.l2vpn.xconnect.pw.state metric.l2vpn.xconnect.state

Rule name (type)	Monitoring functionality	Metrics and subservices
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 name (type)	Monitoring functionality	Metrics and subservices
Rule-L2VPN-MP (Advanced)		

Rule name (type)	Monitoring functionality	Metrics and subservices
	• Checks BGP Neighbor health. • 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. • Groups • together all the PCEP session health subservices. • together all the device subservices. • together all the bridge domain subservices. • together all the transport subservices. • Monitors • bridge domain state on a given endpoint. • MPLS RSVP TE Tunnel Health. Admin, Oper should both be up and if FRR is configured, then backup path should be ready to pickup traffic when primary fails. If failover already 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. • the health of the device. • the policies deployed by the ODN. • PCEP session state to all	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 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)

Rule name (type)	Monitoring functionality	Metrics and subservices
	the peers configured on this device. • whether any routes are present for the given Bridge Domain. • SR Policy • health status reflecting SR-PM SLA (if configured). Admin and Oper should be up. Oper should have stayed up since last polling. Delay and Variance should meet SLA if SR-PM is configured to measure delay. Liveness should be up if SR-PM is configured for Liveness. • health status that include SR-PM. Admin and Oper should be up, and Oper should have stayed up since last polling. Delay and Variance should meet SLA if SR-PM is configured to measure delay. Liveness should be up if SR-PM is configured for Liveness. • Subservice to reflect interface health.	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) 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)

Rule name (type)	Monitoring functionality	Metrics and subservices
		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)

Table 4: Basic and advanced monitoring rules for L3VPN

Rule-L3VPN-NM- Basic	• Monitors the health of the device: CPU and memory utilization. • Reports the overall route connectivity health between the current PE device and its connecting CE device.	subservice.ce.pe.route.health subservice.device.health

Rule-L3VPN-NM (Advanced)	• Checks • BGP Neighbor health. • whether plain LSP route exists within given VRF towards given vpn ip-addresses. • eBGP Session health; Subservice to reflect interface health; Monitors the health of the device • 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. • 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. • PCEP session state to all the peers configured on this device. • Subservice, together with child subservices in L3VPN Rule, reports the overall route health between current PE device and its connecting CE device.	

		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 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
Rule-L3VPN-NM-VPN-Node-Basic	Monitors and presents the health summary for each PE device and the respective features underneath.	subservice.l3vpn.vpn.node.summary subservice.vrf.route.health subservice.device.health
Rule-L3VPN-NM-VPN-Node (Advanced)	Monitors and presents the health summary for each PE device and the respective features underneath.	subservice.l3vpn.vpn.node.summary subservice.vrf.route.health subservice.device.health subservice.interface.health subservice.pcep.session.health subservice.bgp.nbr.health

