



Cisco NSO T-SDN Function Pack Bundle

User Guide

Version 7.0.0

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. YOU MUST TAKE FULL RESPONSIBILITY FOR THE 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.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

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.

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

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

Copyright

© 2023 Cisco Systems, Inc. All rights reserved.

Contents

Preface.....	7
Bias-free Documentation Policy	8
Getting Started	9
Cisco Network Services Orchestrator - Overview.....	9
Cisco NSO Transport-SDN Function Pack Bundle - Overview	10
T-SDN Function Pack Bundle Architecture	12
Working with Core Function Packs	14
Creating SR-TE CFP Services	14
Creating SR-ODN Services	15
Creating SR-Policies	18
Creating CS SR-TE Policies.....	22
Creating IETF-L2VPN-NM Services	32
Creating IETF-L2VPN-NM VPWS Service.....	33
Creating IETF-L2VPN-NM VPWS-EVPN On SRv6	36
Creating IETF-L2VPN-NM VPWS SR-TE Policy Association	37
Creating IETF-L2VPN-NM VPWS IETF-TE Association	38
Creating IETF-L2VPN-NM Service with Automated Assurance.....	39
Creating IETF-L2VPN-NM Service with Y1731	40
Creating IETF-L2VPN-NM EVPN-BGP EVPN-VPWS Service	43
Creating IETF-L2VPN-NM MPLS-EVPN	46
Creating the IETF-L2VPN-NM Service MPLS-EVPN SR-TE Policy Association.....	56
Creating IETF-L2VPN-NM Service with CS-SR-TE-Policy.....	57
Creating RFC Route Policy IETF-L2VPN-NM Service Association	58
Creating IETF-L3VPN-NM Services	63
Creating IETF-L3VPN-NM Services with Automated Assurance.....	64
Creating IETF-L3VPN-NM Multicast VPN	64
Creating RFC Route Policy IETF-L3VPN-NM Service Association	83
Monitoring Network Performance using Accedian Probes.....	100
Creating IETF-L3VPN-NM Service with Direct Routing	107
Working with Action Commands - CFP Services	112
Service Cleanup Actions	112

Error Recovery Action	120
get-modifications Action	135
Service re-deploy Action	157
Supporting MultiVendor Services - CFP Services.....	159
Supporting MultiVendors for SR-TE CFP IOSXE CLI Services	160
Supporting MultiVendors for IETF-L2VPN-NM IOSXE CLI Services	160
Supporting MultiVendors for IETF-L3VPN-NM IOSXE CLI Services	161
Using Custom-Templates - CFP Services	161
Applying Custom-Templates	161
Turning Custom-Templates On/Off	162
Sample Custom-Template Payloads for CFP Services	163
Notifications in Core Function Pack Services	170
SR-TE ODN Notifications	170
SR-TE Policy Notifications	171
CS SR-TE Policy Notifications.....	173
IETF-L2VPN-NM Notifications.....	175
IETF-L3VPN-NM Notifications.....	179
Deleting Core Function Pack Services	183
Working with Example Function Packs.....	184
Creating IETF-TE Services.....	184
Creating IETF Network Slice Services	187
Implementing IETF NSS in L2-any-to-any Shared Topology	196
Implementing IETF NSS in L2-hub-spoke Dedicated Topology	197
Implementing IETF NSS in L2-P2P Topology	199
Implementing IETF NSS in L2-P2PV6-Topology	201
Implementing IETF NSS in L3-any-to-any Shared Topology	203
Implementing IETF NSS in L3-any-to-any-custom	205
Implementing IETF NSS in L3-hub-spoke-Dedicated Topology	208
Implementing IETF NSS in L3 with No Peering Protocol	210
Creating Performance Measurement Services	212
Creating Performance Measurement Service with IETF-TE Attachment	234
Creating Performance Measurement Service with SR Attachment	234
Working with Action Commands - Example Services	239
Service Cleanup Actions	239
Error Recovery Action	244

get-modifications Action	250
Service re-deploy Action	286
Supporting MultiVendor Services - Example Services	287
Supporting MultiVendors for IOSXE CLI Services for Example Services	287
Sample Custom-Template Payloads - Example Services	288
IETF-TE Services	288
PM Services	289
IETF-NSS Services	290
Notifications in Example Services	291
IETF-TE Notifications	291
Network Slice Service Notifications	293
Performance Measurement Notifications	296
Deleting Example Services	299
NSO High Availability in TSDN-FP Bundle	300
Appendix A - Package Categories and Packages	301
Package Categories and Packages on a Single NSO Instance	301
LSA Model - Package Categories and Packages	303
Lower-Level Nodes	303
Upper-Level Node	304
Appendix B - Error Recovery	305
Commit-Queue Error Recovery Flow	305
Error Recovery Configuration	305
Recovery Flow	307
Error Recovery Scenarios	310
Non Commit-Queue Error Recovery Flow	334
Appendix C - Yang Models	336
CFP Services	336
SR-ODN Yang Models	336
SR-Policy Yang Models	339
CS SR-TE Yang Models	345
IETF-L2VPN-NM Yang Models	351
IETF-L3VPN-NM Yang Models	391
Example Function Packs	429
IETF-TE Yang Models	429

PM Yang Models	444
NSS Yang Models.....	452
Appendix D - Handling Zombies	484
Appendix E - Supporting MultiVendors for IOSXR Netconf NEDs.....	486
Supporting MultiVendors for IOSXR Netconf Services for SR-TE CFP	486
Supporting MultiVendors for IETF-L2VPN-NM IOSXR Netconf Services	486
Supporting MultiVendors for IOSXR Netconf Services for IETF-L3VPN-NM	487
Supporting MultiVendors for IETF-TE Netconf Services	487
Supporting MultiVendors for PM Netconf Services.....	488

Preface

Abstract

This document serves as the documentation reference and user guide for the NSO Transport SDN Function Pack (T-SDN FP) Bundle version 6.0.0.

Audience

This document describes how to configure and use the T-SDN function packs. This document is intended for Cisco Advanced Services developers, network engineers, and system engineers who configure and deliver the T-SDN automation functionalities to Cisco customers.

Additional Documentation

This documentation requires the reader to have a good understanding of Cisco NSO and its usage as described in the Cisco NSO documentation. For documentation on additional Cisco products, see the Cisco documentation website.

Sl. No.	Documentation
1.	Cisco NSO Transport SDN FP Bundle Installation Guide
2.	Cisco NSO Installation Guide
3.	Cisco NSO User Guide

Bias-free Documentation Policy

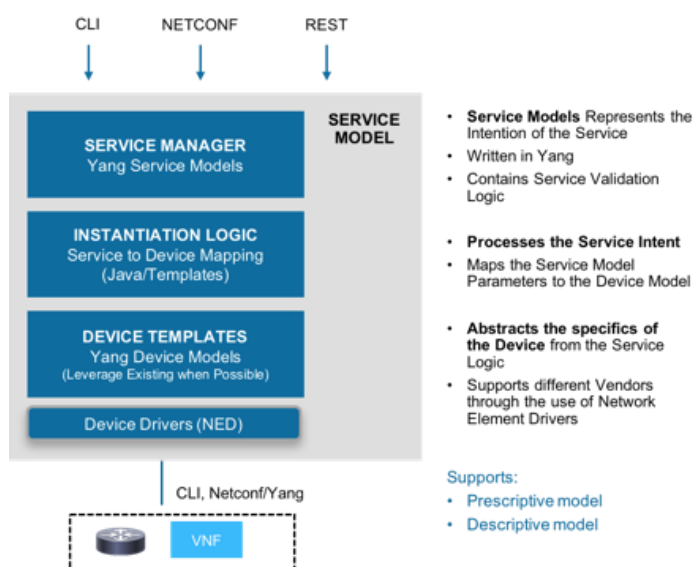
Cisco follows a bias-free documentation policy. According to this policy, Cisco treats all persons with respect—regardless of race, color, ancestry, national origin, age, sex, citizenship, veteran status, marital status, sexual orientation, physical or mental ability, religious creed, or medical condition. Language or graphic elements that offend others violate our business philosophy and our company policy.

Getting Started

This topic provides an overview of Cisco Network Services Orchestration (NSO) and an understanding of Cisco Transport-SDN function packs.

Cisco Network Services Orchestrator - Overview

Network engineers use NSO as a central point of management for the entire network using a network command line interface (CLI), WebUI, and REST. NSO enables service providers to dynamically adapt their orchestration solution with changes in the offered service portfolio. NSO is built on a Model-Driven Architecture that supports the dynamic addition and modification of service definitions. The service models are written in the YANG modeling language (RFC 6020).



You can also upload the configuration information to NSO by using the payloads. These payloads contain the desired service configurations in an XML format. Each service can have a single file or multiple files. Deploy the configuration information by using payloads through the NSO CLI or by invoking the Northbound interface to create and modify service configurations. This documentation uses both network CLI and sample XML payloads.

From a Linux console, invoke the Network Configuration Protocol NETCONF NBI command to push or load the XML payload into NSO.

```
# netconf-console --port=[port_number]--host=[host_IP_address] -u [username] -p [password] --edit-config payload.xml
```

Example

```
# netconf-console --port=830 --host=127.0.0.1 -u admin -p Fr3eB!rd$ --edit-config payload.xml
```

Run the following command to get the configuration details from the configuration database in the NSO. For more information about NETCONF, see **Cisco Networking Services Configuration Guide**.

```
# netconf-console --port=[port_number]--host=[host_IP_address] -u [host_username] -p [host_password] --get-config
```

Example

```
# netconf-console --port=830 --host=127.0.0.1 -u admin -p Fr3eB!rd$ --get-config
```

NETCONF supports the following operations:

- <get-config>
- <edit-config> (operation="create")
- <edit-config> (operation="replace")
- <edit-config> (operation="merge")
- <edit-config> (operation="delete")

The typical workflow when using the network CLI in NSO is as follows:

- All changes are initially made to a (logical) copy of the NSO database of configurations.
- You can view and verify the changes prior to committing them.
- The changes are committed, meaning that the changes are copied to the NSO database and pushed out to the network. Changes that violate integrity constraints or network policies are not committed. The device changes are done in a holistic distributed atomic transaction across all devices in parallel.

Changes either succeed and remain committed or fail and are rolled back as a whole, returning the entire network to the uncommitted state.

Cisco NSO Transport-SDN Function Pack Bundle - Overview

The NSO T-SDN FP Bundle is part of the Cisco Crosswork Network Controller (CNC) solution or can be used as part of a customer solution. NSO T-SDN FP Bundle is based on the SDN architecture to control and manage the transport networks in a multi-layer and multivendor environment. Cisco SDN provides choice in automation and programmability across data centers, campuses, and wide-area networks. Use Cisco software-defined solutions to build intent-based networks (IBN).

IBN transforms a hardware-centric, manual network into a controller-led network that captures business intent and translates it into policies that can be automated and applied consistently across the network. The goal is for the network to continuously monitor and adjust network performance to help ensure desired business outcomes.

NSO T-SDN FPs form an SDN extension, allowing the higher-level network controllers to reroute the network traffic. The NSO T-SDN FPs extend the Cisco Crosswork solution, which is designed to help service providers gain mass awareness, augmented intelligence, and proactive control for a comprehensive, data-driven, intent-based automated network.

The NSO T-SDN FPs are installed on the NSO platform and use orchestration to push services and network configurations. The NSO T-SDN FPs utilize NSO Reactive FastMap (RFM) nano services to manage the full lifecycle of a service.

The NSO T-SDN FP Bundle consists of:

1. Segment Routing-Transport Engineering Core Function Pack (SR-TE CFP) – a productized and supported implementation of SR-TE automation.
2. Internet Engineering Task Force (IETF) implementation of L2VPN (IETF-L2VPN-NM) and L3VPN (IETF-L3VPN-NM) core function packs – a productized and supported implementation of L2NM/L3NM automation.
3. The following Example Function Packs are intended to be customized for each customer deployment by Cisco Customer Experience (CX):
 - IETF-TE (RSVP-TE)
 - IETF-Network-Slice Service (IETF-NSS)
 - Cisco Performance Measurement (Cisco PM)

The IETF - TE function pack is used to push the Resource Reservation Protocol (RSVP) – TE configurations to devices. This function pack uses the IETF-TE model with some modifications for the T-SDN FP Bundle. The IETF Network Slice Service (NSS) Example Function Pack allows you to manage the network slices in your network infrastructure. Performance measurement services help to understand and measure the network performance by using metrics.

SR-TE CFP comprises the following modules/services:

- SR-TE On-Demand-Network (ODN)
- SR-TE Policies
- Circuit Style Segment Routing (CS-SR)

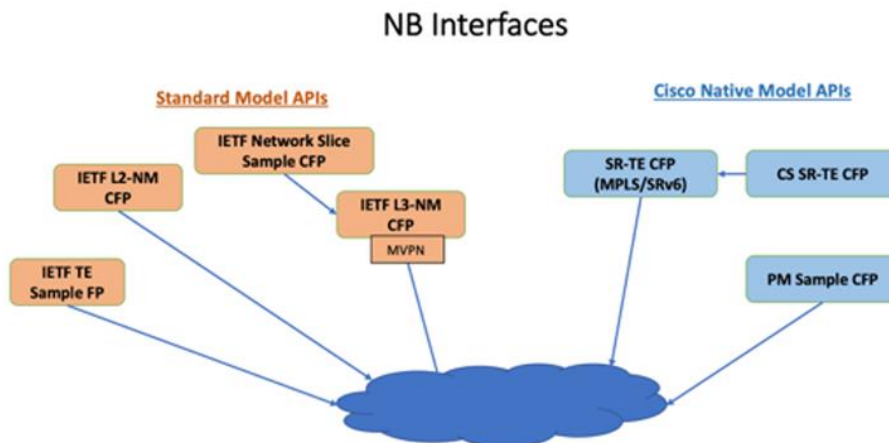
SR policy path computation of SR-TE CFP supports service-specific constraints, such as bandwidth, latency, path diversity, and traffic engineering constraints, such as affinity, bandwidth, cost, and latency coming from the network.

SR-TE CFP configures the SR-TE policies and instantiates services. The SR-TE CFP solution supports the Example Function Packs using the network topology and inventory collected. IETF-L2VPN-NM and IETF-L3VPN-NM services can then be provisioned with a specified segment routing policy. The Example Function Packs utilize SR-TE policies for service instantiation.

Using custom templates, you can add additional configurations to a device without having to modify the SR-TE CFP.

T-SDN Function Pack Bundle Architecture

The following diagram shows the architecture of the T-SDN FP Bundle.



NSO provides Service Provisioning functionality of T-SDN FPs by:

- Provisioning SR-Policy
- Provisioning IETF-L2VPN-NM, IETF-L3VPN-NM over an existing SR-Policy (ODN or Preferred Path option) or IETF-TE

T-SDN FPs are implemented by using the SR-TE CFP, IETF-L2VPN-NM, IETF-L3VPN-NM, and the Example Function Pack IETF-TE.

A router in an SR network can select any path to forward traffic, whether it is explicit or Interior Gateway Protocol (IGP) shortest path.

Each segment is an end-to-end path from the source to the destination and instructs the routers in the provider core network to follow the specified path instead of the shortest path calculated by the IGP. Segments represent sub-paths that a router can combine to form a complete route to a network destination.

SR-TE CFP provides the SR-TE policies and SR-ODN services. T-SDN FPs configure the SR-TE policies and instantiate services, such as the L2NM, L3NM, and IETF-TE function packs that utilize these SR-TE policies.

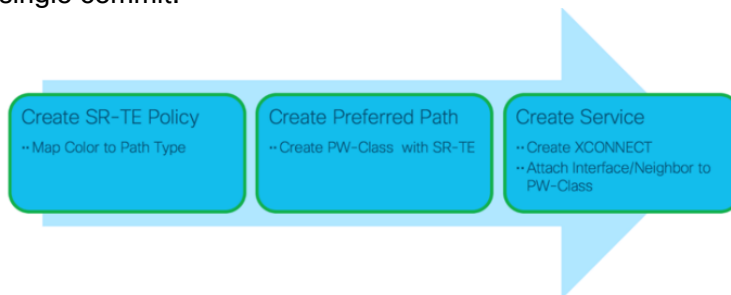
Note: Example Function Packs can be used for reference implementations.

L2VPN allows you to configure point-to-point pseudo wire (PW):

- Static PW or Ethernet VPN Virtual Private Wire Service (EVPN VPWS)

- Association of SR-TE MPLS Policy

The following diagram shows the workflow to instantiate the L2VPN EPVN VPWS. It represents the logical flow of deployment, which can be deployed either separately or in a single commit.



L3VPN allows you to configure:

- VPN
- Interface
- BGP Neighbor
- Association of SR-TE Policy

The following diagram shows the workflow to instantiate L3VPN over SR-TE. It represents the logical flow of deployment, which can be deployed either separately or in a single commit.



The IETF - TE function pack is used to push the Resource Reservation Protocol (RSVP) - TE configurations to devices. RSVP-TE is an extension of RSVP (a standardized IETF protocol) for traffic engineering. With this configuration, you can set up a bidirectional tunnel with both a source and destination for the network traffic.

Working with Core Function Packs

Segment Routing prepares networks for business models where applications can direct network behavior. It provides the right balance between distributed intelligence and centralized optimization and programming. Segment Routing integrates with the multi-service capabilities of MPLS, including Layer 3 VPN (L3VPN), Virtual Private Wire Service (VPWS), Virtual Private LAN Service (VPLS), and Ethernet VPN (EVPN).

The topics in this chapter provide information on how to configure services by using T-SDN FPs.

T-SDN FPs comprise SR-TE CFP. SR-TE CFP consists of the following modules:

- SR-TE ODN module
- SR-TE Policy module
- Circuit-style segment routing module
- IETF-L2VPN-NM module and
- IETF-L3VPN-NM module

IETF-TE, IETF-NSS, and Cisco PM are Example Function Packs used to demonstrate SR-TE CFP. You can choose to install the packages required for these function packs either during or after installing T-SDN FPs.

For more information on the list of packages, see [Appendix A - Package Categories and Packages](#) in this documentation. For more information on how to install the packages, see the *Cisco T-SDN FP Bundle Installation Guide*.

Creating SR-TE CFP Services

SR-TE takes place through a tunnel between a source and a destination pair. In SR-TE, the source calculates the path and encodes it in the packet header as a segment. Each segment is an end-to-end path from the source to the destination and instructs the routers in the provider core network to follow the specified path instead of the shortest path calculated by the IGP.

Segment Routing can be applied on both MPLS and IPv6 data planes. In an SR-MPLS-enabled network, an MPLS label is used as the Segment Identifier (SID), and the source router chooses a path to the destination and encodes the path in the packet header as a stack of labels. However, in a Segment Routing over IPv6 (SRv6) network, an IPv6 address serves as the SID.

SR-TE requires minimal configuration on the source router. The SR-TE CFP provides the SR-TE policies and SR-TE ODN. The SR-ODN and SR-Policy modules allow you to configure the SR-TE policy at various stages.

This chapter discusses each of these SR-TE CFP services. For information about yang models, see [Appendix C - Yang Models](#) in this documentation.

Creating SR-ODN Services

SR-TE ODN service helps configure an SR Policy template for each color for which on-demand SR Policy instantiation is desired.

This topic shows sample payloads to create the SR-ODN service on MPLS and SRv6 networks.

To create the SR-ODN service on MPLS:

1. The following sample payload shows how to create the SR-ODN service. For the yang model, see [Appendix C - Yang Models](#) in this documentation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <odn xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-odn">
      <odn-template>
        <name>SR-CLI-ODN-300</name>
        <head-end>
          <name>PIOSXR-0</name>
        </head-end>
        <head-end>
          <name>PIOSXR-1</name>
        </head-end>
        <color>300</color>
        <bandwidth>200</bandwidth>
        <maximum-sid-depth>100</maximum-sid-depth>
        <dynamic>
          <metric-type>igp</metric-type>
          <metric-margin>
            <absolute>30</absolute>
          </metric-margin>
          <flex-alg>200</flex-alg>
          <pce/>
          <disjoint-path>
            <type>node</type>
            <group-id>10</group-id>
          </disjoint-path>
        </dynamic>
      </odn-template>
    <odn-template>
      <name>SR-CLI-ODN-400</name>
      <head-end>
        <name>PIOSXR-0</name>
      </head-end>
      <color>400</color>
      <dynamic>
```

```

        <metric-type>latency</metric-type>
        <flex-alg>201</flex-alg>
    </dynamic>
</odn-template>
<odn-template>
    <name>SR-ODN-IGP-AFF-20</name>
    <head-end>
        <name>PIOSXR-0</name>
    </head-end>
    <color>20</color>
    <dynamic>
        <metric-type>igp</metric-type>
        <metric-margin>
            <relative>40</relative>
        </metric-margin>
        <affinity>
            <rule>
                <action>exclude-any</action>
                <color>BLUE</color>
            </rule>
        </affinity>
    </dynamic>
</odn-template>
</odn>
</sr-te>
</config>

```

2. Display the plan to view the status of the service. The following table discusses certain plan components unique to this service. For a detailed description of the plan components, see ***NSO documentation***.

Plan Component	Value	Description
Type	head-end	Represents the connected node(device) of the nano plan of a service.
State	cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply	Represents acceptance of service intent and respective configuration application on device


```
admin@ncs> show cisco-sr-te-cfp:sr-te odn odn-template-plan SR-CLI-ODN-300 plan
```

To create SR-ODN service on SRv6:

SRv6 feature is supported only on IOSXR 7.3.2 devices. The following sample payload shows how to create the SR-ODN service with SRv6.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <odn xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-odn">
      <odn-template>
        <name>SR-NC-ODN-SRv6-3000</name>
```

TYPE	NAME	BACK TRACK	GOAL	STATUS CODE	STATE	STATUS	WHEN	ref	POST ACTION STATUS
self	self	false	-	-	init ready	reached	2020-08-03T18:33:12	-	-
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end	PIOSXR-0	false	-	-	init	reached	2020-08-03T18:33:14	-	-
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end	PIOSXR-0				cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply	reached	2020-08-03T18:33:12	-	-
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end	PIOSXR-1	false	-	-	ready	reached	2020-08-03T18:33:12	-	-
					init	reached	2020-08-03T18:33:14	-	-
					cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply	reached	2020-08-03T18:33:12	-	-
					ready	reached	2020-08-03T18:33:14	-	-

```

      <head-end>
        <name>PN73-0</name>
      </head-end>
      <color>3000</color>
      <maximum-sid-depth>100</maximum-sid-depth>
      <dynamic>
        <metric-type>igp</metric-type>
        <metric-margin>
          <absolute>30</absolute>
        </metric-margin>
        <flex-alg>200</flex-alg>
        <pce/>
        <disjoint-path>
          <type>node</type>
          <group-id>10</group-id>
        </disjoint-path>
      </dynamic>
      <srv6>
        <locator>
          <locator-name>Locator-3000</locator-name>
        </locator>
      </srv6>
      <source-address>2001:192:168::7</source-address>
    </odn-template>
  </odn>
</sr-te>
```

```
</config>
```

Display the plan to view the status of the service.

Creating SR-Policies

SR-TE uses policies to steer traffic through the network. Use SR-Policies to instruct the routers in the network to follow a specified path instead of the shortest path calculated by the IGP.

An SR-TE policy path is expressed as a list of segments that specifies the path, called a segment ID (SID) list.

An SR-TE policy uses one or more candidate paths. A candidate path is a single segment list (SID list) or a set of weighted SID lists. If a packet is steered into an SR-TE policy, the head-end pushes the SID list on the packet. The rest of the network executes the instructions embedded in the SID list.

Each segment is an end-to-end path from the source to the destination and instructs the routers in the network to follow the specified path instead of the shortest path calculated by the IGP.

An SR policy is uniquely identified by a tuple - head-end, color, and endpoint. The head-end is where the SR policy is instantiated or implemented.

A tuple (color, endpoint) uniquely identifies an SR policy at a given head-end.

In a Segment Routing over IPv6 (SRv6) network, an IPv6 address serves as the SID. A locator represents the address of a specific SRv6 node and is attached to the policy. The color is mapped to the path type and attached to the VRFs of the VPN service.

There are two types of SR-TE policies:

- **Local Dynamic SR-TE policy:** When configuring the local dynamic SR-TE policy, the head-end calculates the path to the destination address locally. Dynamic path calculation results in a list of interface IP addresses that traffic engineering maps to adj-SID labels. Routes are learned by way of forwarding adjacencies over the TE tunnel.
- **Explicit SR-TE policy:** An explicit path is a list of IP addresses or labels, each representing a node or link in the explicit path. This feature is enabled through the explicit-path command that allows you to create an explicit path and enter a configuration sub-mode to specify the path.

This topic shows sample payloads to create SR-Policies on MPLS and SRv6 networks.

To create SR-Policies on MPLS:

1. The following is a sample payload to create the SR-Policies service. For the yang model, see [Appendix C - Yang Models](#) this documentation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
```

```
<policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-policies">
  <policy>
    <name>SR-Policy-1</name>
    <head-end>
      <name>PIOSXR-0</name>
    </head-end>
    <tail-end>7.7.7.7</tail-end>
    <color>100</color>
    <binding-sid>100</binding-sid>
    <path>
      <preference>100</preference>
      <dynamic>
        <metric-type>te</metric-type>
        <metric-margin>
          <relative>40</relative>
        </metric-margin>
        <constraints>
          <sid-limit>10</sid-limit>
        </constraints>
      </dynamic>
    </path>
    <path>
      <preference>200</preference>
      <explicit>
        <sid-list>
          <name>mysidlist</name>
          <weight>10</weight>
        </sid-list>
        <constraints>
          <affinity>
            <rule>
              <action>include-all</action>
              <color>GREEN</color>
              <color>RED</color>
            </rule>
          </affinity>
        </constraints>
      </explicit>
    </path>
  </policy>
  <policy>
    <name>SR-XE-CLI-ERO</name>
    <head-end>
      <name>XE-CLI-0</name>
```

```

</head-end>
<tail-end>7.7.7.7</tail-end>
<color>100</color>
<binding-sid>100</binding-sid>
<path>
  <preference>200</preference>
  <explicit>
    <sid-list>
      <name>mysidlist-2</name>
    </sid-list>
    <constraints>
      <disjoint-path>
        <type>node</type>
        <group-id>11</group-id>
        <source>2.2.2.2</source>
      </disjoint-path>
      <affinity>
        <rule>
          <action>include-all</action>
          <color>YELLOW</color>
          <color>RED</color>
        </rule>
      </affinity>
    </constraints>
  </explicit>
</path>
<auto-route>
  <auto-route-metric>
    <metric-constant-value>11111</metric-constant-value>
  </auto-route-metric>
  <include-prefixes/>
</auto-route>
<bandwidth>200</bandwidth>
</policy>
<sid-list>
  <name>mysidlist</name>
  <sid>
    <index>1</index>
    <mpls>
      <label>17001</label>
    </mpls>
  </sid>
</sid-list>
<sid-list>

```

```

<name>mysidlist-2</name>
<sid>
  <index>1</index>
  <mpls>
    <label>17002</label>
  </mpls>
  <ipv4>
    <address>1.1.1.1</address>
  </ipv4>
</sid>
<sid>
  <index>2</index>
  <mpls>
    <label>18002</label>
  </mpls>
</sid>
</sid-list>
</policies>
</sr-te>
</config>

```

2. Display the plan to view the status of the service. The following table discusses certain plan components unique to this service. For a detailed description of the plan components, see ***NSO documentation***.

Plan Component	Value	Description
Type	head-end	Represents a connected node (device) of the nano plan of a service.
State	cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply	Represents acceptance of service intent and respective configuration application on the device.

```
admin@ncs> show cisco-sr-te-cfp:sr-te policies policy-plan SR-Policy-1 plan
```

```

-----
TYPE                                     NAME      BACK  TRACK  GOAL  CODE  STATE  STATUS  WHEN  ref  POST ACTION STATUS
-----
self                                     self      false -    -    init  reached 2020-08-17T19:27:17 - -
self                                     self      false -    -    ready reached 2020-08-17T19:27:19 - -
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end P10SKR-0 false -    -    init  reached 2020-08-17T19:27:17 - -
cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply ready reached 2020-08-17T19:27:17 - -
-----

```

To create SR-Policy on SRv6:

SRv6 feature is supported only on IOSXR 7.3.2 devices. The following sample payload shows how to create the SR-Policy service with SRv6.

```

<sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
  <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-policies">
    <policy>
      <name>test</name>
      <head-end>
        <name>XR-SRv6-1</name>
      </head-end>
      <srv6>
        <locator>
          <locator-name>ALG0r7</locator-name>
        </locator>
      </srv6>
      <tail-end>2001:192:168::6</tail-end>
      <color>9004</color>
      <path>
        <preference>100</preference>
        <dynamic>
          <pce/>
          <constraints>
            <affinity>
              <rule>
                <action>include-all</action>
                <color>red</color>
              </rule>
            </affinity>
          </constraints>
        </dynamic>
      </path>
      <source-address>2001:192:168::7</source-address>
    </policy>
  </policies>
</sr-te>

```

Display the plan to view the status of the service.

Creating CS SR-TE Policies

With a Circuit-style (CS) solution, Segment Routing meets the requisites of a connection-oriented transport network, which was earlier delivered over circuit-switched Synchronous Optical Networking (SONET) Synchronous Digital Hierarchy (SDH) networks.

Segment routing allows circuit-centric services to have a circuit, reserved bandwidth, path protection, and path integrity via Operation, Administration, and Management (OAM) path and protection switching.

To use circuit-style service and guarantee end-to-end bandwidth, the network must be partitioned for IP service and circuit-style service.

The CS SR-TE policies offer end-to-end bidirectional transport that provides congruent forward and reverse paths, predictable latency, and disjointness. It offers bandwidth commitment to ensure the service is not impacted due to network load from other services.

Note: CS SR-TE policy is supported only on IOSXR CLI 7.46/7.52 NED, IOSXR NC 7.8 (and later) NED.

CS SR-TE policies contain up to three candidate paths to provide protection for CS SR-TE policies. The candidate path with the highest preference is the working path, which carries traffic.

The second highest preference path is the protect path, which is activated in case of a failure. The protect path can also be established during the policy creation; however, this can set aside bandwidth for use in case of a failure. A third lower preference path, if configured, is called a dynamic restoration path. These paths provide 1:1+R protection for CS SR-TE policies and help to recover from a failure by maintaining the guaranteed bandwidth.

A CS SR-TE policy is of two types – static policy and dynamic policy. A service can use either a static or a dynamic policy at a time. Each policy type can contain up to three candidate paths.

- **Static policy:** In this policy type, the user performs tasks (that are otherwise performed by the Path Computation Element (PCE) controller), such as defining the segment ID list for path computation and bandwidth accounting. The user maintains these characteristics to come up with explicit paths. In a static CS policy with an explicit segment ID, the segment ID list is in the CS SR-TE policy. In a static CS policy with a referenced segment ID, the segment ID that is already available in SR-TE is referenced in the CS policy.
- **Dynamic policy:** In this policy type, the PCE controller performs tasks, such as generating the segment ID list, calculating the path, and maintaining the bi-directional constraint.

CS SR-TE service provisions explicit or dynamic bidirectional SR policies. For each CS SR-TE policy, the disjoint path -> forward-disjoint-id and the reverse-disjoint-id must be globally unique. The tail-end and color for each head-end of the policy must be globally unique.

Note: CS SR-TE policy generates two internal SR-TE policy services: **CS-*<service-name>*-*<endpoint role>*-internal**. You cannot modify these services.

Creating a Static CS SR-TE Policy with Explicit Segment ID

The following is a sample payload to create a static CS SR-TE policy with explicit segment lists.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cs-sr-te-policy xmlns="http://cisco.com/ns/nso/cfp/cisco-cs-sr-te-cfp">
```

```
<name>CS-STATIC-EXPLICIT-SID</name>
<head-end>
  <device>PIOSXR-0</device>
  <ip-address>10.10.10.3</ip-address>
</head-end>
<tail-end>
  <device>PIOSXR-1</device>
  <ip-address>10.10.10.4</ip-address>
</tail-end>
<color>11000</color>
<path-protection></path-protection>
<steering>
  <drop-traffic-on-invalid-path>true</drop-traffic-on-invalid-path>
</steering>
<working-path>
  <explicit>
    <forward-sid-list>
      <sid>
        <index>10</index>
        <mpls>
          <label>15000</label>
        </mpls>
      </sid>
      <sid>
        <index>20</index>
        <mpls>
          <label>15002</label>
        </mpls>
      </sid>
    </forward-sid-list>
    <reverse-sid-list>
      <sid>
        <index>10</index>
        <mpls>
          <label>15000</label>
        </mpls>
      </sid>
      <sid>
        <index>20</index>
        <mpls>
          <label>15000</label>
        </mpls>
      </sid>
    </reverse-sid-list>
```



```
</explicit>
</working-path>
<protect-path>
  <explicit>
    <forward-sid-list>
      <sid>
        <index>10</index>
        <mpls>
          <label>15002</label>
        </mpls>
      </sid>
      <sid>
        <index>20</index>
        <mpls>
          <label>15002</label>
        </mpls>
      </sid>
    </forward-sid-list>
    <reverse-sid-list>
      <sid>
        <index>10</index>
        <mpls>
          <label>15002</label>
        </mpls>
      </sid>
      <sid>
        <index>20</index>
        <mpls>
          <label>15000</label>
        </mpls>
      </sid>
    </reverse-sid-list>
  </explicit>
  <revertive>true</revertive>
  <wait-to-revert-timer>123</wait-to-revert-timer>
</protect-path>
<restore-path>
  <explicit>
    <forward-sid-list>
      <sid>
        <index>10</index>
        <mpls>
          <label>15001</label>
        </mpls>
```

```

    </sid>
    <sid>
      <index>20</index>
      <mpls>
        <label>15003</label>
      </mpls>
    </sid>
  </forward-sid-list>
  <reverse-sid-list>
    <sid>
      <index>10</index>
      <mpls>
        <label>15001</label>
      </mpls>
    </sid>
    <sid>
      <index>20</index>
      <mpls>
        <label>15001</label>
      </mpls>
    </sid>
  </reverse-sid-list>
</explicit>
</restore-path>
</cs-sr-te-policy>
</config>

```

Creating a Static CS SR-TE Policy with Referenced Segment ID

To create a CS SR-TE policy with a referenced SID, you must first create the SID list to reference to in the CS SR-TE policy.

1. The following sample payload shows how to create a SID list to reference later in the policy. The SID list **CS-working-forward** is used to reference the SID list during the CS SR-TE policy creation.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-policies">
      <sid-list>
        <name>CS-working-forward</name>
        <sid>
          <index>10</index>
          <mpls>
            <label>15000</label>

```

```
    </mpls>
  </sid>
<sid>
  <index>20</index>
  <mpls>
    <label>15002</label>
  </mpls>
</sid>
</sid-list>
<sid-list>
  <name>CS-working-reverse</name>
  <sid>
    <index>10</index>
    <mpls>
      <label>15000</label>
    </mpls>
  </sid>
  <sid>
    <index>20</index>
    <mpls>
      <label>15000</label>
    </mpls>
  </sid>
</sid-list>
<sid-list>
  <name>CS-protect-forward</name>
  <sid>
    <index>10</index>
    <mpls>
      <label>15002</label>
    </mpls>
  </sid>
  <sid>
    <index>20</index>
    <mpls>
      <label>15002</label>
    </mpls>
  </sid>
</sid-list>
<sid-list>
  <name>CS-protect-reverse</name>
  <sid>
    <index>10</index>
    <mpls>
```

```
        <label>15002</label>
      </mpls>
    </sid>
    <sid>
      <index>20</index>
      <mpls>
        <label>15000</label>
      </mpls>
    </sid>
  </sid-list>
<sid-list>
  <name>CS-restore-forward</name>
  <sid>
    <index>10</index>
    <mpls>
      <label>15001</label>
    </mpls>
  </sid>
  <sid>
    <index>20</index>
    <mpls>
      <label>15003</label>
    </mpls>
  </sid>
</sid-list>
<sid-list>
  <name>CS-restore-reverse</name>
  <sid>
    <index>10</index>
    <mpls>
      <label>15001</label>
    </mpls>
  </sid>
  <sid>
    <index>20</index>
    <mpls>
      <label>15001</label>
    </mpls>
  </sid>
</sid-list>
</policies>
</sr-te>
</config>
```

2. The following sample payload shows how to reference the SID list **CS-working-forward** created in step 1 and create the static **CS-STATIC** CS SR-TE policy. The **<forward-sid-list-name>** parameter in the payload shows the referenced SID list.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cs-sr-te-policy xmlns="http://cisco.com/ns/nso/cfp/cisco-cs-sr-te-cfp">
    <name>CS-STATIC</name>
    <head-end>
      <device>PN74-0</device>
      <ip-address>10.0.0.1</ip-address>
    </head-end>
    <tail-end>
      <device>PN74-1</device>
      <ip-address>10.0.0.2</ip-address>
    </tail-end>
    <color>10000</color>
    <protection>
      <liveness-detection>
        <active-profile>LIVE_ACTIVE</active-profile>
        <backup-profile>LIVE_BACKUP</backup-profile>
      </liveness-detection>
    </protection>
    <steering>
      <drop-traffic-on-invalid-path>true</drop-traffic-on-invalid-path>
    </steering>
    <working-path>
      <explicit>
        <forward-sid-list-name>CS-working-forward</forward-sid-list-name>
        <reverse-sid-list-name>CS-working-reverse</reverse-sid-list-name>
      </explicit>
    </working-path>
    <protect-path>
      <explicit>
        <forward-sid-list-name>CS-protect-forward</forward-sid-list-name>
        <reverse-sid-list-name>CS-protect-reverse</reverse-sid-list-name>
      </explicit>
      <revertive>true</revertive>
    </protect-path>
    <restore-path>
      <explicit>
        <forward-sid-list-name>CS-restore-forward</forward-sid-list-name>
        <reverse-sid-list-name>CS-restore-reverse</reverse-sid-list-name>
      </explicit>
    </restore-path>
  </cs-sr-te-policy>
```

```
</config>
```

Creating a Dynamic CS SR-TE Policy

The following sample payload shows how to create a dynamic CS SR-TE policy CS-DYNAMIC

It is mandatory to provide path protection and <path> → dynamic → constraints → segments → protection → unprotected-only.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cs-sr-te-policy xmlns="http://cisco.com/ns/nso/cfp/cisco-cs-sr-te-cfp">
    <name>CS-DYNAMIC</name>
    <head-end>
      <device>PIOSXR-0</device>
      <ip-address>10.0.0.1</ip-address>
    </head-end>
    <tail-end>
      <device>PIOSXR-1</device>
      <ip-address>10.0.0.2</ip-address>
    </tail-end>
    <color>10000</color>
    <bandwidth>123</bandwidth>
    <path-protection></path-protection>
    <disjoint-path>
      <forward-path>
        <type>node</type>
        <group-id>5</group-id>
      </forward-path>
      <reverse-path>
        <type>node</type>
        <group-id>10</group-id>
      </reverse-path>
    </disjoint-path>
    <working-path>
      <dynamic>
        <pce></pce>
        <metric-type>igp</metric-type>
        <bidirectional-association-id>100</bidirectional-association-id>
        <constraints>
          <affinity>
            <rule>
              <action>include-all</action>
              <color>123</color>
            </rule>
          </affinity>
        </constraints>
      </dynamic>
    </working-path>
  </cs-sr-te-policy>
</config>
```

```
    </affinity>
    <segments>
      <protection>unprotected-only</protection>
    </segments>
  </constraints>
</dynamic>
</working-path>
<protect-path>
  <dynamic>
    <bidirectional-association-id>50</bidirectional-association-id>
    <metric-type>igp</metric-type>
    <constraints>
      <affinity>
        <rule>
          <action>include-all</action>
          <color>123</color>
        </rule>
      </affinity>
      <segments>
        <protection>unprotected-only</protection>
      </segments>
    </constraints>
  </dynamic>
  <revertive>true</revertive>
  <wait-to-revert-timer>123</wait-to-revert-timer>
</protect-path>
<restore-path>
  <dynamic>
    <bidirectional-association-id>10</bidirectional-association-id>
    <metric-type>igp</metric-type>
    <constraints>
      <affinity>
        <rule>
          <action>include-all</action>
          <color>123</color>
        </rule>
      </affinity>
      <segments>
        <protection>unprotected-only</protection>
      </segments>
    </constraints>
  </dynamic>
</restore-path>
</cs-sr-te-policy>
```

```
</config>
```

Display the plan and verify the policy creation.

```
admin@ncs% run show cs-sr-te-plan CS-DYNAMIC plan
```

TYPE	NAME	BACK		STATUS		STATE	STATUS	WHEN	POST	
		TRACK	GOAL	CODE					ref	ACTION
self	self	false	-	-	init	reached	2022-05-27T23:22:46	-	-	
					ready	reached	2022-05-27T23:22:50	-	-	
cisco-cs-sr-te-cfp-nano-plan:head-end	PIOSKR-0	false	-	-	init	reached	2022-05-27T23:22:46	-	-	
					cisco-cs-sr-te-cfp-nano-plan:config-apply	reached	2022-05-27T23:22:46	-	-	
tail-end	PIOSKR-1	false	-	-	init	reached	2022-05-27T23:22:50	-	-	
					cisco-cs-sr-te-cfp-nano-plan:config-apply	reached	2022-05-27T23:22:46	-	-	
					ready	reached	2022-05-27T23:22:50	-	-	

The following table discusses certain plan components unique to this service. For a detailed description of the plan components, see **NSO documentation**.

Plan Component	Value	Description
Type	head-end	Represents a connected node (device) of the nano plan of a service.
	tail-end	Represents connected tail-end node(device) of the nano plan of a service.
State	cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply	Represents acceptance of service intent and respective configuration application on the device.

Creating IETF-L2VPN-NM Services

The IETF-L2VPN-NM service provides an IETF model overlay of the Flat L2VPN configuration. It implements the **RFC 9291** IETF model. It implements a subset of the IETF YANG model. For the IETF-L2VPN-NM yang model, see **Appendix C - Yang Models** in this documentation.

The IETF-L2VPN-NM has vpws, vpws-evpn, and mpls-evpn service types. The vpws and vpws-evpn types use two vpn-nodes. The mpls-evpn type can have more than two vpn-nodes.

Note: Modifying the vpn-type after creating the service is currently not supported.

In L2NM services, configure and enable the Y1731 probes to monitor the L2NM end-to-end connectivity, delay, and latency, loss, and jitter parameters. Y1731 for a service defines the construct for these parameters for the VPN network services. This construct is applicable only to L2NM service and is currently supported only on XR devices.

Note: Y1731 is not supported for L2NM multipoint service.

Automated Assurance (AA) is an optional feature and is applicable only to L2VPN-L2NM and L3VPN-L3NM services. The AA model, when installed, helps to augment the T-SDN FP Bundle yang models to enable AA. In this model-driven approach, assurance intent is part of service intent.

Installing the AA model enables the service assurance monitoring state for the L2VPN and L3VPN services. When a service intent is defined, NSO CFP sends AA notifications to the North Bound system to indicate changes in the device configuration.

The following topics show how to configure a flat L2VPN-NM service. Make sure the policy exists before associating it with the L2NM service.

Creating IETF-L2VPN-NM VPWS Service

The IETF-L2VPN-NM VPWS service has two types of preferred path policies:

- **SR-TE policy:** An SR-TE policy initiates a single selected path. This is the preferred valid candidate path.

vpn-service → vpn-nodes → vpn-node → te-service-mapping → te-mapping → sr-policy → policy must match the policy type defined in policy-type.

- **RSVP-TE policy:** You may either specify the IETF-TE - RSVP-TE association or manually configure the tunnel-te ID.

<local/remote>-site → te-service-mapping → te-mapping → te-tunnel-list → ietf-te-service: tunnel-te id is derived from the associated IETF-TE → RSVP-TE tunnel id

Note: If the associated IETF-TE service is deleted, the latest cached value of tunnel-te ID remains until it is manually removed/updated.

<local/remote>-site → te-service-mapping → te-mapping → te-tunnel-list → te-tunnel-id: static configuration of tunnel-te id

To create the IETF-L2VPN-NM VPWS service:

1. The following is a sample payload to create the IETF- L2VPN-NM VPWS P2P service.

```
<l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
  <vpn-services>
    <vpn-service>
      <vpn-id>l2nm-p2p</vpn-id>
      <vpn-type>vpn-common:vpws</vpn-type>
      <control-word xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">yes</control-word>
      <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
        <name>CT-CLI-banner</name>
        <variable>
```

```

        <name>BANNER_TEXT</name>
        <value>Welcome_A</value>
    </variable>
</custom-template>
<vpn-nodes>
    <vpn-node>
        <vpn-node-id>PIOSXR-0</vpn-node-id>
        <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
12vpn-ntw">
            <name>CT-CLI-banner</name>
            <variable>
                <name>BANNER_TEXT</name>
                <value>Welcome_B</value>
            </variable>
        </custom-template>
        <signaling-option>
            <ldp-or-l2tp>
                <pw-peer-list>
                    <peer-addr>198.18.1.5</peer-addr>
                    <vc-id>1001</vc-id>
                    <mpls-label xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
12vpn-ntw">101</mpls-label>
                </pw-peer-list>
            </ldp-or-l2tp>
        </signaling-option>
        <vpn-network-accesses>
            <vpn-network-access>
                <id>l2vpn-p2p-acl</id>
                <interface-id>GigabitEthernet0/0/0/1</interface-id>
                <service>
                    <mtu>65</mtu>
                </service>
                <connection>
                    <encapsulation>
                        <encap-type>vpn-common:dot1q</encap-type>
                        <dot1q>
                            <cvlan-id>601</cvlan-id>
                            <tag-operations>
                                <push/>
                                <tag-1>123</tag-1>
                                <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
12vpn-ntw">symmetric</mode>
                            </tag-operations>
                        </dot1q>
                    </encapsulation>
                </connection>
            </vpn-network-access>
        </vpn-network-accesses>
    </vpn-node>
</vpn-nodes>

```

```

        </connection>
    </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <signaling-option>
        <ldp-or-l2tp>
            <pw-peer-list>
                <peer-addr>198.18.1.4</peer-addr>
                <vc-id>1001</vc-id>
                <mpls-label xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">102</mpls-label>
            </pw-peer-list>
        </ldp-or-l2tp>
    </signaling-option>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>l2vpn-p2p-ac1</id>
            <interface-id>GigabitEthernet0/0/0/1</interface-id>
            <service>
                <mtu>64</mtu>
            </service>
            <connection>
                <encapsulation>
                    <encap-type>vpn-common:dot1q</encap-type>
                    <dot1q>
                        <cvlan-id>601</cvlan-id>
                        <tag-operations>
                            <push/>
                            <tag-1>234</tag-1>
                            <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                        </tag-operations>
                    </dot1q>
                </encapsulation>
            </connection>
        </vpn-network-access>
    </vpn-network-accesses>
</vpn-node>
</vpn-nodes>
</vpn-service>
</vpn-services>
</l2vpn-ntw>

```

Display the plan to view the status of the service. For detailed service-state-changes notifications, see chapter **Notifications in Core Function Pack Services** in this documentation.

Creating IETF-L2VPN-NM VPWS-EVPN On SRv6

To configure the IETF-L2VPN-NM service on SRv6, the devices must have an SRv6 network and the evpn-segment routing configured.

Note: When associating SRv6 with L2, define the locator only for te-service-mapping → srv6. Do not define the locator for te-service-mapping → te-mapping → sr-policy → policy.

To create IETF-L2VPN-NM on SRv6:

```
<<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L2NM-EVPN-UNTAGGED</vpn-id>
        <vpn-type>vpn-common:vpws-evpn</vpn-type>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <multi-home xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
              <ethernet-segment-identifier>00.01.00.ac.ce.55.00.0a.00</ethernet-
segment-identifier>
            </multi-home>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>l2vpn-evpn-ac1</id>
                <interface-id>GigabitEthernet0/0/0/1</interface-id>
                <connection>
                  <l2-termination-point>500</l2-termination-point>
                  <encapsulation>
                    <encap-type>vpn-common:untagged-int</encap-type>
                    <untagged xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">
                      <tag-operations>
                        <push/>
                        <tag-1>123</tag-1>
                        <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                      </tag-operations>
                    </untagged>
                  </encapsulation>
                </vpn-network-access>
              </vpn-network-accesses>
            </multi-home>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>
```

```

        </connection>
    </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>l2vpn-evpn-ac1</id>
            <interface-id>GigabitEthernet0/0/0/1</interface-id>
            <connection>
                <l2-termination-point>501</l2-termination-point>
                <encapsulation>
                    <encap-type>vpn-common:untagged-int</encap-type>
                    <untagged xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">
                        <tag-operations>
                            <push/>
                            <tag-1>234</tag-1>
                            <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                        </tag-operations>
                    </untagged>
                </encapsulation>
            </connection>
        </vpn-network-access>
    </vpn-network-accesses>
</vpn-node>
</vpn-nodes>
</vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

Display the plan to view the status of the service.

Creating IETF-L2VPN-NM VPWS SR-TE Policy Association

Any change to an SR-Policy attached to the L2NM service does not reflect automatically on the device. After updating the SR-TE service, manually redeploy the L2NM service to push the corresponding updated L2NM configuration to the device.

To create IETF-L2VPN-NM VPWS SR-TE association:

1. The following is a sample payload to create the IETF- L2VPN-NM VPWS service with a SR-TE policy.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-p2p</vpn-id>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <sr-policy>
                  <policy>SR-CLI-DYNAMIC-P2P-PIOSXR-0</policy>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
          <vpn-node>
            <vpn-node-id>PIOSXR-1</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <sr-policy>
                  <policy>SR-CLI-DYNAMIC-P2P-PIOSXR-1</policy>
                  <fallback>disable</fallback>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>

```

2. Display the plan to view the status of the service.

Creating IETF-L2VPN-NM VPWS IETF-TE Association

Any changes to IETF-TE attached to L2NM service do not automatically reflect on the device. After updating the IETF-TE service, manually redeploy the L2NM service for the corresponding updated L2NM configuration to be pushed to the device.

To create IETF-L2VPN-NM VPWS IETF-TE association:

1. The following is a sample payload to create the IETF-L2VPN-NM VPWS service with IETF-TE association.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-p2p</vpn-id>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <te-tunnel-list>
                  <ietf-te-service>IETF-RSVP-TE</ietf-te-service>
                  <fallback>disable</fallback>
                </te-tunnel-list>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
          <vpn-node>
            <vpn-node-id>PIOSXR-1</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <te-tunnel-list>
                  <te-tunnel-id>321</te-tunnel-id>
                </te-tunnel-list>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>
```

2. Display the plan to view the status of the service.

Creating IETF-L2VPN-NM Service with Automated Assurance

Do the following:

1. The following is a sample payload to create the IETF- L2VPN-NM service with AA.

```
<l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
```

```

<vpn-services>
  <vpn-service>
    <vpn-id>l2nm-p2p-y-1731</vpn-id>
    <service-assurance xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
      <monitoring-state>pause</monitoring-state>
      <profile-name>profile-A custom</profile-name>
      <rule-name>rule-A system</rule-name>
    </service-assurance>
  </vpn-service>
</vpn-services>
</l2vpn-ntw>

```

2. Display the plan to view the status of the service.

Creating IETF-L2VPN-NM Service with Y1731

Currently, only XR devices support Y1731 association with the L2NM service.

1. The following is a sample payload to create the IETF- L2VPN-NM service with Y1731.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-p2p-y-1731</vpn-id>
        <control-word xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">yes</control-word>
        <vpn-type>vpn-common:vpws</vpn-type>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <signaling-option>
              <ldp-or-l2tp>
                <pw-peer-list>
                  <peer-addr>10.10.10.10</peer-addr>
                  <vc-id>1001</vc-id>
                  <mpls-label xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">101</mpls-label>
                </pw-peer-list>
              </ldp-or-l2tp>
            </signaling-option>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>l2vpn-p2p-ac1</id>
                <interface-id>GigabitEthernet0/0/0/1</interface-id>
                <service>
                  <mtu>65</mtu>
                </service>
              </vpn-network-access>
            </vpn-network-accesses>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>

```



```

    </service>
    <connection>
      <encapsulation>
        <encap-type>vpn-common:dot1q</encap-type>
        <dot1q>
          <cvlan-id>601</cvlan-id>
          <tag-operations>
            <push/>
            <tag-1>123</tag-1>
            <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
          </tag-operations>
        </dot1q>
      </encapsulation>
    </connection>
    <ethernet-service-oam>
      <md-name>EVC</md-name>
      <md-level>4</md-level>
      <y-1731>
        <maid>null</maid>
        <mep-id>1</mep-id>
        <id-type xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">number</id-type>
        <message-period>1s</message-period>
        <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
          <name>delay</name>
        </y-1731-profile>
        <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
          <name>loss</name>
        </y-1731-profile>
      </y-1731>
    </ethernet-service-oam>
  </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
  <vpn-node-id>PN73-0</vpn-node-id>
  <signaling-option>
    <ldp-or-l2tp>
      <pw-peer-list>
        <peer-addr>10.10.10.4</peer-addr>
        <vc-id>1001</vc-id>
      </pw-peer-list>
    </ldp-or-l2tp>
  </signaling-option>
</vpn-node>

```

```

        <mpls-label xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">102</mpls-label>
        </pw-peer-list>
    </ldp-or-l2tp>
</signaling-option>
<vpn-network-accesses>
    <vpn-network-access>
        <id>l2vpn-p2p-acl</id>
        <interface-id>GigabitEthernet0/0/0/1</interface-id>
        <service>
            <mtu>64</mtu>
        </service>
        <connection>
            <encapsulation>
                <encap-type>vpn-common:dot1q</encap-type>
                <dot1q>
                    <cvlan-id>601</cvlan-id>
                    <tag-operations>
                        <push/>
                        <tag-1>234</tag-1>
                        <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                    </tag-operations>
                </dot1q>
            </encapsulation>
        </connection>
        <ethernet-service-oam>
            <md-name>EVC</md-name>
            <md-level>4</md-level>
            <y-1731>
                <maid>null</maid>
                <mep-id>2</mep-id>
                <id-type xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">number</id-type>
                <message-period>1s</message-period>
                <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
                    <name>delay</name>
                </y-1731-profile>
                <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
                    <name>loss</name>
                </y-1731-profile>
            </y-1731>
        </ethernet-service-oam>

```

```

        </vpn-network-access>
    </vpn-network-accesses>
</vpn-node>
</vpn-nodes>
</vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

2. Display the plan to view the status of the service.

Creating IETF-L2VPN-NM EVPN-BGP EVPN-VPWS Service

To create an EVPN-VPWS service, you must first create two id-pools and associate the pools with the l2vpn-ntw id-pools. These id-pools are used for evi-id, evi-source, and evi-target configurations, which are allocated by default in the l2vpn-ntw service if no explicit evi-x values are defined.

Note: Resource allocation is an asynchronous process. If the service requires allocation, the service creates a failure (due to device errors) with no commit queue and exhibits similar behavior as the service creates a failure with the commit queue enabled.

The following is a sample payload to associate the pools with the l2vpn-ntw id-pools.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <resource-pools xmlns="http://tail-f.com/pkg/resource-allocator">
    <id-pool xmlns="http://tail-f.com/pkg/id-allocator">
      <name>evi-id-pool</name>
      <range>
        <start>1</start>
        <end>4000</end>
      </range>
    </id-pool>
    <id-pool xmlns="http://tail-f.com/pkg/id-allocator">
      <name>evi-source-target-pool</name>
      <range>
        <start>1</start>
        <end>400</end>
      </range>
    </id-pool>
    <id-pool xmlns="http://tail-f.com/pkg/id-allocator">
      <name>rt-id-pool</name>
      <range>
        <start>1</start>
        <end>65535</end>
      </range>
    </id-pool>
  </resource-pools>
</config>

```

```

    </id-pool>
  </resource-pools>
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <id-pools xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
      <evi-id-pool-name>evi-id-pool</evi-id-pool-name>
      <evi-source-target-pool-name>evi-source-target-pool</evi-source-target-pool-
name>
      <rt-pool-name>rt-id-pool</rt-pool-name>
    </id-pools>
  </l2vpn-ntw>
</config>

```

To define explicit values for `evi-id`, `evi-source`, or `evi-target`, define the respective node under `service` as shown in the following example for `evi-x`. The values are auto-allocated if no explicit value is defined for `evi-x`.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-evpn</vpn-id>
        <evi-id xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">l23</evi-id>
        <evi-source xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">l24</evi-source>
        <evi-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">l25</evi-target>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>

```

To create the IETF-L2VPN-NM EVPN-VPWS service:

1. The following is a sample payload to create the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-evpn</vpn-id>
        <vpn-type>vpn-common:vpws-evpn</vpn-type>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <multi-home xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">

```

```

        <ethernet-segment-identifier>00.01.00.ac.ce.55.00.0a.00</ethernet-
segment-identifier>
    </multi-home>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>l2vpn-evpn-ac1</id>
            <interface-id>GigabitEthernet0/0/0/1</interface-id>
            <connection>
                <encapsulation>
                    <encap-type>vpn-common:dot1q</encap-type>
                    <dot1q>
                        <cvlan-id>601</cvlan-id>
                        <tag-operations>
                            <push/>
                            <tag-1>123</tag-1>
                            <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                        </tag-operations>
                    </dot1q>
                </encapsulation>
            </connection>
        </vpn-network-access>
    </vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>l2vpn-evpn-ac1</id>
            <interface-id>GigabitEthernet0/0/0/1</interface-id>
            <connection>
                <encapsulation>
                    <encap-type>vpn-common:dot1q</encap-type>
                    <dot1q>
                        <cvlan-id>601</cvlan-id>
                        <tag-operations>
                            <push/>
                            <tag-1>234</tag-1>
                            <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                        </tag-operations>
                    </dot1q>
                </encapsulation>
            </connection>
        </vpn-network-access>

```

```

    </vpn-network-accesses>
  </vpn-node>
</vpn-nodes>
</vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

2. Display the plan to view the status of the service.

The following table discusses certain plan components unique to this service. For a detailed description of the plan components, see ***NSO documentation***.