Service and subservice dependency example

The following example demonstrates how 'Rule-L2VPN-NM-P2P-Basic' interacts with its dependent subservices, 'subservice.vpws.ctrlplane.health' and 'subservice.device.health'. The detailed definitions of each subservice explain their metric dependencies and the resulting symptoms.

```
Rule-L2VPN-NM-P2P-Basic
```

```
{
  "name": "Rule-L2VPN-NM-P2P-Basic",
  "namespace": "system",
  "id": "Rule-L2VPN-NM-P2P-Basic system",
  "description": "Rule to generate Assurance Graph for Basic L2VPN NM P2P Services.",
  "matchCriteria": [
    {
      "configSource": "SOURCE_TYPE_NSO",
      "configSubSource": [
        "SUBSOURCE_SERVICE_CONFIG"
      ],
      "matchType": "MATCH_TYPE_XPATH",
      "matchExpression":
"//vpn-service[@xmlns='urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw']/vpn-svc-type[text()='vpn-common:t-ldp']",

      "matchPrefix": "",
      "matchParams": []
    },
    {
      "configSource": "SOURCE_TYPE_NSO",
      "configSubSource": [
        "SUBSOURCE_SERVICE_CONFIG"
      ],
      "matchType": "MATCH_TYPE_XPATH",
      "matchExpression": "//flat-L2vpn/service-type[text()='p2p']",
      "matchPrefix": "",
      "matchParams": []
    },
    {
      "configSource": "SOURCE_TYPE_NSO",
      "configSubSource": [
        "SUBSOURCE_SERVICE_CONFIG"
      ],
      "matchType": "MATCH_TYPE_XPATH",
      "matchExpression":
"//vpn-service[@xmlns='urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw']/vpn-type[text()='vpn-common:t-ldp']",

      "matchPrefix": "",
      "matchParams": []
    },
    {
      "configSource": "SOURCE_TYPE_NSO",
      "configSubSource": [
        "SUBSOURCE_SERVICE_CONFIG"
      ],
      "matchType": "MATCH_TYPE_XPATH",
      "matchExpression": "//vpn-service[not(//bridge-group)]/vpn-type[contains(text(),
':mpls-evpn')]",
      "matchPrefix": "",
      "matchParams": []
    },
    {
      "configSource": "SOURCE_TYPE_NSO",
      "configSubSource": [
        "SUBSOURCE_SERVICE_CONFIG"
      ],
      "matchType": "MATCH_TYPE_XPATH",
      "matchExpression":
"//vpn-service[@xmlns='urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw']/vpn-type[text()='x:vpws']",

      "matchPrefix": "",
      "matchParams": []
    },
    {
      "configSource": "SOURCE_TYPE_NSO",
      "configSubSource": [
        "SUBSOURCE_SERVICE_CONFIG"
      ],
      "matchType": "MATCH_TYPE_XPATH",
      "matchExpression":
"//vpn-service[@xmlns='urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw']/vpn-type[text()='ietf-vpn-common:vpws']",

      "matchPrefix": "",
      "matchParams": []
    }
  ],
  "dependencies": [
    {
      "name": "VPWS-ControlPlane-Health-Summary",
      "id": "subservice.vpws.ctrlplane.health.summary system",
      "ssClass": "subservice.vpws.ctrlplane.health.summary",
      "namespace": "system",
      "type": "DEP_TYPE_NON_LIST",
      "optional": false,
      "paramExtractionMechanism": {
        "mode": "EXTRACT_MODE_XPATH",
        "name": "",
        "namespace": "",
        "version": "",
        "validationHash": "0",
    "pluginMethod": "",
    "extractedParams": [],
    "nativeMethod": ""
  },
  "parameters": [
    {
      "name": "vpnServiceId",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-service/vpn-id"
        },
        {
          "description": "Flat Model",
          "extractValue": "//flat-L2vpn[/flat-L2vpn-p2p]/key"
        }
      ]
    }
  ],
  "subDependencies": [
    "VPWS-ControlPlane-Health-Local-Site",