Plan Component	Value	Description
Type	Self	A service includes a component of type self when it constructs its plan; this component can be used by the upper layer to determine the state of the service.
	vpn-node	This component represents the site in the nano plan for a service.
State	ietf-l2vpn-ntw-nano-services:config-apply	Represents acceptance of service intent and respective config application on device in NSO CDB.

```
admin@ncs% run show l2vpn-ntw vpn-services vpn-service-plan l2nm-evpn plan | tab
```

TYPE	NAME	BACK TRACK GOAL	STATUS CODE	STATE	STATUS	WHEN	ref	POST ACTION STATUS
self	self	false	-	init ready	reached	2020-09-22T18:18:41	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-0	false	-	init ready	reached	2020-09-22T18:18:51	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-1	false	-	init ready	reached	2020-09-22T18:18:41	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-1	false	-	init ready	reached	2020-09-22T18:18:51	-	-
				ready	reached	2020-09-22T18:18:41	-	-
				ready	reached	2020-09-22T18:18:51	-	-

Creating IETF-L2VPN-NM MPLS-EVPN

MPLS-EVPN is only supported on platforms that support VPLS Bridge Domains. This feature allows you to create several sites for L2 and use `vpn-route-target` manipulation to connect the sites. At the service level, pick a **`vpn-service-topology`** that defines how the route-targets are assigned for each site. By default, sites are defined to be hub-role unless the **`role`** is defined within the `vpn-node`.

The following are the service topologies used to define the route-targets and assign them to a site.

- **any-to-any**: This is a full mesh topology where all sites can communicate with one another. The same route-targets are assigned across all nodes.

- **hub-spoke:** This topology provides hub and spoke multipoint connectivity to the sites. In this topology, route targets are assigned in a way that allows the spokes to communicate with the hub but not with each other. A spoke is defined if a **node-role** is defined for a given site.
- **Custom:** In this topology, you must manually define the route-targets for each site.

Examples for each topology

The following sample payloads show the implementation of each topology. Display the service plan to view the configuration for the service.

IETE-L2VPN-NM EVPN-MPLS - any-to-any

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>EVPN-MP-ELAN</vpn-id>
        <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
          <name>CT-CLI-logging</name>
        </custom-template>
        <evi-id xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">12345</evi-id>
        <vpn-type>vpn-common:mpls-evpn</vpn-type>
        <vpn-service-topology>vpn-common:any-to-any</vpn-service-topology>
        <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
          <hub-rt-value>0:65001:100</hub-rt-value>
        </vpn-target>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">
              <name>CT-CLI-syslog</name>
            </custom-template>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>a1</id>
                <interface-id>GigabitEthernet1</interface-id>
                <connection>
                  <encapsulation>
                    <encap-type>vpn-common:untagged-int</encap-type>
                  </encapsulation>
                </connection>
              </vpn-network-access>
            </vpn-node>
          </vpn-nodes>
        </vpn-service>
      </vpn-services>
    </l2vpn-ntw>
  </config>
```

```

    </vpn-network-accesses>
    <control-word-disable xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">false</control-word-disable>
  </vpn-node>
  <vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>a1</id>
        <interface-id>GigabitEthernet1</interface-id>
        <connection>
          <encapsulation>
            <encap-type>vpn-common:untagged-int</encap-type>
          </encapsulation>
        </connection>
      </vpn-network-access>
    </vpn-network-accesses>
    <control-word-disable xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">false</control-word-disable>
  </vpn-node>
  <vpn-node>
    <vpn-node-id>PIOSXR-2</vpn-node-id>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>a1</id>
        <interface-id>GigabitEthernet1</interface-id>
        <connection>
          <encapsulation>
            <encap-type>vpn-common:untagged-int</encap-type>
          </encapsulation>
        </connection>
      </vpn-network-access>
    </vpn-network-accesses>
    <control-word-disable xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">false</control-word-disable>
  </vpn-node>
</vpn-nodes>
  <bridge-group xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">BRIDGE</bridge-group>
</vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

IETE-L2VPN-NM EVPN-PBB - hub-spoke

In this topology, vpn-node PIOSXR-0 acts as the hub while vpn-nodes PIOSXR-1 and PIOSXR-2 act as spokes since they have vpn-node-role defined.

If the vpn-target is not defined and the node role is spoke-single-homing, it will be treated as etree leaf.

```
<l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
  <vpn-services>
    <vpn-service>
      <vpn-id>EVPN-MP-ETREE</vpn-id>
      <evi-id xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">12346</evi-id>
      <vpn-type>vpn-common:mpls-evpn</vpn-type>
      <vpn-service-topology>vpn-common:hub-spoke</vpn-service-topology>
      <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
        <hub-rt-value>0:65001:100</hub-rt-value>
        <spoke-rt-value>0:65001:101</spoke-rt-value>
      </vpn-target>
      <vpn-nodes>
        <vpn-node>
          <vpn-node-id>PIOSXR-0</vpn-node-id>
          <multi-home xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
            <ethernet-segment-identifier>00.01.00.ac.ce.55.00.0a.00</ethernet-segment-identifier>
          </multi-home>
          <te-service-mapping xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
            <te-mapping>
              <odn>
                <route-policy>SET_COLOR_EVPN_MP_service_L2-RP-multipoint</route-policy>
                <attach-point>
                  <parent-rr-route-policy>L2-ATTACH</parent-rr-route-policy>
                </attach-point>
              </odn>
            </te-mapping>
          </te-service-mapping>
          <signaling-option>
            <evpn-policies>
              <mac-learning-mode>control-plane</mac-learning-mode>
            </evpn-policies>
          </signaling-option>
          <vpn-network-accesses>
            <vpn-network-access>
              <id>a1</id>
              <interface-id>GigabitEthernet0/0/0/1</interface-id>
            </vpn-network-access>
          </vpn-network-accesses>
        </vpn-node>
      </vpn-nodes>
    </vpn-service>
  </vpn-services>
</l2vpn-ntw>
```

```

    <service>
      <mtu>1000</mtu>
    </service>
  </connection>
  <encapsulation>
    <encap-type>vpn-common:untagged-int</encap-type>
  </encapsulation>
</connection>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
  <vpn-node-id>PIOSXR-1</vpn-node-id>
  <role>vpn-common:spoke-role</role>
  <signaling-option>
    <evpn-policies>
      <mac-learning-mode>control-plane</mac-learning-mode>
    </evpn-policies>
  </signaling-option>
  <vpn-network-accesses>
    <vpn-network-access>
      <id>a1</id>
      <interface-id>GigabitEthernet2</interface-id>
      <connection>
        <encapsulation>
          <encap-type>vpn-common:dot1q</encap-type>
          <dot1q>
            <cvlan-id>100</cvlan-id>
            <tag-operations>
              <pop>1</pop>
              <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
12vpn-ntw">symmetric</mode>
            </tag-operations>
          </dot1q>
        </encapsulation>
      </connection>
    </vpn-network-access>
  </vpn-network-accesses>
</vpn-node>
<vpn-node>
  <vpn-node-id>PIOSXR-2</vpn-node-id>
  <role>vpn-common:spoke-role</role>
  <multi-home xmlns="http://cisco.com/ns/nso/fp/examples/cisco-12vpn-ntw">
    <spoke-multi-homed/>
  </multi-home>

```

```

        <ethernet-segment-identifier>00.01.00.ac.ce.55.00.0a.00</ethernet-
segment-identifier>
    </multi-home>
    <signaling-option>
        <evpn-policies>
            <mac-learning-mode>control-plane</mac-learning-mode>
        </evpn-policies>
    </signaling-option>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>a1</id>
            <interface-id>GigabitEthernet2</interface-id>
            <connection>
                <encapsulation>
                    <encap-type>vpn-common:dot1q</encap-type>
                    <dot1q>
                        <cvlan-id>100</cvlan-id>
                        <tag-operations>
                            <pop>1</pop>
                            <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">symmetric</mode>
                        </tag-operations>
                    </dot1q>
                </encapsulation>
            </connection>
        </vpn-network-access>
    </vpn-network-accesses>
</vpn-node>
</vpn-nodes>
    <bridge-group xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">BRIDGE</bridge-group>
</vpn-service>
</vpn-services>
</l2vpn-ntw>

```

IETE-L2VPN-NM EVPN-PBB - custom

In this topology, the vpn-targets are set for each vpn-node under vpn-node → signaling-options → evpn-policies. If the vpn-target is not defined and the node role is spoke-single-homing, it will be treated as etree leaf.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
    <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
        <id-pools xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
            <evi-id-pool-name>evi-id-pool</evi-id-pool-name>
        </id-pools>
    </l2vpn-ntw>
</config>

```

```

    <evi-source-target-pool-name>evi-source-target-pool</evi-source-target-pool-
name>
  </id-pools>
  <vpn-services>
    <vpn-service>
      <vpn-id>EVPN-MP-CUSTOM</vpn-id>
      <evi-id xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">12346</evi-id>
      <vpn-type>vpn-common:mpls-evpn</vpn-type>
      <vpn-service-topology>vpn-common:custom</vpn-service-topology>
      <vpn-nodes>
        <vpn-node>
          <vpn-node-id>PIOSXR-0</vpn-node-id>
          <multi-home xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
            <ethernet-segment-identifier>00.01.00.ac.ce.55.00.0a.00</ethernet-
segment-identifier>
            </multi-home>
            <te-service-mapping xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">
              <te-mapping>
                <odn>
                  <route-policy>SET_COLOR_EVPN_MP_service_L2-RP-multipoint</route-
policy>
                  <attach-point>
                    <parent-rr-route-policy>L2-ATTACH</parent-rr-route-policy>
                  </attach-point>
                </odn>
              </te-mapping>
            </te-service-mapping>
            <signaling-option>
              <evpn-policies>
                <mac-learning-mode>control-plane</mac-learning-mode>
                <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
                  <id>0</id>
                  <route-targets>
                    <route-target>0:65000:100</route-target>
                  </route-targets>
                  <route-target-type>both</route-target-type>
                </vpn-target>
                <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
                  <id>1</id>
                  <route-targets>
                    <route-target>0:65000:200</route-target>

```

```

        </route-targets>
        <route-target-type>import</route-target-type>
    </vpn-target>
    <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
        <id>3</id>
        <route-targets>
            <route-target>0:65000:300</route-target>
        </route-targets>
        <route-target-type>export</route-target-type>
    </vpn-target>
</evpn-policies>
</signaling-option>
<vpn-network-accesses>
    <vpn-network-access>
        <id>a1</id>
        <interface-id>GigabitEthernet0/0/0/1</interface-id>
        <service>
            <mtu>1000</mtu>
        </service>
        <connection>
            <encapsulation>
                <encap-type>vpn-common:untagged-int</encap-type>
            </encapsulation>
        </connection>
    </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <role>vpn-common:spoke-role</role>
    <signaling-option>
        <evpn-policies>
            <mac-learning-mode>control-plane</mac-learning-mode>
            <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
                <id>0</id>
                <route-targets>
                    <route-target>0:65000:100</route-target>
                </route-targets>
                <route-target-type>import</route-target-type>
            </vpn-target>
            <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
                <id>1</id>

```

```

        <route-targets>
            <route-target>0:65000:400</route-target>
        </route-targets>
        <route-target-type>export</route-target-type>
    </vpn-target>
</evpn-policies>
</signaling-option>
<vpn-network-accesses>
    <vpn-network-access>
        <id>a1</id>
        <interface-id>GigabitEthernet2</interface-id>
        <connection>
            <encapsulation>
                <encap-type>vpn-common:dot1q</encap-type>
                <dot1q>
                    <cvlan-id>100</cvlan-id>
                    <tag-operations>
                        <pop>1</pop>
                        <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                    </tag-operations>
                </dot1q>
            </encapsulation>
        </connection>
    </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-2</vpn-node-id>
    <multi-home xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
        <spoke-multi-homed/>
        <ethernet-segment-identifier>00.01.00.ac.ce.55.00.0a.00</ethernet-
segment-identifier>
    </multi-home>
    <role>vpn-common:spoke-role</role>
    <signaling-option>
        <evpn-policies>
            <mac-learning-mode>control-plane</mac-learning-mode>
            <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
                <id>0</id>
                <route-targets>
                    <route-target>0:65000:100</route-target>
                </route-targets>
            </vpn-target>
        </evpn-policies>
    </signaling-option>

```

```

        <route-target-type>import</route-target-type>
    </vpn-target>
    <vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
        <id>1</id>
        <route-targets>
            <route-target>0:65000:500</route-target>
        </route-targets>
        <route-target-type>export</route-target-type>
    </vpn-target>
</evpn-policies>
</signaling-option>
<vpn-network-accesses>
    <vpn-network-access>
        <id>a1</id>
        <interface-id>GigabitEthernet2</interface-id>
        <connection>
            <encapsulation>
                <encap-type>vpn-common:dot1q</encap-type>
                <dot1q>
                    <cvlan-id>100</cvlan-id>
                    <tag-operations>
                        <pop>1</pop>
                        <mode xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l2vpn-ntw">symmetric</mode>
                    </tag-operations>
                </dot1q>
            </encapsulation>
        </connection>
    </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
</vpn-nodes>
    <bridge-group xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">BRIDGE</bridge-group>
</vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

Creating the IETF-L2VPN-NM Service MPLS-EVPN SR-TE Policy Association

For every vpn-site, you can configure the preferred path for the evpn-bgp service. SR-TE policy initiates a single (selected) path. This is the preferred valid candidate path.

Any changes to an SR-Policy attached to the L2NM service do not reflect automatically. After updating the SR-TE service, manually redeploy the L2NM service to push the corresponding updated L2NM configuration to the device.

To create the IETF-L2VPN-NM Service MPLS-EVPN SR-TE association:

1. The following is a sample payload to create the service. Display the plan to view the status of the service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-evpn</vpn-id>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <sr-policy>
                  <policy>SR-CLI-ERO-VPWS-PIOSXR-0</policy>
                  <fallback>disable</fallback>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
          <vpn-node>
            <vpn-node-id>PIOSXR-1</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <sr-policy>
                  <policy>SR-CLI-ERO-VPWS-PIOSXR-1</policy>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
```



```

    </l2vpn-ntw>
</config>

```

Creating IETF-L2VPN-NM Service with CS-SR-TE-Policy

An out-of-band CS SR-TE policy or provisioned CS SR-TE service is applicable for EVPN-VPWS type service.

Any changes to an SR-Policy attached to the L2NM service do not reflect automatically. After updating the SR-TE service, manually redeploy the L2NM service to push the corresponding updated L2NM configuration to the device.

To create the IETF-L2VPN-NM Service with CS-SR-TE association:

1. The following is a sample payload to create the service and associate it with the policy.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>l2nm-evpn</vpn-id>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <sr-policy>
                  <policy-type>cs-policy</policy-type>
                  <policy>CS-STATIC-EXPLICIT-SID</policy>
                  <fallback>disable</fallback>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
          <vpn-node>
            <vpn-node-id>PIOSXR-1</vpn-node-id>
            <te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
              <te-mapping>
                <sr-policy>
                  <policy-type>cs-policy</policy-type>
                  <policy>CS-STATIC-EXPLICIT-SID</policy>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>

```

```

    </vpn-nodes>
  </vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

2. Display the plan to view the status of the service.

Creating RFC Route Policy IETF-L2VPN-NM Service Association

The L2 route-policy model is standardized by using the RFC routing policy model RFC#9067. The RFC model provides a routing policy framework to use in the L2NM service. For the routing policy yang model, see [Appendix C - Yang Models](#) in this documentation.

Configure the route policy to filter or prioritize the data traffic in your network infrastructure. You must create defined-sets to create the policy definitions. You can choose to customize the policy with defined-sets that work on your device and meet your infrastructure.

Note: TSDN FP Bundle CFP supports only a subset of the IETF model of RFC#9067. For more information about the RFC routing policy model RFC#9067, see the RFC#9067 model documentation.

Create the RFC route policy and use the policy in the L2NM service.

The route policy model is available under L2VPN-routing-policy. In the route policy model, you must first create the defined sets and use these sets to create policy definitions. Policy definitions create named sequences of policy statements.

Creating Defined-sets

Defined-sets contain sets of attributes to match conditions in a statement or to set values for an action in a statement.

To create defined sets:

A defined-set must contain the rd-sets, evpn-route-type-sets, tag-sets, and tag-value.

The rd-set contains sets of all rd-prefixes you can use. The match-rd-set references the rd-set to match conditions in a policy-definition. The rd-list contains prefixes, such as ASN:nn, IPv4-address:nn, VPN route distinguisher, list of addresses in the RD set, BGP VPN RD in IPv4 format a.b.c.d:n (for example, 10.0.0.2:666, and BGP VPN RD in ASN format asn:n (for example, 10002:666). For more information about routing information and the accepted list of device configurations, see the Routing Configuration Guide on the Cisco documentation website.

The `evpn-route-type-sets` contain sets of all `evpn-route-type` values ranging from zero (0) to 11. The `match-evpn-route-type-set` references the `evpn-route-type-set` to match conditions in a policy-definition.

The **tag-sets** contain the list of colors referenced by `bgp-actions`. These tag-sets are also used to generate values for the **opaque and set** parameters in the **extcommunity-set**. For example, the tag-set value `COLOR_100` in a defined-set sets the **opaque** parameter value to `COLOR_100` and the **set** parameter value to 100 in the **extcommunity-set** configuration on a device.

The **tag-value** represents a color value in the range 1-4294967295 configured within a `bgp-actions` policy statement. A tag-set can have only one tag-value member. For example, a tag-set `COLOR_100` indicates the tag-value is 100.

The following shows a **defined-sets** sample with the discussed parameters to create a route policy.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-routing-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
routing-policy">
    <defined-sets>
      <rd-sets>
        <rd-set>
          <name>rd_set_A</name>
          <rd>1.1.1.1:1</rd>
          <rd>1.1.1.2:1</rd>
        </rd-set>
      </rd-sets>
    <evpn-route-type-sets>
      <evpn-route-type-set>
        <name>set_A</name>
        <evpn-route-type>1</evpn-route-type>
        <evpn-route-type>3</evpn-route-type>
      </evpn-route-type-set>
      <evpn-route-type-set>
        <name>set_B</name>
        <evpn-route-type>1</evpn-route-type>
      </evpn-route-type-set>
      <evpn-route-type-set>
        <name>set_C</name>
        <evpn-route-type>1</evpn-route-type>
        <evpn-route-type>3</evpn-route-type>
        <evpn-route-type>5</evpn-route-type>
      </evpn-route-type-set>
    </evpn-route-type-sets>
  <tag-sets>
```

```

    <tag-set>
      <name>COLOR_100</name>
      <tag-value>100</tag-value>
    </tag-set>
    <tag-set>
      <name>COLOR_101</name>
      <tag-value>101</tag-value>
    </tag-set>
  </tag-sets>
</defined-sets>
</l2vpn-routing-policy>
</config>

```

Creating a Route Policy

A route policy is a policy definition. Route policies are created and attached to an IETF-L2VPN-NM service to prioritize the network traffic.

Policy definitions create named sequences of policy statements. A policy can be applied to other policies. A single policy definition maps to a single route policy. The policy definition name must be the name of the route policy to push to the device. The policy definition name must be alphanumeric and optionally contain the special characters period, hyphen, and an underscore. The policy name must, however, begin with either an alphabet or a number.

Each policy definition contains a list of statements. A statement name identifies a statement list and must be unique. A statement consists of conditions and actions. Each statement is evaluated in the order it was created in the route policy payload.

A statement can optionally contain conditions to match the `rd-sets` or `evpn-route-type-sets` in a route policy. In T-SDN FP Bundle CFP, you can create separate statements to match `rd-sets` and `evpn-route-type-sets`. Each match condition results in an If-then clause in the route-policy.

Note: Do not use the hidden `match-manual` field under policy definitions.

You must mandatorily configure `bgp-actions` in a statement. If the route-policy contains the If-Then clause, the actions appear after the **then** clause. You can also apply the actions to a route-policy without an If-Then clause. The `bgp-actions` may or may not have conditions associated with it.

The `bgp-actions/set-ext-community/ext-community-set-ref` action is used to configure the `set-extcommunity-color` property in the route policy. `Ext-community-set-ref` is mandatory in a policy, and it references tag-sets under defined-sets for color values.

The following is a sample route-policy payload. The route-policy contains two separate statements for `rd-set` condition and `evpn-route-set` condition. Since these conditions are present, it is mandatory to configure `bgp-actions` for each condition.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
```

```
<l2vpn-routing-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
routing-policy">
  <policy-definitions>
    <policy-definition>
      <name>SAMPLE_1</name>
      <statements>
        <statement>
          <name>stmt1</name>
          <conditions>
            <match-evpn-route-type-set>
              <evpn-route-type-set>set_C</evpn-route-type-set>
            </match-evpn-route-type-set>
          </conditions>
          <actions>
            <bgp-actions>
              <set-ext-community>
                <ext-community-set-ref>COLOR_100</ext-community-set-ref>
              </set-ext-community>
            </bgp-actions>
          </actions>
        </statement>
        <statement>
          <name>stmt2</name>
          <conditions>
            <match-rd-set>
              <rd-set>rd_set_A</rd-set>
            </match-rd-set>
          </conditions>
          <actions>
            <bgp-actions>
              <set-ext-community>
                <ext-community-set-ref>COLOR_101</ext-community-set-ref>
              </set-ext-community>
            </bgp-actions>
          </actions>
        </statement>
      </statements>
    </policy-definition>
  </policy-definitions>
</l2vpn-routing-policy>
</config>
```

Creating Routing-Profile-Identifiers

Routing-profile-identifiers are created automatically under l2vpn-ntw/vpn-profiles/valid-provider-identifiers/routing-profile-identifier to use the policy definitions in an L2NM service.

The following example shows how to manually create the routing-profile-identifiers.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-profiles>
      <valid-provider-identifiers>
        <routing-profile-identifier>
          <id>SAMPLE_1</id>
        </routing-profile-identifier>
      </valid-provider-identifiers>
    </vpn-profiles>
  </l2vpn-ntw>
</config>
```

Associating the Route Policy with IETF-L2VPN-NM Service

Before associating SR-TE ODN with l2vpn-ntw, make sure the **parent-rr-route-policy** exists on the device. When the **parent-rr-route-policy attach-point** is defined, the original value of the parent-route-policy is preserved with additional local route policies that are applied after the original route-policy blob.

Note: Do not use the reserved policy name PASS_ALL as the L2 parent-rr-route-policy.

The following is a sample payload to show the association of the route policy with an IETF-L2VPN-NM-MPLS-EVPN service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>EVPN-MP-ELAN</vpn-id>
        <vpn-type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:mpls-evpn</vpn-type>
        <vpn-service-topology xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:any-to-any</vpn-service-topology>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>a1</id>
                <interface-id>GigabitEthernet1</interface-id>
                <connection>
```

```

        <encapsulation>
            <encap-type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:untagged-
int</encap-type>
        </encapsulation>
    </connection>
</vpn-network-access>
</vpn-network-accesses>
<te-service-mapping
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
    <te-mapping>
        <odn>
            <route-policy>SAMPLE_1</route-policy>
            <attach-point>
                <parent-rr-route-policy>L2-ATTACH</parent-rr-route-
policy>
            </attach-point>
        </odn>
    </te-mapping>
</te-service-mapping>
<control-word-disable
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">false</control-word-
disable>
    </vpn-node>
</vpn-nodes>
<evi-id xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">12345</evi-id>
<vpn-target xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
    <hub-rt-value>0:65001:100</hub-rt-value>
</vpn-target>
<bridge-group xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">BRIDGE</bridge-group>
</vpn-service>
</vpn-services>
</l2vpn-ntw>
</config>

```

Creating IETF-L3VPN-NM Services

The IETF-L3VPN-NM service provides an IETF model overlay of the flat L3VPN configuration. It implements the RFC#9182 model. It implements a subset of the IETF YANG model.

L3NM service can configure multiple `vpn-network-access` per `vpn-node`. Each `vpn-network-access` has its own component in the plan in

`<VPN_NODE_ID>_<VPN_NETWORK_ACCESS ID>` format.

Automated Assurance (AA) is an optional feature and is applicable only to Flat IETF-L3VPN-NM Function Pack and Flat IETF-L2VPN-NM Example Function Pack. The AA model, when installed, helps to augment the T-SDN FP Bundle yang models to enable AA. In this model-driven approach, assurance intent is part of service intent.

Installing the AA model enables the service assurance monitoring state for the IETF-L2VPN-NM and IETF-L3VPN-NM services. When a service intent is defined, NSO CFP sends AA notifications to the North Bound system to indicate changes in the device configuration.

Creating IETF-L3VPN-NM Services with Automated Assurance

Do the following:

1. The following is a sample payload to create the IETF-L3VPN-NM service with AA.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>0-65008740</vpn-id>
        <service-assurance>
          <monitoring-state>pause</monitoring-state>
          <profile-name>profile-A system</profile-name>
          <rule-name>rule-A custom</rule-name>
        </service-assurance>
      </vpn-service>
    </vpn-services>
  </l3vpn-ntw>
</config>
```

2. Display the plan to view the status of the service.

Creating IETF-L3VPN-NM Multicast VPN

The Multicast VPN (MVPN) feature provides the ability to support multicast over a Layer 3 VPN. MVPN IP allows a service provider to configure and support multicast traffic in an MPLS VPN environment.

MVPN provides a scalable method to dynamically send information to multiple locations, provides high-speed information delivery, and provides connectivity through a shared infrastructure.

MVPN establishes a static default multicast distribution tree (MDT) for each multicast domain. The default MDT defines the path used by the PE routers to send multicast data and control messages to every other PE router in the multicast domain.

The **mdt** keyword has been added to the `address-family ipv4` command to configure an MDT address-family session.

For more information about MVPN, see *IP Multicast: MVPN Configuration Guide* on the Cisco documentation website.

You must consider the following information to create static and dynamic MVPN.

1. For static TREE SID:

- the static policy name must be unique across a VPN.

IMPORTANT:

For static TREE SID, the group-address must be in IPv4 multicast prefix range 224/4. The source-address must be a unicast address and must not be in the multicast prefix range. If the group-address is in Source-Specific Multicast prefix range 232/8, then there must be at least one source-address.

2. For dynamic MVPN:

- the route distinguisher must be unique per PE per VPN.
- the route policy data-mdt is supported on cisco-iosxr-nc-7.5.2 and later.

3. TSDN FP Bundle CFP does not support configuring PCE.

Prerequisites

1. Make sure your device is configured with the multicast-routing information.

```
multicast-routing
 address-family ipv4
   interface Loopback0
     enable
   !
   mdt source Loopback0
   !
router bgp 1
 address-family ipv4 mvpn
 !
```

2. Create and use the route policy as described in section **Creating RFC Route Policy IETF-L3VPN-NM Service Association**.

Creating Dynamic MVPN with mdt default, route policy, and access control list

The following is a sample payload to demonstrate the service creation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-MVPN-DYNAMIC-DEFAULT</vpn-id>
        <vpn-instance-profiles>
```

```

<vpn-instance-profile>
  <profile-id>ie_00</profile-id>
  <rd>0:120:110</rd>
  <address-family>
    <address-family>ipv4</address-family>
    <vpn-targets>
      <vpn-target>
        <id>1</id>
        <route-targets>
          <route-target>0:120:110</route-target>
        </route-targets>
        <route-target-type>both</route-target-type>
      </vpn-target>
    </vpn-targets>
  </address-family>
  <address-family>
    <address-family>ipv6</address-family>
    <vpn-targets>
      <vpn-target>
        <id>1</id>
        <route-targets>
          <route-target>0:120:110</route-target>
        </route-targets>
        <route-target-type>both</route-target-type>
      </vpn-target>
    </vpn-targets>
  </address-family>
</vpn-instance-profile>
</vpn-instance-profiles>
<vpn-nodes>
  <vpn-node>
    <vpn-node-id>PIOSXR-0</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
      <vpn-instance-profile>
        <profile-id>ie_00</profile-id>
      </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>24</id>
        <interface-id>HundredGigE0/0/0/3</interface-id>
        <ip-connection>
          <ipv4>

```

```

        <local-address>160.160.160.1</local-address>
        <prefix-length>24</prefix-length>
    </ipv4>
</ip-connection>
<routing-protocols>
  <routing-protocol>
    <id>TEST_PROTO</id>
    <type>bgp-routing</type>
    <bgp>
      <address-family>ipv4</address-family>
      <redistribute-connected>
        <address-family>ipv4</address-family>
        <enable>>true</enable>
      </redistribute-connected>
      <neighbor>160.160.160.2</neighbor>
      <peer-as>65003</peer-as>
      <multihop>11</multihop>
    </bgp>
  </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
  <is-receiver-site-only>true</is-receiver-site-only>
</multicast>
</vpn-node>
<vpn-node>
  <vpn-node-id>PIOSXR-1</vpn-node-id>
  <local-as>1</local-as>
  <active-vpn-instance-profiles>
    <vpn-instance-profile>
      <profile-id>ie_00</profile-id>
    </vpn-instance-profile>
  </active-vpn-instance-profiles>
  <vpn-network-accesses>
    <vpn-network-access>
      <id>23</id>
      <interface-id>HundredGigE0/0/0/2</interface-id>
      <ip-connection>
        <ipv4>
          <local-address>161.161.161.1</local-address>
          <prefix-length>24</prefix-length>
        </ipv4>
      </ip-connection>
    </vpn-network-access>
  </vpn-network-accesses>
</vpn-node>

```

```

    <routing-protocols>
      <routing-protocol>
        <id>TEST_PROTO</id>
        <type>bgp-routing</type>
        <bgp>
          <address-family>ipv4</address-family>
          <redistribute-connected>
            <address-family>ipv4</address-family>
            <enable>>true</enable>
          </redistribute-connected>
          <neighbor>161.161.161.2</neighbor>
          <peer-as>65003</peer-as>
          <multihop>11</multihop>
        </bgp>
      </routing-protocol>
    </routing-protocols>
  </vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
  <enable-inter-as/>
  <is-receiver-site-only>true</is-receiver-site-only>
</multicast>
</vpn-node>
</vpn-nodes>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-mvpn">
  <ipv4>
    <mvpn-ipmsi-tunnel-ipv4>
      <tunnel-type>sr-mpls-p2mp</tunnel-type>
      <color>20</color>
      <fast-reroute >
        <lfa/>
      </fast-reroute>
    </mvpn-ipmsi-tunnel-ipv4>
    <mvpn-spmsi-tunnels-ipv4>
      <mvpn-spmsi-tunnel-ipv4>
        <tunnel-type>sr-mpls-p2mp</tunnel-type>
        <switch-wildcard-mode>source-group</switch-wildcard-mode>
        <route-policy>treedyn</route-policy>
        <per-item-tunnel-limit>10</per-item-tunnel-limit>
        <group-acl-ipv4>acl_sample_name1</group-acl-ipv4>
      </mvpn-spmsi-tunnel-ipv4>
    </mvpn-spmsi-tunnels-ipv4>
  </ipv4>
</multicast>

```

```

    </vpn-service>
  </vpn-services>
</l3vpn-ntw>
</config>

```

Creating Dynamic MVPN with mdt default, switch wildcard, and threshold

The following is a sample payload to demonstrate the service creation.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-MVPN-DYNAMIC-DEFAULT-V2</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:120:110</rd>
            <address-family>
              <address-family xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:ipv4</address-family>
                <vpn-targets>
                  <vpn-target>
                    <id>1</id>
                    <route-targets>
                      <route-target>0:120:110</route-target>
                    </route-targets>
                    <route-target-type>both</route-target-type>
                  </vpn-target>
                </vpn-targets>
              </address-family>
              <address-family>
                <address-family xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:ipv6</address-family>
                  <vpn-targets>
                    <vpn-target>
                      <id>1</id>
                      <route-targets>
                        <route-target>0:120:110</route-target>
                      </route-targets>
                      <route-target-type>both</route-target-type>
                    </vpn-target>
                  </vpn-targets>
                </address-family>
              </vpn-instance-profile>
            </vpn-instance-profile>
          </vpn-instance-profiles>
        </vpn-service>
      </vpn-services>
    </l3vpn-ntw>
  </config>

```

```

</vpn-instance-profiles>
<vpn-nodes>
  <vpn-node>
    <vpn-node-id>PIOSXR-0</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
      <vpn-instance-profile>
        <profile-id>ie_00</profile-id>
      </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>24</id>
        <interface-id>HundredGigE0/0/0/3</interface-id>
        <ip-connection>
          <ipv4>
            <local-address>160.160.160.1</local-address>
            <prefix-length>24</prefix-length>
          </ipv4>
        </ip-connection>
        <routing-protocols>
          <routing-protocol>
            <id>TEST_PROTO</id>
            <type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-
common">vpn-common:bgp-routing</type>
            <bgp>
              <peer-as>65003</peer-as>
              <address-family>ipv4</address-family>
              <neighbor>160.160.160.2</neighbor>
              <multihop>11</multihop>
              <redistribute-connected>
                <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                <enable>true</enable>
              </redistribute-connected>
            </bgp>
          </routing-protocol>
        </routing-protocols>
      </vpn-network-access>
    </vpn-network-accesses>
    <multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
      <is-receiver-site-only>true</is-receiver-site-only>
    </multicast>
  </vpn-node>

```

```

<vpn-node>
  <vpn-node-id>PIOSXR-1</vpn-node-id>
  <local-as>1</local-as>
  <active-vpn-instance-profiles>
    <vpn-instance-profile>
      <profile-id>ie_00</profile-id>
    </vpn-instance-profile>
  </active-vpn-instance-profiles>
  <vpn-network-accesses>
    <vpn-network-access>
      <id>23</id>
      <interface-id>HundredGigE0/0/0/2</interface-id>
      <ip-connection>
        <ipv4>
          <local-address>161.161.161.1</local-address>
          <prefix-length>24</prefix-length>
        </ipv4>
      </ip-connection>
      <routing-protocols>
        <routing-protocol>
          <id>TEST_PROTO</id>
          <type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-
common">vpn-common:bgp-routing</type>
          <bgp>
            <peer-as>65003</peer-as>
            <address-family>ipv4</address-family>
            <neighbor>161.161.161.2</neighbor>
            <multihop>11</multihop>
            <redistribute-connected>
              <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                <enable>true</enable>
              </redistribute-connected>
            </bgp>
          </routing-protocol>
        </routing-protocols>
      </vpn-network-access>
    </vpn-network-accesses>
    <multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
      <is-receiver-site-only>true</is-receiver-site-only>
      <enable-inter-as/>
    </multicast>
  </vpn-node>
</vpn-nodes>

```

```

<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-mvpn">
  <ipv4>
    <mvpn-ipmsi-tunnel-ipv4>
      <tunnel-type>sr-mpls-p2mp</tunnel-type>
      <color>20</color>
      <fast-reroute>
        <lfa/>
      </fast-reroute>
    </mvpn-ipmsi-tunnel-ipv4>
    <mvpn-spmsi-tunnels-ipv4>
      <mvpn-spmsi-tunnel-ipv4>
        <tunnel-type>sr-mpls-p2mp</tunnel-type>
        <switch-wildcard-mode>source-group</switch-wildcard-mode>
        <color>44</color>
        <fast-reroute>
          <lfa/>
        </fast-reroute>
        <switch-threshold>2</switch-threshold>
        <per-item-tunnel-limit>10</per-item-tunnel-limit>
      </mvpn-spmsi-tunnel-ipv4>
    </mvpn-spmsi-tunnels-ipv4>
  </ipv4>
</multicast>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Creating Dynamic MVPN with mdt partition and switch wildcard

The following is a sample payload to demonstrate the service creation.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-MVPN-DYNAMIC-PARTITION</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:120:112</rd>
            <address-family>
              <address-family>ipv4</address-family>
            <vpn-targets>
              <vpn-target>

```



```

        <id>1</id>
        <route-targets>
            <route-target>0:120:112</route-target>
        </route-targets>
        <route-target-type>both</route-target-type>
    </vpn-target>
</vpn-targets>
</address-family>
<address-family>
    <address-family>ipv6</address-family>
    <vpn-targets>
        <vpn-target>
            <id>1</id>
            <route-targets>
                <route-target>0:120:112</route-target>
            </route-targets>
            <route-target-type>both</route-target-type>
        </vpn-target>
    </vpn-targets>
</address-family>
</vpn-instance-profile>
</vpn-instance-profiles>
<vpn-nodes>
    <vpn-node>
        <vpn-node-id>PIOSXR-0</vpn-node-id>
        <local-as>1</local-as>
        <active-vpn-instance-profiles>
            <vpn-instance-profile>
                <profile-id>ie_00</profile-id>
            </vpn-instance-profile>
        </active-vpn-instance-profiles>
        <vpn-network-accesses>
            <vpn-network-access>
                <id>24</id>
                <interface-id>HundredGigE0/0/0/3</interface-id>
                <ip-connection>
                    <ipv4>
                        <local-address>160.160.160.1</local-address>
                        <prefix-length>24</prefix-length>
                    </ipv4>
                </ip-connection>
                <routing-protocols>
                    <routing-protocol>
                        <id>TEST_PROTO</id>
                    </routing-protocol>
                </routing-protocols>
            </vpn-network-access>
        </vpn-network-accesses>
    </vpn-node>
</vpn-nodes>

```

```

        <type>bgp-routing</type>
        <bgp>
            <address-family>ipv4</address-family>
            <redistribute-connected>
                <address-family>ipv4</address-family>
                <enable>>true</enable>
            </redistribute-connected>
            <neighbor>160.160.160.2</neighbor>
            <peer-as>65003</peer-as>
            <multihop>11</multihop>
        </bgp>
    </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
    <is-receiver-site-only>true</is-receiver-site-only>
</multicast>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>23</id>
            <interface-id>HundredGigE0/0/0/2</interface-id>
            <ip-connection>
                <ipv4>
                    <local-address>160.160.160.1</local-address>
                    <prefix-length>24</prefix-length>
                </ipv4>
            </ip-connection>
            <routing-protocols>
                <routing-protocol>
                    <id>TEST_PROTO</id>
                    <type>bgp-routing</type>
                    <bgp>
                        <address-family>ipv4</address-family>
                        <redistribute-connected>

```

```

        <address-family>ipv4</address-family>
        <enable>true</enable>
    </redistribute-connected>
    <neighbor>161.161.161.2</neighbor>
    <peer-as>65003</peer-as>
    <multihop>11</multihop>
</bgp>
</routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
    <enable-inter-as/>
    <is-receiver-site-only>true</is-receiver-site-only>
</multicast>
</vpn-node>
</vpn-nodes>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-mvpn">
    <ipv4>
        <mvpn-spmsi-tunnels-ipv4>
            <mvpn-spmsi-tunnel-ipv4>
                <tunnel-type>sr-mpls-p2mp</tunnel-type>
                <switch-wildcard-mode>source-group</switch-wildcard-mode>
                <per-item-tunnel-limit>10</per-item-tunnel-limit>
                <immediate-switch>true</immediate-switch>
            </mvpn-spmsi-tunnel-ipv4>
            <mvpn-spmsi-tunnel-ipv4>
                <tunnel-type>sr-mpls-p2mp</tunnel-type>
                <switch-wildcard-mode>star-star</switch-wildcard-mode>
                <color>22</color>
                <fast-reroute>
                    <lfa/>
                </fast-reroute>
                <per-item-tunnel-limit>10</per-item-tunnel-limit>
            </mvpn-spmsi-tunnel-ipv4>
        </mvpn-spmsi-tunnels-ipv4>
    </ipv4>
</multicast>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Creating Dynamic MVPN with mdt default and immediate switch

The following is a sample payload to demonstrate service creation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-MVPN-DYNAMIC-DEFAULT-V3</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:120:110</rd>
            <address-family>
              <address-family xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:ipv4</address-family>
                <vpn-targets>
                  <vpn-target>
                    <id>1</id>
                    <route-targets>
                      <route-target>0:120:110</route-target>
                    </route-targets>
                    <route-target-type>both</route-target-type>
                  </vpn-target>
                </vpn-targets>
              </address-family>
              <address-family>
                <address-family xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:ipv6</address-family>
                  <vpn-targets>
                    <vpn-target>
                      <id>1</id>
                      <route-targets>
                        <route-target>0:120:110</route-target>
                      </route-targets>
                      <route-target-type>both</route-target-type>
                    </vpn-target>
                  </vpn-targets>
                </address-family>
              </vpn-instance-profile>
            </vpn-instance-profiles>
          <vpn-nodes>
            <vpn-node>
              <vpn-node-id>PIOSXR-0</vpn-node-id>
              <local-as>1</local-as>
              <active-vpn-instance-profiles>
```

```

    <vpn-instance-profile>
      <profile-id>ie_00</profile-id>
    </vpn-instance-profile>
  </active-vpn-instance-profiles>
  <vpn-network-accesses>
    <vpn-network-access>
      <id>24</id>
      <interface-id>HundredGigE0/0/0/3</interface-id>
      <ip-connection>
        <ipv4>
          <local-address>160.160.160.1</local-address>
          <prefix-length>24</prefix-length>
        </ipv4>
      </ip-connection>
      <routing-protocols>
        <routing-protocol>
          <id>TEST_PROTO</id>
          <type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-
common">vpn-common:bgp-routing</type>
          <bgp>
            <peer-as>65003</peer-as>
            <address-family>ipv4</address-family>
            <neighbor>160.160.160.2</neighbor>
            <multihop>11</multihop>
            <redistribute-connected>
              <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                <enable>true</enable>
              </redistribute-connected>
            </bgp>
          </routing-protocol>
        </routing-protocols>
      </vpn-network-access>
    </vpn-network-accesses>
    <multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
      <is-receiver-site-only>true</is-receiver-site-only>
    </multicast>
  </vpn-node>
  <vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
      <vpn-instance-profile>
        <profile-id>ie_00</profile-id>

```

```

    </vpn-instance-profile>
  </active-vpn-instance-profiles>
  <vpn-network-accesses>
    <vpn-network-access>
      <id>23</id>
      <interface-id>HundredGigE0/0/0/2</interface-id>
      <ip-connection>
        <ipv4>
          <local-address>161.161.161.1</local-address>
          <prefix-length>24</prefix-length>
        </ipv4>
      </ip-connection>
      <routing-protocols>
        <routing-protocol>
          <id>TEST_PROTO</id>
          <type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-vpn-
common">vpn-common:bgp-routing</type>
          <bgp>
            <peer-as>65003</peer-as>
            <address-family>ipv4</address-family>
            <neighbor>161.161.161.2</neighbor>
            <multihop>11</multihop>
            <redistribute-connected>
              <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                <enable>true</enable>
              </redistribute-connected>
            </bgp>
          </routing-protocol>
        </routing-protocols>
      </vpn-network-access>
    </vpn-network-accesses>
    <multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
      <is-receiver-site-only>true</is-receiver-site-only>
      <enable-inter-as/>
    </multicast>
  </vpn-node>
</vpn-nodes>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-mvpn">
  <ipv4>
    <mvpn-ipmsi-tunnel-ipv4>
      <tunnel-type>sr-mps-p2mp</tunnel-type>
      <color>20</color>
      <fast-reroute>

```

```

        <lfa/>
    </fast-reroute>
</mvpn-ipmsi-tunnel-ipv4>
<mvpn-spmsi-tunnels-ipv4>
    <mvpn-spmsi-tunnel-ipv4>
        <tunnel-type>sr-mpls-p2mp</tunnel-type>
        <switch-wildcard-mode>source-group</switch-wildcard-mode>
        <per-item-tunnel-limit>10</per-item-tunnel-limit>
        <immediate-switch>true</immediate-switch>
    </mvpn-spmsi-tunnel-ipv4>
</mvpn-spmsi-tunnels-ipv4>
</ipv4>
</multicast>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Creating Static MVPN with One Root and One or Multiple Leafs

The following is a sample payload to demonstrate the service creation with one root and one leaf.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:iETF:params:xml:ns:yang:iETF-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-MVPN-STATIC</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:110:124</rd>
            <address-family>
              <address-family>ipv4</address-family>
              <vpn-targets>
                <vpn-target>
                  <id>1</id>
                  <route-targets>
                    <route-target>0:110:124</route-target>
                  </route-targets>
                  <route-target-type>both</route-target-type>
                </vpn-target>
              </vpn-targets>
            </address-family>
          </address-family>
        </vpn-instance-profile>
      </vpn-service>
    </vpn-services>
  </l3vpn-ntw>
</config>

```

```

<address-family>ipv6</address-family>
<vpn-targets>
  <vpn-target>
    <id>1</id>
    <route-targets>
      <route-target>0:110:124</route-target>
    </route-targets>
    <route-target-type>both</route-target-type>
  </vpn-target>
</vpn-targets>
</address-family>
</vpn-instance-profile>
</vpn-instance-profiles>
<vpn-nodes>
  <vpn-node>
    <vpn-node-id>PIOSXR-0</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
      <vpn-instance-profile>
        <profile-id>ie_00</profile-id>
      </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>24</id>
        <interface-id>HundredGigE0/0/0/3</interface-id>
        <ip-connection>
          <ipv4>
            <local-address>160.160.160.1</local-address>
            <prefix-length>24</prefix-length>
          </ipv4>
        </ip-connection>
        <routing-protocols>
          <routing-protocol>
            <id>TEST_PROTO</id>
            <type>bgp-routing</type>
            <bgp>
              <address-family>ipv4</address-family>
              <redistribute-connected>
                <address-family>ipv4</address-family>
                <enable>>true</enable>
              </redistribute-connected>
              <neighbor>160.160.160.2</neighbor>
              <peer-as>65003</peer-as>
            </bgp>
          </routing-protocol>
        </routing-protocols>
      </vpn-network-access>
    </vpn-network-accesses>
  </vpn-node>
</vpn-nodes>

```



```

        <multihop>11</multihop>
    </bgp>
</routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
    <static-sr-mppls-p2mp>
        <policy-name>TREE-SID-STATIC</policy-name>
        <role>receiver</role>
    </static-sr-mppls-p2mp>
</multicast>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>23</id>
            <interface-id>HundredGigE0/0/0/2</interface-id>
            <ip-connection>
                <ipv4>
                    <local-address>160.160.160.1</local-address>
                    <prefix-length>24</prefix-length>
                </ipv4>
            </ip-connection>
            <routing-protocols>
                <routing-protocol>
                    <id>TEST_PROTO</id>
                    <type>bgp-routing</type>
                    <bgp>
                        <address-family>ipv4</address-family>
                        <redistribute-connected>
                            <address-family>ipv4</address-family>
                            <enable>>true</enable>
                        </redistribute-connected>
                        <neighbor>161.161.161.2</neighbor>
                        <peer-as>65003</peer-as>
                    </bgp>
                </routing-protocol>
            </routing-protocols>
            <multihop>11</multihop>
        </vpn-network-access>
    </vpn-network-accesses>
</vpn-node>
</vpn-node>

```

```

        </bgp>
    </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
    <static-sr-mppls-p2mp>
        <policy-name>TREE-SID-STATIC</policy-name>
        <role>source</role>
    </static-sr-mppls-p2mp>
</multicast>
</vpn-node>
</vpn-nodes>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-mvpn">
    <ipv4>
        <mvpn-ipmsi-tunnel-ipv4>
            <tunnel-type>static-sr-mppls-p2mp</tunnel-type>
        </mvpn-ipmsi-tunnel-ipv4>
        <static-sr-mppls-p2mp xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
13vpn-ntw">
            <policy-name>TREE-SID-STATIC</policy-name>
            <sr-p2mp-policy>
                <group-address>
                    <address>12.0.0.1</address>
                    <source-address>
                        <address>12.0.0.2</address>
                        <inc-mask>
                            <address>0.0.0.1</address>
                            <count>1</count>
                        </inc-mask>
                    </source-address>
                    <inc-mask>
                        <address>0.0.0.2</address>
                        <count>2</count>
                    </inc-mask>
                </group-address>
                <group-address>
                    <address>12.0.0.3</address>
                    <source-address>
                        <address>12.0.0.4</address>
                    </source-address>
                    <inc-mask>
                        <address>0.0.0.3</address>
                        <count>3</count>
                    </inc-mask>
                </group-address>
            </sr-p2mp-policy>
        </static-sr-mppls-p2mp>
    </ipv4>
</multicast>

```

```

        </inc-mask>
    </group-address>
</group-address>
    <address>12.0.0.5</address>
    <source-address>
        <address>12.0.0.6</address>
        <inc-mask>
            <address>0.0.0.4</address>
            <count>4</count>
        </inc-mask>
    </source-address>
</group-address>
</group-address>
    <address>12.0.0.6</address>
    <source-address>
        <address>12.0.0.7</address>
    </source-address>
</group-address>
</group-address>
    <address>12.0.0.8</address>
    <inc-mask>
        <address>0.0.0.8</address>
        <count>8</count>
    </inc-mask>
</group-address>
</group-address>
    <address>12.0.0.9</address>
</group-address>
</sr-p2mp-policy>
</static-sr-mps-p2mp>
</ipv4>
</multicast>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Creating RFC Route Policy IETF-L3VPN-NM Service Association

The L3NM route-policy model is standardized by using the RFC routing policy model RFC#9067. The RFC model provides a routing policy framework to use in the L3NM service. For the routing policy yang model, see [Appendix C - Yang Models](#) in this documentation.

Configure the route policy to filter or prioritize the data traffic in your network infrastructure. You must create defined-sets to create the policy definitions. The defined sets used for the routing policy in the TSDN FP Bundle CFP cover a wide range of constructs to manage and use the route policy. You can choose to customize the policy with defined-sets that work on your device and meet your infrastructure.

Note: TSDN FP Bundle CFP supports only a subset of the IETF model of RFC#9067. For more information about the RFC routing policy model RFC#9067, see the RFC#9067 model documentation.

Create the RFC route policy and use the policy in the L3NM service.

The route policy model is available under L3VPN-routing-policy. In the route policy model, you must first create the defined-sets and use these sets to create policy definitions. Policy definitions create named sequences of policy statements.

Note: TSDN FP Bundle CFP supports only a subset of the IETF model of RFC#9067.

Configure the route policy to filter or prioritize the data traffic in your network infrastructure. You must create defined-sets to create the policy definitions. The defined-sets used for the routing policy in the TSDN FP Bundle CFP cover a wide range of constructs to manage and use the route policy. You can choose to customize the policy with defined-sets that work on your device and meet your infrastructure. For more information about the RFC routing policy model RFC#9067, see the RFC#9067 model documentation.

The route policy model is available under L3VPN-routing-policy. In the route policy model, you must first create the defined sets and use these sets to create policy definitions. Policy definitions create named sequences of policy statements.

Creating Defined Sets

Defined-sets contain sets of attributes to match conditions in a statement or to set values for an action in a statement.

To create defined sets:

The defined-set in a route policy in IETF-L3VPN-NM contains prefix-sets. Prefix-sets further have source-prefix-set, dest-prefix-set, and tag-sets. The source-prefix-set and dest-prefix-set are used only in the MVPN policy type. The import and /or export policy types use only the dest-prefix-set in the model.

The prefixes in the RFC route policy are subnets and not destination IPs.

In the l3vpn-routing-policy model, the **prefix-sets** -> **source-prefix-set** contains **mode** to define the delivery method (IPv4 or IPv6), **prefixes**, and **prefix-list** to match the source and destination conditions in a policy. In the MVPN policy type, only the IPv4 mode is supported in the L3NM service. NSO validates to check the ip-prefix type matches the selected mode.

For example, for an IPv4 mode, the `prefix-list` must contain only IPv4 prefixes. The **match-source-prefix-set** under **policy-definition** references the **source-prefix-set** to match the conditions in a policy.

The **dest-prefix-set** in the model also contains mode, prefixes, and prefix-list to define the destination prefix-set. In the MVPN policy type, this mode can only be IPv4 (since the mode in **source-prefix-set** is applicable and is IPv4). In Import and/or Export policy types, the mode can be either IPv4 or IPv6. The **prefix-list** set is referenced by the **match-dest-prefix-set** to match the conditions in a policy definition. NSO validates to check the ip-prefix type matches the selected mode.

The following is the destination from RFC for IPv4 prefixes.

The canonical format of an IPv4 prefix has all bits of the IPv4 address set to zero that are not part of the IPv4 prefix. Example: 192.0.2.0/24.

The following is the destination from RFC for IPv6 prefixes.

The IPv6 address should have all bits that do not belong to the prefix set to zero. The canonical format of an IPv6 prefix has all bits of the IPv6 address set to zero that are not part of the IPv6 prefix. Example: 2001:DB8::/32.

The **tag-sets** contain the list of colors referenced by bgp-actions and multicast-actions in a policy. These **tag-sets** are also used to generate values for the **opaque and set** parameters in the **extcommunity-set**. For example, the tag-set value COLOR_100 in a defined-set sets the **opaque** parameter value to COLOR_100 and the **set** parameter value to 100 in the **extcommunity-set** configuration on a device.

The **tag-value** represents a color value in the range 1-4294967295 configured within a bgp-actions policy statement. A tag-set can have only one tag-value member. For example, a tag-set COLOR_100 indicates the tag-value is 100.

The following is a sample of defined-sets configuration for an import/export policy type. Only the **dest-prefix-set** is configured in this policy type. The **source-prefix-set** and **match-source-prefix-set** are not supported in this use case.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-routing-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
routing-policy">
    <defined-sets>
      <prefix-sets>
        <dest-prefix-set>
          <name>dest_prefix_set_A</name>
          <mode>ipv4</mode>
          <prefixes>
            <prefix-list>
              <ip-prefix>1.1.1.1/32</ip-prefix>
            </prefix-list>
            <prefix-list>
```

```

        <ip-prefix>1.1.1.2/32</ip-prefix>
    </prefix-list>
</prefixes>
</dest-prefix-set>
<dest-prefix-set>
    <name>dest_prefix_set_B</name>
    <mode>ipv6</mode>
    <prefixes>
        <prefix-list>
            <ip-prefix>2001:db8::/32</ip-prefix>
        </prefix-list>
        <prefix-list>
            <ip-prefix>2001:db9::/32</ip-prefix>
        </prefix-list>
    </prefixes>
</dest-prefix-set>
</prefix-sets>
<tag-sets>
    <tag-set>
        <name>COLOR_100</name>
        <tag-value>100</tag-value>
    </tag-set>
    <tag-set>
        <name>COLOR_101</name>
        <tag-value>101</tag-value>
    </tag-set>
    <tag-set>
        <name>COLOR_102</name>
        <tag-value>102</tag-value>
    </tag-set>
</tag-sets>
</defined-sets>
</l3vpn-routing-policy>
</config>

```

Creating a Route Policy

A route policy is a policy definition. Route policies are created and attached to an IETF-L3VPN-NM service to prioritize the network traffic.

Policy definitions create named sequences of policy statements. A policy can be applied to other policies. A single policy definition maps to a single route policy. The policy definition name must be the name of the route policy to push to the device. The policy definition name must be alphanumeric and optionally contain the special characters period, hyphen, and an underscore. The policy name must, however, begin with either an alphabet or a number.

Each policy definition contains a list of statements. A statement name identifies a statement list and, hence, must be unique. Each statement is evaluated in the order it was created in the route policy payload.

A statement can optionally contain conditions to match the prefix-sets in a route policy. A statement consisting of conditions must have actions to support the conditions. In T-SDN FP Bundle CFP, you can create separate statements for IPv4 and IPv6 prefix-sets. You cannot, however, match mixed prefix-sets. Each match condition results in an If-then clause in the route-policy.

Policies that have conditions must also have both match-source-prefix-set and match-dest-prefix-set configured. The match-source-prefix-set is used only if the mode in the source-prefix-set is IPv4, as in the case of MVPN policy.

Note: Do not use the hidden **match-manual** field under policy definitions.

When a statement contains conditions, you must configure the bgp-actions or multicast-actions. It is mandatory to configure bgp-actions, multicast-actions, or apply-policy action in a policy statement. The policy-result action, which configures the route policy value as **done**, must always be combined with one of these actions in the last statement in a statement list.

Note: Additional statements added after the last statement in the statement list invalidate the policy configuration even though NSO or the device accepts the configuration.

If the route-policy contains the If-Then clause, the actions appear after the **then** clause. You can also apply the actions to a route-policy without an If-Then clause.

The bgp-actions/set-ext-community/ext-community-set-ref action is used to configure the set-extcommunity-color property in the route policy.

Ext-community-set-ref is mandatory in a policy, and it references tag-sets under defined-sets for color values.

The bgp-actions can be combined with the policy-result action and may or may not have conditions associated with it. The bgp-actions cannot be used in MVPN route policy or combined with multicast-actions.

The multicast-actions can be used only in the MVPN route policy. The action cannot be used in a policy that is used by import-export leafs in the L3 model, or cannot be combined with the bgp-actions. Multicast-actions configures the:

- (Optional) Sea-fast-reroute action to set the sea-fast-reroute parameter in the route policy. Use the set-lfa action to optionally set fast-reroute lfa in the route policy.
- (Optional) set-data-mdt to set the set-data-mdt parameter with a string value in the route policy. In Netconf, the action is supported only on cisco-iosxr-nc-7.5.2 or later.

- Set-on-demand-color action is a mandatory action in multicast-actions used to set the set-on-demand-color value in the route policy. This action references a tag-set name, but the actual color value is configured in the route policy.

The apply-policy action is used to apply one policy to another policy. The apply-policy action must be created as a separate statement and cannot be combined with other actions in the statement. It must be the first or last statement in the statement list. It prepends or appends the extra policy to the route-policy. A policy-statement cannot contain conditions with only the apply-policy action.

Depending on the route-policy you want to configure, you can use only a subset of actions to create a valid route-policy.

The following is a sample route-policy payload.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-routing-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
routing-policy">
    <policy-definitions>
      <policy-definition>
        <name>SAMPLE_1</name>
        <statements>
          <statement>
            <name>stmnt1</name>
            <actions>
              <apply-policy>
                <extra-policy>l3-ep1</extra-policy>
              </apply-policy>
            </actions>
          </statement>
          <statement>
            <name>stmnt2</name>
            <actions>
              <bgp-actions>
                <set-ext-community>
                  <ext-community-set-ref>COLOR_100</ext-community-set-ref>
                </set-ext-community>
              </bgp-actions>
            </actions>
          </statement>
          <statement>
            <name>stmnt3</name>
            <actions>
              <policy-result>accept-route</policy-result>
              <apply-policy>
                <extra-policy>l3-ep2</extra-policy>
              </apply-policy>
            </actions>
          </statement>
        </statements>
      </policy-definition>
    </policy-definitions>
  </l3vpn-routing-policy>
</config>
```



```

        </apply-policy>
      </actions>
    </statement>
  </statements>
</policy-definition>
</policy-definitions>
</l3vpn-routing-policy>
</config>

```

Creating Routing-Profile-Identifiers

Routing-profile-identifier are created automatically under l3vpn-ntw/vpn-profiles/valid-provider-identifiers/routing-profile-identifier to use the policy definitions in an L3NM service.

The following example shows how to manually create the routing-profile-identifiers.

See the below payload for the manual creation of routing profile identifiers.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-profiles>
      <valid-provider-identifiers>
        <routing-profile-identifier>
          <id>SAMPLE_1</id>
        </routing-profile-identifier>
      </valid-provider-identifiers>
    </vpn-profiles>
  </l3vpn-ntw>
</config>

```

It is not required to commit the l3vpn-route-policy separately before committing a l3vpn-ntw-service.

Associating a Route Policy with IETF-L3VPN-NM Service

The following sample payloads show how to associate route policy with the L3NM service and MVPN L3NM service.

To associate the route policy with IETF-L3VPN-NM service:

The following is a sample payload to create the IETF-L3VPN-NM service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>0-65008740</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
          </vpn-instance-profile>
        </vpn-instance-profiles>
      </vpn-service>
    </vpn-services>
  </l3vpn-ntw>
</config>

```

```

    <rd>0:65100:87400024</rd>
    <address-family>
      <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>

      <vpn-targets>
        <vpn-target>
          <id>1</id>
          <route-targets>
            <route-target>0:65010:17401</route-target>
          </route-targets>
          <route-targets>
            <route-target>0:65010:17402</route-target>
          </route-targets>
          <route-target-type>both</route-target-type>
        </vpn-target>
        <vpn-target>
          <id>2</id>
          <route-targets>
            <route-target>0:65010:17403</route-target>
          </route-targets>
          <route-target-type>import</route-target-type>
        </vpn-target>
        <vpn-target>
          <id>3</id>
          <route-targets>
            <route-target>0:65010:17404</route-target>
          </route-targets>
          <route-target-type>export</route-target-type>
        </vpn-target>
      <vpn-policies>
        <import-policy>SAMPLE_1</import-policy>
        <export-policy>SAMPLE_1</export-policy>
      </vpn-policies>
    </vpn-targets>
  </address-family>
  <address-family>
    <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv6</address-
family>

    <vpn-targets>
      <vpn-target>
        <id>1</id>
        <route-targets>
          <route-target>0:65010:17401</route-target>
        </route-targets>
      </vpn-target>
    </vpn-targets>
  </address-family>

```

```

        </route-targets>
    </route-targets>
        <route-target>0:65010:17402</route-target>
    </route-targets>
    <route-target-type>both</route-target-type>
</vpn-target>
<vpn-target>
    <id>2</id>
    <route-targets>
        <route-target>0:65010:17403</route-target>
    </route-targets>
    <route-target-type>import</route-target-type>
</vpn-target>
<vpn-target>
    <id>3</id>
    <route-targets>
        <route-target>0:65010:17404</route-target>
    </route-targets>
    <route-target-type>export</route-target-type>
</vpn-target>
<vpn-policies>
    <import-policy>SAMPLE_1</import-policy>
    <export-policy>SAMPLE_1</export-policy>
</vpn-policies>
</vpn-targets>
</address-family>
</vpn-instance-profile>
<vpn-instance-profile>
    <profile-id>ie_01</profile-id>
</vpn-instance-profile>
</vpn-instance-profiles>
<vpn-nodes>
    <vpn-node>
        <vpn-node-id>PIOSXR-0</vpn-node-id>
        <local-as>65001</local-as>
        <active-vpn-instance-profiles>
            <vpn-instance-profile>
                <profile-id>ie_00</profile-id>
                <address-family>
                    <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv6</address-
family>

                <vpn-targets>
                    <vpn-target>
                        <id>1</id>

```

```

        <route-targets>
            <route-target>0:65010:17405</route-target>
        </route-targets>
        <route-targets>
            <route-target>0:65010:17406</route-target>
        </route-targets>
        <route-target-type>both</route-target-type>
    </vpn-target>
    <vpn-policies>
        <import-policy>SAMPLE_1</import-policy>
        <export-policy>SAMPLE_1</export-policy>
    </vpn-policies>
</vpn-targets>
</address-family>
<srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
13vpn-ntw">
    <address-family>
        <name xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv6</name>
        <locator-name>locv6</locator-name>
    </address-family>
</srv6>
</vpn-instance-profile>
</active-vpn-instance-profiles>
<vpn-network-accesses>
    <vpn-network-access>
        <id>25</id>
        <interface-id>GigabitEthernet1/1/1/1</interface-id>
        <connection>
            <encapsulation>
                <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:dot1q</type>
                <dot1q>
                    <cvlan-id>1234</cvlan-id>
                </dot1q>
            </encapsulation>
        </connection>
        <ip-connection>
            <ipv6>
                <local-address>2001:db8::1</local-address>
                <prefix-length>32</prefix-length>
            </ipv6>
        </ip-connection>
    </vpn-network-access>
</vpn-network-accesses>
<routing-protocols>
    <routing-protocol>

```

```

        <id>TEST_PROTO</id>
        <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:bgp-routing</type>
        <bgp>
            <peer-as>65003</peer-as>
            <address-family>dual-stack</address-family>
            <neighbor>2001:db8::2</neighbor>
            <multihop>12</multihop>
            <update-source
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
                <if-type>GigabitEthernet</if-type>
                <if-id>3</if-id>
                <sub-if-id>200</sub-if-id>
            </update-source>
            <mpls-deactivation
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">true</mpls-
deactivation>
        </bgp>
    </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <local-as>65001</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_01</profile-id>
            <address-family>
                <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                    <vpn-targets>
                        <vpn-policies>
                            <import-policy>SAMPLE_1</import-policy>
                            <export-policy>SAMPLE_1</export-policy>
                        </vpn-policies>
                    </vpn-targets>
                </address-family>
                <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l3vpn-ntw">
                    <address-family>
                        <name xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</name>
                        <locator-name>locv4</locator-name>
                    </address-family>
                </srv6>
            </address-family>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
</vpn-node>
</vpn-network-accesses>
</vpn-network-accesses>
</vpn-network-accesses>

```

```

        </address-family>
        <address-family>
            <name xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv6</name>
            <locator-name>locv6</locator-name>
        </address-family>
    </srv6>
</vpn-instance-profile>
</active-vpn-instance-profiles>
<vpn-network-accesses>
    <vpn-network-access>
        <id>23</id>
        <interface-id>GigabitEthernet1/1/1/1</interface-id>
        <connection>
            <encapsulation>
                <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:dot1q</type>
                <dot1q>
                    <cvlan-id>1234</cvlan-id>
                </dot1q>
            </encapsulation>
        </connection>
        <ip-connection>
            <ipv4>
                <local-address>10.1.1.1</local-address>
                <prefix-length>24</prefix-length>
            </ipv4>
            <ipv6>
                <local-address>2002:4601:102::4601:102</local-
address>
                <prefix-length>24</prefix-length>
            </ipv6>
        </ip-connection>
        <routing-protocols>
            <routing-protocol>
                <id>TEST_PROTO</id>
                <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:bgp-routing</type>
                <bgp>
                    <peer-as>65003</peer-as>
                    <address-family>dual-stack</address-family>
                    <neighbor>10.1.1.1</neighbor>
                    <neighbor>2002:4601:102::4601:103</neighbor>
                    <redistribute-connected>

```

```

        <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
            <enable>true</enable>
            <metric
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">123</metric>
            </redistribute-connected>
            <redistribute-connected>
                <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv6</address-
family>
                    <enable>true</enable>
                    </redistribute-connected>
                </bgp>
            </routing-protocol>
        </routing-protocols>
    </vpn-network-access>
    <vpn-network-access>
        <id>24</id>
        <interface-id>GigabitEthernet1/1/1/1</interface-id>
        <connection>
            <encapsulation>
                <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:dot1q</type>
                <dot1q>
                    <cvlan-id>1231</cvlan-id>
                </dot1q>
            </encapsulation>
        </connection>
        <ip-connection>
            <ipv4>
                <local-address>10.1.1.2</local-address>
                <prefix-length>24</prefix-length>
            </ipv4>
            <ipv6>
                <local-address>2002:4601:102::4601:103</local-
address>
                <prefix-length>24</prefix-length>
            </ipv6>
        </ip-connection>
    </routing-protocols>
    <routing-protocol>
        <id>TEST_PROTO_2</id>
        <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:bgp-routing</type>
        <bgp>

```

```

        <peer-as>65005</peer-as>
        <address-family>dual-stack</address-family>
        <neighbor>10.1.1.3</neighbor>
        <neighbor>2002:4601:102::4601:104</neighbor>
        <multihop>11</multihop>
    </bgp>
</routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
</vpn-nodes>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Display the plan to view the status of the service.

The following table describes the IETF-L3VPN-NM plan components.

Plan Component	Description
self	When it constructs its plan, a service includes a component of type 'self'; the upper layer can use this component to determine the service's state.

Self Component State	Description
init	When it constructs its plan, a service includes a component of type 'self'; the upper layer can use this component to determine the service's state.
ietf-l3vpn-ntw-nano-services:config-apply	Represents acceptance of service intent and respective config application on the device in NSO CDB.
ready	The final state in a 'state list' in the plan indicating intent on this device was completed.

Check the service's vpn-service-plan to confirm the overall service provisioning status.

```
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 plan
```

```

          BACK
          ACTION
TYPE      NAME      TRACK  GOAL    STATE  STATUS  WHEN  POST ACTION
-----

```



```

Self          self          false          init          reached          2024-05-29T23:43:34 - -
Ready         reached         2024-05-29T23:43:36 - -

[ok][2024-05-31 00:57:55]
[edit]

```

Check the `vpn-node-plan` to confirm the provisioning status for a specific `vpn-node`.

```
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 vpn-nodes
vpn-node-plan PIOSXR-0 plan
```

TYPE	BACK NAME	STATUS TRACK	GOAL	CODE	STATE	STATUS	WHEN	POST ACTION	
								ref	STATUS
self	self	false			init	reached	2024-05-29T23:43:34	-	-
ietf-l3vpn-ntw-nano-services:config-apply					reached	reached	2024-05-29T23:43:34	-	-
					ready	ready	reached	2024-05-29T23:43:36	- -

```

[ok][2024-05-31 00:58:43]
[edit]

```

To associate the route policy with IETF-L3VPN-NM MVPN service:

The following is a sample payload to create the IETF-L3VPN-NM MVPN service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-MVPN-DYNAMIC-DEFAULT</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:120:110</rd>
            <address-family>
              <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                <vpn-targets>
                  <vpn-target>
                    <id>1</id>
                    <route-targets>
                      <route-target>0:120:110</route-target>
                    </route-targets>
                    <route-target-type>both</route-target-type>
                  </vpn-target>
                </vpn-targets>
              </address-family>
            <address-family>
              <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv6</address-
family>
                <vpn-targets>

```

```

        <vpn-target>
          <id>1</id>
          <route-targets>
            <route-target>0:120:110</route-target>
          </route-targets>
          <route-target-type>both</route-target-type>
        </vpn-target>
      </vpn-targets>
    </address-family>
  </vpn-instance-profile>
</vpn-instance-profiles>
<vpn-nodes>
  <vpn-node>
    <vpn-node-id>PIOSXR-0</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
      <vpn-instance-profile>
        <profile-id>ie_00</profile-id>
      </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>24</id>
        <interface-id>HundredGigE0/0/0/3</interface-id>
        <ip-connection>
          <ipv4>
            <local-address>160.160.160.1</local-address>
            <prefix-length>24</prefix-length>
          </ipv4>
        </ip-connection>
        <routing-protocols>
          <routing-protocol>
            <id>TEST_PROTO</id>
            <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:bgp-routing</type>
          <bgp>
            <peer-as>65003</peer-as>
            <address-family>ipv4</address-family>
            <neighbor>160.160.160.2</neighbor>
            <multihop>11</multihop>
            <redistribute-connected>
              <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
            <enable>true</enable>

```

```

        </redistribute-connected>
    </bgp>
</routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l3vpn-ntw">
    <is-receiver-site-only>false</is-receiver-site-only>
</multicast>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <local-as>1</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
    <vpn-network-accesses>
        <vpn-network-access>
            <id>23</id>
            <interface-id>HundredGigE0/0/0/2</interface-id>
            <ip-connection>
                <ipv4>
                    <local-address>161.161.161.1</local-address>
                    <prefix-length>24</prefix-length>
                </ipv4>
            </ip-connection>
            <routing-protocols>
                <routing-protocol>
                    <id>TEST_PROTO</id>
                    <type xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:bgp-routing</type>
                <bgp>
                    <peer-as>65003</peer-as>
                    <address-family>ipv4</address-family>
                    <neighbor>161.161.161.2</neighbor>
                    <multihop>11</multihop>
                    <redistribute-connected>
                        <address-family xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:ipv4</address-
family>
                            <enable>true</enable>
                    </redistribute-connected>

```

```

        </bgp>
    </routing-protocol>
</routing-protocols>
    </vpn-network-access>
</vpn-network-accesses>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l3vpn-ntw">
    <is-receiver-site-only>true</is-receiver-site-only>
    <enable-inter-as />
</multicast>
</vpn-node>
</vpn-nodes>
<multicast xmlns="http://cisco.com/ns/nso/fp/examples/cisco-mvpn">
    <ipv4>
        <mvpn-ipmsi-tunnel-ipv4>
            <tunnel-type>sr-mpls-p2mp</tunnel-type>
            <color>20</color>
            <fast-reroute>
                <lfa />
            </fast-reroute>
        </mvpn-ipmsi-tunnel-ipv4>
        <mvpn-spmsi-tunnels-ipv4>
            <mvpn-spmsi-tunnel-ipv4>
                <tunnel-type>sr-mpls-p2mp</tunnel-type>
                <switch-wildcard-mode>source-group</switch-wildcard-mode>
                <per-item-tunnel-limit>10</per-item-tunnel-limit>
                <group-acl-ipv4>acl_sample_name1</group-acl-ipv4>
                <route-policy>SAMPLE_1</route-policy>
            </mvpn-spmsi-tunnel-ipv4>
        </mvpn-spmsi-tunnels-ipv4>
    </ipv4>
</multicast>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Monitoring Network Performance using Accedian Probes

Accedian probes help to discover critical focal points for improved network visibility and manage the network performance for the complete lifecycle of the service. It brings together statistical data and performance data from other sources to learn about the network performance.

In TSDN FP Bundle CFP, the L3NM vpn-service contains a probes container that maintains two entities:

- A list of endpoints that reference the vpn node, vpn-network access, and the IP address and vLAN information of the agent.
- The list of available connection-types – point-to-point connection, hub-and-spoke connection, and connection-mesh.

You can define multiple endpoints and connection-types defined in a single probes container. The vpn-nodes and the endpoints are obtained from the associated L3VPN-NM service.

The following example shows a probes container used in an L3NM service to monitor specific nodes and endpoints.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>L3NM-PROBES</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:65100:87400024</rd>
            <address-family>
              <address-family>ipv4</address-family>
            <vpn-targets>
              <vpn-target>
                <id>1</id>
                <route-targets>
                  <route-target>0:65010:17401</route-target>
                </route-targets>
                <route-targets>
                  <route-target>0:65010:17402</route-target>
                </route-targets>
                <route-target-type>both</route-target-type>
              </vpn-target>
              <vpn-target>
                <id>2</id>
                <route-targets>
                  <route-target>0:65010:17403</route-target>
                </route-targets>
                <route-target-type>import</route-target-type>
              </vpn-target>
              <vpn-target>
                <id>3</id>
```

```
        <route-targets>
          <route-target>0:65010:17404</route-target>
        </route-targets>
        <route-target-type>export</route-target-type>
      </vpn-target>
    </vpn-targets>
  </address-family>
</address-family>
<address-family>ipv6</address-family>
<vpn-targets>
  <vpn-target>
    <id>1</id>
    <route-targets>
      <route-target>0:65010:17401</route-target>
    </route-targets>
    <route-targets>
      <route-target>0:65010:17402</route-target>
    </route-targets>
    <route-target-type>both</route-target-type>
  </vpn-target>
<vpn-target>
  <id>2</id>
  <route-targets>
    <route-target>0:65010:17403</route-target>
  </route-targets>
  <route-target-type>import</route-target-type>
</vpn-target>
<vpn-target>
  <id>3</id>
  <route-targets>
    <route-target>0:65010:17404</route-target>
  </route-targets>
  <route-target-type>export</route-target-type>
</vpn-target>
</vpn-targets>
</address-family>
</vpn-instance-profile>
<vpn-instance-profile>
  <profile-id>ie_01</profile-id>
</vpn-instance-profile>
</vpn-instance-profiles>
<vpn-nodes>
  <vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
```

```

<local-as>65001</local-as>
<active-vpn-instance-profiles>
  <vpn-instance-profile>
    <profile-id>ie_01</profile-id>
    <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
      <address-family>
        <name>ipv4</name>
        <locator-name>locv4</locator-name>
      </address-family>
    </srv6>
    <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
      <address-family>
        <name>ipv6</name>
        <locator-name>locv6</locator-name>
      </address-family>
    </srv6>
  </vpn-instance-profile>
</active-vpn-instance-profiles>
<vpn-network-accesses>
  <vpn-network-access>
    <id>23</id>
    <interface-id>GigabitEthernet1/1/1/1</interface-id>
    <connection>
      <encapsulation>
        <type>dot1q</type>
        <dot1q>
          <cvlan-id>1234</cvlan-id>
        </dot1q>
      </encapsulation>
    </connection>
    <ip-connection>
      <ipv4>
        <local-address>10.1.1.1</local-address>
        <prefix-length>24</prefix-length>
      </ipv4>
      <ipv6>
        <local-address>2002:4601:102::4601:102</local-address>
        <prefix-length>24</prefix-length>
      </ipv6>
    </ip-connection>
    <routing-protocols>
      <routing-protocol>
        <id>TEST_PROTO</id>
        <type>bgp-routing</type>
      </routing-protocol>
    </routing-protocols>
  </vpn-network-access>
</vpn-network-accesses>

```

```

    <bgp>
      <address-family>dual-stack</address-family>
      <redistribute-connected>
        <address-family>ipv4</address-family>
        <metric xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
13vpn-ntw">123</metric>
        <enable>>true</enable>
      </redistribute-connected>
      <redistribute-connected>
        <address-family>ipv6</address-family>
        <enable>>true</enable>
      </redistribute-connected>
      <neighbor>10.1.1.1</neighbor>
      <neighbor>2002:4601:102::4601:103</neighbor>
      <peer-as>65003</peer-as>
    </bgp>
  </routing-protocol>
</routing-protocols>
</vpn-network-access>
<vpn-network-access>
  <id>24</id>
  <interface-id>GigabitEthernet1/1/1/1</interface-id>
  <connection>
    <encapsulation>
      <type>dot1q</type>
      <dot1q>
        <cvlan-id>1231</cvlan-id>
      </dot1q>
    </encapsulation>
  </connection>
  <ip-connection>
    <ipv4>
      <local-address>10.1.1.2</local-address>
      <prefix-length>24</prefix-length>
    </ipv4>
    <ipv6>
      <local-address>2002:4601:102::4601:103</local-address>
      <prefix-length>24</prefix-length>
    </ipv6>
  </ip-connection>
</routing-protocols>
  <routing-protocol>
    <id>TEST_PROTO_2</id>
    <type>bgp-routing</type>

```



```

        <bgp>
            <address-family>dual-stack</address-family>
            <neighbor>10.1.1.3</neighbor>
            <neighbor>2002:4601:102::4601:104</neighbor>
            <peer-as>65005</peer-as>
            <multihop>11</multihop>
        </bgp>
    </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-0</vpn-node-id>
    <local-as>65001</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <address-family>
                <address-family>ipv6</address-family>
                <vpn-targets>
                    <vpn-target>
                        <id>1</id>
                        <route-targets>
                            <route-target>0:65010:17405</route-target>
                        </route-targets>
                        <route-targets>
                            <route-target>0:65010:17406</route-target>
                        </route-targets>
                        <route-target-type>both</route-target-type>
                    </vpn-target>
                </vpn-targets>
            </address-family>
            <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
                <address-family>
                    <name>ipv6</name>
                    <locator-name>locv6</locator-name>
                </address-family>
            </srv6>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
</vpn-network-accesses>
    <vpn-network-access>
        <id>25</id>

```

```

<interface-id>GigabitEthernet1/1/1/1</interface-id>
<connection>
  <encapsulation>
    <type>dot1q</type>
    <dot1q>
      <cvlan-id>1234</cvlan-id>
    </dot1q>
  </encapsulation>
</connection>
<ip-connection>
  <ipv6>
    <local-address>2001:db8::1</local-address>
    <prefix-length>32</prefix-length>
  </ipv6>
</ip-connection>
<routing-protocols>
  <routing-protocol>
    <id>TEST_PROTO</id>
    <type>bgp-routing</type>
    <bgp>
      <address-family>ipv6</address-family>
      <neighbor>2001:db8::2</neighbor>
      <peer-as>65003</peer-as>
      <multihop>12</multihop>
      <mpls-deactivation
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">true</mpls-
deactivation>
      <update-source
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
        <if-type>GigabitEthernet</if-type>
        <if-id>3</if-id>
        <sub-if-id>200</sub-if-id>
      </update-source>
    </bgp>
  </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
</vpn-nodes>
<probes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw">
  <endpoint>
    <id>abc</id>
    <vpn-node>PIOSXR-1</vpn-node>
    <vpn-network-access>23</vpn-network-access>

```

```

<interface-id>GigabitEthernet1/1/1/1</interface-id>
  <agent-ipv4>10.1.1.1</agent-ipv4>
  <agent-vlan>v1</agent-vlan>
</endpoint>
<endpoint>
  <id>xyz</id>
  <vpn-node>PIOSXR-0</vpn-node>
  <vpn-network-access>25</vpn-network-access>
<interface-id>GigabitEthernet1/1/1/1</interface-id>
  <agent-ipv6>2001:db8::1</agent-ipv6>
  <agent-vlan>v0</agent-vlan>
</endpoint>
<point-to-point>
  <connection>
    <source>xyz</source>
    <destination>abc</destination>
  </connection>
</point-to-point>
</probes>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Creating IETF-L3VPN-NM Service with Direct Routing

Direct-routing routing-protocol will push bgp → vrf → redistribute-connected for defined ip-connection address-family on device vrf. Direct-routing is used for scenarios where user needs to broadcast local routes without peering protocol.

Do the following:

1. The following is a sample payload to create the IETF-L3VPN-NM service with Direct Routing.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>0-65008740</vpn-id>
        <vpn-instance-profiles>
          <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:65100:87400024</rd>
            <address-family>
              <address-family>ipv4</address-family>
              <vpn-targets>
                <vpn-target>
                  <id>1</id>
                  <route-targets>
                    <route-target>0:65010:17401</route-target>
                  </route-targets>
                </route-targets>
              </route-targets>
            </address-family>
          </vpn-instance-profile>
        </vpn-instance-profiles>
      </vpn-service>
    </vpn-services>
  </l3vpn-ntw>
</config>

```

```

        <route-target>0:65010:17402</route-target>
    </route-targets>
    <route-target-type>both</route-target-type>
</vpn-target>
<vpn-target>
    <id>2</id>
    <route-targets>
        <route-target>0:65010:17403</route-target>
    </route-targets>
    <route-target-type>import</route-target-type>
</vpn-target>
<vpn-target>
    <id>3</id>
    <route-targets>
        <route-target>0:65010:17404</route-target>
    </route-targets>
    <route-target-type>export</route-target-type>
</vpn-target>
<vpn-policies>
    <import-policy>SET_COLORv4_TEST_IMPORT_POLICY</import-policy>
    <export-policy>SET_COLORv4_TEST_EXPORT_POLICY</export-policy>
</vpn-policies>
</vpn-targets>
</address-family>
<address-family>
    <address-family>ipv6</address-family>
    <vpn-targets>
        <vpn-target>
            <id>1</id>
            <route-targets>
                <route-target>0:65010:17401</route-target>
            </route-targets>
            <route-targets>
                <route-target>0:65010:17402</route-target>
            </route-targets>
            <route-target-type>both</route-target-type>
        </vpn-target>
        <vpn-target>
            <id>2</id>
            <route-targets>
                <route-target>0:65010:17403</route-target>
            </route-targets>
            <route-target-type>import</route-target-type>
        </vpn-target>
        <vpn-target>
            <id>3</id>
            <route-targets>
                <route-target>0:65010:17404</route-target>
            </route-targets>
            <route-target-type>export</route-target-type>
        </vpn-target>
        <vpn-policies>
            <import-policy>SET_COLORv6_TEST_IMPORT_POLICY</import-policy>
            <export-policy>SET_COLORv6_TEST_EXPORT_POLICY</export-policy>
        </vpn-policies>
    </vpn-targets>
</address-family>
</vpn-instance-profile>
<vpn-instance-profile>
    <profile-id>ie_01</profile-id>
</vpn-instance-profile>

```

```

</vpn-instance-profiles>
<vpn-nodes>
  <vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <local-as>65001</local-as>
    <active-vpn-instance-profiles>
      <vpn-instance-profile>
        <profile-id>ie_00</profile-id>
        <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
ntw">
          <address-family>
            <name>ipv4</name>
            <locator-name>locv4</locator-name>
          </address-family>
        </srv6>
        <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
ntw">
          <address-family>
            <name>ipv6</name>
            <locator-name>locv6</locator-name>
          </address-family>
        </srv6>
      </vpn-instance-profile>
    </active-vpn-instance-profiles>
  <vpn-network-accesses>
    <vpn-network-access>
      <id>23</id>
      <interface-id>GigabitEthernet1/1/1/1</interface-id>
      <connection>
        <encapsulation>
          <type>dot1q</type>
          <dot1q>
            <cvlan-id>1234</cvlan-id>
          </dot1q>
        </encapsulation>
      </connection>
      <ip-connection>
        <ipv4>
          <local-address>10.1.1.1</local-address>
          <prefix-length>24</prefix-length>
        </ipv4>
        <ipv6>
          <local-address>2002:4601:102::4601:102</local-address>
          <prefix-length>24</prefix-length>
        </ipv6>
      </ip-connection>
      <routing-protocols>
        <routing-protocol>
          <id>TEST_PROTO</id>
          <type>direct-routing</type>
        </routing-protocol>
      </routing-protocols>
    </vpn-network-access>
    <vpn-network-access>
      <id>24</id>
      <interface-id>GigabitEthernet1/1/1/1</interface-id>
      <connection>
        <encapsulation>
          <type>dot1q</type>
          <dot1q>
            <cvlan-id>1231</cvlan-id>

```

```

        </dot1q>
    </encapsulation>
</connection>
<ip-connection>
    <ipv4>
        <local-address>10.1.1.2</local-address>
        <prefix-length>24</prefix-length>
    </ipv4>
    <ipv6>
        <local-address>2002:4601:102::4601:103</local-address>
        <prefix-length>24</prefix-length>
    </ipv6>
</ip-connection>
<routing-protocols>
    <routing-protocol>
        <id>TEST_PROTO_2</id>
        <type>direct-routing</type>
    </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>PIOSXR-0</vpn-node-id>
    <local-as>65001</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <address-family>
                <address-family>ipv6</address-family>
                <vpn-targets>
                    <vpn-target>
                        <id>1</id>
                        <route-targets>
                            <route-target>0:65010:17405</route-target>
                        </route-targets>
                        <route-targets>
                            <route-target>0:65010:17406</route-target>
                        </route-targets>
                        <route-target-type>both</route-target-type>
                    </vpn-target>
                </vpn-targets>
            </address-family>
            <srv6 xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
ntw">
                <address-family>
                    <name>ipv6</name>
                    <locator-name>locv6</locator-name>
                </address-family>
            </srv6>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
</vpn-network-accesses>
<vpn-network-access>
    <id>25</id>
    <interface-id>GigabitEthernet1/1/1/1</interface-id>
    <connection>
        <encapsulation>
            <type>dot1q</type>
            <dot1q>
                <cvlan-id>1234</cvlan-id>

```

```

        </dot1q>
    </encapsulation>
</connection>
<ip-connection>
    <ipv6>
        <local-address>2001:db8::1</local-address>
        <prefix-length>32</prefix-length>
    </ipv6>
</ip-connection>
<routing-protocols>
    <routing-protocol>
        <id>TEST_PROTO</id>
        <type>direct-routing</type>
    </routing-protocol>
</routing-protocols>
</vpn-network-access>
</vpn-network-accesses>
</vpn-node>
<vpn-node>
    <vpn-node-id>XECLI-0</vpn-node-id>
    <local-as>65001</local-as>
    <active-vpn-instance-profiles>
        <vpn-instance-profile>
            <profile-id>ie_00</profile-id>
            <rd>0:65100:87400013</rd>
            <address-family>
                <address-family>ipv6</address-family>
                <vpn-targets>
                    <vpn-target>
                        <id>1</id>
                        <route-targets>
                            <route-target>0:65010:17405</route-target>
                        </route-targets>
                        <route-target-type>both</route-target-type>
                    </vpn-target>
                </vpn-targets>
            </address-family>
        </vpn-instance-profile>
    </active-vpn-instance-profiles>
</vpn-network-accesses>
<vpn-network-access>
    <id>25</id>
    <interface-id>GigabitEthernet1/1/1/1</interface-id>
    <connection>
        <encapsulation>
            <type>dot1q</type>
            <dot1q>
                <cvlan-id>1234</cvlan-id>
                <BDI xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
13vpn-ntw">321</BDI>
            </dot1q>
        </encapsulation>
    </connection>
    <ip-connection>
        <ipv6>
            <local-address>2001:db8::1</local-address>
            <prefix-length>32</prefix-length>
        </ipv6>
    </ip-connection>
    <routing-protocols>
        <routing-protocol>

```

```

        <id>TEST_PROTO</id>
        <type>direct-routing</type>
      </routing-protocol>
    </routing-protocols>
  </vpn-network-access>
</vpn-network-accesses>
</vpn-node>
</vpn-nodes>
</vpn-service>
</vpn-services>
</l3vpn-ntw>
</config>

```

Working with Action Commands – CFP Services

Use the action commands to perform specific tasks on services, such as service cleanup, to get the latest service modifications, and to redeploy services.

Service Cleanup Actions

When you try to perform a regular service deletion, it may fail for several factors, such as the device being unreachable. This leaves the internal CDB with stale service entries, and the plan may display failures. When the regular service deletion fails to delete the service elements, use the Cleanup action to clean up all the data created by the service.

Note: You must use the Cleanup action only if the regular service deletion fails and no other recovery mechanisms are available. If service deletion has not been done, the cleanup action will fail with a message to delete the service configuration.

The following table discusses the command arguments for the different services.

Arguments	Description
service-type	This is the type of the service. This parameter is available only for SR-TE ODN service.
service	This is the name of the service to clean.
device	This is the name of the device to clean under a given service. This parameter is optional. If no device name is provided or if the option is unavailable, the entire service is cleaned up.
no-networking	This flag indicates if NSO must delete the service components from network devices. By default, it is set to true. When true, NSO cleanup does not remove configuration from network devices. When false, NSO cleans up the device configuration.

Arguments	Description
vpn-node	Name of the vpn-node to clean up. If specified, the cleanup is performed only for the vpn-node. This is an optional parameter. This parameter is available only in IETF-L3VPN-NM and IETF-L2VPN-NM services.
vpn-network-access-id	Name of the vpn-network-access ID for input vpn-node to clean up. This parameter is available only in IETF-L3VPN-NM and IETF-L2VPN-NM services. This is a mandatory parameter for IETF-L3VPN-NM if the vpn-node parameter is specified.

Perform the Cleanup action to clean up either a specific device in the service or the entire service. Cleaning up the entire service cleans up all the devices in the service.

SR-ODN Services

SR-CLI-ODN-300 is used as a sample service to demonstrate the command usage.

To clean up SR-ODN service per device

```
admin@ncs> request sr-te cleanup service-type sr-odn service SR-CLI-ODN-300 device
PIOSXR-1 no-networking true
```

```
#####
```

```
#          Warning          #
```

```
#####
```

```
You are about to forcefully cleanup a T-SDN service.
```

```
This will affect the deploying service and leave T-SDN & NSO out-of-sync (for no-
networking=true).
```

```
Are you sure you want to proceed? [no,yes] yes
```

```
success true
```

```
detail Cleaning up SR TE service: SR-CLI-ODN-300Cleaning up SR TE Internal
Services: SR-CLI-ODN-300
```

```
Removed all internal plan components
```

```
Removing side-effect queue: /ncs:side-effect-queue/side-effect{8}
```

```
Removed side-effects
```

```
Removed kickers
```

```
Cleanup Successful for SR TE Internal Services
```

```
Removed all external plan components
```

```
Removed side-effects
```

```
Removed kickers
```

```
Cleanup Successful
```

To clean up the SR-ODN service on all the devices in the service

```

admin@ncs> request sr-te cleanup service-type sr-odn service SR-CLI-ODN-300 no-
networking true

#####
#           Warning           #
#####

You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave T-SDN & NSO out-of-sync (for no-
networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up SR TE service: SR-CLI-ODN-300
  Removed all external plan components
  Removed side-effects
  Removed kickers
  Cleanup Successful

```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

To validate cleanup for SR-ODN service per device

```

admin@ncs% request sr-te cleanup service SR-CLI-ODN-300 device PIOSXR-0 service-
type sr-odn no-networking true

#####
#           Warning           #
#####

You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the device: PIOSXR-0 in service: SR-CLI-ODN-300 before cleanup

```

To validate cleanup for SR-ODN service on all the devices in the service

```

admin@ncs% request sr-te cleanup service SR-CLI-ODN-300 service-type sr-odn no-
networking true

#####
#           Warning           #
#####

You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false

```

detail Delete the service: SR-CLI-ODN-300 before cleanup

SR-Policy Services

A policy service can have only one head-end. Therefore, you can clean up only the service, even if you provide a device name. SR-Policy-1 is used as a sample service to demonstrate the command usage.

To clean up SR-Policy service per device

```
admin@ncs> request sr-te cleanup service-type sr-policy service SR-Policy-1 device
PIOSXR-0 no-networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up SR TE service: SR-Policy-1Cleaning up SR TE Internal Services:
SR-Policy-1
  Removed all internal plan components
  Removing side-effect queue: /ncs:side-effect-queue/side-effect{10}
  Removing side-effect queue: /ncs:side-effect-queue/side-effect{5}
  Removed side-effects
  Removed kickers
Cleanup Successful for SR TE Internal Services
  Removed all external plan components
  Removed side-effects
  Removed kickers
Cleanup Successful
```

To clean up the SR-policy service on all the devices in the service

```
admin@ncs> request sr-te cleanup service-type sr-policy service SR-Policy-1 no-
networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up SR TE service: SR-Policy-1
  Removed all external plan components
  Removed side-effects
```

```
Removed kickers
Cleanup Successful
```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

To validate cleanup for SR-policy service on all the devices in the service

```
admin@ncs% request sr-te cleanup service SR-CLI-ERO service-type sr-policy no-
networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s)
& NSO out-of-sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the service: SR-CLI-ERO before cleanup
```

CS SR-TE Policy Services

CS-DYNAMIC is used as a sample policy service to demonstrate the command usage.

```
admin@ncs% request cs-sr-te-actions cleanup service CS-DYNAMIC no-networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up SR TE service: CS-DYNAMIC-head-end-CS-internalCleaning up SR TE
Internal Services: CS-DYNAMIC-head-end-CS-internal
  Removed all internal plan components
  Removing zombie service: /ncs:zombies/ncs:service{"/cisco-sr-te-cfp-internal:sr-
te/cisco-sr-te-cfp-sr-policies-internal:policies/policy[name='CS-DYNAMIC-head-end-
CS-internal'] [head-end='PIOSXR-0']}
  Removed zombie service
  Removed commit-queue-recovery-data
  Cleanup Successful for SR TE Internal Services
  Removed all external plan components
  Removing zombie service: /ncs:zombies/ncs:service{"/cisco-sr-te-cfp:sr-te/cisco-
sr-te-cfp-sr-policies:policies/policy[name='CS-DYNAMIC-head-end-CS-internal']}
  Removed zombie service
```

```

Removing plan path: /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/cisco-sr-te-cfp-sr-policies:policy-plan{CS-DYNAMIC-head-end-CS-
internal}
Removed plan path
Removing zombie service: /ncs:zombies/ncs:service{"/cisco-sr-te-cfp:sr-te/cisco-
sr-te-cfp-sr-policies:policies/policy[name='CS-DYNAMIC-head-end-CS-internal']}
Removed zombie service
Cleanup SuccessfulCleaning up SR TE service: CS-DYNAMIC-tail-end-CS-internal
Removed all external plan components
Removing zombie service: /ncs:zombies/ncs:service{"/cisco-sr-te-cfp:sr-te/cisco-
sr-te-cfp-sr-policies:policies/policy[name='CS-DYNAMIC-tail-end-CS-internal']}
Removed zombie service
Removing zombie service: /ncs:zombies/ncs:service{"/cisco-sr-te-cfp:sr-te/cisco-
sr-te-cfp-sr-policies:policies/policy[name='CS-DYNAMIC-tail-end-CS-internal']}
Removed zombie service
Cleanup Successful
Removed all plan components
Removing zombie service: /ncs:zombies/ncs:service{"/cs-sr-te-policy[name='CS-
DYNAMIC']"}
Removed zombie service
Removing plan path: /cs-sr-te-plan{CS-DYNAMIC}
Removed plan path
Removing zombie service: /ncs:zombies/ncs:service{"/cs-sr-te-policy[name='CS-
DYNAMIC']"}
Removed zombie service
Cleanup Successful for CS SR-TE

```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

```

admin@ncs% request cs-s-te-actions cleanup service CS-STATIC-EXPLICIT-SID no-
networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s)
& NSO out-of-sync (for no-networking=true).
Are you sure you want to proceed? [no,yes yes
success false
detail Delete the service: CS-STATIC-EXPLICIT-SID before cleanup

```

IETF-L2VPN-NM Services

Use the Cleanup action to clean up the L2NM service as follows:

```
admin@ncs% request l2vpn-ntw l2nm-actions cleanup service l2nm-evpn no-networking
true
```

```
#####
```

```
#          Warning          #
```

```
#####
```

```
You are about to forcefully cleanup a T-SDN service.
```

```
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
```

```
Are you sure you want to proceed? [no,yes] yes
```

```
success true
```

```
detail Cleaning up l2vpn-ntw service: l2nm-evpn
```

```
Cleaning up L2vpn service: L2NM-l2nm-evpn-internal
```

```
Cleaning up L2vpn internal local-site service: L2NM-l2nm-evpn-internal PIOSXR-0
```

```
Removed all plan components
```

```
Removing side-effect queue: /ncs:side-effect-queue/side-effect{15}
```

```
Removing side-effect queue: /ncs:side-effect-queue/side-effect{81}
```

```
Removed side-effects
```

```
Removed kickers
```

```
Cleanup Successful for L2vpn internal local-site service: L2NM-l2nm-evpn-internal
PIOSXR-0
```

```
Removed all external plan components
```

```
Removing zombie service: /ncs:zombies/ncs:service{"/flat-L2vpn[name='L2NM-l2nm-
evpn-internal']"}
```

```
Removed zombie service
```

```
Removing side-effect queue: /ncs:side-effect-queue/side-effect{84}
```

```
Removed side-effects
```

```
Removed kickers
```

```
Removing plan path: /cisco-flat-L2vpn-fp:flat-L2vpn-plan{L2NM-l2nm-evpn-internal}
```

```
Removed plan path
```

```
Cleanup Successful
```

```
Removed all plan components
```

```
Removing side-effect queue: /ncs:side-effect-queue/side-effect{87}
```

```
Removed side-effects
```

```
Removed kickers
```

```
Removing plan path: /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service-plan{l2nm-
evpn}/plan/component{ietf-l2vpn-ntw-nano-services:vpn-node PIOSXR-0}
```

```
Removed plan path
```

```
Removing zombie service: /ncs:zombies/ncs:service{"/l2vpn-ntw/vpn-services/vpn-
service[vpn-id='l2nm-evpn']"}
```

```
Removed zombie service
```

```
Cleanup Successful for L2NM
```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

```
admin@cfs-node% request l2von-ntw l2m-actions cleanup service l2m-2p no-networking
true admin@cfs-node% request l2vpn-ntw l2m-actions cleanup service l2m-p2p no-
networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the service: l2m-p2p before cleanup
```

IETF-L3VPN-NM Services

The following example shows how to use the Cleanup action to clean up the L3NM service.

```
admin@ncs% request l3vpn-ntw l3nm-actions cleanup service 0-65008740 no-networking
true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail
  Removing service /l3vpn-ntw/vpn-services/vpn-service{0-65008740}
  Removed service /l3vpn-ntw/vpn-services/vpn-service{0-65008740}Cleaning up L3vpn
service: L3NM-0-65008740-internal
  Removed all internal plan components
  Removing service /cisco-flat-L3vpn-fp-internal:flat-L3vpn{L3NM-0-65008740-internal
PIOSX0}
  Removed service /cisco-flat-L3vpn-fp-internal:flat-L3vpn{L3NM-0-65008740-internal
PIOSX0}
  Removing service /cisco-flat-L3vpn-fp-internal:flat-L3vpn{L3NM-0-65008740-internal
PIOSX1}
  Removed service /cisco-flat-L3vpn-fp-internal:flat-L3vpn{L3NM-0-65008740-internal
PIOSX1}
  Removing side-effect queue: /ncs:side-effect-queue/side-effect{42}
  Removed side-effects
  Removing side-effect queue: /ncs:side-effect-queue/side-effect{44}
  Removed side-effects
  Removed kickers
```

```

Removed kickers
Cleanup Successful for L3VPN Internal Services
  Removed all plan components
Removed side-effects
Removed kickers
Cleanup Successful
  Removed all plan components
Removed side-effects
Removed kickers
Cleanup Successful for L3NM
[ok]

```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

```

admin@ncs% request l3vpn-ntw l3m-actions cleanup service l3m xr no-networking false
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the service: l3m xr before cleanup

```

Error Recovery Action

Use the error recovery action to manually recover a service from an error when the automatic error recovery fails or is not set up.

The following table shows the error recovery action parameters for a create or a delete failure.

Parameter	Description
service-type	Type of service (SR-ODN) to be recovered.
service	Name of the service to be recovered.
endpoint	Name of the endpoint for which the device connected under a given service is to be recovered. If no device name is specified, the entire service is recovered for all the failed devices in the service. This is an optional parameter.

vpn-node	Name of the vpn-node to be recovered. If specified, the recovery is performed only for the vpn-node. This is an optional parameter. This parameter is available only in IETF-L3VPN-NM and IETF-L2VPN-NM services.
vpn-network-access-id	Name of the vpn-network-access ID for input vpn-node to be recovered. This parameter is available only for IETF-L3VPN-LM service.
device	Name of the device under a given service that needs to be recovered. If no device name is provided or if the option is unavailable, all the failed devices in the service are recovered. This is an optional parameter.
sync-direction	The sync direction to bring the device and NSO in sync. It can be either sync-from or sync-to depending on which configuration is the primary. This is a mandatory parameter.

This topic discusses how to execute the error recovery action on the SR-TE CFP services.

SR-ODN Services

The following is an example of a plan for a create failure. SR-CLI-ODN-300 is used as a sample service to demonstrate the command usage.

```
admin@ncs> show cisco-sr-te-cfp:sr-te odn odn-template-plan SR-CLI-ODN-300 plan
```

```

TYPE                NAME      BACK TRACK GOAL STATUS CODE STATE          STATUS WHEN          ref ACTION STATUS
-----
self                self      false -   -   init         ready          reached 2020-08-03T19:10:15 - -
self                self      false -   -   ready        failed         failed 2020-08-03T19:10:18 - -
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end PIOSXR-0 false -   -   init         ready          reached 2020-08-03T19:10:15 - -
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end PIOSXR-0 false -   -   cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply ready          reached 2020-08-03T19:10:15 - -
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end PIOSXR-1 false -   TSDN-SR-301 init         ready          reached 2020-08-03T19:10:18 - -
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end PIOSXR-1 false -   TSDN-SR-301 cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply ready          reached 2020-08-03T19:10:15 - -
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end PIOSXR-1 false -   TSDN-SR-301 cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply ready          failed 2020-08-03T19:10:18 - -

plan failed

plan error-info message "Failed to connect to device PIOSXR-1: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state"

plan status-code-detail cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end PIOSXR-1
code                TSDN-SR-301
context "Device unreachable"
context-msg "Failed to connect to device PIOSXR-1: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state"
severity            ERROR
recommended-action "Check device connectivity from NSO and perform recovery steps."
impacted-device    PIOSXR-1

```

Error recovery action under service - For create failure

When a plan fails, the device comes back up and you can request error recovery on the service. The plan is successful after the service has been successfully recovered.

```
admin@ncs> request sr-te odn odn-template SR-CLI-ODN-300 error-recovery sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering SR TE service: SR-CLI-ODN-300
Recovered create failure on PIOSXR-1
Recovery Complete for SR TE Internal Services
Recovery Complete
```

Error recovery action under head-end - For create failure

When the device comes back up after a failure, you can request error recovery from under the head-end of the service. The plan is successful after the service has been successfully recovered.

```
admin@ncs> request sr-te odn odn-template SR-CLI-ODN-300 head-end PIOSXR-1 error-
recovery sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering SR TE service: SR-CLI-ODN-300
Recovered create failure on PIOSXR-1
Recovery Complete for SR TE Internal Services
Recovery Complete
```

Error recovery action under service - For Already in Sync Services

If services are already in sync and the user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery is required.

```
admin@ncs% request sr-te odn odn-template SR-CLI-ODN-300 error-recovery sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
```

```
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:SR-CLI-ODN-300 and device are not failed
```

No such Device or Service

```
admin@ncs% request sr-te error-recovery service SR-CLI-ODN-300 service-type sr-odn
device ABC sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. no failed device for SR-CLI-ODN-300
```

Per Device when device Has Not Failed

```
admin@ncs% request sr-te error-recovery service SR-CLI-ODN-300 service-type sr-odn
device PIOSXR-0 sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:SR-CLI-ODN-300 and device:PIOSXR-0 are not
failed
```

SR-Policy Services

The following is an example of a plan for a create failure. SR-Policy-1 is used as a sample service to demonstrate the command usage.

```
admin@ncs> show cisco-sr-te-cfp:sr-te policies policy-plan SR-Policy-1 plan
```

TYPE	NAME	BACK TRACK	GOAL	STATUS	CODE	STATE	STATUS	WHEN	ref	POST ACTION	STATUS
self	self	falsq	-	-	-	init ready	reached failed	2020-08-17T19:41:17	-	-	-
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end	PIOSXR-0	false	-	TSDN-SR-301	-	init cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply ready	reached reached reached failed	2020-08-17T19:41:17	-	-	-

```
plan failed
```

```
plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused (Connection refused) in new state"
```

```
plan status-code-detail cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end
PIOSXR-0
```

```
code           TSDN-SR-301
```

```
context "Device unreachable"
```

```
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
```

```

severity          ERROR
recommended-action "Check device connectivity from NSO and perform recovery
steps."
impacted-device   PIOSXR-0

```

When a plan fails, the device comes back up and you can request error recovery on service. The plan is successful after the service is recovered.

```

admin@ncs> request sr-te error-recovery service-type sr-policy service SR-Policy-1
sync-direction sync-from

```

```

#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering SR TE service: SR-Policy-1
Recovered create failure on PIOSXR-0
Recovery Complete for SR TE Internal Services
Recovery Complete

```

Error Recovery action under service - For create failure

When a plan fails, the device comes back up, and you can request error recovery from under the service. The plan is successful after the service has been successfully recovered.

```

admin@ncs> request sr-te policies policy SR-Policy-1 error-recovery sync-direction
sync-from

```

```

#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering SR TE service: SR-Policy-1
Recovered create failure on PIOSXR-0
Recovery Complete for SR TE Internal Services
Recovery Complete

```

Error recovery action under service - For Already in Sync Services

If services are already in sync and the user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery is required.

```

admin@ncs> request sr-te policies policy SR-Policy-1 error-recovery sync-direction
sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:SR-Policy-1 and device are not failed

```

No such Device or Service

```

admin@ncs% request sr-te error-recovery service SR-CLI-DYNAMIC service-type sr-
policy device ABC sync-direction sync-to
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. no failed device for SR-CLI-DYNAMIC

```

Per Device when device Has Not Failed

```

admin@ncs% request sr-te error-recovery service SR-CLI-DYNAMIC service-type sr-
policy device PIOSXR-0 sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:SR-CLI-DYNAMIC and device:PIOSXR-0 are not
failed

```

CS SR-TE Policy Services

The following is an example of a plan for a create failure. CS-DYNAMIC is used as a sample service to demonstrate the command usage.

```
admin@ncs% run show cs-sr-te-plan CS-DYNAMIC plan
```

```

POST
-----
TYPE          NAME      BACK TRACK_GOAL  STATUS CODE  STATE          STATUS  WHEN          ref  STATUS
-----
self          self      false -      -            join         reached  2022-06-01T22:50:58 - -
ready
failed 2022-06-01T22:51:01 - -
cisco-cs-sr-te-cfg-nano-plan-head-001 PIOSXR-0 false -  TSDM-SR-301 join         reached  2022-06-01T22:50:58 - -
cisco-cs-sr-te-cfg-nano-plan-config-apply reached  2022-06-01T22:50:58 - -
ready
failed 2022-06-01T22:51:01 - -
tail-end     PIOSXR-1 false -      -            join         reached  2022-06-01T22:50:58 - -
cisco-cs-sr-te-cfg-nano-plan-config-apply reached  2022-06-01T22:50:58 - -
ready
reached 2022-06-01T22:51:01 - -

```

plan failed

```

plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused (Connection refused) in new state"

plan status-code-detail cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end
PIOSXR-0

code                TSDN-SR-301
context "Device unreachable"
  context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
  severity          ERROR
  recommended-action "Check device connectivity from NSO and perform recovery
steps."
  impacted-device   PIOSXR-0

```

When a plan fails, the device comes back up and you can request error recovery on service. The plan is successful after the service has been successfully recovered.

```

admin@ncs> request cs-sr-te-actions error-recovery service CS-STATIC-EXPLICIT-SID
sync-direction sync-to
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
Recovering CS SR-TE service: CS-STATIC-EXPLICIT-SID sync_direction: sync-to
endpoint: None
Recovering SR TE service: CS-CS-STATIC-EXPLICIT-SID-head-end-internal
Successful recovery for create failure on device: PIOSXR-0
Recovery Complete for SR TE Internal Services
Recovering SR TE service: CS-CS-STATIC-EXPLICIT-SID-tail-end-internal
Successful recovery for create failure on device: PIOSXR-1
Recovery Complete for SR TE Internal Services

```

Error Recovery action under service - For create failure

When a plan fails, the device comes back up, and you can request error recovery from under the service. The plan is successful after the service has been successfully recovered.

```

admin@ncs% request cs-sr-te-actions error-recovery service CS-DYNAMIC sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success true

```

```

detail Recovering CS SR-TE service: CS-DYNAMIC sync_direction: sync-from endpoint:
None
CS SR-TE plan tail-end component not failed for CS-DYNAMIC, recovery not executed
Recovering SR TE service: CS-DYNAMIC-head-end-CS-internal
Successful recovery for create failure on device: PIOSXR-0
Removed cq_error_path: None
Recovery Complete for SR TE Internal Services
Recovery Complete
Error Recovery Successful for CS SR-TE

```

Error Recovery action under head-end - For create failure

When a plan fails, the device comes back up, and you can request error recovery from under the head-end of the service. The plan is successful after the service has been successfully recovered.

```

admin@ncs% request cs-sr-te-actions error-recovery service CS-DYNAMIC endpoint
head-end sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering CS SR-TE service: CS-DYNAMIC sync_direction: sync-from endpoint:
head-end
Recovering SR TE service: CS-DYNAMIC-head-end-CS-internal
Successful recovery for create failure on device: PIOSXR-0
Removed cq_error_path: None
Recovery Complete for SR TE Internal Services
Recovery Complete
Error Recovery Successful for CS SR-TE

```

Error Recovery action under service - For Already In Sync Services

If services are already in sync and the user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery is required.

```

admin@ncs% request cs-sr-te-actions error-recovery service CS-DYNAMIC sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.

```

```
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service CS-DYNAMIC and devices are in sync
```

No such Device or Service

```
admin@ncs% request cs-sr-te-actions error-recovery service CS-DYNAMIC sync-
direction sync-to endpoint head-end
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service CS-DYNAMIC head-end component not failed
```

```
admin@ncs% request cs-sr-te-actions error-recovery service CS-DYNAMIC sync-
direction sync-to endpoint tail-end
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service CS-DYNAMIC tail-end component not failed
```

IETF-L2VPN-NM Services

The following is an example of a failed plan for the vpn-nodes in a service.

```
admin@ncs% run show l2vpn-ntw vpn-services vpn-service-plan l2nm-evpn plan
```

TYPE	BACK		STATUS CODE	STATE	STATUS	WHEN	ACTION	
	NAME	TRACK_GOAL					ref	STATUS
self	self	false	-	not ready	reached	2020-09-22T19:09:56	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-0	false	TSDN-L2VPN-301	not ready	failed	2020-09-22T19:10:01	-	-
				not ready	reached	2020-09-22T19:09:56	-	-
				not ready	reached	2020-09-22T19:09:56	-	-
				not ready	failed	2020-09-22T19:10:01	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-1	false	TSDN-L2VPN-301	not ready	reached	2020-09-22T19:09:56	-	-
				not ready	reached	2020-09-22T19:09:56	-	-
				not ready	failed	2020-09-22T19:10:01	-	-

```
plan failed
```

```
plan error-info message "Failed to connect to device PIOSXR-1: connection refused:
NEDCOM CONNECT: Connection refused (Connection refused) in new state"
```

```
plan status-code-detail local-site PIOSXR-0
```

```
code TSDN-L2VPN-301
```

```
context "Device unreachable"
```

```
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
```

```
severity ERROR
```

```
recommended-action "Check device connectivity from NSO and perform recovery
steps."
```



```

impacted-device      PIOSXR-0
plan status-code-detail remote-site PIOSXR-1
code                 TSDN-L2VPN-301
context "Device unreachable"
  context-msg "Failed to connect to device PIOSXR-1: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
severity             ERROR
recommended-action "Check device connectivity from NSO and perform recovery
steps."
impacted-device      PIOSXR-1

```

The vpn-node devices come back up and you can request error recovery on the service. The plan is successful after the service has been successfully recovered.

The following shows error recovery for the service.

```

admin@ncs% request l2vpn-ntw l2nm-actions error-recovery service l2nm-evpn sync-
direction sync-from

#####
#           Warning           #
#####

You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering L2NM service: l2nm-evpn
Recovering L2vpn service: L2NM-l2nm-evpn-internal
Recovered create failure on PIOSXR-0
Removed cq_error_path: None
Recovery Complete for L2VPN Internal Service
Recovered create failure on PIOSXR-1
Removed cq_error_path: None
Recovery Complete for L2VPN Internal Service
Recovery Complete

```

Error Recovery action under service - For Already In Sync Services

If services are already in sync and user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery required.

```

admin@ncs% request l2vpn-ntw l2nm-actions error-recovery service l2nm-evpn sync-
direction sync-from

#####
#           Warning           #
#####

```

```

You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service l2nm-evpn and devices are in sync

```

No such Service

```

admin@ncs% request l2vpn-ntw l2nm-actions error-recovery service ABC sync-direction
sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. No such service found for recovery:ABC

```

Error recovery action for a vpn-node

The following is an example of a plan for a creation failure.

```
admin@ncs% run show l2vpn-ntw vpn-services vpn-service-plan l2nm-evpn plan
```

TYPE	BACK NAME	TRACK	GOAL	STATUS	CODE	STATE	STATUS	WHEN	ACTION	
									ref	STATUS
self	self	false	-	-	-	init ready	reached	2020-09-22T18:34:56	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-0	false	-	TSDN-L2VPN-301	-	init ready	failed	2020-09-22T18:35:02	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-0	false	-	-	-	ietf-l2vpn-ntw-nano-services:config-apply	reached	2020-09-22T18:34:56	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-1	false	-	-	-	ready	failed	2020-09-22T18:35:02	-	-
ietf-l2vpn-ntw-nano-services:vpn-node	PIOSXR-1	false	-	-	-	init	reached	2020-09-22T18:34:56	-	-
						ietf-l2vpn-ntw-nano-services:config-apply	reached	2020-09-22T18:34:56	-	-
						ready	reached	2020-09-22T18:35:02	-	-

```
plan failed
```

```
plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused (Connection refused) in new state"
```

```
plan status-code-detail local-site PIOSXR-0
```

```
code TSDN-L2VPN-301
```

```
context "Device unreachable"
```

```
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
```

```
severity ERROR
```

```
recommended-action "Check device connectivity from NSO and perform recovery
steps."
```

```
impacted-device PIOSXR-0
```

When the vpn-node device comes back up, you can request for error recovery on the service as follows:

```

admin@ncs% request l2vpn-ntw l2nm-actions error-recovery service l2nm-evpn vpn-node
PIOSXR-0 sync-direction sync-from
#####

```

```
#          Warning          #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering L2NM service: l2nm-evpn
Recovering L2vpn service: L2NM-l2nm-evpn-internal
Recovered create failure on PIOSXR-0
Removed cq_error_path: None
Recovery Complete for L2VPN Internal Service
Recovery Complete
```

IETF-L3VPN-NM Services

The following is an example of a failed plan for a service.

```
# Check vpn-service-plan for overall service status
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 plan
```

TYPE	NAME	TRACK	GOAL	BACK STATE	STATUS	WHEN	POST ACTION	
							ref	STATUS
Self	self	false	-	init	reached	2023-12-13T00:23:10	-	-
				ready	failed	2023-12-13T00:23:13	-	-

```
plan failed
plan error-info message "Failed to connect to device PIOSXR-0: c
onnection refused: NEDCOM CONNECT: Connection refused in new state"
plan failed-vpn-nodes ['PIOSXR-0']
```

```
# To see more details on failure, check vpn-node-plan for failed-vpn-nodes
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 vpn-nodes
vpn-node-plan PIOSXR-0 plan
```

TYPE	NAME	BACK		GOAL	STATUS CODE	STATE	STATUS	WHEN	POST ACTION	
		TRACK	GOAL						ref	STATUS
Self	self	false	-	TSDN-L3VPN-301	init	reached	2023-12-13T00:23:10	-	-	
				ietf-l3vpn-ntw-nano-services:config-apply	reached	2023-12-13T00:23:10	-	-		
					ready	failed	2023-12-13T00:23:13	-	-	

```
plan failed
plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused in new state"
plan status-code-detail self self
code          TSDN-L3VPN-301
context "Device unreachable"
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused in new state"
severity      ERROR
```

```
recommended-action "Check device connectivity from NSO and perform recovery
steps."
impacted-device      PIOSXR-0
```

The device comes back up and you can request error recovery on the service. The plan is successful after the service is recovered successfully.

The following shows error recovery for the service.

```
admin@ncs% request l3vpn-ntw l3nm-actions error-recovery service 0-65008740 sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering L3VPN service: L3NM-0-65008740-internal
Recovered create failure on PIOSX0
Removed cq_error_path: None
Recovery Complete for L3VPN Internal Services
Recovery Complete
```

Error recovery action for a service – Creation failure

The following is an example of a plan for a creation failure.