    "VPWS-ControlPlane-Health-Remote-Site"
  ],
  "softSubDependencies": []
},
{
  "name": "VPWS-ControlPlane-Health-Local-Site",
  "id": "subservice.vpws.ctrlplane.health system",
  "ssClass": "subservice.vpws.ctrlplane.health",
  "namespace": "system",
  "type": "DEP_TYPE_NON_LIST",
  "optional": false,
  "paramExtractionMechanism": {
    "mode": "EXTRACT_MODE_XPATH",
    "name": "",
    "namespace": "",
    "version": "",
    "validationHash": "0",
    "pluginMethod": "",
    "extractedParams": [],
    "nativeMethod": ""
  },
  "parameters": [
    {
      "name": "device",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-nodes/vpn-node[1]/vpn-node-id"
        }
      ]
    },
    {
      "name": "groupName",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-service/vpn-id"
        },
        {
          "description": "Flat Model",
          "extractValue": "//flat-L2vpn/flat-L2vpn-p2p/local-site/xconnect-group-name"
        }
      ]
    },
    {
      "name": "xconnectName",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-service/vpn-id"
        },
        {
          "description": "Flat Model",
          "extractValue": "//flat-L2vpn/flat-L2vpn-p2p/local-site/xconnect-group-name"
        }
      ]
    }
  ],
  "subDependencies": [],
  "softSubDependencies": [
    "device1"
  ]
},
{
 "name": "VPWS-ControlPlane-Health-Remote-Site",
  "id": "subservice.vpws.ctrlplane.health system",
  "ssClass": "subservice.vpws.ctrlplane.health",
  "namespace": "system",
  "type": "DEP_TYPE_NON_LIST",
  "optional": false,
  "paramExtractionMechanism": {
    "mode": "EXTRACT_MODE_XPATH",
    "name": "",
    "namespace": "",
    "version": "",
    "validationHash": "0",
    "pluginMethod": "",
    "extractedParams": [],
    "nativeMethod": ""
  },
  "parameters": [
    {
      "name": "device",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-nodes/vpn-node[2]/vpn-node-id"
        }
      ]
    },
    {
      "name": "groupName",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-service/vpn-id"
        },
        {
          "description": "Flat Model",
         "extractValue": "//flat-L2vpn/flat-L2vpn-p2p/remote-site/xconnect-group-name"

        }
      ]
    },
    {
      "name": "xconnectName",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
          "extractValue": "//vpn-service/vpn-id"
        },
        {
          "description": "Flat Model",
         "extractValue": "//flat-L2vpn/flat-L2vpn-p2p/remote-site/xconnect-group-name"

        }
      ]
    }
  ],
  "subDependencies": [],
  "softSubDependencies": [
    "device2"
  ]
},
{
  "name": "device1",
  "id": "subservice.device.health system",
  "ssClass": "subservice.device.health",
  "namespace": "system",
  "type": "DEP_TYPE_NON_LIST",
  "optional": false,
  "paramExtractionMechanism": {
    "mode": "EXTRACT_MODE_XPATH",
    "name": "",
    "namespace": "",
    "version": "",
    "validationHash": "0",
    "pluginMethod": "",
    "extractedParams": [],
    "nativeMethod": ""
  },
  "parameters": [
    {
      "name": "device",
      "iterator": false,
      "defaultValue": "",
      "extractionMethod": "DEP_PARAM_XPATH",
      "extractionDetails": [
        {
          "description": "",
              "extractValue": "//vpn-nodes/vpn-node[1]/vpn-node-id"
            }
          ]
        }
      ],
      "subDependencies": [],
      "softSubDependencies": []
    },
    {
      "name": "device2",
      "id": "subservice.device.health system",
      "ssClass": "subservice.device.health",