```
admin@ncs% request l3vpn-ntw vpn-services vpn-service 0-65008740 error-recovery
sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering L3VPN service: L3NM-0-65008740-internal
Recovered create failure on PIOSX0
Removed cq_error_path: None
Recovery Complete for L3VPN Internal Services
Recovery Complete
```

Error recovery action for a device

If a specific device has failed in a service plan, you may invoke error-recovery action for a specific vpn-node. The following is an example of a plan for a creation failure.

```
# Check vpn-service-plan for overall service status
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 plan
```

TYPE	BACK		TRACK	GOAL	STATE	STATUS	WHEN	POST ACTION	
	NAME	TRACK						ref	STATUS
Self	self	false	-		init	reached	2023-12-13T00:23:10	-	-
					ready	failed	2023-12-13T00:23:13	-	-

```
plan failed
plan failed-vpn-nodes ['PIOSXR-0']
```

To see more details on failure, check vpn-node-plan for failed-vpn-nodes

```
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 vpn-nodes vpn-node-plan PIOSXR-0 plan
```

TYPE	BACK		TRACK	GOAL	STATUS CODE	STATE	STATUS	WHEN	POST ACTION	
	NAME	TRACK							ref	STATUS
self	self	false	-	TSDN-L3VPN-301	init	reached	2023-12-13T00:23:10	-	-	
					ietf-l3vpn-ntw-nano-services:config-apply	reached	2023-12-13T00:23:10	-	-	
					ready	failed	2023-12-13T00:23:13	-	-	

```
plan failed
plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused in new state"
plan status-code-detail self self
code TSDN-L3VPN-301
context "Device unreachable"
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused in new state"
severity ERROR
recommended-action "Check device connectivity from NSO and perform recovery
steps."
impacted-device PIOSXR-0
```

The vpn-node plan component is set to success once the service is successfully recovered.

```
admin@ncs% request l3vpn-ntw vpn-services vpn-service 0-65008740 vpn-nodes vpn-node
PIOSXR0 error-recovery vpn-network-access-id 0 sync-direction sync-from
#####
# Warning #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering L3VPN service: L3NM-0-65008740-internal
Recovered create failure on PIOSXR0_0
Removed cq_error_path: None
Recovery Complete for L3VPN Internal Services
Recovery Complete
```

Error recovery action for Error Recovery action under service - For Already In Sync Services

No Failed Endpoints

If services are already in sync and the user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery is required.

```
admin@ncs% request l3vpn-ntw l3nm-actions error-recovery service 0-65008740 sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. No failed endpoints to recover in service: 0-65008740
```

No such Service

```
admin@ncs% request l3vpn-ntw l3nm-actions error-recovery service ABC sync-direction
sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. No such service found for recovery:ABC
```

No such Device

```
admin@ncs% request l3vpn-ntw l3nm-actions error-recovery service 0-65008740 vpn-
node ABCD vpn-network-access-id 25 sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No such failed endpoint: ABCD found for recovery in service: 0-65008740
```

Per Device when device Has Not Failed

```
admin@ncs% request l3vpn-ntw l3nm-actions error-recovery service 0-65008740 vpn-
node ABCD vpn-network-access-id 25 sync-direction sync-from
#####
#           Warning           #
```

```
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Endpoint: PIOSXR-0 and service: 0-65008740 are not
failed
```

get-modifications Action

The **get-modifications** action is like the service get-modifications action. You can perform this action at the service level.

This action returns the data the service modified in the CDB (whether it is device configuration or internal data), either in the CLI curly bracket format or NETCONF XML edit configuration format. By default, the output format is CLI.

If the parameter 'reverse' is given, the modifications required to 'reverse' the effect of the service are shown. This will be applied if the service is deleted. This data is always available. The parameters **shallow** and **deep** control if the modifications must be displayed for the service only or all the modified services, respectively.

This data is available only if the parameter **/services/global-settings/collect-forward-diff** is set to true as follows before you create the service.

```
admin@ncs% show services global-settings collect-forward-diff
collect-forward-diff true;
```

Note: Setting this parameter to true can have severe performance implications. For more information about this parameter, see the NSO documentation.

This section provides the output in CLI format, which is the default format for the SR-TE CFP services.

SR-ODN Services

SR-CLI-ODN-300 is used as a sample service to demonstrate the command usage.

```
admin@ncs> request sr-te odn odn-template SR-CLI-ODN-300 get-modifications
cli {
  local-node {
    data devices {
      device PIOSXR-0 {
        config {
          +       segment-routing {
          +         traffic-eng {
          +           on-demand {
          +             color 300 {
```

```

+         bandwidth 200;
+         dynamic {
+             metric {
+                 type hopcount;
+                 margin {
+                     absolute 30;
+                 }
+             }
+             disjoint-path {
+                 group-id 10;
+                 type link;
+                 sub-id 5;
+             }
+             affinity include-all {
+                 name-list GREEN {
+                     name;
+                 }
+                 name-list RED {
+                     name;
+                 }
+             }
+             maximum-sid-depth 6;
+         }
+     }
+ }
}

device PIOSXR-1 {
    config {
+        segment-routing {
+            traffic-eng {
+                on-demand {
+                    color 300 {
+                        bandwidth 200;
+                        dynamic {
+                            metric {
+                                type hopcount;
+                                margin {
+                                    absolute 30;
+                                }
+                            }
+                        }
+                    }
+                }
+            }
+        }

```



```
+         binding-sid {
+             mpls 100;
+         }
+         color {
+             value 100;
+             end-point {
+                 ipv4 7.7.7.7;
+             }
+         }
+         candidate-paths {
+             preference 100 {
+                 dynamic {
+                     metric {
+                         sid-limit 10;
+                         type te;
+                         margin {
+                             relative 40;
+                         }
+                     }
+                 }
+             }
+             preference 200 {
+                 explicit {
+                     segment-list mysidlist {
+                         weight 10;
+                     }
+                 }
+                 constraints {
+                     affinity {
+                         rule include-all {
+                             name-list GREEN {
+                                 name;
+                             }
+                             name-list RED {
+                                 name;
+                             }
+                         }
+                     }
+                 }
+             }
+         }
+     }
+ }
```

```

    }
  }
}

```

CS SR-TE Policy Services

For CS SR-TE get-modifications, two lower layer SR-TE policy services combine to create a bidirectional CS SR-TE Policy.

The following is a sample for the dynamic CS SR-TE policy **CS-DYNAMIC**.

```

admin@ncs% request cs-sr-te-policy CS-DYNAMIC get-modifications cli {
  local-node {
    data devices {
      device PIOSXR-0 {
        config {
+          segment-routing {
+          traffic-eng {
+            policy srte_c_10000_ep_10.0.0.2 {
+              bandwidth 123;
+              color {
+                value 10000;
+                end-point {
+                  ipv4 10.0.0.2;
+                }
+              }
+            }
+            path-protection;
+            candidate-paths {
+              preference 10 {
+                dynamic {
+                  metric {
+                    type hopcount;
+                  }
+                }
+                lock {
+                  duration 0;
+                }
+                constraints {
+                  segments {
+                    protection unprotected-only;
+                    adjacency-sid-only;
+                  }
+                }
+              }
+            }
+          }
+        }
+      }
+    }
+  }
+}

```

```
+         bidirectional {
+             association-id 10;
+         }
+         backup-ineligible {
+         }
+     }
+ preference 50 {
+     dynamic {
+         metric {
+             type hopcount;
+         }
+     }
+     lock {
+         duration 123;
+     }
+     constraints {
+         segments {
+             protection unprotected-only;
+             adjacency-sid-only;
+         }
+         disjoint-path {
+             group-id 5;
+             type srlg-node;
+         }
+     }
+     bidirectional {
+         association-id 50;
+     }
+ }
+ preference 100 {
+     dynamic {
+         pcep {
+         }
+         metric {
+             type hopcount;
+             margin {
+                 absolute 123;
+             }
+         }
+     }
+     constraints {
+         segments {
+             protection unprotected-only;
+             adjacency-sid-only;
```

```

+                                     }
+                                     disjoint-path {
+                                         group-id 5;
+                                         type srlg-node;
+                                     }
+                                     }
+                                     bidirectional {
+                                         association-id 100;
+                                     }
+                                     }
+                                 }
+                             }
+                         }
+                     }
+                 }
+             }
+         }
+     }
+ }
device PIOSXR-1 {
    config {
+         segment-routing {
+             traffic-eng {
+                 policy srte_c_10000_ep_10.0.0.1 {
+                     bandwidth 123;
+                     color {
+                         value 10000;
+                         end-point {
+                             ipv4 10.0.0.1;
+                         }
+                     }
+                 }
+             }
+             path-protection;
+             candidate-paths {
+                 preference 10 {
+                     dynamic {
+                         metric {
+                             type hopcount;
+                         }
+                     }
+                 }
+                 lock {
+                     duration 0;
+                 }
+                 constraints {
+                     segments {
+                         protection unprotected-only;
+                         adjacency-sid-only;
+                     }
+                 }
+             }
+         }
    }
}

```

```
+         }
+         bidirectional {
+             association-id 10;
+         }
+         backup-ineligible {
+         }
+     }
+     preference 50 {
+         dynamic {
+             metric {
+                 type hopcount;
+             }
+         }
+         lock {
+             duration 123;
+         }
+         constraints {
+             segments {
+                 protection unprotected-only;
+                 adjacency-sid-only;
+             }
+             disjoint-path {
+                 group-id 10;
+                 type srlg-node;
+             }
+         }
+         bidirectional {
+             association-id 50;
+         }
+     }
+     preference 100 {
+         dynamic {
+             pcep {
+             }
+             metric {
+                 type hopcount;
+                 margin {
+                     absolute 123;
+                 }
+             }
+         }
+         constraints {
+             segments {
+                 protection unprotected-only;
```

```

+                                     adjacency-sid-only;
+                                     }
+                                     disjoint-path {
+                                       group-id 10;
+                                       type srlg-node;
+                                     }
+                                     }
+                                     }
+                                     bidirectional {
+                                       association-id 100;
+                                     }
+                                   }
+                                 }
+                               }
+                             }
+                           }
+                         }
+                       }
+                     }
+                   }
+                 }
+               }
+             }
+           }
+         }
+       }
+     }
+   }
+ }

cisco-sr-te-cfp:sr-te {
  policies {
+    policy CS-CS-DYNAMIC-head-end-internal {
+      head-end PIOSXR-0;
+      tail-end 10.0.0.2;
+      color 10000;
+      circuit-style {
+        path-protection {
+          }
+        working-path {
+          dynamic {
+            constraints {
+              segments {
+                protection unprotected-only;
+              }
+            disjoint-path {
+              type srlg-node;
+              group-id 5;
+            }
+          }
+        }
+        pce {
+          }
+        metric-type hopcount;
+        metric-margin {
+          absolute 123;
+        }
+      }
+    }
+  }
+ }

```

```

+         bidirectional-association-id 100;
+     }
+ }
+ protect-path {
+     dynamic {
+         constraints {
+             segments {
+                 protection unprotected-only;
+             }
+             disjoint-path {
+                 type srlg-node;
+                 group-id 5;
+             }
+         }
+         metric-type hopcount;
+         bidirectional-association-id 50;
+     }
+     revertive true;
+     wait-to-revert-timer 123;
+ }
+ restore-path {
+     dynamic {
+         constraints {
+             segments {
+                 protection unprotected-only;
+             }
+         }
+         metric-type hopcount;
+         bidirectional-association-id 10;
+     }
+ }
+ }
+ bandwidth 123;
+ }
+ policy CS-CS-DYNAMIC-tail-end-internal {
+     head-end PIOSXR-1;
+     tail-end 10.0.0.1;
+     color 10000;
+     circuit-style {
+         path-protection {
+         }
+         working-path {
+             dynamic {
+                 constraints {

```



```
+         segments {
+             protection unprotected-only;
+         }
+         disjoint-path {
+             type srlg-node;
+             group-id 10;
+         }
+     }
+     pce {
+     }
+     metric-type hopcount;
+     metric-margin {
+         absolute 123;
+     }
+     bidirectional-association-id 100;
+ }
+ }
+ protect-path {
+     dynamic {
+         constraints {
+             segments {
+                 protection unprotected-only;
+             }
+             disjoint-path {
+                 type srlg-node;
+                 group-id 10;
+             }
+         }
+         metric-type hopcount;
+         bidirectional-association-id 50;
+     }
+     revertive true;
+     wait-to-revert-timer 123;
+ }
+ restore-path {
+     dynamic {
+         constraints {
+             segments {
+                 protection unprotected-only;
+             }
+         }
+         metric-type hopcount;
+         bidirectional-association-id 10;
+     }
+ }
```

```

+         }
+     }
+     bandwidth 123;
+ }
}
}
}
}
}
}

```

IETF-L2VPN-NM Services

Perform the get-modifications action at the service level.

The following is an example of how to use the action.

```

admin@ncs> request l2vpn-ntw vpn-services vpn-service l2nm-p2p get-modifications
cli {
  local-node {
    data devices {
      device PIOSXR-0 {
        config {
          interface {
+           GigabitEthernet 0/0/0/1 {
+             description "T-SDN Interface";
+           }
          GigabitEthernet-subinterface {
+           GigabitEthernet 0/0/0/1.601 {
+             mode l2transport;
+             description "T-SDN Interface";
+             mtu 65;
+             encapsulation {
+             dot1q {
+               vlan-id 601;
+             }
+           }
+           rewrite {
+             ingress {
+               tag {
+                 push;
+                 dot1q 123;
+                 mode symmetric;
+               }
+             }
+           }
+         }
+       }
+     }
+   }
+ }
}

```

```

    }
  }
+   l2vpn {
+     pw-class l2nm-p2p {
+       encapsulation {
+         mpls {
+           control-word;
+         }
+       }
+     }
+     xconnect {
+       group l2nm-p2p {
+         p2p l2nm-p2p {
+           interface GigabitEthernet0/0/0/1.601;
+           neighbor 198.18.1.5 1001 {
+             ip-version ipv4;
+             mpls {
+               static {
+                 label {
+                   local 101;
+                   remote 102;
+                 }
+               }
+             }
+           }
+         }
+       }
+     }
+   }
+ }
+
+ device PIOSXR-1 {
+   config {
+     interface {
+       GigabitEthernet 0/0/0/1 {
+         description "T-SDN Interface";
+       }
+       GigabitEthernet-subinterface {
+         GigabitEthernet 0/0/0/1.601 {
+           mode l2transport;
+           description "T-SDN Interface";
+           mtu 64;
+           encapsulation {

```

```
+             dot1q {
+                 vlan-id 601;
+             }
+         }
+         rewrite {
+             ingress {
+                 tag {
+                     push;
+                     dot1q 234;
+                     mode symmetric;
+                 }
+             }
+         }
+     }
+ }
+ l2vpn {
+     pw-class l2nm-p2p {
+         encapsulation {
+             mpls {
+                 control-word;
+             }
+         }
+     }
+     xconnect {
+         group l2nm-p2p {
+             p2p l2nm-p2p {
+                 interface GigabitEthernet0/0/0/1.601;
+                 neighbor 198.18.1.4 1001 {
+                     ip-version ipv4;
+                     mpls {
+                         static {
+                             label {
+                                 local 102;
+                                 remote 101;
+                             }
+                         }
+                     }
+                 }
+                 pw-class l2nm-p2p;
+             }
+         }
+     }
+ }
+ }
```

```

    }
  }
}
}

```

IETF-L3VPN-NM Services

The following is a sample command to execute the get-modifications action on the IETF-L3VPN-NM service.

```

admin@ncs% request l3vpn-ntw vpn-services vpn-service 0-65008740 get-modifications
cli {
  local-node {
    data devices {
      device PIOSXR-0 {
        config {
          vrf {
+           vrf-list 0-65008740 {
+             address-family {
+               ipv6 {
+                 unicast {
+                   import {
+                     route-target {
+                       address-list 65010:17405;
+                       address-list 65010:17406;
+                     }
+                   }
+                 export {
+                   route-target {
+                     address-list 65010:17405;
+                     address-list 65010:17406;
+                   }
+                 }
+             }
+           }
+         }
+       }
+     }
+   }
+ }
+
+ interface {
+   GigabitEthernet 1/1/1/1 {
+     description "T-SDN Interface";
+   }
+   GigabitEthernet-subinterface {

```

```

+           GigabitEthernet 1/1/1/1.1234 {
+               description "T-SDN Interface";
+               encapsulation {
+                   dot1q {
+                       vlan-id 1234;
+                   }
+               }
+               vrf 0-65008740;
+               ipv6 {
+                   address {
+                       prefix-list 2001:db8::1/32;
+                   }
+               }
+           }
+       }
+   }
+   route-policy PASS_ALL {
+       value " pass\r
+";
+   }
+   router {
+       bgp {
+           bgp-no-instance 65001 {
+               vrf 0-65008740 {
+                   rd 65100:874000024;
+                   address-family {
+                       ipv6 {
+                           unicast {
+                               segment-routing {
+                                   srv6 {
+                                       locator locv6;
+                                       alloc {
+                                           mode per-vrf;
+                                       }
+                                   }
+                               }
+                           }
+                       }
+                   }
+               }
+           }
+           neighbor 2001:db8::2 {
+               remote-as 65003;
+               ebgp-multihop {
+                   ttl-value 12;
+               }
+               mpls;
+           }
+       }
+   }

```



```
+          }
+          GigabitEthernet-subinterface {
+              GigabitEthernet 1/1/1/1.1231 {
+                  description "T-SDN Interface";
+                  encapsulation {
+                      dot1q {
+                          vlan-id 1231;
+                      }
+                  }
+                  vrf 0-65008740;
+                  ipv4 {
+                      address {
+                          ip 10.1.1.2;
+                          mask 255.255.255.0;
+                      }
+                  }
+                  ipv6 {
+                      address {
+2002:4601:102::4601:103/24;
+                          prefix-list
+                      }
+                  }
+              }
+              GigabitEthernet 1/1/1/1.1234 {
+                  description "T-SDN Interface";
+                  encapsulation {
+                      dot1q {
+                          vlan-id 1234;
+                      }
+                  }
+                  vrf 0-65008740;
+                  ipv4 {
+                      address {
+                          ip 10.1.1.1;
+                          mask 255.255.255.0;
+                      }
+                  }
+                  ipv6 {
+                      address {
+2002:4601:102::4601:102/24;
+                          prefix-list
+                      }
+                  }
+              }
+          }
+      }
```



```
    }
+   route-policy PASS_ALL {
+     value " pass\r
";
+   }
  router {
    bgp {
+     bgp-no-instance 65001 {
+       vrf 0-65008740 {
+         rd auto;
+         address-family {
+           ipv4 {
+             unicast {
+               redistribute {
+                 connected {
+                   metric 123;
+                 }
+               }
+             segment-routing {
+               srv6 {
+                 locator locv4;
+                 alloc {
+                   mode per-vrf;
+                 }
+               }
+             }
+           }
+         }
+       }
+     }
+     ipv6 {
+       unicast {
+         redistribute {
+           connected {
+             }
+         }
+       }
+       segment-routing {
+         srv6 {
+           locator locv6;
+           alloc {
+             mode per-vrf;
+           }
+         }
+       }
+     }
+   }
+ }
```

```

+           }
+ neighbor 10.1.1.1 {
+     remote-as 65003;
+     address-family {
+       ipv4 {
+         unicast {
+           route-policy in {
+             name PASS_ALL;
+           }
+           route-policy out {
+             name PASS_ALL;
+           }
+         }
+       }
+     }
+ }
+ neighbor 10.1.1.3 {
+   remote-as 65005;
+   ebgp-multihop {
+     ttl-value 11;
+   }
+   address-family {
+     ipv4 {
+       unicast {
+         route-policy in {
+           name PASS_ALL;
+         }
+         route-policy out {
+           name PASS_ALL;
+         }
+       }
+     }
+   }
+ }
+ neighbor 2002:4601:102::4601:103 {
+   remote-as 65003;
+   address-family {
+     ipv6 {
+       unicast {
+         route-policy in {
+           name PASS_ALL;
+         }
+         route-policy out {
+           name PASS_ALL;

```



```

interface {
+       GigabitEthernet 1/1/1/1 {
+           service {
+               instance 1234 {
+                   ethernet;
+                   encapsulation {
+                       dot1q {
+                           id 1234;
+                       }
+                   }
+                   bridge-domain {
+                       bridge-id 321;
+                   }
+               }
+           }
+       }
+       BDI 321 {
+           vrf {
+               forwarding 0-65008740;
+           }
+           ipv6 {
+               address {
+                   prefix-list 2001:db8::1/32;
+               }
+           }
+       }
+   }
+   mpls {
+       label {
+           mode {
+               vrf 0-65008740 {
+                   protocol {
+                       all-afs per-vrf;
+                   }
+               }
+           }
+       }
+   }
+   router {
+       bgp 65001 {
+           neighbor 2001:db8::2 {
+               remote-as 65003;
+           }
+           address-family {

```

```

+         ipv6-with-vrf {
+             ipv6 unicast {
+                 vrf 0-65008740 {
+                     neighbor 2001:db8::2 {
+                         remote-as 65003;
+                         activate;
+                         ebgp-multihop {
+                             max-hop 12;
+                         }
+                         update-source {
+                             BDI 321;
+                         }
+                     }
+                 }
+             }
+         }
+     }
+     vpnv6 unicast {
+         neighbor 2001:db8::2 {
+             activate;
+         }
+     }
+ }
}

```

Service re-deploy Action

Use the service re-deploy action to re-run the service when a device comes back up after going down during the initial service creation. Use this action to enable re-deploying a service.

Re-deploying a service re-deploys the service on all the devices, or sites, or endpoints. For more information, see the ***NSO documentation***.

This section discusses how to execute the re-deploy action on the SR-TE CFP services.

SR-ODN Services

The following sample commands show how to execute the service re-deploy actions for a SR-ODN service. SR-CLI-ODN-300 is used as a sample service to demonstrate the command usage.

re-deploy

```
admin@ncs> request sr-te odn odn-template SR-CLI-ODN-300 re-deploy
```

re-deploy reconcile

```
admin@ncs> request sr-te odn odn-template SR-CLI-ODN-300 re-deploy reconcile
```

SR-Policy Services

The following sample commands show how to execute the service re-deploy actions for an SR-policy service. SR-Policy-1 is used as a sample service to demonstrate the command usage.

re-deploy

```
admin@ncs> request sr-te policies policy SR-Policy-1 re-deploy
```

re-deploy reconcile

```
admin@ncs> request sr-te policies policy SR-Policy-1 re-deploy reconcile
```

CS SR-TE Policy Services

The following sample commands show how to execute the service re-deploy actions for a cs-sr-te-policy service. CS-DYNAMIC is used as a sample policy to demonstrate the command usage.

re-deploy

```
admin@ncs> request cs-sr-te-policy CS-DYNAMIC re-deploy
```

re-deploy reconcile

```
admin@ncs> request cs-sr-te-policy CS-DYNAMIC re-deploy reconcile
```

IETF-L2VPN-NM Services

The following sample commands show how to execute the service re-deploy actions on L2VPN-NM service.

re-deploy

```
admin@ncs> request l2vpn-ntw vpn-services vpn-service l2nm-evpn re-deploy
```

re-deploy reconcile

```
admin@ncs> request l2vpn-ntw vpn-services vpn-service l2nm-evpn re-deploy reconcile
```

IETF-L3VPN-NM Services

The following sample commands show how to execute the service re-deploy actions on the L3VPN-NM service.

re-deploy

```
admin@ncs> request l3vpn-ntw vpn-services vpn-service 0-65008740 re-deploy
```

re-deploy reconcile

```
admin@ncs> request l3vpn-ntw vpn-services vpn-service 0-65008740 re-deploy
reconcile
```

re-deploy no-networking reconcile

```
admin@ncs> request l3vpn-ntw vpn-services vpn-service 0-65008740 re-deploy no-
networking true reconcile
```

Supporting MultiVendor Services - CFP Services

Cisco NSO T-SDN FP Bundle is an extendable CFP, which allows you to write your own extension package and hook it to the CFP. This allows you to map the CFP service models to device configurations for the NED packages and/or the device versions not supported by CFP. This concept of MultiVendors Support Services (MVSS) thus allows you to provide services to the devices or NED packages you already have.

T-SDN FP Bundle supports multivendors for IOSXR Netconf and IOSXE CLI NED types. For more information about the supported NED versions, see the *Cisco T-SDN FP Bundle Installation Guide*.

Note: All bootstrap files for multivendor settings for Netconf NEDs are in the `tsdn-<version>-nso-<version>/netconf_data` folder. For more information, see [Appendix E - Supporting MultiVendors for IOSXR Netconf NEDs](#).

You must configure the dynamic-device mapping for these NEDs. To create multivendor services for any other NEDs, contact your Cisco representative.

If no dynamic device mapping is configured, the plan displays an error when you try to access non-native devices. The following is an example error message for the SR-ODN service.

```
Aborted: 'cisco-sr-te-cfp-internal:cisco-sr-te-cfp-internal:sr-te policies policy
SR-CLI-ERO-VPWS-PN73-0 PN73-0': Python cb_validate error. STATUS_CODE: TSDN-SR-408
  REASON: Router NED not supported
  CATEGORY: user
  SEVERITY: ERROR
  Context [name = Router NED not supported: ned:netconf, message = Missing
dynamic device mapping
    state = {'Device': 'PN73-0', 'Service': 'SR-CLI-ERO-VPWS-PN73-0', 'Device NED
ID': 'ned:netconf'}]
```

Use the information in this section to support multivendors services for the SR-ODN service, SR-Policy service, and CS-SR-TE Policy service.

To create multivendors services, copy the multivendors package from the **packages** directory, establish the dynamic-device mapping, and create the SR-TE service on the Netconf device.

Note: Make sure the **tsdn-<version>-nso-<version>/standalone/packages/ncs-<version>-<CFP-package>-multi-vendors-<version>.tar.gz** package is installed on NSO. For more information, see the **Cisco T-SDN FP Bundle Installation Guide**.

Supporting MultiVendors for SR-TE CFP IOSXE CLI Services

This section discusses the multivendors support for IOSXE CLI NED for the SR-ODN service and the SR-Policy service.

Note: The CS SR-TE policy is not supported on IOSXE CLI NED.

To support multivendors for the IOSXE CLI services in SR-TE CFP, configure the **cisco-sr-te-cfp dynamic-device** mapping as follows. Make sure to provide the correct NED-ID.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
    <dynamic-device-mapping>
      <ned-id>cisco-ios-cli-6.86:cisco-ios-cli-6.86</ned-id>
      <python-impl-class-name>sr_te_multi_vendors.IosXE</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting MultiVendors for IETF-L2VPN-NM IOSXE CLI Services

To support multivendors for L2NM IOSXE CLI services, configure the **cisco-flat-l2vpn-fp dynamic-device** mapping as follows. Make sure to provide the correct NED-ID.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
      <dynamic-device-mapping>
        <ned-id>cisco-ios-cli-6.86:cisco-ios-cli-6.86</ned-id>
        <python-impl-class-name>flat_l2vpn_multi_vendors.IosXE</python-impl-class-name>
      </dynamic-device-mapping>
    </cfp-configurations>
  </l2vpn-ntw>
</config>
```


Supporting MultiVendors for IETF-L3VPN-NM IOSXE CLI Services

To support multivendors for L3VPN IOSXE CLI services, configure the **cisco-flat-l3vpn-fp** dynamic-device mapping as follows. Make sure to provide the correct NED-ID.

Note: Since L3NM is dependent on the L3VPN cfp-configuration dynamic-device mapping, the procedure to support multivendors is the same for both of these services.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-flat-
L3vpn">
    <dynamic-device-mapping>
      <ned-id>cisco-ios-cli-<version>:cisco-ios-cli-<version></ned-id>
      <python-impl-class-name>flat_l3vpn_multi_vendors.IosXE</python-impl-class-
name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Using Custom-Templates - CFP Services

You can configure the devices directly by using custom-templates. With custom-templates, you can also apply additional configurations that are not supported by the CFP services.

To use this feature, the **apply-custom-template** flag must be set to **true**. By default, this flag is set to **true**. The plan for the service displays the status of the custom-templates.

You can also turn this feature on/off. For more information on turning this flag on/off, see section **Turning Custom-Templates On/Off** in this chapter.

The function packages allow you to use device-templates or feature templates to define custom configurations.

Copy the **custom-templates.tar.gz** (downloaded during the SR-TE CFP installation) to the **/var/opt/ncs/packages** directory. Extract the **custom-templates.tar.gz** file and delete the file after extraction.

Custom-templates must adhere to a naming convention. The custom-templates name must start with either **ct-** or **CT-**. The device name variable in configuration templates must be either **DEVICE_NAME** or **DEVICE**. These variable names are reserved only for the device name in templates and, hence, must not be used for anything else in a template.

For sample custom-templates for each service, see the section **Sample Custom-Template Payloads for CFP Services** in this documentation.

Applying Custom-Templates

You must load merge the custom-templates to NSO before you can apply them.

Load merge the feature templates to NSO as follows:

```
sudo NCS_RELOAD_PACKAGES=true /etc/init.d/ncs restart
```

Load merge the device templates to NSO as follows. The following is an example to load merge.

```
admin@ncs% load merge user-device-template.xml
[ok]
admin@ncs% commit
[ok]
```

Applying global role-based custom-template

The following example shows how to apply a feature custom-template **ct-user-template** to a device.

```
admin@ncs% set sr-te cisco-sr-te-cfp-sr-odn:odn odn-template SR-CLI-ODN-3000
custom-template ct-user-template variable HISTORY_INFO value informational
[ok]
admin@ncs% commit
```

Applying custom-template at the service level

The following example shows how to apply the custom template **CT-Iosxr-syslog** (which is a feature template) to a device.

```
admin@ncs% set cisco-sr-te-cfp:sr-te odn odn-template SR-CLI-ODN-3000 custom-
template CT-Iosxr-syslog
[ok]
admin@ncs% commit
[ok]
```

Applying custom-template at the device level

The following example shows how to apply a device custom-template **CT-CLI-banner** to a device.

```
admin@ncs% set cisco-sr-te-cfp:sr-te odn odn-template SR-CLI-ODN-3000 custom-
template CT-CLI-banner variable BANNER_TEXT value Welcome
[ok]
admin@ncs% commit
[ok]
```

Turning Custom-Templates On/Off

You can turn the custom-templates feature on or off at any time and then redeploy the service for the change to take effect.

To turn off the custom-template feature:

Set the **apply-custom-template** flag to **false** to turn off the custom-template feature. The default value is **true**.

```
admin@ncs% set apply-custom-template false
```

```
commit
```

Sample Custom-Template Payloads for CFP Services

This topic contains sample custom-template payloads for the device templates mentioned in this documentation.

The custom-template payloads for the following topics are discussed in this topic.

- SR-ODN Services
- SR-Policy Services
- CS SR-TE Policy Services
- IETF-L2VPN-NM Services
- IETF-L3VPN-NM Services

SR-ODN Services

Custom-template for all head-ends

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <odn xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-odn">
      <odn-template>
        <name>SR-CLI-ODN-300</name>
        <custom-template>
          =====> APPLYING CUSTOM TEMPLATE AT GLOBAL LEVEL FOR ALL HEAD-ENDS
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome</value>
          </variable>
        </custom-template>
      </odn-template>
    </odn>
  </sr-te>
</config>
```

Custom-template per head-end

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
```

```

<odn xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-odn">
  <odn-template>
    <name>SR-CLI-ODN-300</name>
    <head-end>
      <name>PIOSXR-0</name>
      <custom-template>
===== > APPLYING CUSTOM TEMPLATE AT LOCAL LEVEL FOR GIVEN HEAD-END
        <name>CT-CLI-banner</name>
        <variable>
          <name>BANNER_TEXT</name>
          <value>Welcome</value>
        </variable>
      </custom-template>
    </head-end>
  </odn-template>
</odn>
</sr-te>
</config>

```

Custom-template to authorize configuration for an IOSXE device

Use the permit/deny custom-template to provide authorization to the IP addresses to use the SR-ODN service.

For example, if you load merge a payload that uses the permit custom-template, the IP addresses specified in the payload can access the SR-ODN service. Similarly, IP addresses in the payload that use the deny custom-template cannot access the SR-ODN service.

These templates can be used independently of each other.

custom-template to permit configuration

```

<template>
  <name>CT-authorize-permit</name>
<----- template for permitting access
  <ned-id>
    <id xmlns:cisco-ios-cli-6.86="http://tail-f.com/ns/ned-id/cisco-ios-cli-6.86">cisco-ios-cli-6.86:cisco-ios-cli-6.86</id>
  <config>
    <ip xmlns="urn:ios">
      <prefix-list>
        <?set i = {$Iteration_number} ?>
        <?if {$Permit_IP}?>
          <prefixes>
            <name>{$PREFIX_LIST_NAME}</name>
            <seq>
              <no>{$Iteration_number}</no>

```

```

        <?if {count($Permit_IP) > 0}?>
        <permit>
            <ip>{$Permit_IP}</ip>
        </permit>
        <?end?>
    </seq>
    <?set i={$Iteration_number+5}?>
    </prefixes>
<?end?>
</prefix-list>
</ip>
<segment-routing xmlns="urn:ios">
    <traffic-eng>
        <on-demand>
            <color>
                <id>{$COLOR}</id>
                <authorize>
                    <restrict>
                        <ipv4>
                            <prefix-list>{$PREFIX_LIST_NAME}</prefix-list>
                        </ipv4>
                    </restrict>
                </authorize>
            </color>
        </on-demand>
    </traffic-eng>
</segment-routing>
</config>
</ned-id>
</template>

```

custom-template to deny configuration

```

<template>
    <name>CT-authorize-deny</name>
<----- template for denying access
    <ned-id>
        <id xmlns:cisco-ios-cli-6.86="http://tail-f.com/ns/ned-id/cisco-ios-cli-
6.86">cisco-ios-cli-6.86:cisco-ios-cli-6.86</id>
        <config>
            <ip xmlns="urn:ios">
                <prefix-list>
                    <?set i = {$Iteration_number} ?>
                    <?if {$Deny_IP}?>
                        <prefixes>

```

```

    <name>{$PREFIX_LIST_NAME}</name>
    <seq>
      <no>{$Iteration_number}</no>
      <?if {count($Deny_IP) > 0}?>
        <deny>
          <ip>{$Deny_IP}</ip>
        </deny>
      <?end?>
    </seq>
    <?set i={$Iteration_number+5}?>
  </prefixes>
<?end?>
</prefix-list>
</ip>
<segment-routing xmlns="urn:ios">
  <traffic-eng>
    <on-demand>
      <color>
        <id>{$COLOR}</id>
        <authorize>
          <restrict>
            <ipv4>
              <prefix-list>{$PREFIX_LIST_NAME}</prefix-list>
            </ipv4>
          </restrict>
        </authorize>
      </color>
    </on-demand>
  </traffic-eng>
</segment-routing>
</config>
</ned-id>
</template>

```

SR-Policy Services

Sample custom-template payload for all head-ends

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-policies">
      <policy>
        <name>SR-CLI-DYNAMIC</name>
        <custom-template>

```

```

=====> CUSTOM TEMPLATE CONFIG FOR SR-POLICY
    <name>CT-CLI-logging</name>
  </custom-template>
  ...
</policy>
</policies>
</sr-te>
</config>

```

Sample custom-template payload per head-end

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-policies">
      <policy>
        <name>SR-CLI-DYNAMIC</name>
        <head-end>
          <name>PIOSXR-0</name>
          <custom-template>
=====>>> CUSTOM TEMPLATE CONFIG FOR SR-POLICY
              <name>CT-CLI-logging</name>
            </custom-template>
            ...
          </head-end>
        </policy>
      </policies>
    </sr-te>
  </config>

```

CS SR-TE Policy Services

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <cs-sr-te-policy xmlns="http://cisco.com/ns/nso/cfp/cisco-cs-sr-te-cfp">
    <name>CS-STATIC-EXPLICIT-SID</name>
    <custom-template>
      <name>CT-CLI-logging</name>
    </custom-template>
    <head-end>
      <custom-template>
        <name>CT-CLI-banner</name>
        <variable>
          <name>BANNER_TEXT</name>
          <value>Welcome1</value>
        </variable>
      </custom-template>
    </head-end>
  </cs-sr-te-policy>
</config>

```

```

<tail-end>
  <custom-template>
    <name>CT-CLI-banner</name>
    <variable>
      <name>BANNER_TEXT</name>
      <value>Welcome2</value>
    </variable>
  </custom-template>
</tail-end>
</cs-sr-te-policy>
</config>

```

IETF-L2VPN-NM Services

Sample service payload using custom-template

```

<l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
  <vpn-services>
    <vpn-service>
      <vpn-id>l2nm-p2p</vpn-id>
      <vpn-svc-type>vpn-common:t-ldp</vpn-svc-type>
      <control-word>yes</control-word>
      <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw">
===== > APPLYING CUSTOM TEMPLATE AT GLOBAL LEVEL
FOR ALL vpn-nodes
      <name>CT-CLI-banner</name>
      <variable>
        <name>BANNER_TEXT</name>
        <value>Welcome_A</value>
      </variable>
    </custom-template>
    <vpn-nodes>
      <vpn-node>
        <vpn-node-id>PIOSXR-0</vpn-node-id>
        <ne-id>PIOSXR-0</ne-id>
        <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw"> ===== > APPLYING CUSTOM TEMPLATE AT LOCAL LEVEL
FOR PARTICULAR vpn-node
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome_B</value>
          </variable>
        </custom-template>
        ...
      </vpn-node>
    </vpn-nodes>

```



```

    </vpn-service>
  </vpn-services>
</l2vpn-ntw>

```

IETF-L3VPN-NM Services

Sample custom-template for all head-ends

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-services>
      <vpn-service>
        <vpn-id>0-65008740</vpn-id>
        <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
ntw">
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome_A</value>
          </variable>
        </custom-template>
      <vpn-nodes>
        <vpn-node>
          <vpn-node-id>PIOSXR-1</vpn-node-id>
          <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
l3vpn-ntw">
            <name>CT-CLI-banner</name>
            <variable>
              <name>BANNER_TEXT</name>
              <value>Welcome_B</value>
            </variable>
          </custom-template>
        </vpn-node>
      </vpn-nodes>
    </vpn-service>
  </vpn-services>
</l3vpn-ntw>
</config>

```

Notifications in Core Function Pack Services

The Core Function Pack services generate notifications that communicate vital information about various events. Use the notifications to modify and examine the payload content for a specific event.

SR-TE ODN Notifications

This topic includes the create and delete notifications for SR-ODN service.

Create Notifications

```
notification {
  eventTime 2020-08-27T16:25:47.047107+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-
template[name='SR-CLI-ODN-300']
    component self
    state ready
    operation modified
    status reached
  }
}
notification {
  eventTime 2020-08-27T16:25:47.04728+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-
template[name='SR-CLI-ODN-300']
    component PIOSXR-0
    state ready
    operation modified
    status reached
  }
}
notification {
  eventTime 2020-08-27T16:25:47.04744+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-
template[name='SR-CLI-ODN-300']
    component PIOSXR-1
    state ready
    operation modified
    status reached
  }
}
```

Delete Notifications

```
notification {
  eventTime 2020-08-27T09:29:17.00977+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-
template[name='SR-CLI-ODN-300']
    operation deleted
  }
}
```

The following notification displays if the delete operation fails.

```
notification {
  eventTime 2020-08-27T09:32:01.880002+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-
template[name='SR-CLI-ODN-400']
    operation deleted
    status failed
  }
}
```

SR-TE Policy Notifications

This topic includes the create and delete notifications for SR-policies.

Create Notifications

```
notification {
  eventTime 2020-08-27T17:24:00.72731+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
    component self
    state init
    operation created
    status reached
  }
}

notification {
  eventTime 2020-08-27T17:24:00.727359+00:00
  plan-state-change {
    service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
    component self
    state ready
  }
}
```

```
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-08-27T17:24:00.727483+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        component PIOSXR-0
        state init
        operation created
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:24:00.727525+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        component PIOSXR-0
        state cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply
        operation created
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:24:00.727578+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        component PIOSXR-0
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-08-27T17:24:02.935489+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        component self
        state ready
        operation modified
    }
}
```

```

        status reached
    }
}
notification {
    eventTime 2020-08-27T17:24:02.935676+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        component PIOSXR-0
        state ready
        operation modified
        status reached
    }
}

```

Delete Notifications

```

notification {
    eventTime 2020-08-27T10:25:51.308712+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        operation deleted
    }
}

```

The following notification displays if the delete operation fails.

```

notification {
    eventTime 2020-08-27T10:26:32.958568+00:00
    plan-state-change {
        service /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy[name='SR-CLI-DYNAMIC']
        operation deleted
        status failed
    }
}

```

CS SR-TE Policy Notifications

This topic includes the create and delete notifications for CS SR-policies.

Create Notifications

```

notification {
    eventTime 2023-02-01T22:43:29.082854+00:00
    plan-state-change {
        service /cs-sr-te-policy[name='CS-STATIC-EXPLICIT-SID']
        component self
    }
}

```

```

        state ready
        operation modified
        status reached
        trace-id 03456aa6-05a4-4728-8587-3e521611fa71
    }
}
notification {
    eventTime 2023-02-01T22:43:29.082854+00:00
    plan-state-change {
        service /cs-sr-te-policy[name='CS-STATIC-EXPLICIT-SID']
        component PIOSXR-0
        state ready
        operation modified
        status reached
        trace-id 03456aa6-05a4-4728-8587-3e521611fa71
    }
}
notification {
    eventTime 2023-02-01T22:43:29.082854+00:00
    plan-state-change {
        service /cs-sr-te-policy[name='CS-STATIC-EXPLICIT-SID']
        component PIOSXR-1
        state ready
        operation modified
        status reached
        trace-id 03456aa6-05a4-4728-8587-3e521611fa71
    }
}

```

Delete Notifications

```

notification {
    eventTime 2023-02-01T22:45:05.611362+00:00
    plan-state-change {
        service /cs-sr-te-policy[name='CS-STATIC-EXPLICIT-SID']
        operation deleted
    }
}

```

The following notification displays if the delete operation fails.

```

notification {
    eventTime 2023-02-01T23:36:51.538276+00:00
    plan-state-change {
        service /cs-sr-te-policy[name='CS-STATIC-EXPLICIT-SID']
        operation deleted
        status failed
    }
}

```

```

    }
}

```

IETF-L2VPN-NM Notifications

This topic includes the create notifications, delete notifications and AA notifications for the IETF-L2VPN-NM service.

Create Notifications

```

notification {
    eventTime 2020-09-30T18:22:25.529817+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component self
        state init
        operation created
        status reached
    }
}

notification {
    eventTime 2020-09-30T18:22:25.529876+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component self
        state ready
        operation created
        status not-reached
    }
}

notification {
    eventTime 2020-09-30T18:22:25.529967+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component PIOSXR-0
        state init
        operation created
        status reached
    }
}

notification {
    eventTime 2020-09-30T18:22:25.530023+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component PIOSXR-0
        state ietf-l2vpn-ntw-nano-services:config-apply
    }
}

```

```
        operation created
        status reached
    }
}
notification {
    eventTime 2020-09-30T18:22:25.530069+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component PIOSXR-0
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-09-30T18:22:25.530195+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component PIOSXR-1
        state init
        operation created
        status reached
    }
}
notification {
    eventTime 2020-09-30T18:22:25.53025+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component PIOSXR-1
        state ietf-l2vpn-ntw-nano-services:config-apply
        operation created
        status reached
    }
}
notification {
    eventTime 2020-09-30T18:22:25.530288+00:00
    plan-state-change {
        service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
        component PIOSXR-1
        state ready
        operation created
        status not-reached
    }
}
```



```

notification {
  eventTime 2020-09-30T18:22:32.260433+00:00
  plan-state-change {
    service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
    component self
    state ready
    operation modified
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:22:32.260575+00:00
  plan-state-change {
    service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
    component PIOSXR-0
    state ready
    operation modified
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:22:32.260718+00:00
  plan-state-change {
    service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
    component PIOSXR-1
    state ready
    operation modified
    status reached
  }
}

```

Delete Notifications

```

notification {
  eventTime 2020-09-23T14:10:15.910353+00:00
  plan-state-change {
    service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
    operation deleted
  }
}

```

The following notification displays if the delete operation fails.

```

notification {
  eventTime 2020-09-23T14:11:35.55326+00:00
  plan-state-change {

```

```

    service /l2vpn-ntw/vpn-services/vpn-service[vpn-id='l2nm-p2p']
    operation deleted
    status failed
  }
}

```

AA Notifications

Any configuration change in the container **<service-assurance>** in the service payload triggers notifications for the service.

```

notification {
  eventTime 2024-04-01T20:11:56.355257+00:00
  service-assurance-config-change {
    service /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service[vpn-id=l2nm-p2p-y-
1731]
    monitoring-state enable
    operation created
    notif-type service-assurance
  }
}

notification {
  eventTime 2024-04-01T20:12:43.58002+00:00
  service-assurance-config-change {
    service /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service[vpn-id=l2nm-p2p-y-1731]
    monitoring-state enable
    operation modified
    notif-type service-assurance
  }
}

notification {
  eventTime 2024-05-31T05:25:03.281177+00:00
  service-assurance-config-change {
    service /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service[vpn-id=l2nm-p2p-y-
1731]
    operation deleted
    notif-type service-assurance
  }
}

```

IETF-L3VPN-NM Notifications

This topic includes the creation of notifications, deletion of notifications, and AA notifications for the IETF-L3VPN-NM service.

Create Notification

```
notification {
  eventTime 2020-09-30T18:37:07.675828+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component self
    state init
    operation created
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.675894+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component self
    state ietf-l3vpn-ntw-nano-services:config-apply
    operation created
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.675964+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component self
    state ready
    operation created
    status not-reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.676077+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component PIOSXR-0
    state init
    operation created
    status reached
  }
}
```

```
}
notification {
  eventTime 2020-09-30T18:37:07.676146+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component PIOSXR-0
    state ietf-l3vpn-ntw-nano-services:config-apply
    operation created
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.676205+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component PIOSXR-0
    state ready
    operation created
    status not-reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.676396+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component PIOSXR-1
    state init
    operation created
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.676455+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    component PIOSXR-1
    state ietf-l3vpn-ntw-nano-services:config-apply
    operation created
    status reached
  }
}
notification {
  eventTime 2020-09-30T18:37:07.676505+00:00
  plan-state-change {
```

```
        service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
        component PIOSXR-1
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-09-30T18:37:13.605614+00:00
    plan-state-change {
        service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
        component self
        state ready
        operation modified
        status reached
    }
}
notification {
    eventTime 2020-09-30T18:37:13.605788+00:00
    plan-state-change {
        service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
        component PIOSXR-0
        state ready
        operation modified
        status reached
    }
}
notification {
    eventTime 2020-09-30T18:37:13.605938+00:00
    plan-state-change {
        service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
        component PIOSXR-1
        state ready
        operation modified
        status reached
    }
}
```

Delete Notifications

```
notification {
    eventTime 2020-09-23T13:52:53.159386+00:00
    plan-state-change {
        service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
        operation deleted
    }
}
```

```

    }
}

```

The following notification displays if the delete operation fails.

```

notification {
  eventTime 2020-09-23T13:53:50.561503+00:00
  plan-state-change {
    service /l3vpn-ntw/vpn-services/vpn-service[vpn-id='0-65008740']
    operation deleted
    status failed
  }
}

```

AA Notifications

Any configuration change in the container **<service-assurance>** in the service payload triggers notifications for the service.

```
admin@ncs% run show notification stream service-aa-changes
```

```

notification {
  eventTime 2024-04-01T20:06:24.167346+00:00
  service-assurance-config-change {
    service /l3nm:l3vpn-ntw/vpn-services/vpn-service[vpn-id=0-65008740]
    monitoring-state enable
    operation created
    notif-type service-assurance
  }
}
notification {
  eventTime 2024-04-01T20:07:42.249911+00:00
  service-assurance-config-change {
    service /l3nm:l3vpn-ntw/vpn-services/vpn-service[vpn-id=0-65008740]
    monitoring-state enable
    operation modified
    notif-type service-assurance
  }
}
notification {
  eventTime 2024-04-01T20:08:32.146892+00:00
  service-assurance-config-change {
    service /l3nm:l3vpn-ntw/vpn-services/vpn-service[vpn-id=0-65008740]
    operation deleted
    notif-type service-assurance
  }
}
}

```

```

notification {
  eventTime 2024-04-01T20:06:24.167407+00:00
  service-assurance-config-change {
    service /l3nm:l3vpn-ntw/vpn-services/vpn-service[vpn-id=0-65008740]
    operation created
    notif-type probes
  }
}
notification {
  eventTime 2024-04-01T20:07:42.249956+00:00
  service-assurance-config-change {
    service /l3nm:l3vpn-ntw/vpn-services/vpn-service[vpn-id=0-65008740]
    operation modified
    notif-type probes
  }
}
admin@ncs% run show notification stream service-aa-changes
notification {
  eventTime 2024-04-01T20:08:32.146938+00:00
  service-assurance-config-change {
    service /l3nm:l3vpn-ntw/vpn-services/vpn-service[vpn-id=0-65008740]
    operation deleted
    notif-type probes
  }
}

```

Deleting Core Function Pack Services

You may want to delete a service when the devices in the service are down or when there is a policy discrepancy. Use the following commands to delete the services.

Delete all SR-ODN services

```
admin@ncs> delete sr-te odn odn-template
```

Delete all SR-Policy services

```
admin@ncs> delete sr-te policies policy
```

Delete all CS SR-TE Policy services

```
admin@ncs> delete cs-sr-te-policy
```

Delete all IETF-L2VPN-NM services

```
admin@ncs> delete l2vpn-ntw vpn-services vpn-service
```

Delete all IETF-L3VPN-NM services

```
admin@ncs> delete l3vpn-ntw vpn-services vpn-service
```

Working with Example Function Packs

IETF-TE, IETF Network Service Slice, and Cisco Performance Measurement Example Function Packs illustrate how SR-TE policies are used to instantiate a service with T-SDN FPs. It is optional to install/use these Example Function Packs.

The IETF-TE implementation uses only the IETF model and not the native model.

For more information on how to install these Example Function Packs, see the **Cisco T-SDN FP Bundle Installation Guide**.

For information about the yang models, see **Appendix C - Yang Models** in this documentation.

Creating IETF-TE Services

Create the IETF-TE service to push the Resource Reservation Protocol (RSVP) - TE configurations to devices. RSVP-TE is an extension of RSVP (a standardized IETF protocol) for traffic engineering. With this configuration, you can set up a bidirectional tunnel with both a source and destination for the network traffic.

This function pack uses the IETF-TE model with some modifications for the T-SDN FP Bundle. It implements the **draft-ietf-teas-yang-te-30.txt** model.

The unnumbered IPV4 Loopback0 → Loopback interface configuration is considered the first configuration on the device. While creating the IETF service, the device loopback interface is queried based on the source IP address. If a corresponding loopback is found, it is used for tunnel configuration. If no loopback with the source IP address is configured, an error is displayed. The following is an example error message.

```
Aborted: Python cb_pre_modification error. STATUS_CODE: TSDN-RSVP-TE-404
REASON: Input element's value is not supported
CATEGORY: validation
SEVERITY: ERROR
Context [name = Pre-modification, message = Loopback not set
state = {'Head-end': 'PIOSXR-0', 'Service': '/cisco-rsvp-te-fp:rsvp-te/tunnel-
te{IETF-RSVP-TE-111.1.1.1-internal}'}]
```

To create IETF-TE service:

1. The following is a sample payload to create the IETF-TE service. For the yang model, see **Appendix C - Yang Models** in this documentation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <te xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
    <tunnels>
      <tunnel>
        <name>IETF-RSVP-TE</name>
```



```

    <identifier>1234</identifier>
    <description>RSVP_TE</description>
    <source>111.1.1.1</source>
    <head-end xmlns="http://cisco.com/ns/nso/fp/examples/cisco-te">PIOSXR-
0</head-end>
    <destination>222.2.2.2</destination>
    <tail-end xmlns="http://cisco.com/ns/nso/fp/examples/cisco-te">PIOSXR-
1</tail-end>
    <setup-priority>3</setup-priority>
    <hold-priority>2</hold-priority>
    <te-bandwidth>
      <generic>94967295</generic>
    </te-bandwidth>
    <signaling-type xmlns:te-types="urn:ietf:params:xml:ns:yang:ietf-te-
types">te-types:path-setup-rsvp</signaling-type>
    <primary-paths>
      <primary-path>
        <name>PATH-1</name>
        <path-computation-method xmlns:te-
types="urn:ietf:params:xml:ns:yang:ietf-te-types">te-types:path-explicitly-
defined</path-computation-method>
        <preference>1</preference>
        <explicit-route-objects-always>
          <route-object-include-exclude>
            <index>1</index>
            <numbered-node-hop>
              <node-id>1.1.1.1</node-id>
              <hop-type>loose</hop-type>
            </numbered-node-hop>
          </route-object-include-exclude>
          <route-object-include-exclude>
            <index>2</index>
            <label-hop>
              <te-label>
                <generic>Afw=</generic>
              </te-label>
            </label-hop>
          </route-object-include-exclude>
        </explicit-route-objects-always>
      </primary-path>
      <primary-path>
        <name>PATH-2</name>
        <path-computation-method xmlns:te-
types="urn:ietf:params:xml:ns:yang:ietf-te-types">te-types:path-locally-
computed</path-computation-method>
        <optimizations>

```

```

        <optimization-metric>
            <metric-type xmlns:te-types="urn:ietf:params:xml:ns:yang:ietf-
te-types">te-types:path-metric-te</metric-type>
        </optimization-metric>
    </optimizations>
    <preference>2</preference>
</primary-path>
<primary-path>
    <name>PATH-3</name>
    <path-computation-method xmlns:te-
types="urn:ietf:params:xml:ns:yang:ietf-te-types">te-types:path-externally-
queried</path-computation-method>
    <optimizations>
        <optimization-metric>
            <metric-type xmlns:te-types="urn:ietf:params:xml:ns:yang:ietf-
te-types">te-types:path-metric-te</metric-type>
        </optimization-metric>
    </optimizations>
    <preference>3</preference>
</primary-path>
<primary-path>
    <name>PATH-4</name>
    <path-computation-method xmlns:te-
types="urn:ietf:params:xml:ns:yang:ietf-te-types">te-types:path-externally-
queried</path-computation-method>
    <optimizations>
        <optimization-metric>
            <metric-type xmlns:te-types="urn:ietf:params:xml:ns:yang:ietf-
te-types">te-types:path-metric-te</metric-type>
        </optimization-metric>
    </optimizations>
    <preference>4</preference>
</primary-path>
</primary-paths>
</tunnel>
</tunnels>
</te>
</config>

```

2. Display the plan and verify the service is in the **ready reached** state. The following table describes the IETF-TE plan components.

Plan Component	Value	Description

Type	Source/destination	This component represents the head-end/tail-end of the IETF-TE service.
State	ietf-te-fp-tunnel-nano-plan-services:config-apply	This represents acceptance of service intent and respective configuration application on the device in NSO CDB.

```
admin@ncs% run show te tunnels tunnel-plan plan
```

TYPE	NAME	BACK		STATUS		STATUS	WHEN	POST ACTION	
		TRACK	GOAL	CODE	STATE			ref	STATUS
self	self	false	-	-	init ready	reached	2020-08-24T23:45:50	-	-
source	111.1.1.1	false	-	-	init ready	reached	2020-08-24T23:45:50	-	-
					ietf-te-fp-tunnel-nano-plan-services:config-apply	reached	2020-08-24T23:45:50	-	-
destination	222.2.2.2	false	-	-	init ready	reached	2020-08-24T23:45:50	-	-
					ietf-te-fp-tunnel-nano-plan-services:config-apply	reached	2020-08-24T23:45:50	-	-

3. Verify the configuration is pushed on the devices.

Creating IETF Network Slice Services

Network slicing creates several logical networks and virtual networks in a shared multi-domain infrastructure. Network slicing helps to customize network specifications for tasks and thereby increase network capacity and performance. With network slicing, you can provision services to multiple users in a shared network infrastructure. Slicing a network allows service providers to create, use, and manage various network architectures to deliver applications optimally.

The IETF Network Slice Service (NSS) Example Function Pack allows you to manage the network slices in your network infrastructure. The IETF NSS Example Function Pack implements the **draft-ietf-teas-ietf-network-slice-nbi-yang-02** model. For more information about this model, see the respective IETF model documentation.

Note: The IOSXE CLI NED and the Netconf NED are not supported in the Network Slice Service.

An IETF Network Slice service is specified as a set of Service Demarcation Points (SDPs). An SDP is an endpoint involved in the IETF NSS and belongs to a single slice service. An IETF NSS consists of two or more SDPs. A set of SDPs form a connectivity construct with a communication type that defines the traffic flows between SDPs. The communication type can be point-to-point, point-to-multipoint, or any-to-any. A group of connectivity constructs is a connection group.

IETF Network Slice Service abstracts the following underlying configuration details of the various machinery components required to realize a transport slice:

- The connectivity details drive the VPN configuration.
- The forwarding plane policy details drive the segment routing traffic engineering configurations and BGP prefix coloring.

- The QoS profile details drive the ingress marking (for Per-Hop Behaviors treatment) and the egress scheduling.

IETF NSS uses Service Level Objective-Service Level Expectations templates (slo-sle-templates) container to maintain common network slice templates that apply to one or more IETF NSS. The slo-sle-template is the catalog that the slicing service instances use. The template consists of the following parameters:

- **Forwarding-plane-policy:** This policy references a pre-configured SR-ODN service referenced by the slice services. The slice services use the slo-sle-template definition that contains the forwarding-plane-policy definition. The forwarding-plane-policy uses the **as-is** and **as-blueprint** types in the policy. The **as-is** type uses the policy for slice services using the slo-sle-template. With the policy type **as-blueprint**, the slo-sle-template creates a copy of the policy with a new allocated color. The slice services then use the policy when the slo-sle-template is attached.
- **Custom policy:** The slice service creates and uses a copy of the forwarding plane-policy with a new allocated color. To use a custom policy type, the forwarding-plane-policy-type must be **as-blueprint**. A custom policy type allows:
 - Configuring performance-measurement via **/network-slice-services/slo-sle-templates/slo-sle-template/cisco-nss:service-assurance/performance-measurement**
 - Configuring bandwidth via **/network-slice-services/slice-service/connection-groups/connection-group/service-slo-sle-policy/cisco-nss:bandwidth**
- **QoS:** The qos-policy parameter defined in the slo-sle-template references the policies configured using the qos-catalog. It is assumed the qos-catalog are policy-maps that are pre-configured on the device. The qos-catalog can be defined for both L2NM and L3NM services, where the slice service chooses the QoS policy based on the configured service type.
- **Service assurance:** Certain criteria are set in service assurance to meet the optimal performance requirements in a network infrastructure. The following criteria are used in service assurance:
 - **Heuristics:** Heuristics allows you to configure service-assurance monitoring-state, preservation, and profile-name or rule-name for L2NM and L3NM service slice types. The monitoring-state can be overridden at the service level when creating the slice service. L2NM allows you to define the profile-name or rule-name based on the connectivity-type (point-to-point or multipoint). The slice service chooses the appropriate profile or rules to instantiate the slice service instance.
- **Ethernet-service-oam:** This OAM configures Y.1731 for L2NM point-to-point slice services.
- The maintenance endpoint (MEP) helps to detect the connectivity between two endpoints in the network. The mep-id is pulled from the resource pool defined in **/network-slice-services/cfp-configurations/mep-id-pool-name**. You can view the allocated mep-ids in **/network-slice-services/slice-service-plan/plan/status/mep-id-allocation-data**. A set of MEPs configured with a same identifier is defined as a

Maintenance Association Identifier (MAID). The MAID is configured using the **/network-slice-services/slice-service/service-id**.

The following is a sample payload of the slo-sle-template.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-service">
    <slo-sle-templates>
      <slo-sle-template>
        <id>Dedicated</id>
        <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">
          <L2>
            <input-policy>L2-input</input-policy>
            <output-policy>L2-output</output-policy>
          </L2>
          <L3>
            <input-policy>input-qos</input-policy>
          </L3>
        </qos-policy>
        <forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">min-delay</forwarding-plane-policy>
      </slo-sle-template>
      <slo-sle-template>
        <id>min-delay-all-as-blueprint</id>
        <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">
          <L2>
            <input-policy>L2-input</input-policy>
            <output-policy>L2-output</output-policy>
          </L2>
          <L3>
            <input-policy>input-qos</input-policy>
          </L3>
        </qos-policy>
        <forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">min-delay-all</forwarding-plane-policy>
        <forwarding-plane-policy-type xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">as-blueprint</forwarding-plane-policy-type>
      </slo-sle-template>
      <slo-sle-template>
        <id>min-delay-all-custom</id>
        <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">
          <L2>
            <input-policy>L2-input</input-policy>
```

```

        <output-policy>L2-output</output-policy>
    </L2>
    <L3>
        <input-policy>input-qos</input-policy>
    </L3>
</qos-policy>
<forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">min-delay</forwarding-plane-policy>
<forwarding-plane-policy-type
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">as-
blueprint</forwarding-plane-policy-type>
<custom xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service"/>
<service-assurance xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
    <performance-measurement>
        <sr-te>
            <pm-profile>PM-CLI</pm-profile>
            <liveness-detection>
                <profile>SrPolicyLiveness2</profile>
            </liveness-detection>
        </sr-te>
    </performance-measurement>
</service-assurance>
</slo-sle-template>
<slo-sle-template>
    <id>Shared</id>
    <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
        <L3>
            <input-policy>input-qos</input-policy>
            <output-policy>output-qos</output-policy>
        </L3>
    </qos-policy>
    <service-assurance xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
        <heuristics>
            <monitoring-state>enable</monitoring-state>
            <preservation>remove</preservation>
        </heuristics>
        <L2>
            <point-to-point>
                <profile-name>Rule-L2VPN-NM-P2P system</profile-name>
                <rule-name>Rule-L2VPN-NM-P2P system</rule-name>
            </point-to-point>
            <multipoint>
                <profile-name>Gold_L2VPN_ConfigProfile system</profile-name>
            </multipoint>
        </L2>
    </service-assurance>
</slo-sle-template>

```

```

        <rule-name>Rule-L2VPN-NM system</rule-name>
    </multipoint>
</L2>
<L3>
    <profile-name>Gold_L3VPN_ConfigProfile system</profile-name>
    <rule-name>Rule-L3VPN-NM system</rule-name>
</L3>
</heuristics>
</service-assurance>
</slo-sle-template>
<slo-sle-template>
    <id>P2P</id>
    <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
        <L2>
            <input-policy>L2-input</input-policy>
            <output-policy>L2-output</output-policy>
        </L2>
    </qos-policy>
    <forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">igp</forwarding-plane-policy>
    <service-assurance xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
        <heuristics>
            <monitoring-state>pause</monitoring-state>
            <preservation>preserve</preservation>
        </L2>
            <point-to-point>
                <profile-name>Silver_L2VPN_ConfigProfile system</profile-name>
                <rule-name>Rule-L2VPN-NM-P2P system</rule-name>
            </point-to-point>
            <multipoint>
                <profile-name>Gold_L2VPN_ConfigProfile system</profile-name>
                <rule-name>Rule-L2VPN-NM system</rule-name>
            </multipoint>
        </L2>
    </heuristics>
</ethernet-service-oam>
    <md-name>EVC</md-name>
    <md-level>4</md-level>
    <y-1731>
        <profile-delay>delay</profile-delay>
        <profile-loss>loss</profile-loss>
        <profile-synthetic-loss>synthetic-loss</profile-synthetic-loss>
    </y-1731>

```

```

    </ethernet-service-oam>
  </service-assurance>
</slo-sle-template>
<slo-sle-template>
  <id>P2P-as-blueprint</id>
  <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
    <L2>
      <input-policy>L2-input</input-policy>
      <output-policy>L2-output</output-policy>
    </L2>
  </qos-policy>
  <forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">igp</forwarding-plane-policy>
  <forwarding-plane-policy-type
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">as-
blueprint</forwarding-plane-policy-type>
  <service-assurance xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
    <heuristics>
      <monitoring-state>pause</monitoring-state>
      <preservation>preserve</preservation>
      <L2>
        <point-to-point>
          <profile-name>Silver_L2VPN_ConfigProfile system</profile-name>
          <rule-name>Rule-L2VPN-NM-P2P system</rule-name>
        </point-to-point>
        <multipoint>
          <profile-name>Gold_L2VPN_ConfigProfile system</profile-name>
          <rule-name>Rule-L2VPN-NM system</rule-name>
        </multipoint>
      </L2>
    </heuristics>
  <ethernet-service-oam>
    <md-name>EVC</md-name>
    <md-level>4</md-level>
    <y-1731>
      <profile-delay>delay</profile-delay>
      <profile-loss>loss</profile-loss>
      <profile-synthetic-loss>synthetic-loss</profile-synthetic-loss>
    </y-1731>
  </ethernet-service-oam>
</service-assurance>
</slo-sle-template>
<slo-sle-template>

```



```

<id>P2P-custom</id>
<qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
  <L2>
    <input-policy>L2-input</input-policy>
    <output-policy>L2-output</output-policy>
  </L2>
</qos-policy>
<forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">igp</forwarding-plane-policy>
<forwarding-plane-policy-type
xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service">as-
blueprint</forwarding-plane-policy-type>
<custom xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service"/>
<service-assurance xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
  <heuristics>
    <monitoring-state>pause</monitoring-state>
    <preservation>preserve</preservation>
    <L2>
      <point-to-point>
        <profile-name>Silver_L2VPN_ConfigProfile system</profile-name>
        <rule-name>Rule-L2VPN-NM-P2P system</rule-name>
      </point-to-point>
      <multipoint>
        <profile-name>Gold_L2VPN_ConfigProfile system</profile-name>
        <rule-name>Rule-L2VPN-NM system</rule-name>
      </multipoint>
    </L2>
  </heuristics>
<ethernet-service-oam>
  <md-name>EVC</md-name>
  <md-level>4</md-level>
  <y-1731>
    <profile-delay>delay</profile-delay>
    <profile-loss>loss</profile-loss>
    <profile-synthetic-loss>synthetic-loss</profile-synthetic-loss>
  </y-1731>
</ethernet-service-oam>
<performance-measurement>
  <sr-te>
    <pm-profile>PM-CLI</pm-profile>
    <liveness-detection>
      <profile>SrPolicyLiveness2</profile>
    </liveness-detection>
  </sr-te>
</performance-measurement>

```

```

        </sr-te>
    </performance-measurement>
</service-assurance>
</slo-sle-template>
<slo-sle-template>
    <id>P2P_v6</id>
    <qos-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
        <L2>
            <input-policy>L2-input</input-policy>
            <output-policy>L2-output</output-policy>
        </L2>
    </qos-policy>
    <forwarding-plane-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">igp_v6</forwarding-plane-policy>
</slo-sle-template>
    <qos-catalog xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
        <L2>
            <output-qos-policy>
                <name>L2-output</name>
            </output-qos-policy>
            <input-qos-policy>
                <name>L2-input</name>
            </input-qos-policy>
        </L2>
        <L3>
            <output-qos-policy>
                <name>output-qos</name>
            </output-qos-policy>
            <input-qos-policy>
                <name>input-qos</name>
            </input-qos-policy>
        </L3>
    </qos-catalog>
</slo-sle-templates>
</network-slice-services>
</config>

```

In a sliced network infrastructure, you can isolate network slices to allow coexistence of multiple slices and to share the same network infrastructure. However, for network slice isolation, the performance of each network slice must not impact the performance of another slice in the infrastructure.

The **service-isolation-dedicated (default)** allows all nodes in a dedicated slice to communicate with all other nodes in a shared slice. The dedicated slice service isolation supports the any-to-any and hub-spoke communication types.

The **service-isolation-shared** allows nodes to communicate with each other only within the slice. The nodes in one slice cannot communicate with the nodes in another slice. The shared slice service isolation supports only the any-to-any communication type.

IETF Network Slicing supports and internally creates the L2NM and L3NM services.

The route distinguishers (used to distinguish routes from one another) are configured via **/network-slice-services/cisco-nss:global-settings/rd-type**. With an auto route distinguisher, the slice service pushes the auto route distinguisher configuration to the device. With a manual route distinguisher, the configuration is manually pushed to the device under the slice service or per SDP.

The following are the connectivity types used in the service topologies to define the route targets assigned to each site.

Note: When NSS is created with L2NM service, modifying the connectivity type after creating the service is currently not supported.

- **any-to-any:** This is a full mesh topology in which all sites can communicate with one another. The same route-targets are assigned across all nodes.

This connectivity type imports or exports the same vpn-target across all defined nodes. For L2NM service, the vpn-targets are assigned per Ethernet Virtual Interface (evi) where the evi-id is auto allocated via resource pools.

- **hub-spoke:** This topology provides hub-and-spoke multipoint connectivity to the sites. In this topology, route targets are assigned so that the spokes can communicate with the hub but not with each other. A spoke is defined if a node-role is defined for a given site.

In this connectivity type, you can define roles via the connection-group-sdp-role parameter. Since only node level roles are supported, the Service Demarcation Point (SDP) defined on similar nodes must also have the same connection-group-sdp-role defined. The SDP delivers IETF NSS to a customer through a service provider. The hubs import or export the hub vpn-target and import the spoke vpn-target. Spokes export the spoke vpn-target and import the hub vpn-target.

- **Point-to-point:** In this topology, you must manually define the route-targets for each site.

This connectivity type is applicable only to the L2NM VPWS-EVPN service type. This connectivity type does not support attaching to shared slices. The parent-rr-route-policy can be configured via **/network-slice-services/cisco-nss:global-settings/parent-rr-route-policy**.

Note: Do not use the reserved policy name PASS_ALL as the L2 parent-rr-route-policy.

The following topics discuss the implementation of each topology with sample payloads.

Implementing IETF NSS in L2-any-to-any Shared Topology

The following sample payload shows the implementation of IETF NSS in L2-any-to-any shared topology. Display the service plan to view the configuration for the service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-service">
    <slice-service>
      <service-id>L2-ANY-TO-ANY</service-id>
      <service-tags>
        <tag-type>
          <tag-type>service-tag-customer</tag-type>
          <value>cisco</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-service</tag-type>
          <value>L2</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-opaque</tag-type>
          <value>OPAQUE_VAL</value>
        </tag-type>
      </service-tags>
      <slo-sle-template>Shared</slo-sle-template>
      <isolation xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service" xmlns:ietf-nss="urn:ietf:params:xml:ns:yang:ietf-network-slice-service">ietf-nss:service-isolation-shared</isolation>
      <sdps>
        <sdp>
          <sdp-id>0</sdp-id>
          <node-id>PIOSXR-0</node-id>
          <service-match-criteria>
            <match-criterion>
              <index>0</index>
              <target-connection-group-id>matrix1</target-connection-group-id>
            </match-criterion>
          </service-match-criteria>
          <attachment-circuits>
            <attachment-circuit>
              <ac-id>0</ac-id>
              <ac-tp-id>GigabitEthernet0</ac-tp-id>
              <ac-tags>
                <ac-tags>
```

```

        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>100</value>
      </ac-tags>
    </ac-tags>
  </attachment-circuit>
</attachment-circuits>
</sdp>
<sdp>
  <sdp-id>1</sdp-id>
  <node-id>PIOSXR-1</node-id>
  <service-match-criteria>
    <match-criterion>
      <index>0</index>
      <target-connection-group-id>matrix1</target-connection-group-id>
    </match-criterion>
  </service-match-criteria>
  <attachment-circuits>
    <attachment-circuit>
      <ac-id>0</ac-id>
      <ac-tp-id>GigabitEthernet1</ac-tp-id>
      <ac-tags>
        <ac-tags>
          <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
          <value>100</value>
        </ac-tags>
      </ac-tags>
    </attachment-circuit>
  </attachment-circuits>
</sdp>
</sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type>any-to-any</connectivity-type>
  </connection-group>
</connection-groups>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L2-hub-spoke Dedicated Topology

The following sample payload shows the implementation of IETF NSS in L2-hub-spoke dedicated topology. Display the service plan to view the configuration for the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">
    <slice-service>
      <service-id>L2-HUB-SPOKE</service-id>
      <service-tags>
        <tag-type>
          <tag-type>service-tag-customer</tag-type>
          <value>cisco</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-service</tag-type>
          <value>L2</value>
        </tag-type>
      </service-tags>
      <slo-sle-template>Dedicated</slo-sle-template>
      <sdps>
        <sdp>
          <sdp-id>0</sdp-id>
          <node-id>PIOSXR-2</node-id>
          <service-match-criteria>
            <match-criterion>
              <index>0</index>
              <target-connection-group-id>matrix1</target-connection-group-id>
            </match-criterion>
          </service-match-criteria>
          <attachment-circuits>
            <attachment-circuit>
              <ac-id>0</ac-id>
              <ac-tp-id>GigabitEthernet0</ac-tp-id>
              <ac-tags>
                <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
                <value>100</value>
              </ac-tags>
            </attachment-circuit>
          </attachment-circuits>
        </sdp>
        <sdp>
          <sdp-id>1</sdp-id>
          <node-id>PIOSXR-1</node-id>
          <service-match-criteria>
            <match-criterion>

```

```

        <index>0</index>
        <target-connection-group-id>matrix1</target-connection-group-id>
        <connection-group-sdp-role xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:spoke-
role</connection-group-sdp-role>
    </match-criterion>
</service-match-criteria>
<attachment-circuits>
  <attachment-circuit>
    <ac-id>0</ac-id>
    <ac-tp-id>GigabitEthernet1</ac-tp-id>
    <ac-tags>
      <ac-tags>
        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>100</value>
      </ac-tags>
    </ac-tags>
  </attachment-circuit>
</attachment-circuits>
</sdp>
</sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type>hub-spoke</connectivity-type>
  </connection-group>
</connection-groups>
<shared xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
  <slice>
    <service-id>L2-ANY-TO-ANY</service-id>
  </slice>
  <single-sided-control/>
</shared>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L2-P2P Topology

The following sample payload shows the implementation of IETF NSS in L2-P2P topology. Display the service plan to view the configuration for the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">

```

```

<slice-service>
  <service-id>L2-P2P</service-id>
  <service-tags>
    <tag-type>
      <tag-type>service-tag-customer</tag-type>
      <value>cisco</value>
    </tag-type>
    <tag-type>
      <tag-type>service-tag-service</tag-type>
      <value>L2</value>
    </tag-type>
  </service-tags>
  <slo-sle-template>P2P</slo-sle-template>
  <sdps>
    <sdp>
      <sdp-id>0</sdp-id>
      <node-id>PIOSXR-0</node-id>
      <service-match-criteria>
        <match-criterion>
          <index>0</index>
          <target-connection-group-id>matrix1</target-connection-group-id>
        </match-criterion>
      </service-match-criteria>
      <attachment-circuits>
        <attachment-circuit>
          <ac-id>0</ac-id>
          <ac-tp-id>GigabitEthernet0</ac-tp-id>
          <ac-tags>
            <ac-tags>
              <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
              <value>100</value>
            </ac-tags>
          </ac-tags>
        </attachment-circuit>
      </attachment-circuits>
    </sdp>
    <sdp>
      <sdp-id>1</sdp-id>
      <node-id>PIOSXR-1</node-id>
      <service-match-criteria>
        <match-criterion>
          <index>0</index>
          <target-connection-group-id>matrix1</target-connection-group-id>
        </match-criterion>
      </service-match-criteria>
    </sdp>
  </sdps>
</slice-service>

```



```

    </service-match-criteria>
  <attachment-circuits>
    <attachment-circuit>
      <ac-id>0</ac-id>
      <ac-tp-id>GigabitEthernet1</ac-tp-id>
      <ac-tags>
        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>100</value>
      </ac-tags>
    </attachment-circuit>
  </attachment-circuits>
</sdp>
</sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type>point-to-point</connectivity-type>
  </connection-group>
</connection-groups>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L2-P2PV6-Topology

The following sample payload shows the implementation of IETF NSS in the L2-P2PV6 topology. Display the service plan to view the configuration for the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-service">
    <slice-service>
      <service-id>L2-P2P</service-id>
      <service-tags>
        <tag-type>
          <tag-type>service-tag-customer</tag-type>
          <value>cisco</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-service</tag-type>
          <value>L2</value>
        </tag-type>
      </service-tags>
      <slo-sle-template>P2P_v6</slo-sle-template>
    </slice-service>
  </network-slice-services>
</config>

```

```

<sdps>
  <sdp>
    <sdp-id>0</sdp-id>
    <node-id>PIOSXR-0</node-id>
    <service-match-criteria>
      <match-criterion>
        <index>0</index>
        <target-connection-group-id>matrix1</target-connection-group-id>
      </match-criterion>
    </service-match-criteria>
    <attachment-circuits>
      <attachment-circuit>
        <ac-id>0</ac-id>
        <ac-tp-id>GigabitEthernet0</ac-tp-id>
        <ac-tags>
          <ac-tags>
            <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
            <value>100</value>
          </ac-tags>
        </ac-tags>
      </attachment-circuit>
    </attachment-circuits>
  </sdp>
  <sdp>
    <sdp-id>1</sdp-id>
    <node-id>PIOSXR-1</node-id>
    <service-match-criteria>
      <match-criterion>
        <index>0</index>
        <target-connection-group-id>matrix1</target-connection-group-id>
      </match-criterion>
    </service-match-criteria>
    <attachment-circuits>
      <attachment-circuit>
        <ac-id>0</ac-id>
        <ac-tp-id>GigabitEthernet1</ac-tp-id>
        <ac-tags>
          <ac-tags>
            <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
            <value>100</value>
          </ac-tags>
        </ac-tags>
      </attachment-circuit>
    </attachment-circuits>
  </sdp>

```

```

    </sdp>
  </sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type>point-to-point</connectivity-type>
  </connection-group>
</connection-groups>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L3-any-to-any Shared Topology

The following sample payload shows the implementation of IETF NSS in the L3-any-to-any shared topology. Display the service plan to view the configuration for the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">
    <slice-service>
      <service-id>L3-Shared</service-id>
      <service-tags>
        <tag-type>
          <tag-type>service-tag-service</tag-type>
          <value>L3</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-customer</tag-type>
          <value>cisco</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-opaque</tag-type>
          <value>OPAQUE_VAL</value>
        </tag-type>
      </service-tags>
      <slo-sle-template>Shared</slo-sle-template>
      <isolation xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service" xmlns:ietf-nss="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">ietf-nss:service-isolation-shared</isolation>
      <sdps>
        <sdp>
          <sdp-id>0</sdp-id>
          <node-id>PIOSXR-0</node-id>
          <service-match-criteria>

```

```

    <match-criterion>
      <index>0</index>
      <target-connection-group-id>matrix1</target-connection-group-id>
    </match-criterion>
  </service-match-criteria>
<attachment-circuits>
  <attachment-circuit>
    <ac-id>0</ac-id>
    <ac-tp-id>GigabitEthernet0</ac-tp-id>
    <ac-ip-address>11.0.0.1</ac-ip-address>
    <ac-ip-prefix-length>24</ac-ip-prefix-length>
    <ac-tags>
      <ac-tags>
        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>100</value>
      </ac-tags>
    </ac-tags>
    <sdp-peering>
      <protocol>
        <protocol-type>peering-protocol-bgp</protocol-type>
        <bgp-attributes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
          <neighbor>11.0.0.2</neighbor>
          <remote-as>100</remote-as>
        </bgp-attributes>
      </protocol>
    </sdp-peering>
  </attachment-circuit>
</attachment-circuits>
</sdp>
<sdp>
  <sdp-id>1</sdp-id>
  <node-id>PIOSXR-1</node-id>
  <service-match-criteria>
    <match-criterion>
      <index>0</index>
      <target-connection-group-id>matrix1</target-connection-group-id>
    </match-criterion>
  </service-match-criteria>
<attachment-circuits>
  <attachment-circuit>
    <ac-id>0</ac-id>
    <ac-tp-id>GigabitEthernet1</ac-tp-id>
    <ac-ip-address>11.0.0.2</ac-ip-address>

```

```

    <ac-ip-prefix-length>24</ac-ip-prefix-length>
    <ac-tags>
      <ac-tags>
        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>100</value>
      </ac-tags>
    </ac-tags>
    <sdp-peering>
      <protocol>
        <protocol-type>peering-protocol-bgp</protocol-type>
        <bgp-attributes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
          <neighbor>11.0.0.1</neighbor>
          <remote-as>100</remote-as>
        </bgp-attributes>
      </protocol>
    </sdp-peering>
  </attachment-circuit>
</attachment-circuits>
</sdp>
</sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:any-to-any</connectivity-type>
  </connection-group>
</connection-groups>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L3-any-to-any-custom

The following sample payload shows the implementation of IETF NSS in the L3-any-to-any-custom topology. Display the service plan to view the configuration for the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">
    <slice-service>
      <service-id>L3-Shared</service-id>
      <service-tags>
        <tag-type>
          <tag-type>service-tag-opaque</tag-type>
          <value>OPAQUE_VAL</value>
        </tag-type>
      </service-tags>
    </slice-service>
  </network-slice-services>
</config>

```

```

</tag-type>
<tag-type>
  <tag-type>service-tag-customer</tag-type>
  <value>cisco</value>
</tag-type>
<tag-type>
  <tag-type>service-tag-service</tag-type>
  <value>L3</value>
</tag-type>
</service-tags>
<slo-sle-template>min-delay-all-custom</slo-sle-template>
<isolation xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service" xmlns:ietf-nss="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">ietf-nss:service-isolation-shared</isolation>
<sdps>
  <sdp>
    <sdp-id>0</sdp-id>
    <node-id>PIOSXR-0</node-id>
    <service-match-criteria>
      <match-criterion>
        <index>0</index>
        <target-connection-group-id>matrix1</target-connection-group-id>
      </match-criterion>
    </service-match-criteria>
    <attachment-circuits>
      <attachment-circuit>
        <ac-id>0</ac-id>
        <ac-tp-id>GigabitEthernet0</ac-tp-id>
        <ac-ip-address>11.0.0.1</ac-ip-address>
        <ac-ip-prefix-length>24</ac-ip-prefix-length>
        <ac-tags>
          <ac-tags>
            <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
            <value>100</value>
          </ac-tags>
        </ac-tags>
        <sdp-peering>
          <protocol>
            <protocol-type>peering-protocol-bgp</protocol-type>
            <bgp-attributes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
              <neighbor>11.0.0.2</neighbor>
              <remote-as>100</remote-as>
            </bgp-attributes>
          </protocol>

```

```

        </sdp-peering>
    </attachment-circuit>
</attachment-circuits>
</sdp>
<sdp>
    <sdp-id>1</sdp-id>
    <node-id>PIOSXR-1</node-id>
    <service-match-criteria>
        <match-criterion>
            <index>0</index>
            <target-connection-group-id>matrix1</target-connection-group-id>
        </match-criterion>
    </service-match-criteria>
    <attachment-circuits>
        <attachment-circuit>
            <ac-id>0</ac-id>
            <ac-tp-id>GigabitEthernet1</ac-tp-id>
            <ac-ip-address>11.0.0.2</ac-ip-address>
            <ac-ip-prefix-length>24</ac-ip-prefix-length>
            <ac-tags>
                <ac-tag>
                    <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
                    <value>100</value>
                </ac-tag>
            </ac-tags>
            <sdp-peering>
                <protocol>
                    <protocol-type>peering-protocol-bgp</protocol-type>
                    <bgp-attributes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
                        <neighbor>11.0.0.1</neighbor>
                        <remote-as>100</remote-as>
                    </bgp-attributes>
                </protocol>
            </sdp-peering>
        </attachment-circuit>
    </attachment-circuits>
</sdp>
</sdps>
<connection-groups>
    <connection-group>
        <connection-group-id>matrix1</connection-group-id>
        <connectivity-type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:any-to-any</connectivity-type>
        <service-slo-sle-policy>

```

```

        <bandwidth xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-
slice-service">100</bandwidth>
    </service-slo-sle-policy>
</connection-group>
</connection-groups>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L3-hub-spoke-Dedicated Topology

The following sample payload shows the implementation of IETF NSS in the L3-hub-spoke-dedicated topology. Display the service plan to view the configuration for the service.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">
    <slice-service>
      <service-id>L3-Dedicated</service-id>
      <service-tags>
        <tag-type>
          <tag-type>service-tag-service</tag-type>
          <value>L3</value>
        </tag-type>
        <tag-type>
          <tag-type>service-tag-customer</tag-type>
          <value>Customer-A</value>
        </tag-type>
      </service-tags>
      <slo-sle-template>Dedicated</slo-sle-template>
      <isolation xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service" xmlns:ietf-nss="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">ietf-nss:service-isolation-dedicated</isolation>
      <sdps>
        <sdp>
          <sdp-id>0</sdp-id>
          <node-id>PIOSXR-2</node-id>
          <service-match-criteria>
            <match-criterion>
              <index>0</index>
              <target-connection-group-id>matrix1</target-connection-group-id>
            </match-criterion>
          </service-match-criteria>
          <attachment-circuits>
            <attachment-circuit>
              <ac-id>0</ac-id>
            </attachment-circuit>
          </attachment-circuits>
        </sdp>
      </sdps>
    </slice-service>
  </network-slice-services>
</config>

```



```

    <ac-tp-id>GigabitEthernet0</ac-tp-id>
    <ac-ip-address>10.0.0.2</ac-ip-address>
    <ac-ip-prefix-length>24</ac-ip-prefix-length>
    <ac-tags>
      <ac-tags>
        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>200</value>
      </ac-tags>
    </ac-tags>
    <sdp-peering>
      <protocol>
        <protocol-type>peering-protocol-bgp</protocol-type>
        <bgp-attributes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
          <neighbor>10.0.0.1</neighbor>
          <remote-as>100</remote-as>
        </bgp-attributes>
      </protocol>
    </sdp-peering>
  </attachment-circuit>
</attachment-circuits>
</sdp>
<sdp>
  <sdp-id>1</sdp-id>
  <node-id>PIOSXR-1</node-id>
  <service-match-criteria>
    <match-criterion>
      <index>0</index>
      <target-connection-group-id>matrix1</target-connection-group-id>
      <connection-group-sdp-role xmlns:vpn-
common="urn:ietf:params:xml:ns:yang:ietf-vpn-common">vpn-common:spoke-
role</connection-group-sdp-role>
    </match-criterion>
  </service-match-criteria>
  <attachment-circuits>
    <attachment-circuit>
      <ac-id>0</ac-id>
      <ac-tp-id>GigabitEthernet0</ac-tp-id>
      <ac-ip-address>10.0.0.2</ac-ip-address>
      <ac-ip-prefix-length>24</ac-ip-prefix-length>
      <ac-tags>
        <ac-tags>
          <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
          <value>200</value>
        </ac-tags>
      </ac-tags>
    </attachment-circuit>
  </attachment-circuits>
</sdp>

```

```

    </ac-tags>
    <sdp-peering>
      <protocol>
        <protocol-type>peering-protocol-bgp</protocol-type>
        <bgp-attributes xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
network-slice-service">
          <neighbor>10.0.0.1</neighbor>
          <remote-as>100</remote-as>
        </bgp-attributes>
      </protocol>
    </sdp-peering>
  </attachment-circuit>
</attachment-circuits>
</sdp>
</sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:hub-spoke</connectivity-type>
  </connection-group>
</connection-groups>
  <shared xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service">
    <slice>
      <service-id>L3-Shared</service-id>
    </slice>
    <single-sided-control/>
  </shared>
</slice-service>
</network-slice-services>
</config>

```

Implementing IETF NSS in L3 with No Peering Protocol

For TSDN 7.0, we are adding support for direct-routing routing-protocol which will push bgp → vrf → redistribute-connected for defined sdp attachment-circuit address-family on device vrf. Direct-routing is used for scenarios where the user needs to broadcast local routes without peering protocol. To configure NSS in L3 with direct-routing, you must omit sdp-peering, as shown in the payload below.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">
    <slice-service>
      <service-id>L3-Shared-NC</service-id>

```

```

<service-tags>
  <tag-type>
    <tag-type>service-tag-service</tag-type>
    <value>L3</value>
  </tag-type>
  <tag-type>
    <tag-type>service-tag-customer</tag-type>
    <value>cisco</value>
  </tag-type>
  <tag-type>
    <tag-type>service-tag-opaque</tag-type>
    <value>OPAQUE_VAL</value>
  </tag-type>
</service-tags>
<slo-sle-template>Shared</slo-sle-template>
<isolation xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-slice-
service" xmlns:ietf-nss="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">ietf-nss:service-isolation-shared</isolation>
<s dps>
  <sdp>
    <sdp-id>0</sdp-id>
    <node-id>PN76-0</node-id>
    <service-match-criteria>
      <match-criterion>
        <index>0</index>
        <target-connection-group-id>matrix1</target-connection-group-id>
      </match-criterion>
    </service-match-criteria>
    <attachment-circuits>
      <attachment-circuit>
        <ac-id>0</ac-id>
        <ac-tp-id>HundredGigE0</ac-tp-id>
        <ac-ip-address>11.0.0.1</ac-ip-address>
        <ac-ip-prefix-length>24</ac-ip-prefix-length>
        <ac-tags>
          <ac-tags>
            <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
            <value>100</value>
          </ac-tags>
        </ac-tags>
      </attachment-circuit>
    </attachment-circuits>
  </sdp>
  <sdp>
    <sdp-id>1</sdp-id>

```

```

<node-id>PN76-1</node-id>
<service-match-criteria>
  <match-criterion>
    <index>0</index>
    <target-connection-group-id>matrix1</target-connection-group-id>
  </match-criterion>
</service-match-criteria>
<attachment-circuits>
  <attachment-circuit>
    <ac-id>0</ac-id>
    <ac-tp-id>HundredGigE1</ac-tp-id>
    <ac-ip-address>11.0.0.2</ac-ip-address>
    <ac-ip-prefix-length>24</ac-ip-prefix-length>
    <ac-tags>
      <ac-tags>
        <ac-tag-type>attachment-circuit-tag-vlan-id</ac-tag-type>
        <value>100</value>
      </ac-tags>
    </ac-tags>
  </attachment-circuit>
</attachment-circuits>
</sdp>
</sdps>
<connection-groups>
  <connection-group>
    <connection-group-id>matrix1</connection-group-id>
    <connectivity-type xmlns:vpn-common="urn:ietf:params:xml:ns:yang:ietf-
vpn-common">vpn-common:any-to-any</connectivity-type>
  </connection-group>
</connection-groups>
</slice-service>
</network-slice-services>
</config>

```

Creating Performance Measurement Services

Performance measurement (PM) services help to understand and measure the network performance by using metrics. Network operators use this information to evaluate network performance and to comply with service level agreements, mitigate any potential network issues, and allocate the required resources as a response to provide quality service to customers.

For more information about network performance metrics in traffic engineering, see the ***Segment Routing Configuration Guide*** on the Cisco documentation website.

Note: In TSDN FP Bundle CFP, performance measurement is currently fully supported on IOSXR Netconf NED 7.5 and later and is offered with limited support on IOSXE CLI NED 6.77/6.86.

A performance measurement service requires a catalog of profiles. You can configure different profiles for different types of delay measurements. This catalog is saved only to CDB. You must load it to NSO to use the catalog in the performance measurement services.

Use the "interfaces" delay profile type for link-delay measurement. The "sr-policy" delay profile type is used for SR policy delay measurements.

Note: With IOSXE CLI NED, Performance measurement services currently allow only creation of delay profiles, except RSVP-TE delay profiles. The service currently does not support liveness-detection profiles and pushing loss profiles on a device. The loss profile models are created only as a roadmap. IETF-TE and RSVP-TE support only the creation of default profiles. Named profiles are currently not supported.

To create a performance measurement service:

1. Create a performance measurement profile. Delay profile allows you to schedule probes and configure metric advertisement parameters for delay measurement. The following is a sample payload to demonstrate configurations for the delay profile, loss profile, and liveness profile, and the different ways they are used in the service payload.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <pm xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-pm">
    <pm-profiles>
      <protocol>
        <profile>
          <name>PmProtocol</name>
          <twamp-light>
            <measurement>
              <delay>
                <unauthenticated>
                  <querier-dst-port>1026</querier-dst-port>
                  <querier-src-port>1029</querier-src-port>
                  <ipv4>
                    <timestamp1>
                      <label>2577</label>
                    </timestamp1>
                    <timestamp2>
                      <label>350</label>
                    </timestamp2>
                  </ipv4>
                  <ipv6>
                    <timestamp1>
```

```

        <label>900</label>
      </timestamp1>
      <timestamp2>
        <label>980</label>
      </timestamp2>
    </ipv6>
  </unauthenticated>
</delay>
</measurement>
</twamp-light>
</profile>
</protocol>
<delay-profile>
  <interface>
    <profile>
      <name>DelayInterface</name>
      <probe>
        <computation-interval>90</computation-interval>
        <tx-interval>80000</tx-interval>
        <tos>
          <dscp>50</dscp>
          <traffic-class>3</traffic-class>
        </tos>
        <measurement-mode>one-way</measurement-mode>
      </probe>
    <advertisement>
      <logging>
        <delay-exceeded/>
      </logging>
      <periodic>
        <disabled/>
        <interval>3000</interval>
        <threshold>90</threshold>
        <minimum-change>10000</minimum-change>
      </periodic>
      <accelerated>
        <threshold>90</threshold>
        <minimum-change>900</minimum-change>
      </accelerated>
      <anomaly-check>
        <upper-bound>200000</upper-bound>
        <lower-bound>100000</lower-bound>
      </anomaly-check>
    </advertisement>
  </interface>
</delay-profile>

```

```
</profile>
<profile>
  <name>DelayInterface2</name>
  <probe>
    <computation-interval>90</computation-interval>
    <tx-interval>80000</tx-interval>
    <tos>
      <dscp>50</dscp>
      <traffic-class>3</traffic-class>
    </tos>
    <measurement-mode>one-way</measurement-mode>
  </probe>
  <advertisement>
    <logging>
      <delay-exceeded/>
    </logging>
    <periodic>
      <disabled/>
      <interval>3000</interval>
      <threshold>90</threshold>
      <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
      <threshold>90</threshold>
      <minimum-change>900</minimum-change>
    </accelerated>
    <anomaly-check>
      <upper-bound>200000</upper-bound>
      <lower-bound>100000</lower-bound>
    </anomaly-check>
  </advertisement>
</profile>
<profile>
  <name>DelayInterface3</name>
  <probe>
    <computation-interval>90</computation-interval>
    <tx-interval>80000</tx-interval>
    <tos>
      <dscp>50</dscp>
      <traffic-class>3</traffic-class>
    </tos>
    <measurement-mode>one-way</measurement-mode>
  </probe>
  <advertisement>
```

```

    <logging>
      <delay-exceeded/>
    </logging>
    <periodic>
      <disabled/>
      <interval>3000</interval>
      <threshold>90</threshold>
      <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
      <threshold>90</threshold>
      <minimum-change>900</minimum-change>
    </accelerated>
    <anomaly-check>
      <upper-bound>200000</upper-bound>
      <lower-bound>100000</lower-bound>
    </anomaly-check>
  </advertisement>
</profile>
</interface>
<rsvp-te>
  <profile>
    <name>DelayRsvpTe</name>
    <probe>
      <computation-interval>90</computation-interval>
      <tx-interval>80000</tx-interval>
      <tos>
        <dscp>50</dscp>
        <traffic-class>2</traffic-class>
      </tos>
      <measurement-mode>one-way</measurement-mode>
    </probe>
  <advertisement>
    <logging>
      <delay-exceeded/>
    </logging>
    <periodic>
      <disabled/>
      <interval>3000</interval>
      <threshold>90</threshold>
      <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
      <threshold>90</threshold>

```



```

        <minimum-change>900</minimum-change>
    </accelerated>
</advertisement>
</profile>
<profile>
    <name>DelayRsvpTe2</name>
    <probe>
        <computation-interval>90</computation-interval>
        <tx-interval>80000</tx-interval>
        <tos>
            <dscp>50</dscp>
            <traffic-class>2</traffic-class>
        </tos>
        <measurement-mode>one-way</measurement-mode>
    </probe>
    <advertisement>
        <logging>
            <delay-exceeded/>
        </logging>
        <periodic>
            <disabled/>
            <interval>3000</interval>
            <threshold>90</threshold>
            <minimum-change>10000</minimum-change>
        </periodic>
        <accelerated>
            <threshold>90</threshold>
            <minimum-change>900</minimum-change>
        </accelerated>
        <threshold-check>
            <average-delay/>
            <maximum-delay/>
            <minimum-delay/>
        </threshold-check>
    </advertisement>
</profile>
</rsvp-te>
<sr-policy>
    <profile>
        <name>DelaySrPolicy</name>
        <probe>
            <computation-interval>90</computation-interval>
            <tx-interval>80000</tx-interval>
            <tos>

```

```

        <dscp>50</dscp>
        <traffic-class>4</traffic-class>
    </tos>
    <sweep>
        <destination>
            <ipv4>127.0.0.3</ipv4>
            <range>5</range>
        </destination>
    </sweep>
    <measurement-mode>one-way</measurement-mode>
</probe>
<advertisement>
    <logging>
        <delay-exceeded/>
    </logging>
    <periodic>
        <disabled/>
        <interval>3000</interval>
        <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
        <threshold>90</threshold>
        <minimum-change>900</minimum-change>
    </accelerated>
    <anomaly-check>
        <upper-bound>200000</upper-bound>
        <lower-bound>100000</lower-bound>
    </anomaly-check>
    <threshold-check>
        <average-delay/>
        <maximum-delay/>
        <minimum-delay/>
    </threshold-check>
</advertisement>
</profile>
<profile>
    <name>DelaySrPolicy2</name>
    <probe>
        <computation-interval>90</computation-interval>
        <tx-interval>80000</tx-interval>
        <tos>
            <dscp>50</dscp>
            <traffic-class>4</traffic-class>
        </tos>
    </probe>
</profile>

```

```

    <sweep>
      <destination>
        <ipv4>127.0.0.3</ipv4>
        <range>5</range>
      </destination>
    </sweep>
    <measurement-mode>one-way</measurement-mode>
  </probe>
</advertisement>
  <logging>
    <delay-exceeded/>
  </logging>
  <periodic>
    <disabled/>
    <interval>3000</interval>
    <minimum-change>10000</minimum-change>
  </periodic>
  <accelerated>
    <threshold>90</threshold>
    <minimum-change>900</minimum-change>
  </accelerated>
  <anomaly-check>
    <upper-bound>200000</upper-bound>
    <lower-bound>100000</lower-bound>
  </anomaly-check>
  <threshold-check>
    <average-delay/>
    <maximum-delay/>
    <minimum-delay/>
  </threshold-check>
</advertisement>
</profile>
<profile>
  <name>DelaySrPolicy3</name>
  <probe>
    <computation-interval>90</computation-interval>
    <tx-interval>80000</tx-interval>
    <tos>
      <dscp>50</dscp>
      <traffic-class>4</traffic-class>
    </tos>
    <sweep>
      <destination>
        <ipv4>127.0.0.3</ipv4>

```

```

        <range>5</range>
    </destination>
</sweep>
    <measurement-mode>one-way</measurement-mode>
</probe>
<advertisement>
    <logging>
        <delay-exceeded/>
    </logging>
    <periodic>
        <disabled/>
        <interval>3000</interval>
        <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
        <threshold>90</threshold>
        <minimum-change>900</minimum-change>
    </accelerated>
    <anomaly-check>
        <upper-bound>200000</upper-bound>
        <lower-bound>100000</lower-bound>
    </anomaly-check>
    <threshold-check>
        <average-delay/>
        <maximum-delay/>
        <minimum-delay/>
    </threshold-check>
</advertisement>
</profile>
</sr-policy>
<endpoint>
    <profile>
        <name>DelayEndpointProfile</name>
    <probe>
        <computation-interval>90</computation-interval>
        <tx-interval>80000</tx-interval>
        <tos>
            <dscp>50</dscp>
        </tos>
        <sweep>
            <destination>
                <ipv4>127.0.0.3</ipv4>
                <range>5</range>
            </destination>

```

```
    </sweep>
    <measurement-mode>one-way</measurement-mode>
  </probe>
<advertisement>
  <logging>
    <delay-exceeded/>
  </logging>
  <periodic>
    <disabled/>
    <interval>3000</interval>
    <threshold>90</threshold>
    <minimum-change>10000</minimum-change>
  </periodic>
  <accelerated>
    <threshold>90</threshold>
    <minimum-change>900</minimum-change>
  </accelerated>
  <threshold-check>
    <average-delay/>
    <maximum-delay/>
    <minimum-delay/>
  </threshold-check>
</advertisement>
</profile>
<profile>
  <name>DelayEndpointProfile2</name>
  <probe>
    <computation-interval>90</computation-interval>
    <tx-interval>80000</tx-interval>
    <tos>
      <dscp>40</dscp>
    </tos>
    <sweep>
      <destination>
        <ipv4>127.0.0.3</ipv4>
        <range>5</range>
      </destination>
    </sweep>
    <measurement-mode>one-way</measurement-mode>
  </probe>
  <advertisement>
    <logging>
      <delay-exceeded/>
    </logging>
```

```

    <periodic>
      <disabled/>
      <interval>3000</interval>
      <threshold>90</threshold>
      <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
      <threshold>90</threshold>
      <minimum-change>900</minimum-change>
    </accelerated>
    <threshold-check>
      <average-delay/>
      <maximum-delay/>
      <minimum-delay/>
    </threshold-check>
  </advertisement>
</profile>
</endpoint>
</delay-profile>
<liveness-profile>
  <sr-policy>
    <profile>
      <name>SrPolicyLiveness</name>
      <probe>
        <tx-interval>600000</tx-interval>
        <tos>
          <dscp>48</dscp>
        </tos>
        <sweep>
          <destination>
            <ipv4>127.0.0.3</ipv4>
            <range>5</range>
          </destination>
        </sweep>
      </probe>
      <liveness-detection>
        <multiplier>5</multiplier>
      </liveness-detection>
    </profile>
    <profile>
      <name>SrPolicyLiveness2</name>
      <probe>
        <tx-interval>600000</tx-interval>
        <tos>

```

```
        <dscp>48</dscp>
    </tos>
    <sweep>
        <destination>
            <ipv4>127.0.0.3</ipv4>
            <range>5</range>
        </destination>
    </sweep>
</probe>
<liveness-detection>
    <multiplier>5</multiplier>
</liveness-detection>
</profile>
<profile>
    <name>SrPolicyLiveness3</name>
    <probe>
        <tx-interval>600000</tx-interval>
        <tos>
            <dscp>48</dscp>
        </tos>
        <sweep>
            <destination>
                <ipv4>127.0.0.3</ipv4>
                <range>5</range>
            </destination>
        </sweep>
    </probe>
    <liveness-detection>
        <multiplier>5</multiplier>
    </liveness-detection>
</profile>
</sr-policy>
<endpoint>
    <profile>
        <name>EndpointLiveness</name>
        <probe>
            <tx-interval>800000</tx-interval>
            <tos>
                <dscp>48</dscp>
            </tos>
        </probe>
        <liveness-detection>
            <multiplier>7</multiplier>
        </liveness-detection>
    </profile>
</endpoint>
```

```

</profile>
<profile>
  <name>EndpointLiveness2</name>
  <probe>
    <tx-interval>800000</tx-interval>
    <tos>
      <dscp>48</dscp>
    </tos>
  </probe>
  <liveness-detection>
    <multiplier>7</multiplier>
    <logging>
      <state-change/>
    </logging>
  </liveness-detection>
</profile>
<profile>
  <name>EndpointLiveness3</name>
  <probe>
    <tx-interval>800000</tx-interval>
    <tos>
      <dscp>48</dscp>
    </tos>
  </probe>
  <liveness-detection>
    <multiplier>7</multiplier>
  </liveness-detection>
</profile>
</endpoint>
</liveness-profile>
<loss-profile>
  <interface>
    <profile>
      <name>LossInterfaceProfile</name>
      <probe>
        <computation-interval>20</computation-interval>
        <tx-interval>70000</tx-interval>
        <tos>
          <dscp>60</dscp>
          <traffic-class>5</traffic-class>
        </tos>
      </probe>
      <advertisement>
        <periodic>

```



```
        <disabled/>
        <interval>500</interval>
        <threshold>80</threshold>
        <minimum-change>4000</minimum-change>
    </periodic>
    <accelerated>
        <threshold>80</threshold>
        <minimum-change>2000</minimum-change>
    </accelerated>
</advertisement>
</profile>
<profile>
    <name>LossInterfaceProfile2</name>
    <probe>
        <computation-interval>20</computation-interval>
        <tx-interval>70000</tx-interval>
        <tos>
            <dscp>60</dscp>
            <traffic-class>5</traffic-class>
        </tos>
    </probe>
    <advertisement>
        <periodic>
            <disabled/>
            <interval>500</interval>
            <threshold>80</threshold>
            <minimum-change>4000</minimum-change>
        </periodic>
        <accelerated>
            <threshold>80</threshold>
            <minimum-change>2000</minimum-change>
        </accelerated>
    </advertisement>
</profile>
<profile>
    <name>LossInterfaceProfile3</name>
    <probe>
        <computation-interval>20</computation-interval>
        <tx-interval>70000</tx-interval>
        <tos>
            <dscp>60</dscp>
            <traffic-class>5</traffic-class>
        </tos>
    </probe>
```

```
<advertisement>
  <periodic>
    <disabled/>
    <interval>500</interval>
    <threshold>80</threshold>
    <minimum-change>4000</minimum-change>
  </periodic>
  <accelerated>
    <threshold>80</threshold>
    <minimum-change>2000</minimum-change>
  </accelerated>
</advertisement>
</profile>
</interface>
<sr-policy>
  <profile>
    <name>LossSrPolicyProfile</name>
    <probe>
      <computation-interval>70</computation-interval>
      <tx-interval>80000</tx-interval>
      <tos>
        <dscp>60</dscp>
        <traffic-class>5</traffic-class>
      </tos>
    </probe>
    <advertisement>
      <periodic>
        <disabled/>
        <interval>90</interval>
        <threshold>70</threshold>
        <minimum-change>10000</minimum-change>
      </periodic>
      <accelerated>
        <threshold>80</threshold>
        <minimum-change>1000</minimum-change>
      </accelerated>
    </advertisement>
  </profile>
  <profile>
    <name>LossSrPolicyProfile2</name>
    <probe>
      <computation-interval>70</computation-interval>
      <tx-interval>80000</tx-interval>
      <tos>
```

```

        <dscp>60</dscp>
        <traffic-class>5</traffic-class>
    </tos>
</probe>
<advertisement>
    <periodic>
        <disabled/>
        <interval>90</interval>
        <threshold>70</threshold>
        <minimum-change>10000</minimum-change>
    </periodic>
    <accelerated>
        <threshold>80</threshold>
        <minimum-change>1000</minimum-change>
    </accelerated>
</advertisement>
</profile>
<profile>
    <name>LossSrPolicyProfile3</name>
    <probe>
        <computation-interval>70</computation-interval>
        <tx-interval>80000</tx-interval>
        <tos>
            <dscp>60</dscp>
            <traffic-class>5</traffic-class>
        </tos>
    </probe>
    <advertisement>
        <periodic>
            <disabled/>
            <interval>90</interval>
            <threshold>70</threshold>
            <minimum-change>10000</minimum-change>
        </periodic>
        <accelerated>
            <threshold>80</threshold>
            <minimum-change>1000</minimum-change>
        </accelerated>
    </advertisement>
</profile>
</sr-policy>
<rsvp-te>
    <profile>
        <name>LossRsvpTeProfile</name>

```

```
<probe>
  <computation-interval>70</computation-interval>
  <tx-interval>80000</tx-interval>
  <tos>
    <dscp>60</dscp>
    <traffic-class>5</traffic-class>
  </tos>
</probe>
<advertisement>
  <periodic>
    <disabled/>
    <interval>90</interval>
    <threshold>80</threshold>
    <minimum-change>1000</minimum-change>
  </periodic>
  <accelerated>
    <threshold>80</threshold>
    <minimum-change>8000</minimum-change>
  </accelerated>
</advertisement>
</profile>
<profile>
  <name>LossRsvpTeProfile2</name>
  <probe>
    <computation-interval>70</computation-interval>
    <tx-interval>80000</tx-interval>
    <tos>
      <dscp>60</dscp>
      <traffic-class>5</traffic-class>
    </tos>
  </probe>
  <advertisement>
    <periodic>
      <disabled/>
      <interval>90</interval>
      <threshold>80</threshold>
      <minimum-change>1000</minimum-change>
    </periodic>
    <accelerated>
      <threshold>80</threshold>
      <minimum-change>8000</minimum-change>
    </accelerated>
  </advertisement>
</profile>
```

```

<profile>
  <name>LossRsvpTeProfile3</name>
  <probe>
    <computation-interval>70</computation-interval>
    <tx-interval>80000</tx-interval>
    <tos>
      <dscp>60</dscp>
      <traffic-class>5</traffic-class>
    </tos>
  </probe>
  <advertisement>
    <periodic>
      <disabled/>
      <interval>90</interval>
      <threshold>80</threshold>
      <minimum-change>1000</minimum-change>
    </periodic>
    <accelerated>
      <threshold>80</threshold>
      <minimum-change>8000</minimum-change>
    </accelerated>
  </advertisement>
</profile>
</rsvp-te>
</loss-profile>
</pm-profiles>
</pm>
</config>

```

2. Create a performance measurement service. The service pushes the chosen catalogs into the list of devices.

Note: When the profiles catalog is updated, the services using the profile are automatically redeployed. The performance measurement service is used to update the devices with the latest profile configuration.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <pm xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-pm">
    <svc-profiles>
      <name>PM-CLI</name>
      <performance-measurement>
        <protocol-profile>
          <name>PmProtocol</name>
        </protocol-profile>
        <delay-profile>

```

```
<interface>
  <default-profile>DelayInterface</default-profile>
  <profile>
    <name>DelayInterface2</name>
  </profile>
  <profile>
    <name>DelayInterface3</name>
  </profile>
</interface>
<rsvp-te>
  <default-profile>DelayRsvpTe</default-profile>
</rsvp-te>
<sr-policy>
  <default-profile>DelaySrPolicy</default-profile>
  <profile>
    <name>DelaySrPolicy2</name>
  </profile>
  <profile>
    <name>DelaySrPolicy3</name>
  </profile>
</sr-policy>
<endpoint>
  <default-profile>DelayEndpointProfile</default-profile>
  <profile>
    <name>DelayEndpointProfile2</name>
  </profile>
</endpoint>
</delay-profile>
<liveness-profile>
  <sr-policy>
    <default-profile>SrPolicyLiveness</default-profile>
    <profile>
      <name>SrPolicyLiveness2</name>
    </profile>
    <profile>
      <name>SrPolicyLiveness3</name>
    </profile>
  </sr-policy>
  <endpoint>
    <default-profile>EndpointLiveness</default-profile>
    <profile>
      <name>EndpointLiveness2</name>
    </profile>
  </profile>
</liveness-profile>
```

```

        <name>EndpointLiveness3</name>
    </profile>
</endpoint>
</liveness-profile>
<loss-profile>
    <interface>
        <default-profile>LossInterfaceProfile</default-profile>
        <profile>
            <name>LossInterfaceProfile2</name>
        </profile>
        <profile>
            <name>LossInterfaceProfile3</name>
        </profile>
    </interface>
    <rsvp-te>
        <default-profile>LossRsvpTeProfile</default-profile>
        <profile>
            <name>LossRsvpTeProfile2</name>
        </profile>
        <profile>
            <name>LossRsvpTeProfile3</name>
        </profile>
    </rsvp-te>
    <sr-policy>
        <default-profile>LossSrPolicyProfile</default-profile>
        <profile>
            <name>LossSrPolicyProfile2</name>
        </profile>
        <profile>
            <name>LossSrPolicyProfile3</name>
        </profile>
    </sr-policy>
</loss-profile>
</performance-measurement>
<devices>
    <name>PIOSXR-0</name>
    <interface-profile-attachment>
        <interfaces>
            <if-type>HundredGigE</if-type>
            <if-id>0/0/0/0</if-id>
            <delay-measurement>
                <profile-name>DelayInterface2</profile-name>
                <advertise-delay>10000</advertise-delay>
            </delay-measurement>
        </interfaces>
    </interface-profile-attachment>
</devices>

```

```

        <next-hop-ip-addr>10.1.1.2</next-hop-ip-addr>
    </interfaces>
</interfaces>
    <if-type>HundredGigE</if-type>
    <if-id>0/0/0/1</if-id>
    <delay-measurement>
        <explicit-delay-profile>explicitInterfaceDelayProfile</explicit-
delay-profile>
        <advertise-delay>20000</advertise-delay>
    </delay-measurement>
    <next-hop-ipv6-addr>110:101::102</next-hop-ipv6-addr>
</interfaces>
</interface-profile-attachment>
</devices>
<devices>
    <name>PIOSXR-1</name>
    <interface-profile-attachment>
        <interfaces>
            <if-type>HundredGigE</if-type>
            <if-id>0/0/0/0</if-id>
            <delay-measurement>
                <profile-name>DelayInterface</profile-name>
                <advertise-delay>10000</advertise-delay>
            </delay-measurement>
            <next-hop-ip-addr>10.1.1.2</next-hop-ip-addr>
        </interfaces>
        <interfaces>
            <if-type>HundredGigE</if-type>
            <if-id>0/0/0/1</if-id>
            <delay-measurement>
                <explicit-delay-profile>explicitInterfaceDelayProfile</explicit-
delay-profile>
                <advertise-delay>20000</advertise-delay>
            </delay-measurement>
            <next-hop-ipv6-addr>110:101::102</next-hop-ipv6-addr>
        </interfaces>
    </interface-profile-attachment>
</devices>
</svc-profiles>
</pm>
</config>

```

3. Display the plan to view the status of the service. The following table discusses certain plan components unique to this service. For a detailed description of the plan components, see ***NSO documentation***.