      "namespace": "system",
      "type": "DEP_TYPE_NON_LIST",
      "optional": false,
      "paramExtractionMechanism": {
        "mode": "EXTRACT_MODE_XPATH",
        "name": "",
        "namespace": "",
        "version": "",
        "validationHash": "0",
        "pluginMethod": "",
        "extractedParams": [],
        "nativeMethod": ""
      },
      "parameters": [
        {
          "name": "device",
          "iterator": false,
          "defaultValue": "",
          "extractionMethod": "DEP_PARAM_XPATH",
          "extractionDetails": [
            {
              "description": "",
              "extractValue": "//vpn-nodes/vpn-node[2]/vpn-node-id"
            }
          ]
        }
      ],
      "subDependencies": [],
      "softSubDependencies": []
    }
  ],
  "softRootDependencies": [],
  "createTimestamp": "1697841637567500247",
  "updateTimestamp": "0",
  "monitoringType": "BASIC",
  "version": "1.1"
}
```

```
Sub service: 'subservice.vpws.ctrlplane.health'
```

```
{
  "id": "subservice.vpws.ctrlplane.health.summary system",
  "name": "subservice.vpws.ctrlplane.health.summary",
  "namespace": "system",
  "description": "Groups together all the VPWS Ctrlplane health subservices.",
  "params": [
    {
      "name": "vpnServiceId",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    }
  ],
  "liveMetrics": {},
  "rootExpressions": [],
  "dynamicConfig": null,
  "symptom": null,
  "dependencies": [],
  "exprCid": "",
  "createTimestamp": "1697841637373426164",
  "updateTimestamp": "0",
  "tags": [],
  "version": "1.0"
}

{
  "id": "subservice.vpws.ctrlplane.health system",
  "name": "subservice.vpws.ctrlplane.health",
  "namespace": "system",
  "description": "check the health of the VPWS state",
  "params": [
    {
      "name": "device",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    },
    {
      "name": "groupName",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    },
    {
      "name": "xconnectName",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    }
  ],
  "liveMetrics": {},
  "rootExpressions": [
    {
      "evalExpression": "xconnect_state == 'up' && ac_state == 'up' && evpn_state == 'up'",

      "activateCondition": ""
    }
  ],
  "dynamicConfig": null,
  "symptom": {
    "formatString": "VPWS State degraded. Device: {device}, XConnectGroup: {groupName},
XconnectName: {xconnectName}",
    "level": "DEGRADED",
    "priority": 15,
    "condition": false
  },
  "dependencies": [
    {
      "type": "DEP_TYPE_METRIC",
      "label": "xconnect_state",
      "evalExpression": "metric.l2vpn.xconnect.state",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device",
        "groupName": "groupName",
        "xconnectName": "xconnectName"
      },
      "id": ""
    },
    {
      "type": "DEP_TYPE_METRIC",
      "label": "ac_state",
      "evalExpression": "metric.l2vpn.xconnect.ac.state",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device",
        "groupName": "groupName",
        "xconnectName": "xconnectName"
      },
      "id": ""
    },
    {
      "type": "DEP_TYPE_METRIC",
      "label": "evpn_state",
      "evalExpression": "metric.l2vpn.xconnect.pw.state",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device",
        "groupName": "groupName",
        "xconnectName": "xconnectName"
      },
      "id": ""
    }
  ],
  "exprCid": "",
  "createTimestamp": "1697841637370064741",
  "updateTimestamp": "0",
  "tags": [],
  "version": "1.0"
}
```