Plan Component	Value	Description
Type	Device	Represents the connected node(device) of the nano plan of a service.
State	cisco-pm-fp-nano-plan-services:config-apply	Represents acceptance of service intent and respective configuration application on device

```
admin@ncs% run show pm pm-plan
```

```

              BACK      STATUS
TYPE  NAME  TRACK_GOAL  CODE  STATE
-----
self  self  false      -      init
      ready
device PIOSXR-0 false - -      init
      cisco-pm-fp-nano-plan-services:config-apply reached 2022-09-02T18:47:10 - -
      ready
      cisco-pm-fp-nano-plan-services:config-apply reached 2022-09-02T18:47:13 - -
device PIOSXR-1 false - -      init
      cisco-pm-fp-nano-plan-services:config-apply reached 2022-09-02T18:47:10 - -
      ready
      cisco-pm-fp-nano-plan-services:config-apply reached 2022-09-02T18:47:13 - -

```

Creating Performance Measurement Service with IETF-TE Attachment

IETF-TE uses only default profiles. No named profile creation is available in IETF-TE. The following sample payload creates a PM service with the default delay profile enabled on the tunnel. Display the plan to view the status of the service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <te xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
    <tunnels>
      <tunnel>
        <name>IETF-RSVP-TE</name>
        <identifier>1234</identifier>
        <description>RSVP_TE</description>
        <source>111.1.1.1</source>
        <destination>222.2.2.2</destination>
        <te-bandwidth>
          <generic>94967295</generic>
        </te-bandwidth>
        <setup-priority>3</setup-priority>
        <hold-priority>2</hold-priority>
        <signaling-type>te-types:path-setup-rsvp</signaling-type>
        <performance-measurement xmlns="http://cisco.com/ns/nso/fp/examples/cisco-te"/>
          <head-end xmlns="http://cisco.com/ns/nso/fp/examples/cisco-te">PIOSXR-0</head-end>
          <tail-end xmlns="http://cisco.com/ns/nso/fp/examples/cisco-te">PIOSXR-1</tail-end>
        </tunnel>
      </tunnels>
    </te>
  </config>
```

Creating Performance Measurement Service with SR Attachment

In segment routing, it is not required to associate the profile name to a performance measurement service. You can set the profile name to a performance measurement service, or the profile name can be an existing profile on the device.

The following payloads show a performance measurement service attached to an SR-ODN service and an SR-policy.

Performance measurement parameters under SR-ODN service

The following is a sample payload using performance measurement parameters under SR-ODN service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
```

```

<sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
  <odn xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te-sr-odn">
    <odn-template>
      <name>SR-CLI-ODN-300</name>
      <performance-measurement>
        <delay-measurement>
          <profile>testDelayProfile</profile>
          <logging>
            <delay-exceeded/>
          </logging>
        </delay-measurement>
        <reverse-path-label>900</reverse-path-label>
      </performance-measurement>
    <head-end>
      <name>PIOSXR-0</name>
    </head-end>
    <head-end>
      <name>PIOSXR-1</name>
    </head-end>
    <maximum-sid-depth>9</maximum-sid-depth>
    <color>300</color>
    <bandwidth>200</bandwidth>
    <dynamic>
      <pce/>
      <flex-alg>200</flex-alg>
      <metric-type>igp</metric-type>
      <metric-margin>
        <absolute>30</absolute>
      </metric-margin>
      <disjoint-path>
        <type>node</type>
        <group-id>10</group-id>
      </disjoint-path>
    </dynamic>
  </odn-template>
  <odn-template>
    <name>SR-CLI-ODN-400</name>
    <performance-measurement>
      <liveness-detection>
        <profile>testLivenessProfile</profile>
        <backup>testBackupProfile</backup>
        <invalidation-action>down</invalidation-action>
      </liveness-detection>
      <logging>
        <session-state-change/>
      </logging>
    </performance-measurement>
  </odn-template>
</sr-te>

```

```

        </logging>
    </liveness-detection>
    <reverse-path-label>600</reverse-path-label>
</performance-measurement>
<head-end>
    <name>PIOSXR-0</name>
</head-end>
<color>400</color>
<dynamic>
    <flex-alg>201</flex-alg>
    <metric-type>latency</metric-type>
</dynamic>
</odn-template>
</odn>
</sr-te>
</config>

```

Performance measurement service under SR-policy

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te">
    <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tdsn-sr-te-sr-policies">
      <policy>
        <name>SR-CLI-DYNAMIC</name>
        <head-end>
          <name>PIOSXR-0</name>
        </head-end>
        <tail-end>127.0.0.1</tail-end>
        <color>111</color>
        <binding-sid>111</binding-sid>
        <performance-measurement>
          <delay-measurement>
            <profile>testDelayProfile</profile>
            <logging>
              <delay-exceeded/>
            </logging>
          </delay-measurement>
          <reverse-path-label>90</reverse-path-label>
        </performance-measurement>
        <path>
          <preference>111</preference>
          <dynamic>
            <constraints>
              <affinity>

```

```
<rule>
  <action>include-all</action>
  <color>GREEN</color>
  <color>RED</color>
</rule>
</affinity>
<segments>
  <sid-algorithm>128</sid-algorithm>
</segments>
<sid-limit>10</sid-limit>
</constraints>
<metric-type>te</metric-type>
<metric-margin>
  <absolute>30</absolute>
</metric-margin>
</dynamic>
</path>
<auto-route>
  <auto-route-metric>
    <metric-constant-value>11111</metric-constant-value>
  </auto-route-metric>
  <include-prefixes>
    <include-prefix>
      <prefix-address>1.1.1.2/16</prefix-address>
    </include-prefix>
    <include-prefix>
      <prefix-address>1.1.1.3/32</prefix-address>
    </include-prefix>
  </include-prefixes>
  <force-sr-include/>
  <forward-class>1</forward-class>
</auto-route>
</policy>
<policy>
  <name>SR-CLI-ERO</name>
  <head-end>
    <name>PIOSXR-1</name>
  </head-end>
  <tail-end>7.7.7.7</tail-end>
  <color>100</color>
  <binding-sid>100</binding-sid>
  <performance-measurement>
    <liveness-detection>
      <profile>testLivenessProfile</profile>
```

```

    <invalidation-action>none</invalidation-action>
    <logging>
      <session-state-change/>
    </logging>
  </liveness-detection>
  <reverse-path-label>50</reverse-path-label>
</performance-measurement>
<path>
  <preference>100</preference>
  <dynamic>
    <constraints>
      <sid-limit>10</sid-limit>
    </constraints>
    <metric-type>te</metric-type>
    <metric-margin>
      <relative>40</relative>
    </metric-margin>
  </dynamic>
</path>
<path>
  <preference>200</preference>
  <explicit>
    <sid-list>
      <name>mysidlist</name>
      <weight>10</weight>
    </sid-list>
    <constraints>
      <affinity>
        <rule>
          <action>include-all</action>
          <color>GREEN</color>
          <color>RED</color>
        </rule>
      </affinity>
    </constraints>
  </explicit>
</path>
</policy>
<sid-list>
  <name>mysidlist</name>
  <sid>
    <index>1</index>
    <mpls>
      <label>17001</label>

```

```

    </mpls>
  </sid>
  <sid>
    <index>2</index>
    <mpls>
      <label>18001</label>
    </mpls>
  </sid>
</sid-list>
</policies>
</sr-te>
</config>

```

Working with Action Commands – Example Services

Use the action commands to perform specific tasks on services, such as service cleanup, to get the latest service modifications, and to redeploy services.

Service Cleanup Actions

Perform the Cleanup action to clean up either a specific device in the service or the entire service. Cleaning up the entire service cleans up all the devices in the service. For more information about service cleanup actions and the description of the various parameters, see [Working with Action Commands – CFP Services](#).

IETF-TE Example Services

The following example shows how to use the Cleanup action to clean up the IETF-TE service.

```

admin@ncs> request te tunnels actions cleanup service IETF-RSVP-TE no-networking
true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up IETF TE service: IETF-RSVP-TECleaning up RSVP TE Internal
Services: IETF-RSVP-TE-111.1.1.1-internal
  Removed all internal plan components
  Removing side-effect queue: /ncs:side-effect-queue/side-effect{37}
  Removed side-effects
  Removed kickers
Cleanup Successful for RSVP TE Internal Services
  Removed all external plan components

```

```

Removing zombie service: /ncs:zombies/ncs:service{"/te/tunnels/tunnel[name='IETF-
RSVP-TE']}
Removed zombie service
Removed side-effects
Removed kickers
Removing plan path: /te:te/tunnels/tunnel-plan{IETF-RSVP-TE}
Removed plan path
Removing zombie service: /ncs:zombies/ncs:service{"/te/tunnels/tunnel[name='IETF-
RSVP-TE']}
Removed zombie service
Cleanup Successful

```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

```

admin@ncs% request te tunnels actions cleanup service IETF-RSVP-TE no-networking
true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s)
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the service: IETF-RSVP-TE before cleanup

```

Cisco PM Example Services

The following example shows how to use the Cleanup action to clean up the Cisco PM service.

To clean up Cisco PM service per device

```

admin@ncs> request pm pm-actions cleanup service PM-CLI device PIOSXR-0 no-
networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up PM service: PM-CLI
  Removing service /pm/svc-profiles{PM-CLI}/devices{PIOSXR-0}
  Removed service /pm/svc-profiles{PM-CLI}/devices{PIOSXR-0}Cleaning up PM Internal
Services: PM-CLI

```



```

Removed all internal plan components
Removing zombie service: /ncs:zombies/ncs:service{"/cisco-pm-fp-internal:pm-internal/device-profiles[name='PM-CLI'][device='PIOSXR-0']}
Removed zombie service
Removed commit-queue-recovery-data
Cleanup Successful for PM Internal Services
Removed all external plan components
Cleanup Successful

```

To clean up Cisco PM service on all devices

```

admin@ncs% request pm pm-actions cleanup service PM-CLI no-networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up PM service: PM-CLI
  Removing service /pm/svc-profiles{PM-CLI}
  Removed service /pm/svc-profiles{PM-CLI}Cleaning up PM Internal Services: PM-CLI
  Removed all internal plan components
  Removing zombie service: /ncs:zombies/ncs:service{"/cisco-pm-fp-internal:pm-internal/device-profiles[name='PM-CLI'][device='PIOSXR-0']}
  Removed zombie service
  Removing zombie service: /ncs:zombies/ncs:service{"/cisco-pm-fp-internal:pm-internal/device-profiles[name='PM-CLI'][device='PIOSXR-1']}
  Removed zombie service
  Removed commit-queue-recovery-data
  Cleanup Successful for PM Internal Services
  Removed all external plan components
  Removing zombie service: /ncs:zombies/ncs:service{"/pm/svc-profiles[name='PM-CLI']}
  Removed zombie service
  Removing zombie service: /ncs:zombies/ncs:service{"/pm/svc-profiles[name='PM-CLI']}
  Removed zombie service
  Cleanup Successful

```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

To validate cleanup for Cisco PM service per device

```

admin@cfs-node% request pm pm-actions cleanup service PM-CLI device PIOSXR-0 no-
networking true
#####
#           Warning           #
#####
are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the device: PIOSXR-0 in service: PM-CLI before cleanup

```

To validate cleanup for Cisco PM service on all devices

```

admin@cfs-node% request pm pm-actions cleanup service PM-CLI no-networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the service: PM-CLI before cleanup

```

IETF-NSS Example Services

The following example shows how to use the Cleanup action to clean up the IETF-NSS service.

```

admin@ncs% request network-slice-services actions cleanup service L2-HUB-SPOKE no-
networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-
sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success true
detail Cleaning up IETF NSS service: L2-HUB-SPOKE
Cleaning up l2vpn-ntw service: NSS-L2-HUB-SPOKE-internal

Cleaning up L2vpn service: NSS-L2-HUB-SPOKE-internal
Cleaning up L2vpn internal site service: NSS-L2-HUB-SPOKE-internal PIOSXR-1_1-0
  Removed all plan components
  Removing zombie service: /ncs:zombies/ncs:service{"/flat-L2vpn-internal-site-
service[name='NSS-L2-HUB-SPOKE-internal'][site-name='PIOSXR-1_1-0']}

```

```

Removed zombie service
Removed commit-queue-recovery-data
Cleanup Successful for L2vpn internal site service: NSS-L2-HUB-SPOKE-internal
PIOSXR-1_1-0

Removed all external plan components
Removing zombie service: /ncs:zombies/ncs:service{"/flat-L2vpn[name='NSS-L2-HUB-SPOKE-internal']"}
Removed zombie service
Removing plan path: /cisco-flat-L2vpn-fp:flat-L2vpn-plan{NSS-L2-HUB-SPOKE-internal}
Removed plan path
Removing zombie service: /ncs:zombies/ncs:service{"/flat-L2vpn[name='NSS-L2-HUB-SPOKE-internal']"}
Removed zombie service
Cleanup Successful
Removed all plan components
Removing plan path: /l2vpn-ntw/vpn-services/vpn-service-plan{NSS-L2-HUB-SPOKE-internal}
Removed plan path
Removing zombie service: /ncs:zombies/ncs:service{"/l2vpn-ntw/vpn-services/vpn-service[vpn-id='NSS-L2-HUB-SPOKE-internal']"}
Removed zombie service
Cleanup Successful for L2NM
Removed all plan components
Removing zombie service: /ncs:zombies/ncs:service{"/network-slice-services/slice-service[service-id='L2-HUB-SPOKE']"}
Removed zombie service
Cleanup Successful for IETF NSS

```

The user must delete the service before executing the cleanup action.

Validation will be triggered as shown below if the service is not deleted before executing cleanup:

To validate cleanup for Cisco NSS service on all devices

```

admin@cfs-node% request network-slice-services actions cleanup service L2-ANY-TO-ANY no-networking true
#####
#           Warning           #
#####
You are about to forcefully cleanup a T-SDN service.
This will affect the deploying service and leave network device(s) & NSO out-of-sync (for no-networking=true).
Are you sure you want to proceed? [no,yes] yes
success false
detail Delete the service: L2-ANY-TO-ANY before cleanup

```

Error Recovery Action

Use the error recovery action to manually recover a service from an error when the automatic error recovery fails or is not set up. For more information about the error recovery action, see [Working with Action Commands – CFP Services](#).

This topic discusses how to execute the error recovery action on the example services.

IETF-TE Example Services

The following is an example of a failed plan for a service.

```
admin@ncs> show te tunnels tunnel-plan
```

TYPE	NAME	BACK		STATUS CODE	STATE	STATUS	WHEN	POST ACTION	
		TRACK	GOAL					ref	STATUS
self	<u>self</u>	false	-	-	init, ready	reached failed	2020-08-25T00:11:24 2020-08-25T00:11:26	-	-
source	<u>111.1.1.1</u>	false	-	TSDN-IETF-TE-301	init, ready	reached reached	2020-08-25T00:11:24 2020-08-25T00:11:24	-	-
destination	<u>222.2.2.2</u>	false	-	-	init, ready	reached reached	2020-08-25T00:11:24 2020-08-25T00:11:26	-	-

```
plan failed
```

```
plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused (Connection refused) in new state"
```

```
plan status-code-detail source 111.1.1.1
```

```
code TSDN-IETF-TE-301
```

```
context "Device unreachable"
```

```
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
```

```
severity ERROR
```

```
recommended-action "Check device connectivity from NSO and perform recovery
steps."
```

```
impacted-device PIOSXR-0
```

The device comes back up and you can request error recovery on the service. The plan is successful after the service is recovered successfully.

The following shows error recovery for the service.

```
admin@ncs> request te tunnels actions error-recovery service IETF-RSVP-TE source
111.1.1.1 sync-direction sync-from
```

```
#####
```

```
# Warning #
```

```
#####
```

```
You are about to recover a T-SDN service.
```

```
This will issue a sync-from on the device.
```

```
Are you sure you want to proceed? [no,yes] y
```

```
success true
```

```
detail Recovering IETF TE service: IETF-RSVP-TE
```

```
Recovered create failure on PIOSXR-0
  Removed cq_error_path: None
Recovery Complete for RSVP TE Services
Recovery Complete
```

Error recovery action under a service - Create failure

```
admin@ncs> request te tunnels tunnel IETF-RSVP-TE error-recovery sync-direction
sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync-from on the device.
Are you sure you want to proceed? [no,yes] yes
success true
detail Recovering IETF TE service: IETF-RSVP-TE
Recovered create failure on PIOSXR-0
  Removed cq_error_path: None
Recovery Complete for RSVP TE Services
Recovery Complete
```

Error recovery action under a service - For Already in Sync Service

If services are already in sync and user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery required.

```
admin@ncs> request te tunnels actions error-recovery service IETF-RSVP-TE sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:IETF-RSVP-TE and device are not failed
```

No Such Device in a Service

```
admin@ncs> request te tunnels actions error-recovery service IETF-RSVP-TE source
3.3.3.3 sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
```

```
success false
detail No recovery required. No such failed node 3.3.3.3 found for recovery in
service
```

Per Source/Destination when device has not Failed

```
admin@ncs> request te tunnels actions error-recovery service IETF-RSVP-TE source
111.1.1.1 sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:IETF-RSVP-TE and device:111.1.1.1 are not
failed
```

IETF-NSS Example Services

The following is an example of a failed plan for a service.

```
admin@ncs% run show network-slice-services slice-service-plan L2-ANY-TO-ANY plan
```

TYPE	NAME	BACK TRACK	GOAL	STATUS CODE	NODE	STATE	STATUS	WHEN	ref	POST ACTION STATUS
self	self	false	-	-	-	init	reached	2022-09-02T17:45:26	-	-
						ready	failed	2022-09-02T17:45:28	-	-
sdp	0	false	-	TSDN-L2VPN-301	PIOSXR-0	init	reached	2022-09-02T17:45:26	-	-
						ietf-nss-nano:config-apply	reached	2022-09-02T17:45:26	-	-
						ready	failed	2022-09-02T17:45:28	-	-
sdp	1	false	-	-	PIOSXR-1	init	reached	2022-09-02T17:45:26	-	-
						ietf-nss-nano:config-apply	reached	2022-09-02T17:45:26	-	-
						ready	reached	2022-09-02T17:45:28	-	-

```
plan failed
plan error-info message "Failed to connect to device PIOSXR-0: connection refused:
NEDCOM CONNECT: Connection refused (Connection refused) in new state"
plan status service-tag-service [ L2 ]
plan status rt-allocation-data hub-rt 0:1:1
plan status evi-allocation-data evi-id 1
plan status-code-detail site PIOSXR-0_0-0
code TSDN-L2VPN-301
context "Device unreachable"
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM
CONNECT: Connection refused (Connection refused) in new state"
severity ERROR
recommended-action "Check device connectivity from NSO and perform recovery
steps."
impacted-device PIOSXR-0
```

The device comes back up and you can request error recovery on the service. The plan is successful after the service is recovered successfully.

The following shows error recovery for the service.

```
admin@ncs% request network-slice-services actions error-recovery service L2-ANY-TO-
ANY sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes detail Recovering IETF NSS service:
L2-ANY-TO-ANY sync_direction: sync-from

Recovering L2NM service: NSS-L2-ANY-TO-ANY-internal sync_direction: sync-from

site:
L2VPN Recovery action for internal site : L2NM-NSS-L2-ANY-TO-ANY-internal-internal
PIOSXR-0_0-0 sync-from
Successful recovery for create failure on device: PIOSXR-0
Removed cq_error_path: None

Recovery Complete for L2VPN Internal Service for device PIOSXR-0

site:
L2VPN Recovery action for internal site : L2NM-NSS-L2-ANY-TO-ANY-internal-internal
PIOSXR-1_1-0 sync-from
No failure found for site : L2NM-NSS-L2-ANY-TO-ANY-internal-internal PIOSXR-1

Recovery Complete for L2VPN Internal Service for device PIOSXR-1

Recovery Complete
```

Error recovery action under a service – For Already in Sync Service

If services are already in sync and the user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery is required.

```
admin@ncs% request network-slice-services actions error-recovery service L2-P2P
sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
```

```
detail Recovering IETF NSS service: L2-P2P sync_direction: sync-from
```

No recovery required. Service:igp and device are not failed

No recovery required. Service NSS-L2-P2P-internal and devices are in sync. No failure found in plan for NSS-L2-P2P-internal

Performance Measurement Example Services

The following is an example of a failed plan for a service.

```
admin@ncs% run show pm pm-plan
```

TYPE	NAME	BACK		STATUS CODE	STATE	STATUS	WHEN	POST ACTION	
		TRACK	GOAL					ref	STATUS
--									
self	self	false	-	-	init ready	reached	2022-09-02T19:16:40	-	-
device	PIOSXR-0	false	-	TSDN-PM-301	init ready	failed	2022-09-02T19:18:18	-	-
					cisco-pm-fp-nano-plan-services:config:apply	reached	2022-09-02T19:16:40	-	-
device	PIOSXR-1	false	-	-	init ready	failed	2022-09-02T19:18:18	-	-
					cisco-pm-fp-nano-plan-services:config:apply	reached	2022-09-02T19:16:40	-	-
					ready	reached	2022-09-02T19:16:44	-	-

```
plan failed
```

```
plan error-info message "Failed to connect to device PIOSXR-0: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state"
```

```
plan status-code-detail device PIOSXR-0
```

```
code TSDN-PM-301
context "Device unreachable"
context-msg "Failed to connect to device PIOSXR-0: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state"
severity ERROR
recommended-action "Check device connectivity from NSO and perform recovery steps."
impacted-device PIOSXR-0
```

The device comes back up and you can request error recovery on the service. The plan is successful after the service is recovered successfully.

Error recovery action under a service - Create failure

```
admin@ncs> request pm pm-actions error-recovery service PM-CLI sync-direction sync-from
```

```
#####
# Warning #
#####
```

```
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success true
```

```
detail Recovering PM service: PM-CLI
```

```
Successful recovery for create failure on device: PIOSXR-0
```



```
Removed cq_error_path: None
Recovery Complete for PM Internal Services
Recovery Complete
```

Error recovery action under a service – For Already in Sync Service

If services are already in sync and user invokes error recovery, the command will execute as a Fail with a message as shown below that the service is already in sync and no recovery required.

```
admin@ncs> request pm pm-actions error-recovery service PM-CLI sync-direction sync-
from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:PM-CLI and device are not failed
```

No Such Device in a Service

```
admin@ncs> request pm pm-actions error-recovery service PM-CLI device ABC sync-
direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. no failed device for PM-CLI
```

Per Source/Destination when device has not Failed

```
admin@ncs> request pm pm-actions error-recovery service PM-CLI device PIOSXR-0
sync-direction sync-from
#####
#           Warning           #
#####
You are about to recover a T-SDN service.
This will issue a sync on the device.
Are you sure you want to proceed? [no,yes] yes
success false
detail No recovery required. Service:PM-CLI and device:PIOSXR-0 are not failed
```

get-modifications Action

The **get-modifications** action is like the service get-modifications action. You can perform this action at the service level. By default, the output format is CLI.

This action returns the data the service modified in the CDB (whether it is device configuration or internal data), either in the CLI curly bracket format or NETCONF XML edit configuration format.

If the parameter 'reverse' is given, the modifications required to 'reverse' the effect of the service is shown. This will be applied if the service is deleted. This data is always available. The parameters **shallow** and **deep** control if the modifications must be displayed for the service only or for all the modified services, respectively.

This data is available only if the parameter **/services/global-settings/collect-forward-diff** is set to true as follows before you create the service.

```
admin@ncs% show services global-settings collect-forward-diff
collect-forward-diff true;
```

Note: Setting this parameter to true can have severe performance implications. For more information about this parameter, see the NSO documentation.

IETF-TE Example Services

Perform the get-modifications action at the service level. The following is a sample command for the action.

```
admin@ncs> request te tunnels tunnel IETF-RSVP-TE get-modifications
cli {
  local-node {
    data devices {
      device PIOSXR-0 {
        config {
          explicit-path {
+           name IETF-RSVP-TE-PATH-1 {
+             index 1 {
+               keyword next-address;
+               hop-type loose;
+               ipv4 {
+                 unicast 1.1.1.1;
+               }
+             }
+             index 2 {
+               keyword next-label;
+               label 508;
+             }
+           }
+         }
      }
    }
  }
}
```

```
interface {
+     tunnel-te 1234 {
+         description RSVP_TE;
+         ipv4 {
+             unnumbered {
+                 Loopback 3;
+             }
+         }
+         signalled-name IETF-RSVP-TE;
+         signalled-bandwidth {
+             bandwidth 94967295;
+         }
+         priority {
+             setup 3;
+             hold-value 2;
+         }
+         autoroute {
+             announce {
+                 metric {
+                     relative 7;
+                 }
+             }
+         }
+         destination 222.2.2.2;
+         path-selection {
+             metric {
+                 metric-type te;
+             }
+         }
+         path-option 1 {
+             explicit {
+                 name IETF-RSVP-TE-PATH-1;
+             }
+         }
+         path-option 2 {
+             dynamic {
+             }
+         }
+         path-option 3 {
+             dynamic {
+             }
+         }
+         path-option 4 {
+             dynamic {
+             }
+         }
+     }
+ }
```

```

+         }
+         address {
+             ipv4 1.2.3.4;
+         }
+     }
+ }
+ }
+ }
+ }
}
}
}
}

```

IETF NSS Example Services

Perform the get-modifications action at the service level. The following is a sample command for the action.

```
admin@ncs% request network-slice-services slice-service L2-HUB-SPOKE get-modifications
```

```
cli {
    local-node {
        data +flat-L2vpn NSS-L2-ANY-TO-ANY-internal {
            + service-type evpn-multipoint;
            + flat-L2vpn-evpn-multipoint {
            +     evi-id 1;
            +     topology E-LAN;
            +     bridge-group L2-ANY-TO-ANY;
            +     vpn-target {
            +         root-rt-value 1:1;
            +     }
            +     site PIOSXR-0_0-0 {
            +         pe PIOSXR-0;
            +         if-type GigabitEthernet;
            +         if-id 0;
            +         if-encap dot1q;
            +         vlan-id 100;
            +         sub-if-id 100;
            +         rewrite {
            +             ingress {
            +                 pop 1;
            +                 mode symmetric;
            +             }
            +         }
            +     }
            + }
            + }
            + }
        }
    }
}

```

```

+         advertise-mac {
+             enable true;
+         }
+         control-word-disable true;
+         vpn-target 2:2 {
+             rt-type import;
+         }
+         vpn-target 3:3 {
+             rt-type import;
+         }
+     }
+     site PIOSXR-1_1-0 {
+         pe PIOSXR-1;
+         if-type GigabitEthernet;
+         if-id 1;
+         if-encap dot1q;
+         vlan-id 100;
+         sub-if-id 100;
+         rewrite {
+             ingress {
+                 pop 1;
+                 mode symmetric;
+             }
+         }
+         advertise-mac {
+             enable true;
+         }
+         control-word-disable true;
+         vpn-target 2:2 {
+             rt-type import;
+         }
+         vpn-target 3:3 {
+             rt-type import;
+         }
+     }
+ }
+}
+flat-L2vpn NSS-L2-HUB-SPOKE-internal {
+     service-type evpn-multipoint;
+     flat-L2vpn-evpn-multipoint {
+         evi-id 2;
+         topology E-TREE;
+         bridge-group L2-HUB-SPOKE;
+         vpn-target {

```

```
+         root-rt-value 2:2;
+         leaf-rt-value 3:3;
+     }
+     site PIOSXR-1_1-0 {
+         pe PIOSXR-1;
+         if-type GigabitEthernet;
+         if-id 1;
+         qos {
+             input-policy L2-input;
+             output-policy L2-output;
+         }
+         if-encap dot1q;
+         vlan-id 100;
+         sub-if-id 100;
+         rewrite {
+             ingress {
+                 pop 1;
+                 mode symmetric;
+             }
+         }
+         leaf single-homing;
+         advertise-mac {
+             enable true;
+         }
+         control-word-disable true;
+         vpn-target 1:1 {
+             rt-type import;
+         }
+         split-horizon {
+             enable true;
+         }
+         sr-te {
+             odn {
+                 route-policy NSS-Dedicated-min-delay-internal;
+                 route-policy-type both;
+             }
+         }
+     }
+     site PIOSXR-2_0-0 {
+         pe PIOSXR-2;
+         if-type GigabitEthernet;
+         if-id 0;
+         qos {
+             input-policy L2-input;
```

```

+         output-policy L2-output;
+     }
+     if-encap dot1q;
+     vlan-id 100;
+     sub-if-id 100;
+     rewrite {
+         ingress {
+             pop 1;
+             mode symmetric;
+         }
+     }
+     advertise-mac {
+         enable true;
+     }
+     control-word-disable true;
+     vpn-target 1:1 {
+         rt-type import;
+     }
+     sr-te {
+         odn {
+             route-policy NSS-Dedicated-min-delay-internal;
+             route-policy-type both;
+         }
+     }
+ }
+ }
+}
devices {
    device PIOSXR-0 {
        config {
            interface {
                GigabitEthernet 0 {
+                 description "T-SDN Interface";
                }
                GigabitEthernet-subinterface {
+                 GigabitEthernet 0.100 {
+                     mode l2transport;
+                     description "T-SDN Interface";
+                     encapsulation {
+                         dot1q {
+                             vlan-id 100;
+                         }
+                     }
+                 }
+                 rewrite {

```

```

+             ingress {
+                 tag {
+                     pop 1;
+                     mode symmetric;
+                 }
+             }
+         }
+     }
+ }
+
+ evpn {
+     evi 1 {
+         bgp {
+             route-target {
+                 import 1:1;
+                 import 2:2;
+                 import 3:3;
+                 export 1:1;
+             }
+         }
+         control-word-disable;
+         advertise-mac {
+         }
+     }
+ }
+
+ l2vpn {
+     bridge {
+         group L2-ANY-TO-ANY {
+             bridge-domain L2-ANY-TO-ANY_evi_1 {
+                 interface GigabitEthernet0.100;
+                 evi 1;
+             }
+         }
+     }
+ }
+
+ segment-routing {
+     traffic-eng {
+         on-demand {
+             color 30 {
+                 dynamic {
+                     metric {
+                         type latency;
+                     }
+                 }
+             }
+         }
+     }
+ }

```



```

+           }
+         }
+       }
+     }
+   }
+ }
+
+ device PIOSXR-1 {
+   config {
+     interface {
+       GigabitEthernet 1 {
+         description "T-SDN Interface";
+       }
+       GigabitEthernet-subinterface {
+         GigabitEthernet 1.100 {
+           mode l2transport;
+           description "T-SDN Interface";
+           encapsulation {
+             dot1q {
+               vlan-id 100;
+             }
+           }
+           service-policy {
+             input {
+               name L2-input;
+             }
+             output {
+               name L2-output;
+             }
+           }
+           rewrite {
+             ingress {
+               tag {
+                 pop 1;
+                 mode symmetric;
+               }
+             }
+           }
+         }
+       }
+     }
+   }
+   evpn {
+     evi 1 {
+       bgp {
+         route-target {

```

```

+             import 1:1;
+             import 2:2;
+             import 3:3;
+             export 1:1;
+         }
+     }
+     control-word-disable;
+     advertise-mac {
+     }
+ }
+     evi 2 {
+         bgp {
+             route-target {
+                 import 1:1;
+                 import 2:2;
+                 export 3:3;
+             }
+             route-policy {
+                 export SET_COLOR_EVPN_MP_service_NSS-
Dedicated-min-delay-internal;
+                 import SET_COLOR_EVPN_MP_service_NSS-
Dedicated-min-delay-internal;
+             }
+         }
+     control-word-disable;
+     advertise-mac {
+     }
+ }
+     l2vpn {
+         bridge {
+             group L2-ANY-TO-ANY {
+                 bridge-domain L2-ANY-TO-ANY_evi_1 {
+                     interface GigabitEthernet1.100;
+                     evi 1;
+                 }
+             }
+             group L2-HUB-SPOKE {
+                 bridge-domain L2-HUB-SPOKE_evi_2 {
+                     interface GigabitEthernet1.100 {
+                         split-horizon {
+                             group;
+                         }
+                     }
+                 }
+                 evi 2;

```



```

+         service-policy {
+             input {
+                 name L2-input;
+             }
+             output {
+                 name L2-output;
+             }
+         }
+         rewrite {
+             ingress {
+                 tag {
+                     pop 1;
+                     mode symmetric;
+                 }
+             }
+         }
+     }
+ }
+
+     evpn {
+         evi 2 {
+             bgp {
+                 route-target {
+                     import 1:1;
+                     import 2:2;
+                     import 3:3;
+                     export 2:2;
+                 }
+                 route-policy {
+                     export SET_COLOR_EVPN_MP_service_NSS-
Dedicated-min-delay-internal;
+                     import SET_COLOR_EVPN_MP_service_NSS-
Dedicated-min-delay-internal;
+                 }
+             }
+         }
+         control-word-disable;
+         advertise-mac {
+         }
+     }
+ }
+
+     l2vpn {
+         bridge {
+             group L2-HUB-SPOKE {
+                 bridge-domain L2-HUB-SPOKE_evi_2 {
+                     interface GigabitEthernet0.100;

```

```

+                 evi 2;
+                 }
+             }
+         }
+
+         extcommunity-set {
+             opaque COLOR_30 {
+                 set 30;
+             }
+         }
+
+         route-policy SET_COLOR_EVPN_MP_service_NSS-Dedicated-
min-delay-internal {
+             value " set extcommunity color COLOR_30\r
";
+         }
+         segment-routing {
+             traffic-eng {
+                 on-demand {
+                     color 30 {
+                         dynamic {
+                             metric {
+                                 type latency;
+                             }
+                         }
+                     }
+                 }
+             }
+         }
+     }
+ }
+
+ cisco-sr-te-cfp:sr-te {
+     odn {
+         odn-template min-delay {
+             head-end PIOSXR-0;
+             head-end PIOSXR-1;
+             head-end PIOSXR-2;
+         }
+     }
+ }
+
+ resource-pools {
+     id-pool evi-id-pool {
+         allocation NSS-L2-ANY-TO-ANY-internal-evi_id {
+             username admin;
+         }
+     }
+ }

```

```

+           allocating-service /l2vpn-ntw/vpn-services/vpn-
service[vpn-id='NSS-L2-ANY-TO-ANY-internal'];
+           redeploy-type default;
+           request {
+               sync false;
+           }
+       }
+       allocation NSS-L2-HUB-SPOKE-internal-evi_id {
+           username admin;
+           allocating-service /l2vpn-ntw/vpn-services/vpn-
service[vpn-id='NSS-L2-HUB-SPOKE-internal'];
+           redeploy-type default;
+           request {
+               sync false;
+           }
+       }
+   }
+   id-pool rt-id-pool {
+       allocation NSS-L2-ANY-TO-ANY-internal-hub_rt_value {
+           username admin;
+           allocating-service /l2vpn-ntw/vpn-services/vpn-
service[vpn-id='NSS-L2-ANY-TO-ANY-internal'];
+           redeploy-type default;
+           request {
+               sync false;
+           }
+       }
+       allocation NSS-L2-HUB-SPOKE-internal-hub_rt_value {
+           username admin;
+           allocating-service /l2vpn-ntw/vpn-services/vpn-
service[vpn-id='NSS-L2-HUB-SPOKE-internal'];
+           redeploy-type default;
+           request {
+               sync false;
+           }
+       }
+       allocation NSS-L2-HUB-SPOKE-internal-spoke_rt_value {
+           username admin;
+           allocating-service /l2vpn-ntw/vpn-services/vpn-
service[vpn-id='NSS-L2-HUB-SPOKE-internal'];
+           redeploy-type default;
+           request {
+               sync false;
+           }
+       }
+   }

```

```

+   }
+ }
l2vpn-ntw {
    vpn-profiles {
        valid-provider-identifiers {
+           qos-profile-identifier L2-input {
+           }
+           qos-profile-identifier L2-output {
+           }
        }
    }
    vpn-services {
+       vpn-service NSS-L2-ANY-TO-ANY-internal {
+           vpn-nodes {
+               vpn-node PIOSXR-0;
+               vpn-node PIOSXR-1;
+           }
+       }
+       vpn-service NSS-L2-HUB-SPOKE-internal {
+           vpn-type mpls-evpn;
+           vpn-service-topology hub-spoke;
+           vpn-nodes {
+               vpn-node PIOSXR-1 {
+                   role spoke-role;
+                   signaling-option {
+                       evpn-policies {
+                           mac-learning-mode control-plane;
+                           vpn-target 0 {
+                               route-targets 0:1:1;
+                               route-target-type import;
+                           }
+                           vpn-policies {
+                               import-policy NSS-Dedicated-min-
delay-internal;
+                               export-policy NSS-Dedicated-min-
delay-internal;
+                           }
+                       }
+                   }
+               }
+           }
+           vpn-network-accesses {
+               vpn-network-access 1-0 {
+                   interface-id GigabitEthernet1;
+                   connection {
+                       encapsulation {
+                           encap-type dot1q;

```

```
+          dot1q {
+              cvlan-id 100;
+              tag-operations {
+                  pop 1;
+                  mode symmetric;
+              }
+          }
+      }
+  }
+  service {
+      qos {
+          qos-profile {
+              qos-profile L2-input {
+                  direction wan-to-site;
+              }
+              qos-profile L2-output {
+                  direction site-to-wan;
+              }
+          }
+      }
+  }
+  }
+  }
+  }
+  te-service-mapping {
+      te-mapping {
+          odn {
+              route-policy NSS-Dedicated-min-
delay-internal;
+          }
+      }
+  }
+  }
+  }
+  }
+  vpn-node PIOSXR-2 {
+      role hub-role;
+      signaling-option {
+          evpn-policies {
+              mac-learning-mode control-plane;
+              vpn-target 0 {
+                  route-targets 0:1:1;
+                  route-target-type import;
+              }
+          }
+          vpn-policies {
+              import-policy NSS-Dedicated-min-
delay-internal;
```



```

+
+           export-policy NSS-Dedicated-min-
delay-internal;
+
+       }
+
+   }
+
+   vpn-network-accesses {
+       vpn-network-access 0-0 {
+           interface-id GigabitEthernet0;
+           connection {
+               encapsulation {
+                   encap-type dot1q;
+                   dot1q {
+                       cvlan-id 100;
+                       tag-operations {
+                           pop 1;
+                           mode symmetric;
+                       }
+                   }
+               }
+           }
+       }
+       service {
+           qos {
+               qos-profile {
+                   qos-profile L2-input {
+                       direction wan-to-site;
+                   }
+                   qos-profile L2-output {
+                       direction site-to-wan;
+                   }
+               }
+           }
+       }
+   }
+
+   te-service-mapping {
+       te-mapping {
+           odn {
+               route-policy NSS-Dedicated-min-
delay-internal;
+           }
+       }
+   }
+
+   }
+
+   auto-evi-id;

```

```

+         vpn-target {
+             auto-hub-rt;
+             auto-spoke-rt;
+         }
+         bridge-group L2-HUB-SPOKE;
+     }
+ }
}
}
}
}
}
}

```

Performance Measurement Example Services

Perform the get-modifications action at the service level. The following is a sample command for the action.

```

admin@ncs% request pm svc-profiles PM-CLI get-modifications
cli {
    local-node {
        data devices {
            device PIOSXR-0 {
                config {
+                 performance-measurement {
+                     interface HundredGigE0/0/0/0 {
+                         delay-measurement {
+                             advertise-delay 10000;
+                             delay-profile {
+                                 name DelayInterface2;
+                             }
+                         }
+                     }
+                     next-hop {
+                         ipv4 10.1.1.2;
+                     }
+                 }
+                 interface HundredGigE0/0/0/1 {
+                     delay-measurement {
+                         advertise-delay 20000;
+                         delay-profile {
+                             name explicitInterfaceDelayProfile;
+                         }
+                     }
+                 }
+                 next-hop {
+                     ipv6 110:101::102;
+                 }

```

```

+         }
+     protocol {
+         twamp-light {
+             measurement {
+                 delay {
+                     unauthenticated {
+                         ipv4 {
+                             timestamp1 {
+                                 label 2577;
+                             }
+                             timestamp2 {
+                                 label 350;
+                             }
+                         }
+                         ipv6 {
+                             timestamp1 {
+                                 label 900;
+                             }
+                             timestamp2 {
+                                 label 980;
+                             }
+                         }
+                         querier-dst-port 1026;
+                         querier-src-port 1029;
+                     }
+                 }
+             }
+         }
+     }
+     delay-profile {
+         sr-policy-name {
+             sr-policy DelaySrPolicy2 {
+                 advertisement {
+                     accelerated {
+                         minimum-change 900;
+                         threshold 90;
+                     }
+                     periodic {
+                         disabled;
+                         minimum-change 10000;
+                         interval 3000;
+                         threshold 10;
+                     }
+                 }
+                 logging {

```

```
+             delay-exceeded;
+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 50;
+             traffic-class 4;
+         }
+         sweep {
+             destination {
+                 ipv4 127.0.0.3;
+                 range 5;
+             }
+         }
+     }
+ }
+ sr-policy DelaySrPolicy3 {
+     advertisement {
+         accelerated {
+             minimum-change 900;
+             threshold 90;
+         }
+         periodic {
+             disabled;
+             minimum-change 10000;
+             interval 3000;
+             threshold 10;
+         }
+         logging {
+             delay-exceeded;
+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 50;
+             traffic-class 4;
+         }
+         sweep {
```

```
+           destination {
+             ipv4 127.0.0.3;
+             range 5;
+           }
+         }
+       }
+     }
+   sr-policy-default {
+     sr-policy {
+       default {
+         advertisement {
+           accelerated {
+             minimum-change 900;
+             threshold 90;
+           }
+           periodic {
+             disabled;
+             minimum-change 10000;
+             interval 3000;
+             threshold 10;
+           }
+           logging {
+             delay-exceeded;
+           }
+         }
+       probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+           dscp 50;
+           traffic-class 4;
+         }
+         sweep {
+           destination {
+             ipv4 127.0.0.3;
+             range 5;
+           }
+         }
+       }
+     }
+   }
+ }
```

```
+         interfaces-name {
+             interfaces DelayInterface2 {
+                 advertisement {
+                     accelerated {
+                         minimum-change 900;
+                         threshold 90;
+                     }
+                     periodic {
+                         disabled;
+                         minimum-change 10000;
+                         interval 3000;
+                         threshold 90;
+                     }
+                     logging {
+                         delay-exceeded;
+                     }
+                 }
+                 probe {
+                     measurement-mode one-way;
+                     computation-interval 90;
+                     burst-interval 80;
+                     tos {
+                         dscp 50;
+                         traffic-class 3;
+                     }
+                 }
+             }
+             interfaces DelayInterface3 {
+                 advertisement {
+                     accelerated {
+                         minimum-change 900;
+                         threshold 90;
+                     }
+                     periodic {
+                         disabled;
+                         minimum-change 10000;
+                         interval 3000;
+                         threshold 90;
+                     }
+                     logging {
+                         delay-exceeded;
+                     }
+                 }
+                 probe {
```

```
+           measurement-mode one-way;
+           computation-interval 90;
+           burst-interval 80;
+           tos {
+             dscp 50;
+             traffic-class 3;
+           }
+         }
+       }
+     }
+   interfaces-default {
+     interfaces {
+       default {
+         advertisement {
+           accelerated {
+             minimum-change 900;
+             threshold 90;
+           }
+           periodic {
+             disabled;
+             minimum-change 10000;
+             interval 3000;
+             threshold 90;
+           }
+           logging {
+             delay-exceeded;
+           }
+         }
+         probe {
+           measurement-mode one-way;
+           computation-interval 90;
+           burst-interval 80;
+           tos {
+             dscp 50;
+             traffic-class 3;
+           }
+         }
+       }
+     }
+   }
+ }
+ rsvp-te {
+   default {
+     advertisement {
+       accelerated {
```

```

+             minimum-change 900;
+             threshold 90;
+         }
+     periodic {
+         disabled;
+         minimum-change 10000;
+         interval 3000;
+         threshold 90;
+     }
+     logging {
+         delay-exceeded;
+     }
+ }
+ probe {
+     measurement-mode one-way;
+     computation-interval 90;
+     burst-interval 80;
+     tos {
+         dscp 50;
+         traffic-class 2;
+     }
+ }
+ }
+ }
+ endpoint {
+     default {
+         advertisement {
+             accelerated {
+                 minimum-change 900;
+                 threshold 90;
+             }
+             periodic {
+                 disabled;
+                 minimum-change 10000;
+                 interval 3000;
+                 threshold 90;
+             }
+             logging {
+                 delay-exceeded;
+             }
+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;

```



```
+         burst-interval 80;
+         tos {
+             dscp 50;
+         }
+     }
+ }
+ name DelayEndpointProfile2 {
+     advertisement {
+         accelerated {
+             minimum-change 900;
+             threshold 90;
+         }
+         periodic {
+             disabled;
+             minimum-change 10000;
+             interval 3000;
+             threshold 90;
+         }
+         logging {
+             delay-exceeded;
+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 40;
+         }
+     }
+ }
+ }
+ }
+ liveness-profile {
+     endpoint {
+         default {
+             liveness-detection {
+                 multiplier 7;
+                 logging {
+                     state-change;
+                 }
+             }
+         }
+         probe {
+             burst-interval 800;
```

```
+           measurement-mode loopback;
+           tos {
+             dscp 48;
+           }
+         }
+       }
+     name EndpointLiveness2 {
+       liveness-detection {
+         multiplier 7;
+         logging {
+           state-change;
+         }
+       }
+     probe {
+       burst-interval 800;
+       measurement-mode loopback;
+       tos {
+         dscp 48;
+       }
+     }
+   }
+   name EndpointLiveness3 {
+     liveness-detection {
+       multiplier 7;
+       logging {
+         state-change;
+       }
+     }
+     probe {
+       burst-interval 800;
+       measurement-mode loopback;
+       tos {
+         dscp 48;
+       }
+     }
+   }
+ }
+ sr-policy {
+   default {
+     liveness-detection {
+       multiplier 5;
+     }
+     probe {
+       burst-interval 600;
```

```
+           measurement-mode loopback;
+           tos {
+             dscp 48;
+           }
+           sweep {
+             destination {
+               ipv4 127.0.0.3;
+               range 5;
+             }
+           }
+         }
+       }
+     name SrPolicyLiveness2 {
+       liveness-detection {
+         multiplier 5;
+       }
+       probe {
+         burst-interval 600;
+         measurement-mode loopback;
+         tos {
+           dscp 48;
+         }
+         sweep {
+           destination {
+             ipv4 127.0.0.3;
+             range 5;
+           }
+         }
+       }
+     }
+     name SrPolicyLiveness3 {
+       liveness-detection {
+         multiplier 5;
+       }
+       probe {
+         burst-interval 600;
+         measurement-mode loopback;
+         tos {
+           dscp 48;
+         }
+         sweep {
+           destination {
+             ipv4 127.0.0.3;
+             range 5;
+           }
+         }
+       }
+     }
```

```

+                                     }
+                                     }
+                                     }
+                                 }
+                             }
+                         }
+                     }
+                 }
+             }
+         }
+     }
+ }
+
+ device PIOSXR-1 {
+     config {
+         performance-measurement {
+             interface HundredGigE0/0/0/0 {
+                 delay-measurement {
+                     advertise-delay 10000;
+                     delay-profile {
+                         name DelayInterface;
+                     }
+                 }
+                 next-hop {
+                     ipv4 10.1.1.2;
+                 }
+             }
+             interface HundredGigE0/0/0/1 {
+                 delay-measurement {
+                     advertise-delay 20000;
+                     delay-profile {
+                         name explicitInterfaceDelayProfile;
+                     }
+                 }
+                 next-hop {
+                     ipv6 110:101::102;
+                 }
+             }
+         }
+         protocol {
+             twamp-light {
+                 measurement {
+                     delay {
+                         unauthenticated {
+                             ipv4 {
+                                 timestamp1 {
+                                     label 2577;
+                                 }
+                             }
+                             timestamp2 {

```

```
+                label 350;
+            }
+        }
+        ipv6 {
+            timestamp1 {
+                label 900;
+            }
+            timestamp2 {
+                label 980;
+            }
+        }
+        querier-dst-port 1026;
+        querier-src-port 1029;
+    }
+}
+}
+}
+    delay-profile {
+        sr-policy-name {
+            sr-policy DelaySrPolicy2 {
+                advertisement {
+                    accelerated {
+                        minimum-change 900;
+                        threshold 90;
+                    }
+                    periodic {
+                        disabled;
+                        minimum-change 10000;
+                        interval 3000;
+                        threshold 10;
+                    }
+                    logging {
+                        delay-exceeded;
+                    }
+                }
+                probe {
+                    measurement-mode one-way;
+                    computation-interval 90;
+                    burst-interval 80;
+                    tos {
+                        dscp 50;
+                        traffic-class 4;
+                    }
+                }
+            }
+        }
+    }
+}
```

```

+         sweep {
+             destination {
+                 ipv4 127.0.0.3;
+                 range 5;
+             }
+         }
+     }
+ }
+ sr-policy DelaySrPolicy3 {
+     advertisement {
+         accelerated {
+             minimum-change 900;
+             threshold 90;
+         }
+         periodic {
+             disabled;
+             minimum-change 10000;
+             interval 3000;
+             threshold 10;
+         }
+         logging {
+             delay-exceeded;
+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 50;
+             traffic-class 4;
+         }
+         sweep {
+             destination {
+                 ipv4 127.0.0.3;
+                 range 5;
+             }
+         }
+     }
+ }
+ sr-policy-default {
+     sr-policy {
+         default {

```

```
+         advertisement {
+             accelerated {
+                 minimum-change 900;
+                 threshold 90;
+             }
+             periodic {
+                 disabled;
+                 minimum-change 10000;
+                 interval 3000;
+                 threshold 10;
+             }
+             logging {
+                 delay-exceeded;
+             }
+         }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 50;
+             traffic-class 4;
+         }
+         sweep {
+             destination {
+                 ipv4 127.0.0.3;
+                 range 5;
+             }
+         }
+     }
+ }
+ }
+ }
+ interfaces-name {
+     interfaces DelayInterface2 {
+         advertisement {
+             accelerated {
+                 minimum-change 900;
+                 threshold 90;
+             }
+             periodic {
+                 disabled;
+                 minimum-change 10000;
+                 interval 3000;
```

```

+         threshold 90;
+     }
+     logging {
+         delay-exceeded;
+     }
+ }
+ probe {
+     measurement-mode one-way;
+     computation-interval 90;
+     burst-interval 80;
+     tos {
+         dscp 50;
+         traffic-class 3;
+     }
+ }
+ }
+ interfaces DelayInterface3 {
+     advertisement {
+         accelerated {
+             minimum-change 900;
+             threshold 90;
+         }
+         periodic {
+             disabled;
+             minimum-change 10000;
+             interval 3000;
+             threshold 90;
+         }
+         logging {
+             delay-exceeded;
+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 50;
+             traffic-class 3;
+         }
+     }
+ }
+ }
+ interfaces-default {

```



```
+         interfaces {
+             default {
+                 advertisement {
+                     accelerated {
+                         minimum-change 900;
+                         threshold 90;
+                     }
+                     periodic {
+                         disabled;
+                         minimum-change 10000;
+                         interval 3000;
+                         threshold 90;
+                     }
+                     logging {
+                         delay-exceeded;
+                     }
+                 }
+                 probe {
+                     measurement-mode one-way;
+                     computation-interval 90;
+                     burst-interval 80;
+                     tos {
+                         dscp 50;
+                         traffic-class 3;
+                     }
+                 }
+             }
+         }
+     }
+     rsvp-te {
+         default {
+             advertisement {
+                 accelerated {
+                     minimum-change 900;
+                     threshold 90;
+                 }
+                 periodic {
+                     disabled;
+                     minimum-change 10000;
+                     interval 3000;
+                     threshold 90;
+                 }
+                 logging {
+                     delay-exceeded;
+                 }
+             }
+         }
+     }
+ }
```

```

+         }
+     }
+     probe {
+         measurement-mode one-way;
+         computation-interval 90;
+         burst-interval 80;
+         tos {
+             dscp 50;
+             traffic-class 2;
+         }
+     }
+ }
+ }
+ endpoint {
+     default {
+         advertisement {
+             accelerated {
+                 minimum-change 900;
+                 threshold 90;
+             }
+             periodic {
+                 disabled;
+                 minimum-change 10000;
+                 interval 3000;
+                 threshold 90;
+             }
+             logging {
+                 delay-exceeded;
+             }
+         }
+         probe {
+             measurement-mode one-way;
+             computation-interval 90;
+             burst-interval 80;
+             tos {
+                 dscp 50;
+             }
+         }
+     }
+ }
+ name DelayEndpointProfile2 {
+     advertisement {
+         accelerated {
+             minimum-change 900;
+             threshold 90;

```



```

+         }
+     }
+     probe {
+         burst-interval 800;
+         measurement-mode loopback;
+         tos {
+             dscp 48;
+         }
+     }
+ }
+
+ name EndpointLiveness3 {
+     liveness-detection {
+         multiplier 7;
+         logging {
+             state-change;
+         }
+     }
+     probe {
+         burst-interval 800;
+         measurement-mode loopback;
+         tos {
+             dscp 48;
+         }
+     }
+ }
+ }
+
+ sr-policy {
+     default {
+         liveness-detection {
+             multiplier 5;
+         }
+         probe {
+             burst-interval 600;
+             measurement-mode loopback;
+             tos {
+                 dscp 48;
+             }
+             sweep {
+                 destination {
+                     ipv4 127.0.0.3;
+                     range 5;
+                 }
+             }
+         }
+     }
+ }

```

```

+         }
+         name SrPolicyLiveness2 {
+             liveness-detection {
+                 multiplier 5;
+             }
+             probe {
+                 burst-interval 600;
+                 measurement-mode loopback;
+                 tos {
+                     dscp 48;
+                 }
+                 sweep {
+                     destination {
+                         ipv4 127.0.0.3;
+                         range 5;
+                     }
+                 }
+             }
+         }
+         name SrPolicyLiveness3 {
+             liveness-detection {
+                 multiplier 5;
+             }
+             probe {
+                 burst-interval 600;
+                 measurement-mode loopback;
+                 tos {
+                     dscp 48;
+                 }
+                 sweep {
+                     destination {
+                         ipv4 127.0.0.3;
+                         range 5;
+                     }
+                 }
+             }
+         }
+     }
+ }

```

```

    }
}

```

Service re-deploy Action

Use the service re-deploy action to re-run the service when a device comes back up after going down during the initial service creation. Use this action to enable the re-deployment of a service.

Re-deploying a service re-deploys the service on all the devices, or sites, or endpoints. For more information, see the ***NSO documentation***.

This section discusses how to execute the re-deploy action on the example services.

IETF-TE Example Services

The following sample commands show how to execute the service re-deploy actions on the IETF-TE service.

re-deploy

```
admin@ncs> request te tunnels tunnel IETF-RSVP-TE re-deploy
```

re-deploy reconcile

```
admin@ncs> request te tunnels tunnel IETF-RSVP-TE re-deploy reconcile
```

IETF-NSS Example Services

The following sample commands show how to execute the service re-deploy actions on the IETF-NSS service. L2-HUB-SPOKE is used as a sample service to show the command implementation.

re-deploy

```
admin@ncs% request network-slice-services slice-service L2-HUB-SPOKE re-deploy
```

re-deploy reconcile

```
admin@ncs% request network-slice-services slice-service L2-HUB-SPOKE re-deploy
reconcile
```

Performance Measurement Example Services

The following sample commands show how to execute the service re-deploy actions on performance measurement services. PM-CLI is used as a sample service to show the command implementation.

re-deploy

```
admin@ncs% request pm svc-profiles PM-CLI re-deploy
```

re-deploy reconcile

```
admin@ncs% request pm svc-profiles PM-CLI re-deploy reconcile
```

Supporting MultiVendor Services - Example Services

Use the information in this section to support multivendors services for the Example Function Packs.

To create the multivendors services, you must copy the multivendors package to the TSDN FP Bundle installation folder, establish the dynamic-device mapping, and create the example service on the Netconf device.

Note: Make sure the `tsdn-<version>-nso-<version>/standalone/packages/ncs-<version>-<example-package>-multi-vendors-EXAMPLE-<version>.tar.gz` package is installed on your NSO. For more information, see the *Cisco T-SDN FP Bundle Installation Guide*.

Supporting MultiVendors for IOSXE CLI Services for Example Services

This section discusses the multivendors support for IOSXE CLI NED for the example services.

Supporting MultiVendors for IETF-TE IOSXE CLI Services

To support multivendors for IETF-TE IOSXE CLI services, configure the `cisco-rsvp-te-fp` dynamic-device mapping as follows. Make sure to provide the correct NED-ID. Ensure the XE device's tunnel-te interface is configured as a tunnel interface and not a tunnel-tp interface.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
    <dynamic-device-mapping>
      <ned-id>cisco-ios-cli-6.86:cisco-ios-cli-6.86</ned-id>
      <python-impl-class-name>rsvp_te_multi_vendors.IosXE</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting MultiVendors for Cisco Performance Measurement IOSXE CLI Services

To support multivendors for Cisco PM IOSXE CLI services, configure the **cisco-tdsn-pm** dynamic-device mapping as follows. Make sure to provide the correct NED-ID.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-pm">
    <dynamic-device-mapping>
      <ned-id>cisco-ios-cli-6.86:cisco-ios-cli-6.86</ned-id>
      <python-impl-class-name>pm_multi_vendors.IosXE</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Sample Custom-Template Payloads – Example Services

You can configure the devices directly by using custom-templates. With custom-templates, you can also apply additional configurations that are not supported by the function pack. For more information about custom-templates, see [Using Custom-Templates – CFP Services](#).

The custom template payloads for the following services are discussed in this topic.

- IETF-TE Services
- PM Services
- IETF-NSS Services

IETF-TE Services

Sample custom-template for all head-ends

```
<te xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
  <tunnels>
    <tunnel>
      <name>IETF-RSVP-TE</name>
      <custom-template>
=====> APPLYING CUSTOM TEMPLATE FOR ALL HEAD-END/TAIL-END
      <name>CT-CLI-banner</name>
      <variable>
        <name>BANNER_TEXT</name>
        <value>Welcome</value>
      </variable>
      </custom-template>
    </tunnel>
  </tunnels>
```



```
</te>
```

PM Services

Sample custom-template for pm devices

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <pm xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-pm">
    <svc-profiles>
      <name>PM-CLI</name>
      <custom-template>
        <name>CT-CLI-banner</name>
        <variable>
          <name>BANNER_TEXT</name>
          <value>WELCOME_A</value>
        </variable>
      </custom-template>
      <custom-template>
        <name>CT-CLI-time</name>
        <variable>
          <name>TIME_ZONE</name>
          <value>CST</value>
        </variable>
      </custom-template>
    </svc-profiles>
  </pm>
</config>
<config xmlns="http://tail-f.com/ns/config/1.0">
  <pm xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-pm">
    <svc-profiles>
      <name>PM-CLI</name>
      <devices>
        <name>PIOSXR-1</name>
        <interface-profile-attachment>
          <interfaces>
            <if-type>HundredGigE</if-type>
            <if-id>0/0/0/0</if-id>
            <delay-measurement>
              <profile-name>DelayInterface</profile-name>
              <advertise-delay>10000</advertise-delay>
            </delay-measurement>
            <next-hop-ip-addr>10.1.1.2</next-hop-ip-addr>
          </interfaces>
          <interfaces>
            <if-type>HundredGigE</if-type>
```

```

    <if-id>0/0/0/1</if-id>
    <delay-measurement>
      <explicit-delay-profile>explicitInterfaceDelayProfile</explicit-
delay-profile>
      <advertise-delay>20000</advertise-delay>
    </delay-measurement>
    <next-hop-ipv6-addr>110:101::102</next-hop-ipv6-addr>
  </interfaces>
</interface-profile-attachment>
<custom-template>
  <name>CT-CLI-banner</name>
  <variable>
    <name>BANNER_TEXT</name>
    <value>WELCOME_B</value>
  </variable>
</custom-template>
<custom-template>
  <name>CT-CLI-time</name>
  <variable>
    <name>TIME_ZONE</name>
    <value>PST</value>
  </variable>
</custom-template>
</devices>
</svc-profiles>
</pm>
</config>

```

IETF-NSS Services

```

<network-slice-services xmlns="urn:ietf:params:xml:ns:yang:ietf-network-slice-
service">
  <slice-service>
    <service-id>L2-ANY-TO-ANY</service-id>
    <sdps>
      <sdp>
        <sdp-id>0</sdp-id>
        <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-
slice-service">
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome_b</value>
          </variable>
        </custom-template>
      </sdp>
    </sdps>
  </slice-service>
</network-slice-services>

```

```

    </sdp>
  </sdps>
  <custom-template xmlns="http://cisco.com/ns/nso/fp/examples/cisco-network-
slice-service">
    <name>CT-CLI-banner</name>
    <variable>
      <name>BANNER_TEXT</name>
      <value>Welcome_A</value>
    </variable>
  </custom-template>
</slice-service>
</network-slice-services>

```

Notifications in Example Services

The services generate notifications that communicate vital information about various events. Use the notifications to modify and examine the payload content for a specific event.

IETF-TE Notifications

This topic includes creating and deleting notifications for the IETF-TE service.

Create Notifications

```

notification {
  eventTime 2020-09-16T20:56:53.748949+00:00
  plan-state-change {
    service /te/tunnels/tunnel[name='IETF-RSVP-TE']
    component self
    state init
    operation created
    status reached
  }
}
notification {
  eventTime 2020-09-16T20:56:53.749013+00:00
  plan-state-change {
    service /te/tunnels/tunnel[name='IETF-RSVP-TE']
    component self
    state ready
    operation created
    status not-reached
  }
}
notification {
  eventTime 2020-09-16T20:56:53.749149+00:00
  plan-state-change {

```

```
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        component 111.1.1.1
        state init
        operation created
        status reached
    }
}
notification {
    eventTime 2020-09-16T20:56:53.749197+00:00
    plan-state-change {
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        component 111.1.1.1
        state ietf-te-fp-tunnel-nano-plan-services:config-apply
        operation created
        status reached
    }
}
notification {
    eventTime 2020-09-16T20:56:53.749258+00:00
    plan-state-change {
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        component 111.1.1.1
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-09-16T20:56:55.614024+00:00
    plan-state-change {
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        component self
        state ready
        operation modified
        status reached
    }
}
notification {
    eventTime 2020-09-16T20:56:55.614147+00:00
    plan-state-change {
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        component 111.1.1.1
        state ready
        operation modified
    }
}
```

```

        status reached
    }
}

```

Delete Notifications

```

notification {
    eventTime 2020-09-23T13:43:21.058294+00:00
    plan-state-change {
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        operation deleted
    }
}

```

The following notification displays if the delete operation fails.

```

notification {
    eventTime 2020-09-23T13:44:43.26129+00:00
    plan-state-change {
        service /te/tunnels/tunnel[name='IETF-RSVP-TE']
        operation deleted
        status failed
    }
}

```

Network Slice Service Notifications

This topic includes creating and deleting notifications for the NSS service.

Create Notifications

```

notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component self
        state init
        operation created
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component self
        state ready
        operation created
    }
}

```

```
        status not-reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 0
        state init
        operation created
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 0
        state ietf-nss-nano:config-apply
        operation created
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 0
        state ready
        operation created
        status not-reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 1
        state init
        operation created
    }
}
```

```
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 1
        state ietf-nss-nano:config-apply
        operation created
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:42.166106+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 1
        state ready
        operation created
        status not-reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:44.97416+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component self
        state ready
        operation modified
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:44.97416+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 0
        state ready
        operation modified
    }
}
```

```

        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}
notification {
    eventTime 2022-09-02T23:32:44.97416+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        component 1
        state ready
        operation modified
        status reached
        trace-id 6e55d9cd-2aeb-41ec-8009-bcd62fe1ef4c
    }
}

```

Delete Notifications

```

notification {
    eventTime 2022-09-02T23:34:04.362247+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        operation deleted
    }
}

```

The following notification displays if the delete operation fails.

```

notification {
    eventTime 2022-09-02T23:35:05.798932+00:00
    plan-state-change {
        service /network-slice-services/slice-service[service-id='L3-Shared']
        operation deleted
        status failed
    }
}

```

Performance Measurement Notifications

This topic includes creating and deleting notifications for the PM service.

Create Notifications

```

notification {
    eventTime 2023-02-23T23:59:56.599365+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component PIOSXR-0
    }
}

```



```

        state ready
        operation created
        status not-reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}
notification {
    eventTime 2023-02-23T23:59:56.599365+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component PIOSXR-1
        state init
        operation created
        status reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}
notification {
    eventTime 2023-02-23T23:59:56.599365+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component PIOSXR-1
        state cisco-pm-fp-nano-plan-services:config-apply
        operation created
        status reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}
notification {
    eventTime 2023-02-23T23:59:56.599365+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component PIOSXR-1
        state ready
        operation created
        status not-reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}
notification {
    eventTime 2023-02-24T00:00:01.18508+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component self
    }
}

```

```

        state ready
        operation modified
        status reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}
notification {
    eventTime 2023-02-24T00:00:01.18508+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component PIOSXR-0
        state ready
        operation modified
        status reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}
notification {
    eventTime 2023-02-24T00:00:01.18508+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        component PIOSXR-1
        state ready
        operation modified
        status reached
        trace-id dde379e8-3cc0-43d4-8baf-c77c800fd762
    }
}

```

Delete Notifications

```

notification {
    eventTime 2023-02-24T00:00:40.266499+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        operation deleted
    }
}

```

The following notification displays if the delete operation fails.

```

notification {
    eventTime 2023-02-24T00:10:00.126769+00:00
    plan-state-change {
        service /pm/svc-profiles[name='PM-CLI']
        operation deleted
        status failed
    }
}

```

```
}  
}
```

Deleting Example Services

Delete all IETF-TE services

```
admin@ncs> delete te tunnels tunnel
```

Delete all IETE - NSS services

```
admin@ncs> delete network-slice-services slice-service
```

Delete all Cisco PM services

```
admin@ncs> delete pm svc-profiles
```

NSO High Availability in TSDN-FP Bundle

The TSDN-FP supports NSO High Availability like the other NSO Core Function Packs. For more information on how to configure NSO HA, see the ***NSO documentation***.

Appendix A - Package Categories and Packages

The following table lists the package categories and the associated packages. These packages are extracted either during or after SR-TE CFP-IOSXR CLI installation. For more information on SR-TE CFP-IOSXR CLI installation, see the **Cisco T-SDN FP Bundle Installation Guide**.

The Core Function Pack packages are required to install SR-TE CFP. The Example packages comprise the packages for the Example Function Packs (such as L2VPN) you choose to install.

Package Categories and Packages on a Single NSO Instance

This section discusses the required package categories and their associated packages, which are extracted either during or after SR-TE CFP-IOSXR CLI installation.

For more information on SR-TE CFP-IOSXR CLI installation, see the **T-SDN FP Bundle Installation Guide**.

The following tables show the packages in the Core Function Pack and Example Packages categories.

The IOSXR CLI NED is the default NED. The IOSXR CLI and IOSXE CLI NEDs are packaged with the installation tar file. The IOSXR Netconf NEDs are downloadable from the Cisco website. For more information about supported NED versions, see the **T-SDN FP Installation Guide**.

The Core Function Pack packages are required to install SR-TE CFP. The example packages comprise the packages for the Example Function Packs you choose to install.

Core Function Pack	
Package Category	Packages
T-SDN FP Bundle packages	ncs-6.1.10-cisco-sr-te-cfp-7.0.0.tar.gz ncs-6.1.10-cisco-sr-te-cfp-internal-7.0.0.tar.gz ncs-6.1.10-cisco-cs-sr-te-cfp-7.0.0.tar.gz
T-SDN FP Bundle L2NM Packages	ncs-6.1.10-cisco-L2vpn-fp-internal-7.0.0.tar.gz ncs-6.1.10-ietf-l2vpn-nm-7.0.0.tar.gz
T-SDN FP Bundle L3NM Packages	ncs-6.1.10-cisco-L3vpn-fp-internal-7.0.0.tar.gz ncs-6.1.10-ietf-l3vpn-nm-7.0.0.tar.gz
T-SDN FP Bundle common packages	ncs-6.1-core-fp-plan-notif-generator-1.0.10.tar.gz ncs-6.1-custom-template-utils-2.0.13.tar.gz ncs-6.1-lsa-utils-1.0.4.tar.gz

	<p>ncs-6.1-core-fp-common-1.33.0.tar.gz ncs-6.1.10-cisco-tdsn-core-fp-common-7.0.0.tar.gz</p> <p>ncs-6.1.8-resource-manager-4.2.5.tar.gz ncs-6.1-core-fp-delete-tag-service-1.0.6.tar.gz ncs-6.1.3-cisco-cfp-jwt-auth-1.0.0.tar.gz (RBAC PAM package) ncs-6.1.6-phased-provisioning-1.1.0.tar.gz</p>
T-SDN FP Bundle MultiVendor for IOSXE	<p>ncs-6.1.10-sr-te-multi-vendors-7.0.0.tar.gz ncs-6.1.10-l2vpn-multi-vendors-7.0.0.tar.gz ncs-6.1.10-l3vpn-multi-vendors-7.0.0.tar.gz ncs-6.1.10-rsvp-te-multi-vendors-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-pm-multi-vendors-EXAMPLE-7.0.0.tar.gz</p>
CNC user specific packages	<p>ncs-6.1.10-dlm-svc-7.0.0-9.tar.gz ncs-6.1.10-cw-device-auth-7.0.0.tar.gz ncs-6.1.10-cisco-tm-tc-fp-7.0.0-4.tar.gz</p>
NEDs	<p>IOS XR CLI NED: ncs-6.1.4-cisco-iosxr-7.52.2.tar.gz ncs-6.0.3-cisco-iosxr-7.46.3.tar.gz</p> <p>IOS XE CLI NED: ncs-6.1.5-cisco-ios-6.100.8.tar.gz ncs-6.1-cisco-ios-6.86.6.tar.gz</p>

Example Packages	
Package Category	Package Name
Automated Assurance Example package for L2NM,L3NM	ncs-6.1.10-cisco-aa-service-assurance-7.0.0.tar.gz
IETF Network Slice Service Example package	ncs-6.1.10-ietf-network-slice-service-EXAMPLE-7.0.0.tar.gz
IETF-TE Example package with IOSXR CLI NED	ncs-6.1.10-ietf-te-fp-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-cisco-rsvp-te-fp-EXAMPLE-7.0.0.tar.gz
Performance Measurement Example package	ncs-6.1.10-cisco-pm-fp-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-cisco-pm-fp-internal-EXAMPLE-7.0.0.tar.gz

LSA Model – Package Categories and Packages

Lower-Level Nodes

The following table shows the package categories and the packages extracted on the lower-level nodes.

The IOSXR CLI NED is the default NED. The IOSXR CLI and IOSXE CLI NEDs are packaged with the installation tar file. The IOSXR Netconf NEDs are downloadable from the Cisco website. For more information about supported NED versions, see the *T-SDN FP Installation Guide*.

RFS Node Packages	
Package Category	Packages
T-SDN FP Bundle packages	ncs-6.1.10-cisco-sr-te-cfp-internal-7.0.0.tar.gz ncs-6.1.10-sr-te-multi-vendors-7.0.0.tar.gz ncs-6.1.10-cisco-L2vpn-fp-internal-7.0.0.tar.gz ncs-6.1.10-l2vpn-multi-vendors-7.0.0.tar.gz ncs-6.1.10-l3vpn-multi-vendors-7.0.0.tar.gz ncs-6.1.10-cisco-L3vpn-fp-internal-7.0.0.tar.gz
T-SDN FP Bundle common packages	ncs-6.1-core-fp-common-1.33.0.tar.gz ncs-6.1-custom-template-utils-2.0.13.tar.gz ncs-6.1-core-fp-delete-tag-service-1.0.6.tar.gz ncs-6.1-core-fp-plan-notif-generator-1.0.10.tar.gz ncs-6.1-lsa-utils-1.0.4.tar.gz ncs-6.1.10-cisco-tdsn-core-fp-common-7.0.0.tar.gz ncs-6.1.6-phased-provisioning-1.1.0.tar.gz
LSA NED packages	IOS XR CLI NED: ncs-6.1.4-cisco-iosxr-7.52.2.tar.gz ncs-6.10.3-cisco-iosxr-7.46.3.tar.gz IOS XE CLI NED: ncs-6.1-cisco-ios-6.86.6.tar.gz ncs-6.1.5-cisco-ios-6.100.8.tar.gz
CNC User Specific Packages	ncs-6.1.10-cisco-tm-tc-fp-6.0.0-4.tar.gz ncs-6.1.10-dlm-svc-6.0.0-9.tar.gz ncs-6.1.10-cw-device-auth-6.0.0.tar.gz
Example packages	ncs-6.1.10-cisco-rsvp-te-fp-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-rsvp-te-multi-vendors-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-cisco-pm-fp-internal-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-pm-multi-vendors-EXAMPLE-7.0.0.tar.gz

Upper-Level Node

The following table shows the package categories and the packages extracted on the upper-level node.

The IOSXR CLI NED is the default NED. The IOSXR CLI and IOSXE CLI NEDs are packaged with the installation tar file. The IOSXR Netconf NEDs are downloadable from the Cisco website. For more information about supported NED versions, see the ***T-SDN FP Installation Guide***.

The cs-sr-te-cfp package in SR-TE CFP-IOSXR CLI is supported only on IOSXR CLI 7.46/7.52 NED, IOSXR NC 7.8 9 or later) NED.

CFS Node Packages	
Package Category	Packages
T-SDN FP Bundle packages	ncs-6.1.10-cisco-sr-te-cfp-7.0.0.tar.gz ncs-6.1.10-cisco-cs-sr-te-cfp-7.0.0.tar.gz ncs-6.1.10-ietf-l2vpn-nm-7.0.0.tar.gz ncs-6.1.10-ietf-l3vpn-nm-7.0.0.tar.gz ncs-6.1.3-cisco-cfp-jwt-auth-1.0.0.tar.gz
T-SDN FP Bundle common packages	ncs-6.1-core-fp-plan-notif-generator-1.0.10.tar.gz ncs-6.1.10-custom-template-utils-ned-1.0.tar.gz ncs-6.1-core-fp-common-1.33.0.tar.gz ncs-6.1-lsa-utils-1.0.4.tar.gz ncs-6.1.10-cisco-tdsn-core-fp-common-7.0.0.tar.gz ncs-6.1.8-resource-manager-4.2.5.tar.gz ncs-6.1.6-phased-provisioning-1.1.0.tar.gz
LSA NED packages	ncs-6.1.10-cisco-L2vpn-fp-internal-ned-7.0.0.tar.gz ncs-6.1.10-cisco-L3vpn-fp-internal-ned-7.0.0.tar.gz ncs-6.1.10-cisco-pm-fp-internal-ned-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-cisco-rsvp-te-fp-internal-ned-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-lsa-utils-ned-1.0.tar.gz ncs-6.1.10-custom-template-utils-ned-1.0.tar.gz ncs-6.1.10-core-fp-common-ned-1.0.tar.gz ncs-6.1.10-cisco-sr-te-cfp-internal-ned-7.0.0.tar.gz ncs-6.1.10-cisco-nso-nc-6.1.tar.gz
Example packages	ncs-6.1.10-cisco-aa-service-assurance-7.0.0.tar.gz ncs-6.1.10-cisco-pm-fp-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-ietf-network-slice-service-EXAMPLE-7.0.0.tar.gz ncs-6.1.10-ietf-te-fp-EXAMPLE-7.0.0.tar.gz

Appendix B - Error Recovery

Errors are classified as:

- Transient errors or failures
- Persistent errors or failures

Transient failures are self-correcting failures. The device may come back up in such failures, and you can reconcile the services. For example, a failure to connect to a device is a transient failure. The device can be temporarily out of order and can come back up without any changes being performed on NSO to bring it back up.

Persistent failures require manual intervention to recover from the error condition. These failures include errors such as the device being out-of-sync or device configuration errors. For example, in a device configuration error, a device may not accept a configuration pushed to it. You may manually revise the service intent to come out of the failure condition.

Commit-Queue Error Recovery Flow

This topic discusses the error recovery configuration and the recovery flow when the commit-queue async flag is enabled.

Error Recovery Configuration

The following yang model shows the error recovery parameters to configure.

```

module: cisco-tdsn-core-fp-common
  +--rw commit-queue-recovery-data
  | +--rw auto-cleanup?                boolean
  | +--rw enable-polling-recovery?    boolean
  | +--ro failed-device* [name]
  | | +--ro name                      string
  | | +--ro impacted-service-path* [service]
  | |   +--ro service                 string
  | |   +--ro failed-commit-queue-id? string
  | |   +--ro poller-recovery-result? string
  | +--ro current-device-poller* [name]
  | | +--ro name                      string
  | +--rw device-poller-configurations
  | | +--rw poll-wait-time?           uint32
  | | +--rw poll-time?               uint32
  | | +--rw poll-time-multiplier?    uint32
  | | +--rw sync-direction?          enumeration
  | +---x trigger-device-poller
  | | +---w input
  | | | +---w device                  string
  | | +--ro output

```

```

| |    +---ro success    boolean
| |    +---ro detail?   string
| +---x purge-failed-device
|   +---w input
|     | +---w device           string
|     | +---w impacted-service-path* [service]
|     | | +---w service      string
|     | +---w force?         empty
|   +---ro output
|     +---ro success    boolean
|     +---ro detail?   String

```

The following table provides the parameter descriptions.

Parameter	Description
enable-polling-recovery	By default, this flag is set to false. If set to true, the CFP tries to automatically recover the failed services for transient errors. The CFP starts a connection check poller toward the failed device. The recovery flow is executed if the device comes back up within the configured time.
auto-cleanup	If set to true, deleting a service from an offline device automatically removes all the data related to the device when you use the no-networking option with auto-cleanup. You must clean up the configuration on the device once it is back online. If set to false, manually run the Cleanup action to remove the device from the service if the device becomes permanently unreachable.
failed-device	This list contains information about the devices that have failed due to transient errors when creating/deleting the service along with a reference to the impacted services and failed commit-queue. CFP uses this data internally to drive auto-recovery flow. The poller-recovery-result states the poller results for recovering the impacted service once the device is back online. If the recovery is successful, the entry is removed from the failed-device list.
current device poller (hidden CFP internal data)	If auto-recovery is enabled, the current device poller lists the devices being polled by the CFP currently.
poll-wait-time	This is the wait time in seconds between each polling request to the device. The default wait time is 30 seconds.
poll-time	This is the duration in minutes for which the device polls for connection. The default value is 30 minutes.
poll-time-multiplier	This is a numeric value that is multiplied by the poll-time to calculate the total poll-time. The default value is set to 1.

sync-direction	Sync the device using sync-from or sync-to. The default setting is sync-from.
trigger-device-poller	Use this action to manually trigger the poller on a device if no poller is running. Use this action when the poller has timed out and given up on the device that did not become active within the polling time frame.
purge-failed-device	Use this action to purge commit-queue error failed device or remove services from the failed list. This action helps to manually remove the failed device entry from the failed list. Use this action when all other recovery mechanisms have failed and if the data gets corrupted in the operational data store of the failed-device list.

Recovery Flow

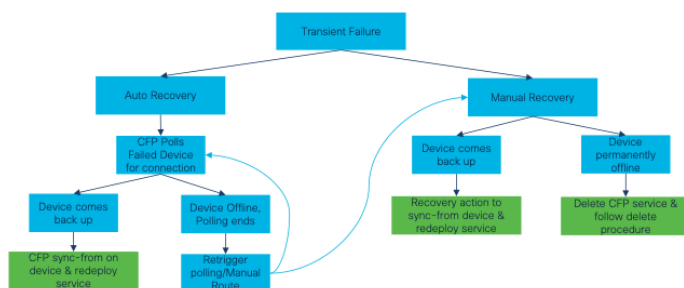
This section discusses the flow of recovery of a device from a create failure and a delete failure when the error is transient or persistent.

Create failure when the error is transient

The following diagram shows the recovery flow from a create failure when the error is transient.

The service plan notification shows the device failure. As required, you can choose to perform either the auto-recovery process or the manual recovery process.

Error Recovery - Create Failure (Transient)



© 2020 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

Auto-recovery

TSDN CFP provides the auto-recovery feature to automatically recover the services from transient failures based on the auto-recovery flow.

When you try to reconfigure a failed device, the CFP polls it to connect to it. During the polling period, the device may or may not come back up. If it does, the poller tries to issue a sync-from on the device to push the configuration and redeploy the service on the device. On successful redeployment of the service, the plan reaches the **reached** state.

If the device does not come back after the polling period, you can either retrigger polling and wait for it to come back up or manually recover it once it does.

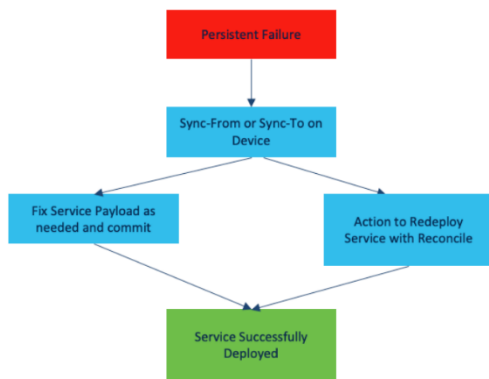
Manual recovery

In case of manual recovery, request the recovery action once the device comes back up. Manual recovery actions are required for transient services and are available for all TSDN services. The recovery action performs a sync-from and redeploys the service. For more information about this action, see section **Error Recovery Action**.

If the device is permanently offline and fails to come back up even after performing a manual recovery, delete the service.

Create failure when the error is persistent

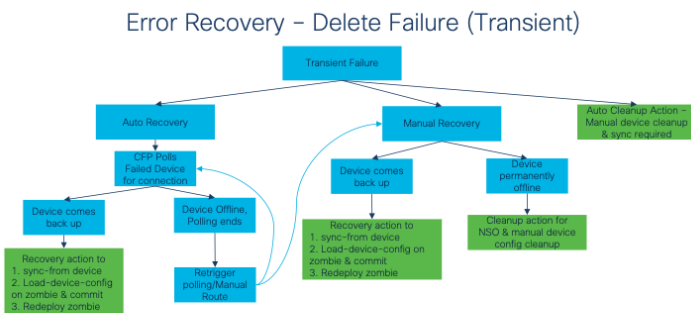
The following diagram shows the flow of recovery of a device from a create failure when the error is persistent.



Persistent failures are recovered manually. You perform a sync-from or sync-to on the device as required, fix the service payload, and trigger service redeploy with the reconcile option.

Delete failure when the error is transient/persistent

The following diagram shows the flow of recovery from a delete failure when the error is transient.



Recovery methods include auto-recovery, manual recovery, and the auto Cleanup action.

Auto-recovery

TSDN CFP provides the auto-recovery feature to automatically recover the services from transient failures based on the auto-recovery flow.

When you try to reconfigure a failed device, the CFP polls the failed device to connect to it. During the polling period, the device may or may not come back up.

If the device comes back up, it follows a 3-step recovery process. It performs:

- a sync-from the device
- tries to load the zombie (delete) configuration on the device and commits the zombie configuration and
- redeploys the zombie

If the device does not come back up after the polling period, you can either retrigger polling and wait for it to or manually recover it once it does.

Manual recovery

During the manual recovery process, if the device comes back up, the recovery action performs:

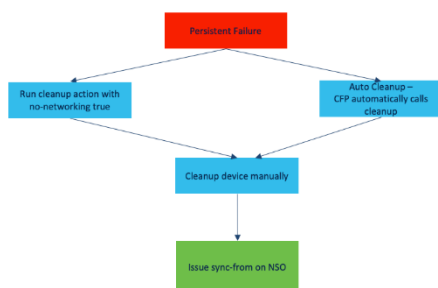
- a sync-from device
- tries to load the zombie (delete) configuration and commits it and
- redeploys the zombie

Redeploying the zombie again pushes a delete on the device configuration.

Use the Cleanup action if the device is permanently offline and you must clean up NSO.

Delete failure when the error is persistent

The following diagram shows the flow of recovery from a delete failure when the error is persistent.



To recover from this error, run the auto Cleanup action or perform the manual recovery process. To manually recover, use the Cleanup action by setting the **no-networking** option to true. Clean up the device manually and issue a sync-from on NSO.

Error Recovery Scenarios

This section creates possible error scenarios, and the steps to recover from the errors are discussed.

The error scenarios were created with the following NSO configuration.

```
***Global Settings on NSO **
devices global-settings connect-timeout 240
devices global-settings read-timeout 240
devices global-settings write-timeout 240
devices global-settings trace raw
devices global-settings commit-queue enabled-by-default false
devices global-settings commit-queue async
devices global-settings commit-queue atomic false
devices global-settings commit-queue retry-attempts 0
devices global-settings commit-queue retry-timeout 30
devices global-settings commit-queue error-option stop-on-error
devices global-settings out-of-sync-commit-behaviour reject
```

Scenario 1: Unable to provision multiple services on the same device when one of the services failed

Explanation

This problem occurs when you are trying to provision multiple services on the same device, and one of the services deployed on the device has failed due to an incorrect AS number.

Creating the scenario

The following configuration was considered to create the error scenario.

1. **Service 1** (IETF-L3NM service) is deployed on the PE-5 device and CE-1 device, and (**SR-TE policy**) is deployed on the PE-5 device.
2. Make sure the devices are in sync with NSO.

```
admin@ncs# devices check-sync device [ PE-5 CE-1]
sync-result {
  device PE-5
  result in-sync
}
sync-result {
  device CE-1
  result in-sync
}
```

3. The correct BGP AS for PE5 and CE1 is 1.

```
admin@ncs% show devices device PE-5 config router bgp bgp-no-instance
```

```

bgp-no-instance 1 {
  bgp {
    router-id 200.200.200.5;
  }
}

admin@ncs% show devices device CE-1 config router bgp bgp-no-instance
bgp-no-instance 1 {
  bgp {
    router-id 200.200.200.1;
  }
}

```

Do the following to create the error scenario:

1. Push a service payload with an incorrect AS number 100.

```

admin@ncs(config)# show full-configuration l3vpn-ntw vpn-services vpn-service 0-
65008740 vpn-nodes vpn-node * local-as
l3vpn-ntw vpn-services vpn-service 0-65008740
  vpn-nodes vpn-node CE-1
    local-as 100
  !
  vpn-nodes vpn-node PE-5
    local-as 100
  !
  !

```

2. Perform a commit dry-run to view the configuration on the CE-1 and PE-5.

```

admin@ncs% commit dry-run outformat native
native {
  device {
    name CE-1
    data vrf 0-65008740
      address-family ipv6 unicast
        import route-target
          65010:17405
          65010:17406
        exit
      export route-target
        65010:17405
        65010:17406
      exit
    exit
  exit
  interface GigabitEthernet 0/0/0/0
    description T-SDN Interface
  exit
}

```

```
interface GigabitEthernet 0/0/0/0.1234
  description T-SDN Interface
  encapsulation dot1q 1234
  vrf 0-65008740
  ipv6 address 2001:db8::1/32
  no shutdown
exit
extcommunity-set opaque COLOR_100
  100
end-set
!
extcommunity-set opaque COLOR_101
  101
end-set
!
route-policy PASS_ALL
  pass
end-policy
!
route-policy SET_COLORv4_TEST_POLICY
  if destination in (1.1.1.1/32, 1.1.1.2/32) then
    set extcommunity color COLOR_100
  endif
  if destination in (2.1.1.1/32, 2.1.1.2/32) then
    set extcommunity color COLOR_101
  endif
end-policy
!
router bgp 100 ---> User Error of providing incorrect AS
vrf 0-65008740
  rd 65100:87400024
  address-family ipv6 unicast
  exit
  neighbor 2001:db8::2
  remote-as 65003
  ebgp-multihop 12
  address-family ipv6 unicast
  route-policy PASS_ALL in
  route-policy PASS_ALL out
  exit
  exit
  exit
exit
}
```



```
device {
  name PE-5
  data vrf 0-65008740
    address-family ipv4 unicast
      import route-target
        65010:17401
        65010:17402
        65010:17403
      exit
      export route-policy SET_COLORv4_TEST_POLICY
      export route-target
        65010:17401
        65010:17402
        65010:17404
      exit
    exit
  exit
  extcommunity-set opaque COLOR_101
    101
  end-set
  !
  route-policy PASS_ALL
    pass
  end-policy
  !
  route-policy SET_COLORv4_TEST_POLICY
    if destination in (1.1.1.1/32, 1.1.1.2/32) then
      set extcommunity color COLOR_100
    endif
    if destination in (2.1.1.1/32, 2.1.1.2/32) then
      set extcommunity color COLOR_101
    endif
  end-policy
  !
  router bgp 100 ----> User Error of providing incorrect AS
  vrf 0-65008740
    rd 65100:87400024
    address-family ipv4 unicast
      exit
    neighbor 10.1.1.1
      remote-as 65003
      ebgp-multihop 11
    address-family ipv4 unicast
      route-policy PASS_ALL in
```

```

        route-policy PASS_ALL out
    exit
    exit
    exit
    exit
}
}

```

3. Commit the configuration in commit-queue.

```

admin@ncs% commit commit-queue async
Commit complete.

```

4. Display the plan for service 1. The plan displays as failed with an error message.

```

admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 plan
                                           POST
                                           ACTION
TYPE  NAME  TRACK  GOAL  STATE  STATUS  WHEN                ref  STATUS
-----
self  self  false  -     init   reached  2024-06-11T01:04:45  -   -
                                           ready  failed   2024-06-11T01:04:53  -   -

```

```

plan failed
plan error-info message "External error in the NED implementation for device PE-
5: Thu Mar 25 21:55:02.441 UTC\r\n\r\n% Failed to commit one or more
configuration items during a pseudo-atomic operation. All changes made have been
reverted.\r\n !! SEMANTIC ERRORS: This configuration was rejected by \r\n!! the
system due to semantic errors. The individual \r\n!! errors with each failed
configuration command can be \r\n!! found below.\r\n\r\n\r\nrouter bgp
100\r\n!!% The instance name is used already: asn 0.1 inst-name default\r\n vrf
0-65008740\r\n neighbor 10.1.1.1\r\n remote-as 65003\r\n!!% The instance name
is used already: asn 0.1 inst-name default\r\n address-family ipv4 unicast\r\n
route-policy PASS_ALL in\r\n!!% The instance name is used already: asn 0.1 inst-
name default\r\n route-policy PASS_ALL out\r\n!!% The instance name is used
already: asn 0.1 inst-name default\r\n  !\r\n  !\r\n  !\r\n  !\r\nend"
plan failed-vpn-nodes ['PE-5']

```

```

aadmin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 vpn-
nodes vpn-node-plan PE-1 plan

```

```

                                           POST
                                           ACTION
TYPE  NAME  TRACK  GOAL  STATUS CODE    STATE                STATUS  WHEN                ref  STATUS
-----
self  self  false  -     TSDN-IETF-L3NM-303  init                 reached  2024-06-11T01:04:45  -   -   |
                                           ietf-l3vpn-ntw-nano-services:config-apply  reached  2024-06-11T01:04:45  -   -
                                           ready                 failed   2024-06-11T01:04:53  -   -

```

```

plan failed
plan error-info message "External error in the NED implementation for device PE-
5: Thu Mar 25 22:03:32.057 UTC\r\n\r\n% Failed to commit one or more
configuration items during a pseudo-atomic operation. All changes made have been

```

```

reverted.\r\n !! SEMANTIC ERRORS: This configuration was rejected by \r\n!! the
system due to semantic errors. The individual \r\n!! errors with each failed
configuration command can be \r\n!! found below.\r\n\r\n\r\n\r\nrouter bgp
100\r\n!!% The instance name is used already: asn 0.1 inst-name default\r\n vrf
0-65008740\r\n neighbor 2001:db8::2\r\n remote-as 65003\r\n!!% The instance
name is used already: asn 0.1 inst-name default\r\n address-family ipv6
unicast\r\n route-policy PASS_ALL in\r\n!!% The instance name is used
already: asn 0.1 inst-name default\r\n route-policy PASS_ALL out\r\n!!% The
instance name is used already: asn 0.1 inst-name default\r\n !\r\n !\r\n
!\r\n!\r\nend"

```

```
plan status-code-detail self self
```

```
code TSDN-IETF-L3NM-303
```

```
context "Config push failed"
```

```

context-msg "External error in the NED implementation for device PE-5: Thu Mar
25 22:03:32.057 UTC\r\n\r\n% Failed to commit one or more configuration items
during a pseudo-atomic operation. All changes made have been reverted.\r\n !!
SEMANTIC ERRORS: This configuration was rejected by \r\n!! the system due to
semantic errors. The individual \r\n!! errors with each failed configuration
command can be \r\n!! found below.\r\n\r\n\r\n\r\nrouter bgp 100\r\n!!% The instance
name is used already: asn 0.1 inst-name default\r\n vrf 0-65008740\r\n neighbor
2001:db8::2\r\n remote-as 65003\r\n!!% The instance name is used already: asn
0.1 inst-name default\r\n address-family ipv6 unicast\r\n route-policy
PASS_ALL in\r\n!!% The instance name is used already: asn 0.1 inst-name
default\r\n route-policy PASS_ALL out\r\n!!% The instance name is used
already: asn 0.1 inst-name default\r\n !\r\n !\r\n !\r\n!\r\nend"

```

```
severity ERROR
```

```
recommended-action "Device configuration rejected, fix the service payload and
perform recovery steps."
```

```
impacted-device PE-5
```

5. View the NSO device CDB. The configuration on the device is not rolled back.

```
admin@ncs# devices device PE-5 compare-config outformat cli
```

```
diff
```

```

devices {
  device PE-5 {
    config {
      vrf {
-       vrf-list 0-65008740 {
-         address-family {
-           ipv4 {
-             unicast {
-               import {
-                 route-target {
-                   address-list 65010:17401;
-                   address-list 65010:17402;
-                   address-list 65010:17403;
-                 }
-               }
-             export {

```

```
-                                route-policy SET_COLORv4_TEST_POLICY;
-                                route-target {
-                                    address-list 65010:17401;
-                                    address-list 65010:17402;
-                                    address-list 65010:17404;
-                                }
-                                }
-                                }
-                            }
-                    }
-                }
-            }
-        }
-    }
-    }
-
-    extcommunity-set {
-        opaque COLOR_101 {
-            set 101;
-        }
-    }
-
-    route-policy PASS_ALL {
-        value pass;
+        value " pass\r\n";
-    }
-
-    route-policy SET_COLORv4_TEST_POLICY {
-        value " if destination in (1.1.1.1/32, 1.1.1.2/32) then\r\n
set extcommunity color COLOR_100\r\n endif\r\n if destination in (2.1.1.1/32,
2.1.1.2/32) then\r\n set extcommunity color COLOR_101\r\n endif\r\n";
-    }
-
-    router {
-        bgp {
-            bgp-no-instance 100 {
-                vrf 0-65008740 {
-                    rd 65100:8740002 4;
-                    address-family {
-                        ipv4 {
-                            unicast {
-                                }
-                            }
-                        }
-                    }
-                }
-                neighbor 10.1.1.1 {
-                    remote-as 65003;
-                    ebgp-multihop {
-                        ttl-value 11;
-                    }
-                    address-family {
-                        ipv4 {
-                            unicast {
```

```
-
-                                     route-policy in {
-                                         name PASS_ALL;
-                                     }
-                                     route-policy out {
-                                         name PASS_ALL;
-                                     }
-                                     }
-                                 }
-                                 }
-                             }
-                             }
-                         }
-                         }
-                     }
-                     }
-                 }
-                 }
-             }
-             }
-         }
-         }
-     }
-     }
- }
```

6. Deploy the SR-TE policy service on the PE-5 device.

7. Perform a dry-run to view the SR-TE policy service configuration on the PE-5 device.

```
Service Payload - Commit Dry Run outformat native.
admin@ncs% commit dry-run outformat native
native {
    device {
        name PE-5
        data segment-routing
            traffic-eng
                policy srte_c_550_ep_100.100.100.6
                color 550 end-point ipv4 100.100.100.6
                candidate-paths
                preference 100
                dynamic
                metric
                    type te
                margin absolute 100
                !
                !
                !
                !
                !
                !
                !
            }
        }
    }
```

8. Commit the SR-TE policy service configuration in commit-queue.

```
admin@ncs% commit commit-queue async
Commit complete.
[ok]
```

9. Display the plan for SR-TE policy service. The plan displays as failed with an error message that the alarms for the device are out of sync.

```
admin@ncs% *** ALARM out-of-sync: Device PE-5 is out of sync
admin@ncs% *** ALARM commit-through-queue-failed: Commit queue item
1616709575292 has failed: Network Element Driver: device PE-5: out of sync
admin@ncs% run show cisco-sr-te-cfp:sr-te policies policy-plan sr-policy-11 plan
```

TYPE	BACK			POST			STATUS	WHEN	ACTION	
	NAME	TRACK	GOAL	STATUS	CODE	STATE			ref	STATUS
self	self	false	-	-	-	init	reached	2021-03-25T21:59:35	-	-
						ready	failed	2021-03-25T21:59:38	-	-
cisco-sr-te-cfp:sr-policies:nano-plan-services:head-end	PE-5	false	-	TSDN-SR-302	-	init	reached	2021-03-25T21:59:35	-	-
						cisco-sr-te-cfp:sr-policies:nano-plan-services:config-apply	reached	2021-03-25T21:59:35	-	-
						ready	failed	2021-03-25T21:59:38	-	-

```
plan failed
plan error-info message "Network Element Driver: device PE-5: out of sync"
plan status-code-detail cisco-sr-te-cfp:sr-policies:nano-plan-services:head-end
PE-5
code TSDN-SR-302
context "Device out of sync"
context-msg "Network Element Driver: device PE-5: out of sync"
severity ERROR
recommended-action "Check sync between device and NSO, and perform recovery
steps."
```

Now, any successive services deployed on the PE-5 device or CE-1 device fail.

Solution

This section discusses the steps to recover from the error condition.

Note: The recovery steps vary based on the errors or scenarios. Be sure to be aware of the actions performed on the service and device.

The following is the workflow to recover from the error condition:

1. Perform a sync-from on devices PE-5 and CE-1.
2. Correct the AS number in the payload for service 1 and redeploy the service.
3. Re-deploy service 2.

The detailed procedure to recover from the error condition is as follows:

1. Log in to NSO UI with your login credentials.
2. Navigate to the **Device Manager** window and perform a sync-from on the PE-5 and CE-1 devices.

<input type="checkbox"/>	name	address	port	type	services	ping	connect	check-sync	sync-from
<input type="checkbox"/>	ASR920	172.22.141.158	22	cisco-ios-ctl-6.67:cisco-ios-ctl-6.67	0	ping	connect	check-sync	sync-from
<input checked="" type="checkbox"/>	CE-1	172.25.86.23	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	2	ping	connect	check-sync	sync-from
<input type="checkbox"/>	CE-2	172.25.86.24	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	CE-3	172.25.86.26	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	CE-4	172.25.86.27	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	IOSXE-1	10.85.69.221	22	cisco-ios-ctl-6.67:cisco-ios-ctl-6.67	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	IOSXR731	192.168.66.21	22	cisco-iosxr-nc-7.3:cisco-iosxr-nc-7.3	0	ping	connect	check-sync	sync-from
<input checked="" type="checkbox"/>	PE-5	172.25.86.29	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	3	ping	connect	check-sync	sync-from
<input type="checkbox"/>	PE-6	172.25.86.30	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	XE-PE1	172.22.141.36	22	cisco-ios-ctl-6.67:cisco-ios-ctl-6.67	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	XE-PE2	172.22.141.37	22	cisco-ios-ctl-6.67:cisco-ios-ctl-6.67	0	ping	connect	check-sync	sync-from
<input type="checkbox"/>	XR-PE-1	172.22.141.30	22	cisco-iosxr-ctl-7.33...cisco-iosxr-ctl-7.33	1	ping	connect	check-sync	sync-from

3. Navigate to **Configuration editor** window. Update the value of the **local-autonomous-system** field for both the PE-5 device and CE-1 device with the correct AS number 1.

Configuration editor
NSO VERSION:5.4.2

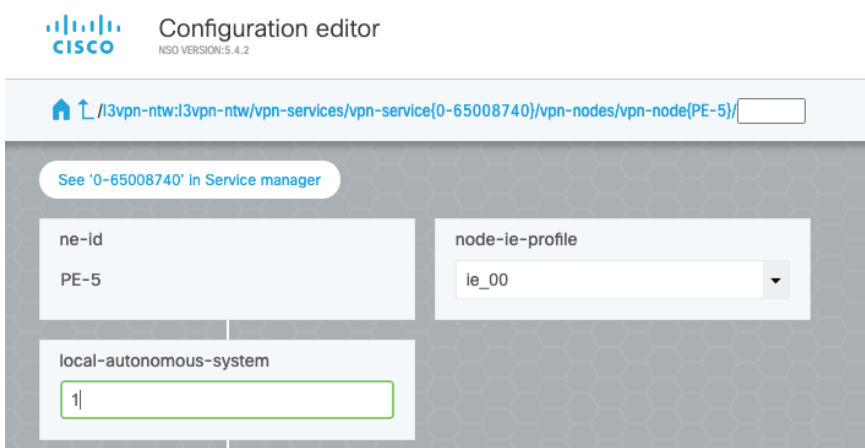
Home / /3vpn-ntw:/3vpn-ntw/vpn-services/vpn-service(0-65008740)/vpn-nodes/vpn-node(CE-1)

See '0-65008740' in Service manager

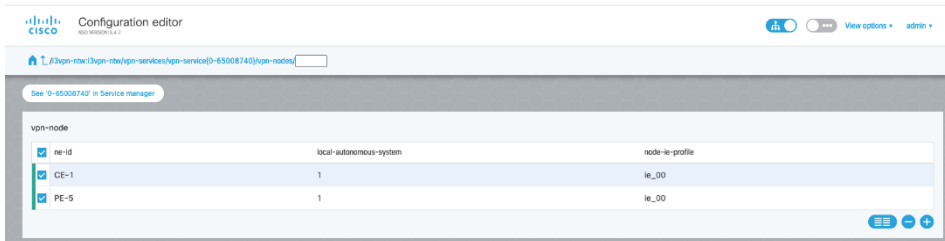
ne-id: CE-1

node-ie-profile: ie_00

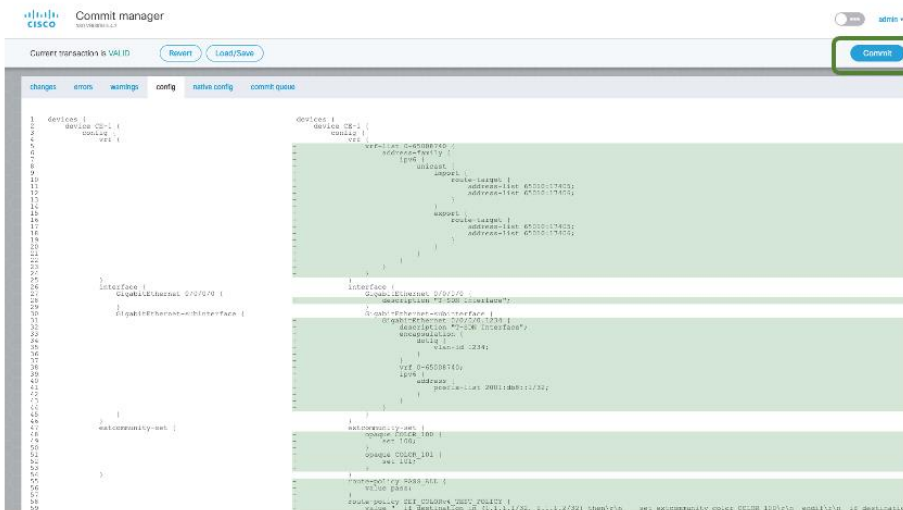
local-autonomous-system: 1



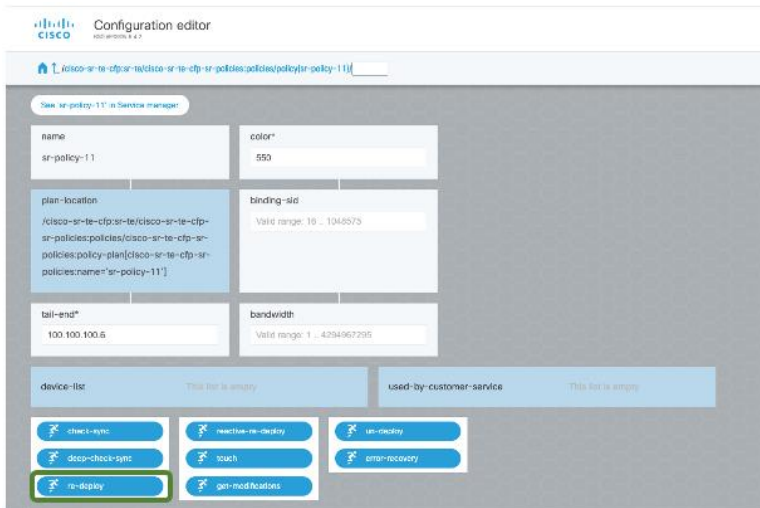
4. Verify the AS number for both devices is updated.



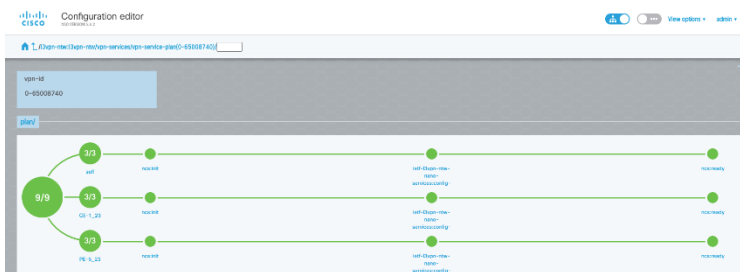
5. In the **Commit Manager** window, commit the configuration to deploy the service.



6. Navigate to the **Configuration editor** window and click **re-deploy** to redeploy the sr-te policy service (service 2).



7. Verify service 1 and service 2 are deployed successfully.



Scenario 2: Pushing a service update for a leaf-list in a configuration to an offline device results in a transient error

Explanation

This issue occurs when you try to update a service on a device that is down/offline, and the sync-direction for auto-recovery is set to the default value, sync-from. The sync-from option does not update the service as required in an auto-recovery process.

Auto-recovery or manual recovery performs the sync (sync-from or sync-to) from the device to sync the configuration between the device and the NSO CDB. The sync action to perform (sync-from or sync-to) varies with the use case and the configuration that is pushed to the device in one commit.

Auto-recovery has a global-level configuration to set the sync direction. The default value is sync-from.

Creating the scenario

The following configuration was considered to create the error scenario.

1. SR-TE policy is deployed on XR device PE-1.

2. The leaf-list **preference** value is set to 10.
3. When the device PE-1 is offline, an update to the service is pushed to modify the **preference** value to 20.

Do the following to create the error scenario:

1. Set the Commit-Queue Recovery Data (NSO auto-recovery) to **sync-from** before provisioning the services.

```
admin@ncs% show commit-queue-recovery-data
enable-polling-recovery true;
device-poller-configurations {
    poll-wait-time 20;
    sync-direction sync-from;
}
```

2. Deploy the SR-TE policy service on the PE-1 device with path 10 and verify the deployment.

- a. Push the following payload to deploy the SR-TE policy service

```
cisco-sr-te-cfp:sr-te policies policy sr-policy-12
head-end XR-PE-1
!
tail-end 100.100.100.44
color 25
path 10
dynamic metric-type te
dynamic metric-margin relative 100
dynamic constraints affinity rule include-all
!
!
!
```

- b. Verify the service configuration is pushed to the device.

```
segment-routing
traffic-eng
policy srte_c_25_ep_100.100.100.44
color 25 end-point ipv4 100.100.100.44
candidate-paths
preference 10
dynamic
metric
type te
margin relative 100
!
!
!
```

```

!
!
!
!
end

```

c. Display and view the plan for the service deployed.

```

admin@ncs# show cisco-sr-te-cfp:sr-te policies policy-plan sr-policy-12
plan | tab

```

TYPE	NAME	BACK TRACK	STATUS GOAL	CODE	STATE	STATUS	WHEN	POST ref	ACTION STATUS
Self	self	false	-		init	reached	2021-03-26T19:52:41	-	-
					Ready	reached	2021-03-26T19:52:50	-	-
	cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end		XR-PE-1	false	-	init	reached	2021-03-26T19:52:41	- -
	cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply					reached	2021-03-26T19:52:41	-	-
					ready	reached	2021-03-26T19:52:50	-	-

3. At this point, the device goes down or is offline.

4. While the device is down, update the SR-TE Policy service that you deployed on PE-1 to modify the path value from 10 to 20.

** Service Update - delete Path-10 and add Path-20

```

admin@ncs(config)# commit dry-run
cli {
  local-node {
    data devices {
      device XR-PE-1 {
        config {
          segment-routing {
            traffic-eng {
              policy srte_c_25_ep_100.100.100.44 {
                candidate-paths {
                  preference 10 {
                    dynamic {
                      metric {
                        type te;
                        margin {
                          relative 100;
                        }
                      }
                    }
                  }
                constraints {
                  affinity {
                    rule include-all;
                  }
                }
              }
            }
          }
        }
      }
    }
  }
  + preference 20 {

```

```

+           constraints {
+               segments {
+                   sid-algorithm 129;
+               }
+               disjoint-path {
+                   group-id 10;
+                   type node;
+                   sub-id 15;
+               }
+           }
+       }
+   }
+   }
+   }
}

cisco-sr-te-cfp:sr-te {
  policies {
    policy sr-policy-12 {
-     path 10 {
-       dynamic {
-         metric-type te;
-         metric-margin {
-           relative 100;
-         }
-         constraints {
-           affinity {
-             rule include-all;
-           }
-         }
-       }
+     path 20 {
+       explicit {
+         constraints {
+           disjoint-path {
+             type node;
+             group-id 10;
+             sub-id 15;
+           }
+         }
+         segments {
+           sid-algorithm 129;

```

```
      +
      +
      +
      +
      }
    }
  }
}
```

5. Commit the update.

```
admin@ncs(config)# commit
Commit complete
```

The following errors occur upon commit. This is because the device PE-1 is down or offline and hence the service has failed to update as required.

```
admin@ncs(config)# *** ALARM connection-failure: Failed to connect to device XR-PE-1: connection refused: NEDCOM CONNECT: The kexTimeout (3000 ms) expired. in new state
admin@ncs(config)# *** ALARM commit-through-queue-failed: Commit queue item 1616788869473 has failed: Failed to connect to device XR-PE-1: connection refused: NEDCOM CONNECT: The kexTimeout (3000 ms) expired. in new state
```

6. Display and view the plan for the service update. The plan shows as failed.

```
admin@ncs# show cisco-sr-te-cfp:sr-te policies policy-plan sr-policy-12 plan
```

TYPE	NAME	BACK		STATUS	CODE	STATE	STATUS	WHEN	POST ACTION	
		TRACK	GOAL						ref	STATUS
self	self	false	-	-	init	ready	reached	2021-03-26T19:52:41	-	-
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end	XR-PE-1	false	-	TSDN-SR-301	init	ready	failed	2021-03-26T20:01:13	-	-
					cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply	ready	reached	2021-03-26T19:52:41	-	-
					ready	ready	failed	2021-03-26T19:52:50	-	-

```

plan failed
plan error-info message "Failed to connect to device XR-PE-1: connection
refused: NEDCOM CONNECT: The kexTimeout (3000 ms) expired. in new state"
plan status-code-detail cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end
XR-PE-1
code TSDN-SR-301
context "Device unreachable"
context-msg "Failed to connect to device XR-PE-1: connection refused: NEDCOM
CONNECT: The kexTimeout (3000 ms) expired. in new state"
severity ERROR
recommended-action "Check device connectivity from NSO and perform recovery
steps."
    
```

7. At this stage, the device is back online and auto-recovery is triggered.

8. Display the service plan. The plan is in the **reached** state.

```
admin@ncs# show cisco-sr-te-cfp:sr-te policies policy-plan sr-policy-12 plan
```

TYPE	NAME	BACK		STATUS	CODE	STATE	STATUS	WHEN	POST ACTION	
		TRACK	GOAL						ref	STATUS
self	self	false	-	-	init	ready	reached	2021-03-26T19:52:41	-	-
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end	XR-PE-1	false	-	-	init	ready	reached	2021-03-26T20:04:14	-	-
					cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply	ready	reached	2021-03-26T19:52:41	-	-
					ready	ready	reached	2021-03-26T20:04:08	-	-

9. Validate the configuration pushed on the device after auto-recovery.

Note that the configuration displays the old value and the new value for the leaf-list preference. The old value for preference is not deleted and the new value for the same is also added.

```

RP/0/RP0/CPU0:r1#show running-config segment-routing traffic-eng policy
srte_c_25_ep_100.100.100.44
Fri Mar 26 13:04:43.868 PDT
segment-routing
traffic-eng
policy srte_c_25_ep_100.100.100.44
color 25 end-point ipv4 100.100.100.44
candidate-paths
preference 10 -----> Configuration not deleted on device.
dynamic
metric
    
```

```

    type te
    margin relative 100
    !
    !
    !
  preference 20
  constraints
  segments
    sid-algorithm 129
    !
    disjoint-path group-id 10 type node sub-id 15
    !
    !
    !
    !
    !
  !
!
```

The Refcount parameter in service metadata is incremented.

```

admin@ncs(config)# show full-configuration devices device XR-PE-1
config segment-routing traffic-eng policy srte_c_25_ep_100.100.100.44 | display
service-meta-data
devices device XR-PE-1
  config
    ! Refcount: 2
    ! Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-policies-
internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-plan[cisco-sr-te-
cfp-sr-policies-internal:name='sr-policy-12'] [cisco-sr-te-cfp-sr-policies-
internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-internal:plan/cisco-sr-
te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-policies-
internal:type='ncs:self'] [cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ]
    segment-routing
      ! Refcount: 2
      ! Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-policies-
internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-plan[cisco-sr-te-
cfp-sr-policies-internal:name='sr-policy-12'] [cisco-sr-te-cfp-sr-policies-
internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-internal:plan/cisco-sr-
te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-policies-
internal:type='ncs:self'] [cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ]
      traffic-eng
        ! Refcount: 2
        ! Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-
policies-internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-
plan[cisco-sr-te-cfp-sr-policies-internal:name='sr-policy-12'] [cisco-sr-te-cfp-
```

```

sr-policies-internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-
internal:plan/cisco-sr-te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-
policies-internal:type='ncs:self'][cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ]
    policy srte_c_25_ep_100.100.100.44
        ! Refcount: 1
        color 25 end-point ipv4 100.100.100.44
        ! Refcount: 2 -----> Ref
Count to 2 , means config already existed on device. On deleting service this
config for preference10 will not be removed from device.
        ! Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-
policies-internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-
plan[cisco-sr-te-cfp-sr-policies-internal:name='sr-policy-12'][cisco-sr-te-cfp-
sr-policies-internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-
internal:plan/cisco-sr-te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-
policies-internal:type='ncs:self'][cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ]
    candidate-paths
    preference 10
    dynamic
    metric
    type te
    margin relative 100
    !
    !
    !
    ! Refcount: 1
    ! Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-
policies-internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-
plan[cisco-sr-te-cfp-sr-policies-internal:name='sr-policy-12'][cisco-sr-te-cfp-
sr-policies-internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-
internal:plan/cisco-sr-te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-
policies-internal:type='ncs:self'][cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ]
    preference 20
    constraints
    segments
    ! Refcount: 1
    sid-algorithm 129
    !
    ! Refcount: 1
    disjoint-path group-id 10 type node sub-id 15
    !
    !
    !

```


!
!
!
!
!

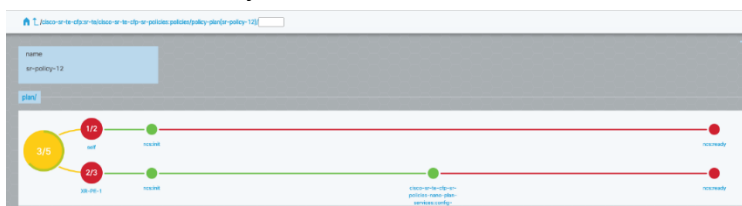
Solution

You can recover from this error condition in any of the following ways after the service update has failed and before bringing the device back online:

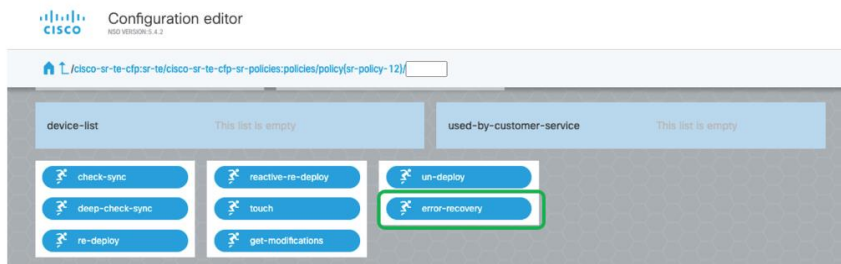
- Use manual recovery
- Change the sync-direction in auto-recovery from sync-from to sync-to.

Manual Recovery

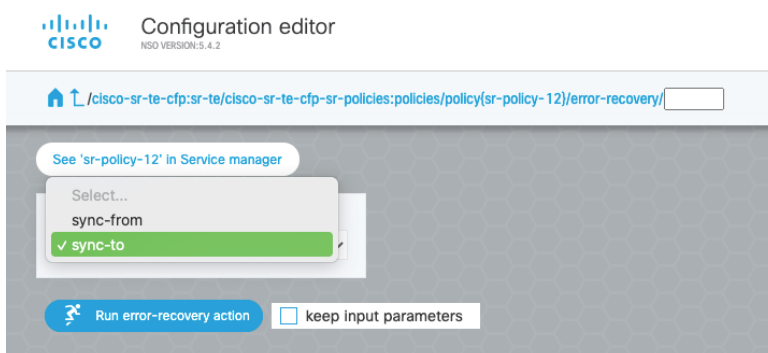
1. In the NSO UI, verify that the service is marked failed and that the device PE-1 is down/offline.

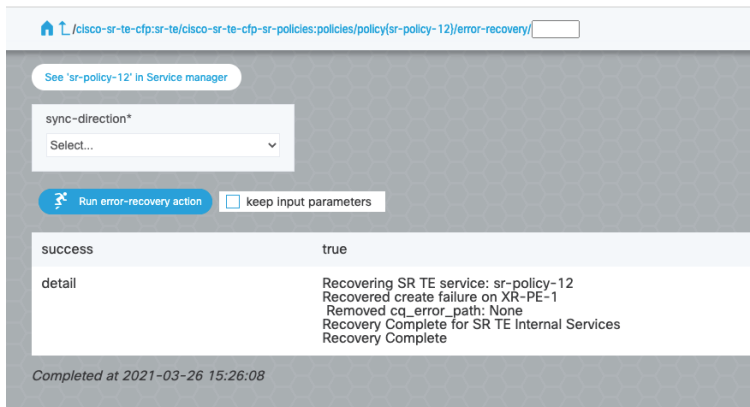


2. Bring back the device online and verify if you can connect to the device.
3. Navigate to the **Configuration editor** window and click **error-recovery** to perform the recovery at the service level.

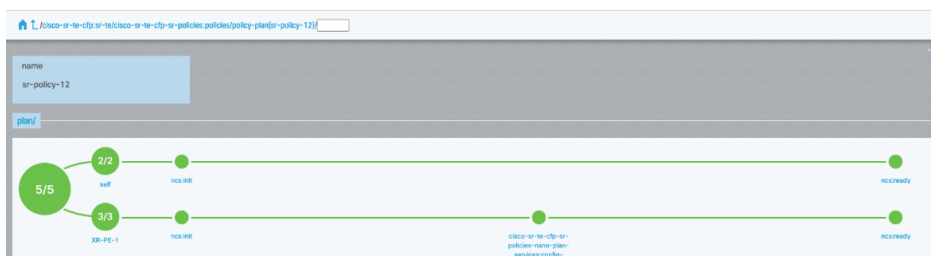


4. Set error-recovery to **sync-to**, then click **Run error-recovery action**. After successful recovery, a message is displayed.





5. Verify the service is successfully deployed.



6. Verify the configuration is correctly applied on the device.

```

** SR-TE Policy Device **
RP/0/RP0/CPU0:r1#show run segment-routing traffic-eng policy
srte_c_25_ep_100.100.100.44
Fri Mar 26 13:51:23.669 PDT
segment-routing
 traffic-eng
  policy srte_c_25_ep_100.100.100.44
   color 25 end-point ipv4 100.100.100.44
   candidate-paths
    preference 20
    constraints
     segments
      sid-algorithm 129
     !
    disjoint-path group-id 10 type node sub-id 15
   !
  !
 !
 !
 !
 !
 !
 !
 !
 !
 !