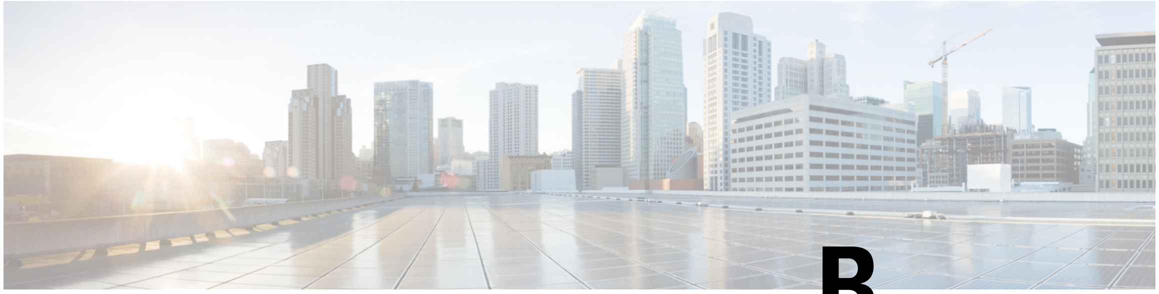
```
Sub service: 'subservice.device.health'
```

```
{
  "id": "subservice.device.health system",
  "name": "subservice.device.health",
  "namespace": "system",
  "description": "Monitor the health of the device.",
  "params": [
    {
      "name": "device",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    }
  ],
  "liveMetrics": {},
  "rootExpressions": [
    {
      "evalExpression": "cpu_healthy && memory_healthy",
      "activateCondition": ""
    }
  ],
  "dynamicConfig": null,
  "symptom": {
    "formatString": "Heavier than expected resource consumption on the Device: {device}",
    "level": "DEGRADED",
    "priority": 100,
    "condition": false
  },
  "dependencies": [
    {
      "type": "DEP_TYPE_EXPRESSION",
      "label": "cpu_healthy",
      "evalExpression": "ListElemsAverage(cpu_load) <= CPU_THRESHOLD_MAX",
      "namespace": "",
      "symptom": null,
      "paramMap": {},
      "id": ""
    },
    {
      "type": "DEP_TYPE_EXPRESSION",
      "label": "memory_healthy",
      "evalExpression": "ListElemsSum(memory_free) > MEMFREE_THRESHOLD_MIN",
      "namespace": "",
      "symptom": null,
      "paramMap": {},
      "id": ""
    },
    {
      "type": "DEP_TYPE_METRIC",
      "label": "cpu_load",
      "evalExpression": "metric.device.cpu.load",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device"
      },
      "id": ""
    },
    {
      "type": "DEP_TYPE_METRIC",
      "label": "memory_free",
      "evalExpression": "metric.device.memory.free",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device"
      },
      "id": ""
    }
  ],
  "exprCid": "",
  "createTimestamp": "1697841637256704609",
  "updateTimestamp": "0",
  "tags": [
    "DEVICE_SUBSERVICES"
  ],
  "version": "1.1"
}

{
  "id": "subservice.device.summary system",
  "name": "subservice.device.summary",
  "namespace": "system",
  "description": "Groups together all the Device subservices",
  "params": [
    {
      "name": "vpnServiceId",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    }
  ],
  "liveMetrics": {},
  "rootExpressions": [],
  "dynamicConfig": null,
  "symptom": null,
  "dependencies": [],
  "exprCid": "",
  "createTimestamp": "1697841637260108075",
  "updateTimestamp": "0",
  "tags": [],
  "version": "1.0"
}

{
  "id": "subservice.device.health system",
  "name": "subservice.device.health",
  "namespace": "system",
  "description": "Monitor the health of the device.",
  "params": [
    {
      "name": "device",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    }
  ],
  "liveMetrics": {},
  "rootExpressions": [
    {
      "evalExpression": "cpu_healthy && memory_healthy",
      "activateCondition": ""
    }
  ],
  "dynamicConfig": null,
  "symptom": {
    "formatString": "Heavier than expected resource consumption on the Device: {device}",
    "level": "DEGRADED",
    "priority": 100,
    "condition": false
  },
  "dependencies": [
    {
      "type": "DEP_TYPE_EXPRESSION",
      "label": "cpu_healthy",
      "evalExpression": "ListElemsAverage(cpu_load) <= CPU_THRESHOLD_MAX",
      "namespace": "",
      "symptom": null,
      "paramMap": {},
      "id": ""
    },
    {
      "type": "DEP_TYPE_EXPRESSION",
      "label": "memory_healthy",
      "evalExpression": "ListElemsSum(memory_free) > MEMFREE_THRESHOLD_MIN",
      "namespace": "",
      "symptom": null,
      "paramMap": {},
      "id": ""
    },
    {
      "type": "DEP_TYPE_METRIC",
      "label": "cpu_load",
      "evalExpression": "metric.device.cpu.load",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device"
      },
      "id": ""
    },
    {
      "type": "DEP_TYPE_METRIC",
      "label": "memory_free",
      "evalExpression": "metric.device.memory.free",
      "namespace": "",
      "symptom": null,
      "paramMap": {
        "device": "device"
      },
      "id": ""
    }
  ],
  "exprCid": "",
  "createTimestamp": "1697841637256704609",
  "updateTimestamp": "0",
  "tags": [
    "DEVICE_SUBSERVICES"
  ],
  "version": "1.1"
}

{
  "id": "subservice.device.summary system",
  "name": "subservice.device.summary",
  "namespace": "system",
  "description": "Groups together all the Device subservices",
  "params": [
    {
      "name": "vpnServiceId",
      "description": "",
      "type": "PARAM_TYPE_NON_LIST"
    }
  ],
  "liveMetrics": {},
  "rootExpressions": [],
  "dynamicConfig": null,
  "symptom": null,
  "dependencies": [],
  "exprCid": "",
  "createTimestamp": "1697841637260108075",
  "updateTimestamp": "0",
  "tags": [],
  "version": "1.0"
}
```



APPENDIX B

Supported VPN services with associated subservices

The Service Health L2VPN and L3VPN types with their associated subservices provide a detailed, service-centric monitoring framework that benefits users by improving visibility, troubleshooting efficiency, and operational assurance for IOS XE and IOS XR devices in Crosswork Network Controller.

Supported VPN services with associated subservices for IOS XE devices

Supported VPN services	Associated subservices	Details
L2VPN P2P Plain	• Device Health • Path Reachability • Summary (aggregator) nodes • VPN Interface Health • Y.1731 Health	IOS XE does not support SNMP or 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.
L2VPN Point to Point over MPLS LDP	• Device Health • Path Reachability • Summary (aggregator) nodes • VPN Interface Health • VPWS Control Plane health • Y.1731 Health	IOS XE does not support SNMP or gNMI collection type for this subservice (CEF route; XConnect).
L2VPN Point to Point with SR underlay	• Device Health • Path Reachability • Summary (aggregator) nodes • VPN Interface Health • Y.1731 Health	IOS XE does not support SNMP or gNMI collection type for this subservice (CEF route; PCEP Session State; SRPolicy State; XConnect).

Supported VPN services	Associated subservices	Details
L3VPN SR	• BGP Neighbor Health (DynExp) • CE-PE Route Health • eBGP Neighbor Health • Path Reachability • Summary (aggregator) nodes • VPN Interface Health	IOS XE does not support SNMP or 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 Plain	• BGP • Neighbor Health (DynExp) • Nexthop Health (DynExp) • Device Health • EVPN Health • Path • Reachability • SLA • Plain LSP Path Health (DynExp) • Summary (aggregator) nodes • VPN Interface Health • VPWS Control Plane health **Note** The reference to ‘Plain’ implies that L2VPN and L3VPN traffic takes the IGP path and does not use any transports, like SR.

Supported VPN services	Associated subservices
L2VPN EVPN SR	• BGP • Neighbor Health (DynExp) • Nexthop Health (DynExp) • Device Health • EVPN Health • Fallback Enabled or Disabled (DynExp) • Path • Reachability • SLA • PCEP Session Health (DynExp) • SR Policy • PCC • PCE • Summary (aggregator) nodes • VPN Interface Health • VPWS Control Plane Health • Y.1731 Health
L2VPN P2P Plain	• Device Health • Path Reachability • Path SLA • Plain LSP Path Health • Summary (aggregator) nodes • VPN Interface Health • VPWS Control Plane Health • Y.1731 Health **Note** The reference to ‘Plain’ implies that L2VPN and L3VPN traffic takes the IGP path and does not use any transports, like SR.

Supported VPN services	Associated subservices
L2VPN Point to Point over MPLS LDP	• Device Health • Fallback Enabled/Disabled • Path Reachability • Path SLA • Summary (aggregator) nodes • VPN Interface Health • VPWS Control Plane Health • Y.1731 Health
L2VPN Point to Point over RSVP	• Device Health • Fallback Enabled/Disabled • Path Reachability • Path SLA • RSVP-TE Health • VPN Interface Health • VPWS Control Plane Health/Xconnect Health • Y.1731 Health
L2VPN Point to Point with SR underlay	• Device Health • Fallback Enabled/Disabled • Path • Reachability • SLA • PCEP Session Health (DynExp) • SR Policy • PCC • PCE • Summary (aggregator) nodes • VPN Interface Health • VPWS Control Plane Health • Y.1731 Health

Supported VPN services	Associated subservices
L3VPN SR	• BGP Neighbor Health (DynExp) • CE-PE Route Health • Device Health • eBGP Neighbor Health • Path Reachability • PCEP Session Health (DynExp) • SR and SRv6 polices • Summary (aggregator) nodes • VPN Interface Health • Vrf Plain LSP Path Health