```

7. Display the service metadata and verify the **Refcount** value is 1 and the leaf-list **Preference** parameter is updated to 20.

```

** Service Meta Data **
admin@ncs% show devices device XR-PE-1 config segment-routing traffic-eng policy
srte_c_25_ep_100.100.100.44 | display service-meta-data
color {
    /* Refcount: 1 */
    value 25;
    end-point {
        /* Refcount: 1 */
        ipv4 100.100.100.44;
    }
}
/* Refcount: 1 */
/* Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-policies-
internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-plan[cisco-sr-te-
cfp-sr-policies-internal:name='sr-policy-12'] [cisco-sr-te-cfp-sr-policies-
internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-internal:plan/cisco-sr-
te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-policies-
internal:type='ncs:self'] [cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ] */
candidate-paths {
    /* Refcount: 1 */ -----> Present for path preference 20
    /* Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-
policies-internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-
plan[cisco-sr-te-cfp-sr-policies-internal:name='sr-policy-12'] [cisco-sr-te-cfp-
sr-policies-internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-
internal:plan/cisco-sr-te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-
policies-internal:type='ncs:self'] [cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ] */
    preference 20 {
        constraints {
            segments {
                /* Refcount: 1 */
                sid-algorithm 129;
            }
            disjoint-path {
                /* Refcount: 1 */
                group-id 10;
                /* Refcount: 1 */
                type node;
                /* Refcount: 1 */
                sub-id 15;
            }
        }
    }
}

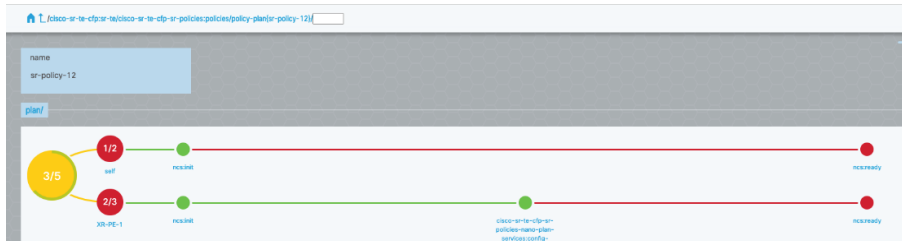
```

```
}
}
```

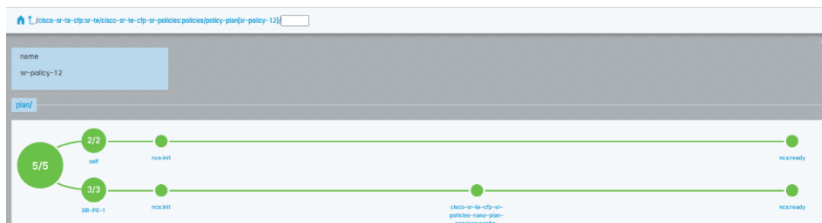
Auto Recovery

Set the sync-direction to sync-to to use the auto-recovery method to resolve the transient error.

1. In NSO UI, make sure the service has failed and the device is down/offline.



2. Set up auto-recovery to **sync-to** in the **device-poller-configurations** pane.
3. Bring back the device online.
4. Verify the service is deployed successfully.



5. Verify the configuration is correctly applied on the device.

** SR-TE Policy Device **

```

RP/0/RP0/CPU0:r1#show run segment-routing traffic-eng policy
srte_c_25_ep_100.100.100.44
Fri Mar 26 13:51:23.669 PDT
segment-routing
  traffic-eng
    policy srte_c_25_ep_100.100.100.44
      color 25 end-point ipv4 100.100.100.44
      candidate-paths
        preference 20
        constraints
          segments
            sid-algorithm 129
          !
          disjoint-path group-id 10 type node sub-id 15
          !
          !
          !
          !
          !
          !

```

6. Display the service metadata and verify the **Refcount** value is 1 and the leaf-list **Preference** parameter is updated to 20.

```

** Service Meta Data **
admin@ncs% show devices device XR-PE-1 config segment-routing traffic-eng policy
srte_c_25_ep_100.100.100.44 | display service-meta-data
color {
  /* Refcount: 1 */
  value 25;
  end-point {
    /* Refcount: 1 */
    ipv4 100.100.100.44;
  }
}
/* Refcount: 1 */
/* Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-policies-
internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-plan[cisco-sr-te-
cfp-sr-policies-internal:name='sr-policy-12'] [cisco-sr-te-cfp-sr-policies-
internal:head-end='XR-PE-1'] /cisco-sr-te-cfp-sr-policies-internal:plan/cisco-sr-
te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-policies-
internal:type='ncs:self'] [cisco-sr-te-cfp-sr-policies-
internal:name='self'] /cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ] */
candidate-paths {
  /* Refcount: 1 */ -----> Present for path preference 20
  /* Backpointer: [ /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-
policies-internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-

```

```

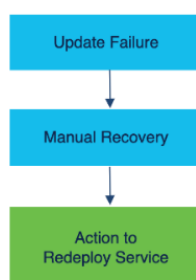
plan[cisco-sr-te-cfp-sr-policies-internal:name='sr-policy-12'][cisco-sr-te-cfp-
sr-policies-internal:head-end='XR-PE-1']/cisco-sr-te-cfp-sr-policies-
internal:plan/cisco-sr-te-cfp-sr-policies-internal:component[cisco-sr-te-cfp-sr-
policies-internal:type='ncs:self'][cisco-sr-te-cfp-sr-policies-
internal:name='self']/cisco-sr-te-cfp-sr-policies-internal:state[cisco-sr-te-
cfp-sr-policies-internal:name='cisco-sr-te-cfp-sr-policies-nano-services:config-
apply'] ] */
  preference 20 {
    constraints {
      segments {
        /* Refcount: 1 */
        sid-algorithm 129;
      }
      disjoint-path {
        /* Refcount: 1 */
        group-id 10;
        /* Refcount: 1 */
        type    node;
        /* Refcount: 1 */
        sub-id  15;
      }
    }
  }
}

```

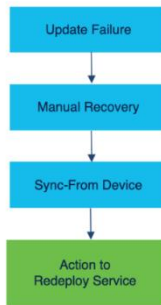
Non Commit-Queue Error Recovery Flow

This topic shows the error recovery flow when the `async` flag is set to `false`. The following flow charts show `sync-from` as an example. Depending on where the primary/golden config resides, perform either `sync-from` or `sync-to` on the device, as required.

Transient Update Failure Recovery Flow

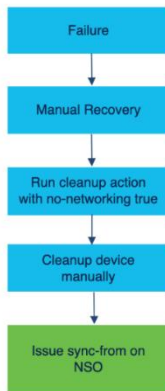


Persistent Update Failure Recovery Flow



Delete Failure Recovery Flow

Delete failure can occur when the device is out of sync or unreachable and when the error is transient or persistent.



Appendix C - Yang Models

This topic contains the yang models for the CFP and Example Function Packs services.

IMPORTANT:

Any modifications to the T-SDN FP Bundle Core Function Pack source code, including the YANG models, invalidate the support contract. You can customize by using supported third party API and custom templates.

CFP Services

This topic discusses the yang models for the T-SDN FP Bundle CFP services.

- SR-ODN Yang Models
- SR-Policy Yang Models
- CS SR-TE Yang Models
- IETF-L2VPN-NM Yang Models
- IETF-L3VPN-NM Yang Models

SR-ODN Yang Models

This section includes:

- SR-ODN Service Model
- SR-ODN Service Plan Model

SR-ODN Service Model

```

module: cisco-sr-te-cfp-sr-odn
  augment /cisco-sr-te-cfp:sr-te:
    +--rw odn
      +--rw odn-template* [name]
        | +--rw name?                                cisco-sr-te-cfp-sr-types:Cisco-ios-xr-
string
        | +--rw srv6!
        | | +--rw locator!
        | |   +--rw locator-name                    string
        | |   +--rw behavior?                      enumeration
        | |   +--rw binding-sid-type?              enumeration
        | +--rw performance-measurement!
        | | +--rw (profile-type)?
        | | | +--:(delay)
        | | | | +--rw delay-measurement!
        | | | |   +--rw profile?                  string
        | | | |   +--rw logging!
        | | | |   +--rw delay-exceeded?          empty

```



```

| | | +---:(liveness)
| | |     +---rw liveness-detection!
| | |     +---rw profile?           string
| | |     +---rw backup?           string
| | |     +---rw invalidation-action? enumeration
| | |     +---rw logging!
| | |     +---rw session-state-change? empty
| | +---rw reverse-path-label?      uint32
| +---rw head-end* [name]
| | +---rw name?                    -> /ncs:devices/lsa-utils:lsa/dispatch-
map/device/name
| | +---rw custom-template* [name]
| | | +---rw name?                  string
| | | +---rw variable* [name]
| | | | +---rw name?               string
| | | | +---rw value               string
| | | +---rw iteration* [number]
| | |     +---rw number?           uint16
| | |     +---rw variable* [name]
| | |         +---rw name?         string
| | |         +---rw value         string
| | +---x error-recovery
| |     +---w input
| |     | +---w sync-direction     enumeration
| |     +---ro output
| |         +---ro success         boolean
| |         +---ro detail?        string
| +---rw maximum-sid-depth?        uint8
| +---rw pce-group?                string
| +---rw color                      uint32
| +---rw bandwidth?                uint32
| +---rw dynamic!
| | +---rw pce!
| | +---rw flex-alg?                uint32
| | +---rw metric-type?            enumeration
| | +---rw metric-margin!
| | | +---rw (metric-margin)?
| | |     +---:(absolute)
| | |     | +---rw absolute?       uint32
| | |     +---:(relative)
| | |         +---rw relative?    uint32
| | +---rw affinity!
| | | +---rw rule* [action]
| | |     +---rw action?          enumeration

```

```

| | | +--rw color*    string
| | +--rw segments!
| | | +--rw sid-algorithm?  uint16
| | +--rw disjoint-path!
| |   +--rw type          enumeration
| |   +--rw group-id      uint16
| |   +--rw sub-id?       uint16
| |   +--rw source?       inet:ipv4-address
| +--rw source-address?   inet:ip-address
| +---x error-recovery
|   +---w input
|     | +---w sync-direction  enumeration
|     +--ro output
|       +--ro success         boolean
|       +--ro detail?        String

```

SR-ODN Service Plan Model

```

module: cisco-sr-te-cfp-sr-odn
  augment /cisco-sr-te-cfp:sr-te: augment /cisco-sr-te-cfp:sr-te:
    +--rw odn
      +--ro odn-template-plan* [name]
        | +--ro name?    string
        | +--ro plan
        |   +--ro component* [type name]
        |     | +--ro name?          string
        |     | +--ro type           plan-component-type-t
        |     | +--ro state* [name]
        |     | | +--ro name?        plan-state-name-t
        |     | | +--ro status?      plan-state-status-t
        |     | | +--ro when?       yang:date-and-time
        |     | | +--ro service-reference?  instance-identifier
        |     +--ro failed?         empty
        |     +--ro error-info!
        |       | +--ro message?    string
        |       | +--ro log-entry?  instance-identifier
        |       +--ro deleting?     empty
        |       +--ro service-location?  instance-identifier
      augment /cisco-sr-te-cfp:sr-te/odn/odn-template-plan/plan:
        +---- status-code-detail* [type name]
          +---- type?          ncs:plan-component-type-t
          +---- name?         string
          +---- code?         string

```

```

+---- context* [context-name]
| +---- context-name?   string
| +---- context-msg?    string
+---- severity?         enumeration
+---- recommended-action? string
+---- impacted-device?  string
augment /cisco-sr-te-cfp:sr-te/odn/odn-template-plan/plan/component:
+---- status-code?     string

```

SR-Policy Yang Models

This section includes:

- SR-Policy Service Model
- SR-Policy Service Plan Model

SR-Policy Service Model

```

module: cisco-sr-te-cfp-sr-policies
  augment /cisco-sr-te-cfp:sr-te:
    +--rw policies
      +--rw policy* [name]
        | +--rw name?                               cisco-sr-te-cfp-sr-types:Cisco-ios-xr-
string
        | +--rw custom-policy-name?                 cisco-sr-te-cfp-sr-types:Cisco-ios-xr-
string
        | +--rw head-end* [name]
        | | +--rw name?   -> /ncs:devices/lsa-utils:lsa/dispatch-map/device/name
        | +--rw srv6!
        | | +--rw locator!
        | | | +--rw locator-name                     string
        | | | +--rw behavior?                         enumeration
        | | | +--rw binding-sid-type?                 enumeration
        | +--rw tail-end                             inet:ip-address
        | +--rw color                                 uint32
        | +--rw binding-sid?                          uint32
        | +--rw performance-measurement!
        | | +--rw (profile-type)?
        | | | +--:(delay)
        | | | | +--rw delay-measurement!
        | | | | | +--rw profile?   string
        | | | | | +--rw logging!
        | | | | | +--rw delay-exceeded?  empty
        | | | +--:(liveness)

```

```

| | |      +---rw liveness-detection!
| | |      +---rw profile?           string
| | |      +---rw backup?            string
| | |      +---rw invalidation-action? enumeration
| | |      +---rw logging!
| | |      +---rw session-state-change? empty
| | +---rw reverse-path-label?       uint32
| +---rw circuit-style!
| | +---rw path-protection!
| | +---rw steering!
| | | +---rw drop-traffic-on-invalid-path? boolean
| | +---rw working-path
| | | +---rw (sr-te-path-choice)?
| | |   +---:(explicit-path)
| | |   | +---rw explicit
| | |   |   +---rw constraints
| | |   |   | +---rw affinity!
| | |   |   | | +---rw rule* [action]
| | |   |   | |   +---rw action?  enumeration
| | |   |   | |   +---rw color*   string
| | |   |   | | +---rw segments!
| | |   |   | | | +---rw sid-algorithm?  uint16
| | |   |   | | | +---rw disjoint-path!
| | |   |   | |   +---rw type           enumeration
| | |   |   | |   +---rw group-id      uint16
| | |   |   | |   +---rw sub-id?      uint16
| | |   |   | |   +---rw forward-sid-list-name  -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | |   |   | |   +---rw reverse-sid-list-name  -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | |   +---:(dynamic-path)
| | |   +---rw dynamic!
| | |   +---rw constraints
| | |   | +---rw affinity!
| | |   | | +---rw rule* [action]
| | |   | |   +---rw action?  enumeration
| | |   | |   +---rw color*   string
| | |   | | +---rw segments!
| | |   | | | +---rw sid-algorithm?  uint16
| | |   | | | +---rw protection?     enumeration
| | |   | |   +---rw sid-limit?     uint32
| | |   | |   +---rw disjoint-path!
| | |   | |   +---rw type           enumeration
| | |   | |   +---rw group-id      uint16
| | |   | |   +---rw sub-id?      uint16

```

```

| | | +---rw pce!
| | | +---rw metric-type? enumeration
| | | +---rw metric-margin!
| | | | +---rw (metric-margin)?
| | | | | +---:(absolute)
| | | | | | +---rw absolute? uint32
| | | | | +---:(relative)
| | | | | | +---rw relative? uint32
| | | +---rw bidirectional-association-id uint16
| | +---rw protect-path!
| | | +---rw (sr-te-path-choice)?
| | | | +---:(explicit-path)
| | | | | +---rw explicit
| | | | | | +---rw constraints
| | | | | | | +---rw affinity!
| | | | | | | | +---rw rule* [action]
| | | | | | | | | +---rw action? enumeration
| | | | | | | | | +---rw color* string
| | | | | | | | +---rw segments!
| | | | | | | | | +---rw sid-algorithm? uint16
| | | | | | | | +---rw disjoint-path!
| | | | | | | | | +---rw type enumeration
| | | | | | | | | +---rw group-id uint16
| | | | | | | | | +---rw sub-id? uint16
| | | | | | | +---rw forward-sid-list-name -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | | | +---rw reverse-sid-list-name -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | +---:(dynamic-path)
| | | | | +---rw dynamic!
| | | | | +---rw constraints
| | | | | | +---rw affinity!
| | | | | | | +---rw rule* [action]
| | | | | | | | +---rw action? enumeration
| | | | | | | | +---rw color* string
| | | | | | | +---rw segments!
| | | | | | | | +---rw sid-algorithm? uint16
| | | | | | | | +---rw protection? enumeration
| | | | | | | | +---rw sid-limit? uint32
| | | | | | | | +---rw disjoint-path!
| | | | | | | | | +---rw type enumeration
| | | | | | | | | +---rw group-id uint16
| | | | | | | | | +---rw sub-id? uint16
| | | | | +---rw pce!
| | | | | +---rw metric-type? enumeration

```

```

| | | | +---rw metric-margin!
| | | | | +---rw (metric-margin)?
| | | | | +---:(absolute)
| | | | | | +---rw absolute? uint32
| | | | | +---:(relative)
| | | | | +---rw relative? uint32
| | | | +---rw bidirectional-association-id uint16
| | | +---rw revertive? boolean
| | | +---rw wait-to-revert-timer? uint32
| | +---rw restore-path!
| | +---rw (sr-te-path-choice)?
| | | +---:(explicit-path)
| | | | +---rw explicit
| | | | +---rw constraints
| | | | | +---rw affinity!
| | | | | | +---rw rule* [action]
| | | | | | +---rw action? enumeration
| | | | | | +---rw color* string
| | | | | +---rw segments!
| | | | | +---rw sid-algorithm? uint16
| | | | | +---rw forward-sid-list-name -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | +---rw reverse-sid-list-name -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | +---:(dynamic-path)
| | | | +---rw dynamic!
| | | | +---rw constraints
| | | | | +---rw affinity!
| | | | | | +---rw rule* [action]
| | | | | | +---rw action? enumeration
| | | | | | +---rw color* string
| | | | | +---rw segments!
| | | | | | +---rw sid-algorithm? uint16
| | | | | | +---rw protection? enumeration
| | | | | +---rw sid-limit? uint32
| | | | +---rw pce!
| | | | +---rw metric-type? enumeration
| | | | +---rw metric-margin!
| | | | | +---rw (metric-margin)?
| | | | | +---:(absolute)
| | | | | | +---rw absolute? uint32
| | | | | +---:(relative)
| | | | | +---rw relative? uint32
| | | | +---rw bidirectional-association-id uint16
| | | +---rw revertive? boolean

```

```

| |     +---rw wait-to-revert-timer?   uint32
| +---rw path* [preference]
| |     +---rw preference?             uint16
| |     +---rw (sr-te-path-choice)?
| |         +---:(explicit-path)
| |             | +---rw explicit!
| |             |     +---rw sid-list* [name]
| |             |         | +---rw name?      -> /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-
sr-policies:policies/sid-list/name
| |             |         | +---rw weight?   uint32
| |             |         +---rw constraints
| |             |             +---rw affinity!
| |             |                 | +---rw rule* [action]
| |             |                 |     +---rw action?   enumeration
| |             |                 |     +---rw color*    string
| |             |                 +---rw segments!
| |             |                     | +---rw sid-algorithm?  uint16
| |             |                     +---rw disjoint-path!
| |             |                         +---rw type          enumeration
| |             |                         +---rw group-id      uint16
| |             |                         +---rw sub-id?       uint16
| |             |                         +---rw source?       inet:ipv4-address
| |         +---:(dynamic-path)
| |             +---rw dynamic!
| |                 +---rw constraints
| |                     | +---rw affinity!
| |                     | | +---rw rule* [action]
| |                     | |     +---rw action?   enumeration
| |                     | |     +---rw color*    string
| |                     | +---rw segments!
| |                     | | +---rw sid-algorithm?  uint16
| |                     | +---rw disjoint-path!
| |                     | | +---rw type          enumeration
| |                     | | +---rw group-id      uint16
| |                     | | +---rw sub-id?       uint16
| |                     | | +---rw source?       inet:ipv4-address
| |                     | +---rw sid-limit?      uint32
| |                 +---rw pce!
| |                 +---rw metric-type?          enumeration
| |                 +---rw metric-margin!
| |                     +---rw (metric-margin)?
| |                         +---:(absolute)
| |                             | +---rw absolute?  uint32
| |                             +---:(relative)

```

```

| |                                     +---rw relative?   uint32
| +---rw bandwidth?                    uint32
| +---rw auto-route!
| |   +---rw auto-route-metric!
| | |   +---rw (metric-choice)?
| | |   |   +---:(metric-relative-value)
| | |   |   |   +---rw metric-relative-value?   int32
| | |   |   |   +---:(metric-constant-value)
| | |   |   |   +---rw metric-constant-value?   uint32
| | +---rw include-prefixes!
| | |   +---rw include-prefix* [prefix-address]
| | |   |   +---rw prefix-address?   tailf:ipv4-address-and-prefix-length
| | +---rw force-sr-include?   empty
| | +---rw forward-class?   uint8
| +---rw source-address?   inet:ip-address
| +---x error-recovery
| | +---w input
| | |   +---w sync-direction   enumeration
| | +---ro output
| | |   +---ro success   boolean
| | |   +---ro detail?   string
| +---rw action
|   +---x self-test
|   +---ro output
|   |   +---ro status?   string
|   |   +---ro message?   string
+---rw sid-list* [name]
| +---rw name?   cisco-sr-te-cfp-sr-types:Cisco-ios-xr-string
| +---rw sid* [index]
| |   +---rw index?   uint32
| |   +---rw (type)
| |   |   +---:(mpls)
| |   |   |   +---rw mpls
| |   |   |   |   +---rw label   uint32
| |   |   |   +---:(ipv4)
| |   |   |   +---rw ipv4
| |   |   |   +---rw address   inet:ipv4-address
| +---rw discovered?   empty

```

SR-Policy Service Plan Model

```

module: cisco-sr-te-cfp-sr-policies
  augment /cisco-sr-te-cfp:sr-te:
    +---rw policies

```



```

+---ro policy-plan* [name]
  | +---ro name?          string
  | +---ro plan
  | | +---ro component* [type name]
  | | | +---ro name?          string
  | | | +---ro type          plan-component-type-t
  | | | +---ro state* [name]
  | | | | +---ro name?          plan-state-name-t
  | | | | +---ro status?       plan-state-status-t
  | | | | +---ro when?        yang:date-and-time
  | | | | +---ro service-reference? instance-identifier
  | | +---ro failed?          empty
  | | +---ro error-info!
  | | | +---ro message?       string
  | | | +---ro log-entry?     instance-identifier
augment /cisco-sr-te-cfp:sr-te/policies/policy-plan/plan:
+----- status-code-detail* [type name]
  +----- type?              ncs:plan-component-type-t
  +----- name?              string
  +----- code?              string
  +----- context* [context-name]
  | +----- context-name?    string
  | +----- context-msg?     string
  +----- severity?          enumeration
  +----- recommended-action? string
  +----- impacted-device?   string
augment /cisco-sr-te-cfp:sr-te/policies/policy-plan/plan/component:
+----- status-code?        string

```

CS SR-TE Yang Models

This section includes:

- CS SR-TE Service Model
- CS SR-TE Service Plan Model

CS SR-TE Service Model

```

module: cisco-cs-sr-te-cfp
  +---rw cs-sr-te-policy* [name]
  | +---rw custom-template* [name]
  | | +---rw name          -> /ct-info:custom-template-info/template-name
  | | +---rw variable* [name]
  | | | +---rw name        -> deref(..../name)/../ct-info:variables
  | | | +---rw value       string
  | | +---rw iteration* [number]

```

```

| |     +---rw number      uint16
| |     +---rw variable* [name]
| |         +---rw name      -> deref(..../name)/../ct-info:variables
| |         +---rw value     string
| +---rw name                string
| +---rw head-end
| | +---rw device            -> /ncs:devices/lsa-utils:lsa/dispatch-
map/device/name
| | +---rw ip-address        inet:ip-address
| | +---rw source-address?   inet:ip-address
| | +---rw custom-template* [name]
| |     +---rw name          -> /ct-info:custom-template-info/template-name
| |     +---rw variable* [name]
| |         | +---rw name      -> deref(..../name)/../ct-info:variables
| |         | +---rw value     string
| |     +---rw iteration* [number]
| |         +---rw number      uint16
| |         +---rw variable* [name]
| |             +---rw name      -> deref(..../name)/../ct-info:variables
| |             +---rw value     string
| +---rw tail-end
| | +---rw device            -> /ncs:devices/lsa-utils:lsa/dispatch-
map/device/name
| | +---rw ip-address        inet:ip-address
| | +---rw source-address?   inet:ip-address
| | +---rw custom-template* [name]
| |     +---rw name          -> /ct-info:custom-template-info/template-name
| |     +---rw variable* [name]
| |         | +---rw name      -> deref(..../name)/../ct-info:variables
| |         | +---rw value     string
| |     +---rw iteration* [number]
| |         +---rw number      uint16
| |         +---rw variable* [name]
| |             +---rw name      -> deref(..../name)/../ct-info:variables
| |             +---rw value     string
| +---rw (color-choice)?
| | +---:(auto-color)
| | | +---rw auto-color?      empty
| | +---:(color)
| | | +---rw color?          uint32
| +---rw bandwidth?         uint32
| +---rw disjoint-path!
| | +---rw forward-path
| | | +---rw type            enumeration
| | | +---rw (group-id-type)?
| | | | +---:(group-id)
| | | | | +---rw group-id?   uint16
| | | | +---:(auto-group-id)
| | | | | +---rw auto-group-id? empty
| | | | +---rw sub-id?       uint16
| | +---rw reverse-path
| | | +---rw type            enumeration
| | | +---rw (group-id-type)?
| | | | +---:(group-id)
| | | | | +---rw group-id?   uint16
| | | | +---:(auto-group-id)
| | | | | +---rw auto-group-id? empty
| | | | +---rw sub-id?       uint16
| +---rw steering!
| | +---rw drop-traffic-on-invalid-path?  boolean
| +---rw path-protection!

```

```

| +--rw performance-measurement!
| | +--rw (profile-type)?
| | | +--:(delay)
| | | | +--rw delay-measurement!
| | | | | +--rw profile? string
| | | | | +--rw logging!
| | | | | +--rw delay-exceeded? empty
| | | +--:(liveness)
| | | | +--rw liveness-detection!
| | | | | +--rw profile? string
| | | | | +--rw backup? string
| | | | | +--rw invalidation-action? enumeration
| | | | | +--rw logging!
| | | | | +--rw session-state-change? empty
| | +--rw reverse-path-label? uint32
| +--rw working-path
| | +--rw (sr-te-path-choice)
| | | +--:(explicit-path)
| | | | +--rw explicit
| | | | | +--rw constraints
| | | | | | +--rw affinity!
| | | | | | | +--rw rule* [action]
| | | | | | | | +--rw action enumeration
| | | | | | | | +--rw color* string
| | | | | | +--rw segments!
| | | | | | | +--rw sid-algorithm? uint16
| | | | | +--rw (forward-sid-list-type)?
| | | | | | +--:(forward-sid-list-name)
| | | | | | | +--rw forward-sid-list-name? -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | | | +--:(forward-sid-list)
| | | | | | | | +--rw forward-sid-list!
| | | | | | | | | +--rw sid* [index]
| | | | | | | | | | +--rw index uint32
| | | | | | | | | | +--rw (type)
| | | | | | | | | | | +--:(mpls)
| | | | | | | | | | | | +--rw mpls
| | | | | | | | | | | | | +--rw label uint32
| | | | | | | | | | | +--:(ipv4)
| | | | | | | | | | | | +--rw ipv4
| | | | | | | | | | | | | +--rw address inet:ipv4-address
| | | | | | +--rw (reverse-sid-list-type)?
| | | | | | | +--:(reverse-sid-list-name)
| | | | | | | | +--rw reverse-sid-list-name -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | | | | +--:(reverse-sid-list)
| | | | | | | | | +--rw reverse-sid-list!
| | | | | | | | | | +--rw sid* [index]
| | | | | | | | | | | +--rw index uint32
| | | | | | | | | | | +--rw (type)
| | | | | | | | | | | | +--:(mpls)
| | | | | | | | | | | | | +--rw mpls
| | | | | | | | | | | | | | +--rw label uint32
| | | | | | | | | | | | +--:(ipv4)
| | | | | | | | | | | | | +--rw ipv4
| | | | | | | | | | | | | | +--rw address inet:ipv4-address
| | | | | | +--:(dynamic-path)
| | | | | | | +--rw dynamic!
| | | | | | | | +--rw constraints
| | | | | | | | | +--rw affinity!
| | | | | | | | | | +--rw rule* [action]

```

```

| | | | | +---rw action enumeration
| | | | | +---rw color* string
| | | | | +---rw segments!
| | | | | | +---rw sid-algorithm? uint16
| | | | | | +---rw protection? enumeration
| | | | | +---rw sid-limit? uint32
| | | | +---rw pce!
| | | | +---rw metric-type? enumeration
| | | | +---rw metric-margin!
| | | | | +---rw (metric-margin)?
| | | | | | +---:(absolute)
| | | | | | | +---rw absolute? uint32
| | | | | | | +---:(relative)
| | | | | | | +---rw relative? uint32
| | | | +---rw (bidirectional-association-choice)?
| | | | | +---:(auto-bidirectional-association)
| | | | | | +---rw auto-bidirectional-association? empty
| | | | | +---:(bidirectional-association-id)
| | | | | | +---rw bidirectional-association-id? uint16
| | +---rw protect-path!
| | | +---rw (sr-te-path-choice)
| | | | +---:(explicit-path)
| | | | | +---rw explicit
| | | | | | +---rw constraints
| | | | | | | +---rw affinity!
| | | | | | | | +---rw rule* [action]
| | | | | | | | | +---rw action enumeration
| | | | | | | | | +---rw color* string
| | | | | | | +---rw segments!
| | | | | | | | +---rw sid-algorithm? uint16
| | | | | +---rw (forward-sid-list-type)?
| | | | | | +---:(forward-sid-list-name)
| | | | | | | +---rw forward-sid-list-name? -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | | | | +---:(forward-sid-list)
| | | | | | | | +---rw forward-sid-list!
| | | | | | | | | +---rw sid* [index]
| | | | | | | | | | +---rw index uint32
| | | | | | | | | | +---rw (type)
| | | | | | | | | | | +---:(mpls)
| | | | | | | | | | | | +---rw mpls
| | | | | | | | | | | | | +---rw label uint32
| | | | | | | | | | | +---:(ipv4)
| | | | | | | | | | | | +---rw ipv4
| | | | | | | | | | | | | +---rw address inet:ipv4-address
| | | | | +---rw (reverse-sid-list-type)?
| | | | | | +---:(reverse-sid-list-name)
| | | | | | | +---rw reverse-sid-list-name -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | | | | | | +---:(reverse-sid-list)
| | | | | | | | +---rw reverse-sid-list!
| | | | | | | | | +---rw sid* [index]
| | | | | | | | | | +---rw index uint32
| | | | | | | | | | +---rw (type)
| | | | | | | | | | | +---:(mpls)
| | | | | | | | | | | | +---rw mpls
| | | | | | | | | | | | | +---rw label uint32
| | | | | | | | | | | +---:(ipv4)
| | | | | | | | | | | | +---rw ipv4
| | | | | | | | | | | | | +---rw address inet:ipv4-address
| | | | | | +---:(dynamic-path)

```

```

| | | +---rw dynamic!
| | |   +---rw constraints
| | |     | +---rw affinity!
| | |       | | +---rw rule* [action]
| | |         | |   +---rw action    enumeration
| | |         | |   +---rw color*   string
| | |         | +---rw segments!
| | |         | | +---rw sid-algorithm? uint16
| | |         | | +---rw protection?  enumeration
| | |         | +---rw sid-limit?    uint32
| | |       +---rw pce!
| | |       +---rw metric-type?      enumeration
| | |       +---rw metric-margin!
| | |         | +---rw (metric-margin)?
| | |           | +---:(absolute)
| | |             | +---rw absolute?  uint32
| | |             | +---:(relative)
| | |               +---rw relative?  uint32
| | |       +---rw (bidirectional-association-choice)?
| | |         +---:(auto-bidirectional-association)
| | |         | +---rw auto-bidirectional-association? empty
| | |         +---:(bidirectional-association-id)
| | |           +---rw bidirectional-association-id?  uint16
| | | +---rw revertive?                boolean
| | | +---rw wait-to-revert-timer?    uint32
| +---rw restore-path!
| | +---rw (sr-te-path-choice)
| | | +---:(explicit-path)
| | | | +---rw explicit
| | | |   +---rw constraints
| | | |     | +---rw affinity!
| | | |       | | +---rw rule* [action]
| | | |         | |   +---rw action    enumeration
| | | |         | |   +---rw color*   string
| | | |         | +---rw segments!
| | | |         | | +---rw sid-algorithm? uint16
| | | |         | | +---rw (forward-sid-list-type)?
| | | |         | | +---:(forward-sid-list-name)
| | | |         | | +---rw forward-sid-list-name?  -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | |         | | +---:(forward-sid-list)
| | | |         | |   +---rw forward-sid-list!
| | | |         | |     +---rw sid* [index]
| | | |         | |       +---rw index          uint32
| | | |         | |     +---rw (type)
| | | |         | |       +---:(mpls)
| | | |         | |         | +---rw mpls
| | | |         | |           | +---rw label    uint32
| | | |         | |       +---:(ipv4)
| | | |         | |         +---rw ipv4
| | | |         | |           +---rw address    inet:ipv4-address
| | | |         | | +---rw (reverse-sid-list-type)?
| | | |         | | +---:(reverse-sid-list-name)
| | | |         | | +---rw reverse-sid-list-name  -> /cisco-sr-te-cfp:sr-
te/cisco-sr-te-cfp-sr-policies:policies/sid-list/name
| | | |         | | +---:(reverse-sid-list)
| | | |         | |   +---rw reverse-sid-list!
| | | |         | |     +---rw sid* [index]
| | | |         | |       +---rw index          uint32
| | | |         | |     +---rw (type)
| | | |         | |       +---:(mpls)

```

```

| | | | | +---rw mpls
| | | | | | +---rw label uint32
| | | | | +---:(ipv4)
| | | | | | +---rw ipv4
| | | | | | | +---rw address inet:ipv4-address
| | | | +---:(dynamic-path)
| | | | | +---rw dynamic!
| | | | | | +---rw constraints
| | | | | | | +---rw affinity!
| | | | | | | | +---rw rule* [action]
| | | | | | | | | +---rw action enumeration
| | | | | | | | | +---rw color* string
| | | | | | | +---rw segments!
| | | | | | | | +---rw sid-algorithm? uint16
| | | | | | | | +---rw protection? enumeration
| | | | | | | | +---rw sid-limit? uint32
| | | | | | +---rw pce!
| | | | | | +---rw metric-type? enumeration
| | | | | | +---rw metric-margin!
| | | | | | | +---rw (metric-margin)?
| | | | | | | | +---:(absolute)
| | | | | | | | | +---rw absolute? uint32
| | | | | | | | | +---:(relative)
| | | | | | | | | +---rw relative? uint32
| | | | | | +---rw (bidirectional-association-choice)?
| | | | | | | +---:(auto-bidirectional-association)
| | | | | | | | +---rw auto-bidirectional-association? empty
| | | | | | | +---:(bidirectional-association-id)
| | | | | | | | +---rw bidirectional-association-id? uint16
| | | | +---rw revertive? boolean
| | | | +---rw wait-to-revert-timer? uint32
| +---x error-recovery
| | +---w input
| | | +---w endpoint? enumeration
| | | +---w sync-direction enumeration
| | +---ro output
| | | +---ro success boolean
| | | +---ro detail? string
+---rw cs-sr-te-resource-pools
| | +---rw color-pool-name? -> /ralloc:resource-
pools/idalloc:id-pool/name
| | +---rw bidirectional-association-id-pool-name? -> /ralloc:resource-
pools/idalloc:id-pool/name
| | +---rw disjoint-group-id-pool-name? -> /ralloc:resource-
pools/idalloc:id-pool/name
+---rw cs-sr-te-actions
| | +---x internal-plan-change-handler
| | | +---w input
| | | | +---w kicker-id? string
| | | | +---w path? tailf:node-instance-identifier
| | | | +---w tid? uint32
| | +---x error-recovery
| | | +---w input
| | | | +---w service string
| | | | +---w endpoint? enumeration
| | | | +---w sync-direction enumeration
| | | +---ro output
| | | | +---ro success boolean
| | | | +---ro detail? string
+---x cleanup
| | +---w input

```

```

| | +---w service          string
| | +---w no-networking   boolean
| +---ro output
|   +---ro success        boolean
|   +---ro detail?       string

```

CS SR-TE Service Plan Model

```

module: cisco-cs-sr-te-cfp
+---ro cs-sr-te-plan* [name]
  +---ro name          string
  +---ro plan
    +---ro component* [type name]
      | +---ro name          string
      | +---ro type          plan-component-type-t
      | +---ro state* [name]
      | | +---ro name          plan-state-name-t
      | | +---ro status?      plan-state-status-t
      | | +---ro when?        yang:date-and-time
    +---ro commit-queue!
      | +---ro queue-item* [id]
      |   +---ro id          uint64
    +---ro failed?          empty
    +---ro error-info!
      | +---ro message?      string
      | +---ro log-entry?    instance-identifier
    +---ro deleting?        empty
    +---ro service-location? instance-identifier
    +---ro color-oper-data
      | +---ro color?        uint32
    +---ro bidirectional-association-id-oper-data
      | +---ro working-path-bidirectional-association-id? uint16
      | +---ro protect-path-bidirectional-association-id? uint16
      | +---ro restore-path-bidirectional-association-id? uint16
    +---ro cs-policy-oper-data
      | +---ro policies* [device]
      | | +---ro device          string
      | | +---ro policy-name?    string
      | +---ro sid-lists*        string
    +---ro disjoint-group-id-oper-data
      | +---ro forward-group-id? uint16
      | +---ro reverse-group-id? uint16
    +---ro status-code-detail* [type name]
      +---ro type          ncs:plan-component-type-t
      +---ro name          string
      +---ro code?         string
      +---ro context* [context-name]
      | +---ro context-name    string
      | +---ro context-msg?    string
      +---ro severity?      enumeration
      +---ro recommended-action? string
      +---ro impacted-device? string

```

IETF-L2VPN-NM Yang Models

This section includes:

- IETF-L2VPN-NM Service Model

- IETF-L2VPN-NM Service Plan Model
- IETF-L2VPN-NM Deviation Model
- IETF-L2VPN-NM Augmentations
- IETF-L2VPN-NM Route Policy Model

IETF-L2VPN-NM Service Model

```

module: ietf-l2vpn-ntw
  +--rw l2vpn-ntw
    +--rw vpn-profiles
      | +--rw valid-provider-identifiers
      |   +--rw qos-profile-identifier* [id]
      |     | +--rw id string
      |     +--rw routing-profile-identifier* [id]
      |       +--rw id -> /cisco-l2vpn-routing-policy:l2vpn-routing-
policy/policy-definitions/policy-definition/name
    +--rw vpn-services
      | +--rw vpn-service* [vpn-id]
      |   | +--rw vpn-id string
      |   | +--rw vpn-type? identityref
      |   | +--rw vpn-service-topology? identityref
      |   | +--rw underlay-transport
      |   |   +--ro cisco-l2vpn-ntw:discovered-underlay-transport
      |   |   +--ro cisco-l2vpn-ntw:te-tunnel-ref* [tunnel-id source]
      |   |     | +--ro cisco-l2vpn-ntw:tunnel-id? string
      |   |     | +--ro cisco-l2vpn-ntw:source? inet:ip-address
      |   |     | +--ro cisco-l2vpn-ntw:destination? inet:ip-address
      |   |     +--ro cisco-l2vpn-ntw:sr-policy-ref* [headend color endpoint]
      |   |       +--ro cisco-l2vpn-ntw:headend? string
      |   |       +--ro cisco-l2vpn-ntw:color? uint32
      |   |       +--ro cisco-l2vpn-ntw:endpoint? inet:ip-address
      |   +--rw status
      |     +--ro oper-status
      |     +--ro status? identityref
      |     +--ro last-change? yang:date-and-time
      |   +--rw vpn-nodes
      |     | +--rw vpn-node* [vpn-node-id]
      |     | +--rw vpn-node-id -> /ncs:devices/l2vpn-
utils:lsa/dispatch-map/device/name
      |     | +--rw role? identityref
      |     | +--rw signaling-option
      |     |   +--rw (signaling-option)?
      |     |     +--:(bgp)
      |     |     | +--rw (bgp-type)?
      |     |     |   +--:(evpn-bgp)
      |     |     |   +--rw evpn-policies
      |     |     |   +--rw mac-learning-mode?
      |     |     +--rw cisco-l2vpn-ntw:vpn-target* [id]
      |     |     | +--rw cisco-l2vpn-ntw:id?
      |     |     | +--rw cisco-l2vpn-ntw:route-targets* [route-
target]
      |     |     | | +--rw cisco-l2vpn-ntw:route-target?
      |     |     | +--rw cisco-l2vpn-ntw:route-target-type

```



```

rt-types:route-target-type
  |   |   |   |   |
  |   |   |   |   |   +---rw cisco-l2vpn-ntw:vpn-policies
  |   |   |   |   |   +---rw cisco-l2vpn-ntw:import-policy?   ->
  ..../..../..../te-service-mapping/te-mapping/odn/route-policy
  |   |   |   |   |   +---rw cisco-l2vpn-ntw:export-policy?   ->
  ..../..../..../te-service-mapping/te-mapping/odn/route-policy
  |   |   |   |   |   +---:(ldp-or-l2tp)
  |   |   |   |   |   +---rw ldp-or-l2tp
  |   |   |   |   |   +---rw (ldp-or-l2tp)?
  |   |   |   |   |   +---:(ldp)
  |   |   |   |   |   +---rw pw-peer-list* [peer-addr vc-id]
  |   |   |   |   |   +---rw peer-addr                               inet:ip-
address
  |   |   |   |   |   +---rw vc-id                               uint32
  |   |   |   |   |   +---rw cisco-l2vpn-ntw:mpls-label?       uint32
  |   |   |   |   | +---rw vpn-network-accesses
  |   |   |   |   |   +---rw vpn-network-access* [id]
  |   |   |   |   |   +---rw id                               vpn-common:vpn-id
  |   |   |   |   |   +---rw interface-id                   string
  |   |   |   |   |   +---rw connection
  |   |   |   |   |   | +---rw l2-termination-point?       string
  |   |   |   |   |   | +---rw encapsulation
  |   |   |   |   |   | | +---rw encaps-type                       identityref
  |   |   |   |   |   | | +---rw dot1q
  |   |   |   |   |   | | | +---rw cvlan-id                   dot1q-types:vlanid
  |   |   |   |   |   | | | +---rw tag-operations
  |   |   |   |   |   | | |   +---rw (op-choice)?
  |   |   |   |   |   | | | | +---:(pop)
  |   |   |   |   |   | | | | | +---rw pop?                     enumeration
  |   |   |   |   |   | | | | | +---:(push)
  |   |   |   |   |   | | | | | | +---rw push?                   empty
  |   |   |   |   |   | | | | | | +---:(translate)
  |   |   |   |   |   | | | | | | | +---rw translate?           enumeration
  |   |   |   |   |   | | | | | | | +---rw tag-1?                 dot1q-types:vlanid
  |   |   |   |   |   | | | | | | | +---rw cisco-l2vpn-ntw:mode? enumeration
  |   |   |   |   |   +---rw cisco-l2vpn-ntw:untagged
  |   |   |   |   |   +---rw cisco-l2vpn-ntw:tag-operations
  |   |   |   |   |   +---rw (op-choice)?
  |   |   |   |   |   | +---:(push)
  |   |   |   |   |   | | +---rw cisco-l2vpn-ntw:push?          empty
  |   |   |   |   |   | | +---:(translate)
  |   |   |   |   |   | | +---rw cisco-l2vpn-ntw:translate?
enumeration
  |   |   |   |   |   | | +---rw cisco-l2vpn-ntw:tag-1?         dot1q-
types:vlanid
  |   |   |   |   |   | | +---rw cisco-l2vpn-ntw:mode?
enumeration
  |   |   |   |   |   | | +---rw lag-interface {vpn-common:lag-interface}?
  |   |   |   |   |   | | +---rw split-horizon
  |   |   |   |   |   | |   +---rw cisco-l2vpn-ntw:enable?       boolean
  +---rw ethernet-service-oam
  |   +---rw md-name?       string
  |   +---rw md-level?     uint8
  |   +---rw y-1731* [maid]
  |   +---rw maid
string
  |   |   |   |   |   |   +---rw mep-id
uint16
  |   |   |   |   |   |   +---rw message-period?
string
  |   |   |   |   |   |   +---rw cisco-l2vpn-ntw:id-type?

```

```

enumeration
| | | | | +---rw cisco-l2vpn-ntw:sman-id?
string
| | | | | +---ro cisco-l2vpn-ntw:sman-id-allocation-data
| | | | | | +---ro cisco-l2vpn-ntw:icc-based-id? string
| | | | | | +---ro cisco-l2vpn-ntw:number-id? string
| | | | | +---rw cisco-l2vpn-ntw:y-1731-profile* [name]
| | | | | +---rw cisco-l2vpn-ntw:schedule
| | | | | | +---rw cisco-l2vpn-ntw:interval? uint8
| | | | | | +---rw cisco-l2vpn-ntw:duration? union
| | | | | +---rw cisco-l2vpn-ntw:name? -> /l2vpn-ntw/y-
1731-profile/name
| | | | | +---rw service
| | | | | | +---rw mtu? uint16
| | | | | | +---rw qos {vpn-common:qos}?
| | | | | | | +---rw qos-profile
| | | | | | | | +---rw qos-profile* [profile]
| | | | | | | | +---rw profile -> /l2vpn-ntw/vpn-
profiles/valid-provider-identifiers/qos-profile-identifier/id
| | | | | | | | +---rw direction? identityref
| | | | | | +---rw cisco-l2vpn-ntw:custom-template* [name]
| | | | | | +---rw cisco-l2vpn-ntw:name? -> /ct-info:custom-template-
info/template-name
| | | | | | +---rw cisco-l2vpn-ntw:variable* [name]
| | | | | | | +---rw cisco-l2vpn-ntw:name? -> deref(..../name)/../ct-
info:variables
| | | | | | | +---rw cisco-l2vpn-ntw:value string
| | | | | | | +---rw cisco-l2vpn-ntw:iteration* [number]
| | | | | | | +---rw cisco-l2vpn-ntw:number? uint16
| | | | | | | +---rw cisco-l2vpn-ntw:variable* [name]
| | | | | | | +---rw cisco-l2vpn-ntw:name? ->
deref(..../name)/../ct-info:variables
| | | | | | | +---rw cisco-l2vpn-ntw:value string
| | | | | | +---x cisco-l2vpn-ntw:error-recovery
| | | | | | | +---w cisco-l2vpn-ntw:input
| | | | | | | | +---w cisco-l2vpn-ntw:sync-direction enumeration
| | | | | | | +---ro cisco-l2vpn-ntw:output
| | | | | | | | +---ro cisco-l2vpn-ntw:success boolean
| | | | | | | | +---ro cisco-l2vpn-ntw:detail? string
| | | | | | +---rw cisco-l2vpn-ntw:multi-home!
| | | | | | | +---rw cisco-l2vpn-ntw:spoke-multi-homed? empty
| | | | | | | +---rw cisco-l2vpn-ntw:ethernet-segment-identifier string
| | | | | | +---rw cisco-l2vpn-ntw:te-service-mapping
| | | | | | | +---rw cisco-l2vpn-ntw:srv6!
| | | | | | | | +---rw cisco-l2vpn-ntw:locator? string
| | | | | | +---rw cisco-l2vpn-ntw:te-mapping
| | | | | | | +---rw (te)?
| | | | | | | | +---:(sr-policy)
| | | | | | | | | +---rw cisco-l2vpn-ntw:sr-policy!
| | | | | | | | | | +---rw cisco-l2vpn-ntw:policy-type? enumeration
| | | | | | | | | | +---rw cisco-l2vpn-ntw:policy string
| | | | | | | | | | +---rw cisco-l2vpn-ntw:fallback? enumeration
| | | | | | | | +---:(te-tunnel-list)
| | | | | | | | | +---rw cisco-l2vpn-ntw:te-tunnel-list!
| | | | | | | | | | +---rw (tunnel-te-id-source)
| | | | | | | | | | | +---:(te-tunnel-id)
| | | | | | | | | | | | +---rw cisco-l2vpn-ntw:te-tunnel-id?
uint16
| | | | | | | | +---:(ietf-te-service)
| | | | | | | | | +---rw cisco-l2vpn-ntw:ietf-te-service?
string

```

```

| | | | | +---rw cisco-l2vpn-ntw:fallback?
enumeration
| | | | | +---:(odn)
| | | | | +---rw cisco-l2vpn-ntw:odn!
| | | | | +---rw cisco-l2vpn-ntw:route-policy* -> /l2vpn-
ntw/vpn-profiles/valid-provider-identifiers/routing-profile-identifier/id
| | | | | +---rw cisco-l2vpn-ntw:attach-point
| | | | | +---rw (parent-rr-route-policy-choice)?
| | | | | +---:(parent-rr-route-policy)
| | | | | +---rw cisco-l2vpn-ntw:parent-rr-route-
policy? string
| | | | | +---rw cisco-l2vpn-ntw:control-word-disable? boolean
| | | | | +---rw cisco-l2vpn-ntw:custom-template* [name]
| | | | | +---rw cisco-l2vpn-ntw:name? -> /ct-info:custom-template-
info/template-name
| | | | | +---rw cisco-l2vpn-ntw:variable* [name]
| | | | | +---rw cisco-l2vpn-ntw:name? -> deref(..../name)/../ct-
info:variables
| | | | | +---rw cisco-l2vpn-ntw:value string
| | | | | +---rw cisco-l2vpn-ntw:iteration* [number]
| | | | | +---rw cisco-l2vpn-ntw:number? uint16
| | | | | +---rw cisco-l2vpn-ntw:variable* [name]
| | | | | +---rw cisco-l2vpn-ntw:name? -> deref(..../name)/../ct-
info:variables
| | | | | +---rw cisco-l2vpn-ntw:value string
| | | | | +---rw (evi-id-choice)?
| | | | | +---:(auto-evi-id)
| | | | | | +---rw cisco-l2vpn-ntw:auto-evi-id? empty
| | | | | +---:(evi-id)
| | | | | +---rw cisco-l2vpn-ntw:evi-id uint16
| | | | | +---rw (evi-source-choice)?
| | | | | +---:(auto-evi-source)
| | | | | | +---rw cisco-l2vpn-ntw:auto-evi-source? empty
| | | | | +---:(evi-source)
| | | | | +---rw cisco-l2vpn-ntw:evi-source uint32
| | | | | +---rw (evi-target-choice)?
| | | | | +---:(auto-evi-target)
| | | | | | +---rw cisco-l2vpn-ntw:auto-evi-target? empty
| | | | | +---:(evi-target)
| | | | | +---rw cisco-l2vpn-ntw:evi-target uint32
| | | | | +---x cisco-l2vpn-ntw:self-test
| | | | | +---ro cisco-l2vpn-ntw:output
| | | | | +---ro cisco-l2vpn-ntw:status? string
| | | | | +---ro cisco-l2vpn-ntw:message? string
| | | | | +---rw cisco-l2vpn-ntw:control-word? enumeration
| | | | | +---rw cisco-l2vpn-ntw:vpn-target!
| | | | | +---rw (hub-rt-choice)
| | | | | | +---:(auto-hub-rt)
| | | | | | | +---rw cisco-l2vpn-ntw:auto-hub-rt? empty
| | | | | | +---:(hub-rt-value)
| | | | | | +---rw cisco-l2vpn-ntw:hub-rt-value? rt-types:route-target
| | | | | +---rw (spoke-rt-choice)
| | | | | +---:(auto-spoke-rt)
| | | | | | +---rw cisco-l2vpn-ntw:auto-spoke-rt? empty
| | | | | +---:(spoke-rt-value)
| | | | | +---rw cisco-l2vpn-ntw:spoke-rt-value? rt-types:route-target
| | | | | +---rw cisco-l2vpn-ntw:service-assurance!
| | | | | +---rw cisco-l2vpn-ntw:monitoring-state? aa-monitoring-state
| | | | | +---rw cisco-l2vpn-ntw:preservation? historical-data-options
| | | | | +---rw cisco-l2vpn-ntw:profile-name string

```

```

| | | +--rw cisco-l2vpn-ntw:rule-name          string
| | | +--rw cisco-l2vpn-ntw:bridge-group?     string

```

IETF-L2VPN-NM Service Plan Model

```

module: ietf-l2vpn-ntw
  +--rw l2vpn-ntw
    | +--ro cisco-l2vpn-ntw:vpn-service-plan* [vpn-id]
    |   +--ro cisco-l2vpn-ntw:vpn-id?         string
    |   +--ro cisco-l2vpn-ntw:local-sites*    string
    |   +--ro cisco-l2vpn-ntw:plan
    |     +--ro cisco-l2vpn-ntw:component* [type name]
    |       | +--ro cisco-l2vpn-ntw:name?         string
    |       | +--ro cisco-l2vpn-ntw:type         plan-
  component-type-t
    | | +--ro cisco-l2vpn-ntw:state* [name]
    | | | +--ro cisco-l2vpn-ntw:name?         plan-state-name-t
    | | | +--ro cisco-l2vpn-ntw:status?      plan-state-status-
  t
    | | | +--ro cisco-l2vpn-ntw:when?        yang:date-and-time
    | +--ro cisco-l2vpn-ntw:commit-queue!
    | | +--ro cisco-l2vpn-ntw:queue-item* [id]
    | | | +--ro cisco-l2vpn-ntw:id?          uint64
    | +--ro cisco-l2vpn-ntw:failed?          empty
    +--ro cisco-l2vpn-ntw:error-info!
    | +--ro cisco-l2vpn-ntw:message?         string
    | +--ro cisco-l2vpn-ntw:log-entry?       instance-identifier
    +--ro cisco-l2vpn-ntw:deleting?          empty
    +--ro cisco-l2vpn-ntw:service-location?  instance-identifier
    +--ro cisco-l2vpn-ntw:status-code-detail* [type name]
    | +--ro cisco-l2vpn-ntw:type?           ncs:plan-component-
  type-t
    | | +--ro cisco-l2vpn-ntw:name?           string
    | | +--ro cisco-l2vpn-ntw:code?           string
    | | +--ro cisco-l2vpn-ntw:context* [context-name]
    | | | +--ro cisco-l2vpn-ntw:context-name? string
    | | | +--ro cisco-l2vpn-ntw:context-msg? string
    | | +--ro cisco-l2vpn-ntw:severity?      enumeration
    | | +--ro cisco-l2vpn-ntw:recommended-action? string
    | | +--ro cisco-l2vpn-ntw:impacted-device? string
    +--ro cisco-l2vpn-ntw:rt-allocation-data
    | +--ro cisco-l2vpn-ntw:hub-rt?          string
    | +--ro cisco-l2vpn-ntw:spoke-rt?       string
    +--ro cisco-l2vpn-ntw:evi-allocation-data
    | +--ro cisco-l2vpn-ntw:evi-id?          string
    | +--ro cisco-l2vpn-ntw:evi-source?     string
    | +--ro cisco-l2vpn-ntw:evi-target?     string

```

IETF-L2VPN-NM Deviation Model

```

module ietf-l2vpn-ntw-deviations {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-ntw-deviations";
  prefix l2vpn-ntw-deviations;

  import ietf-l2vpn-ntw { prefix l2vpn-ntw; }
  import cisco-tsdn-core-fp-common { prefix tsdn-core-fp-common; }

```

```

import tailf-common { prefix tailf; }
import tailf-ncs { prefix ncs; }
import lsa-utils { prefix lsa-utils; }
import resource-allocator { prefix ralloc; }
import id-allocator { prefix idalloc; }
import custom-template-hook { prefix ct-hook; }
import tailf-kicker { prefix kicker; }
import ietf-vpn-common { prefix vpn-common; }
import ietf-routing-types { prefix rt-types; }
import ieee802-dot1q-types { prefix dot1q-types; }
import cisco-l2vpn-ntw { prefix cisco-l2vpn-ntw; }
import cisco-l2vpn-routing-policy { prefix cisco-l2vpn-routing-policy; }

description "IETF L2VPN NTW Cisco NSO Deviations";

...

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-profiles/l2vpn-ntw:valid-provider-
identifiers/l2vpn-ntw:routing-profile-identifier/l2vpn-ntw:id" {
    deviate replace {
        type leafref {
            path "/cisco-l2vpn-routing-policy:l2vpn-routing-policy/cisco-l2vpn-routing-
policy:policy-definitions/cisco-l2vpn-routing-policy:policy-definition/cisco-l2vpn-
routing-policy:name";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-service-topology" {
    deviate add {
        must ". = 'vpn-common:any-to-any' or . = 'vpn-common:hub-spoke' or . = 'vpn-
common:custom'" {
            error-message "vpn-service-topology must be one of the following: vpn-
common:any-to-any,"
                + " vpn-common:hub-spoke, vpn-common:custom";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-type" {
    deviate add {
        default "vpn-common:vpws";
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-
l2tp/l2vpn-ntw:ldp/l2vpn-ntw:pw-peer-list/l2vpn-ntw:vc-id" {
    deviate replace {
        type uint32 {
            range "1..4294967295";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:role" {
    deviate add {

```

```

    must ". = 'vpn-common:spoke-role' or . = 'vpn-common:hub-role'" {
        error-message "role must be one of the following: vpn-common:spoke-role, "
            + "vpn-common:hub-role";
    }
}
deviate replace {
    default "vpn-common:hub-role";
}
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:encap-type" {
    deviate add {
        must ". = 'vpn-common:dot1q' or . = 'vpn-common:untagged-int'" {
            error-message "encap-type must be one of the following: vpn-common:dot1q, "
                + "vpn-common:untagged-int";
        }
        mandatory true;
    }
    deviate delete {
        default "vpn-common:priority-tagged";
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-id" {
    deviate replace {
        type string {
            pattern "[a-zA-Z0-9\\-\\_]+";
            length "1..32";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-node-id" {
    deviate replace {
        type leafref {
            path "/ncs:devices/lsa-utils:lsa/l2vpn-ntw:dispatch-map/l2vpn-
utils:device/l2vpn-ntw:lsa-utils:name";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-
ntw:evpn-policies/cisco-l2vpn-ntw:vpn-policies/cisco-l2vpn-ntw:import-policy" {
    deviate replace {
        type leafref {
            path "../.../te-service-mapping/te-mapping/odn/route-policy";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-
ntw:evpn-policies/cisco-l2vpn-ntw:vpn-policies/cisco-l2vpn-ntw:export-policy" {

```

```

    deviate replace {
      type leafref {
        path "../../../te-service-mapping/te-mapping/odn/route-policy";
      }
    }
  }

  // L2VPN does not support assigned number field being optional. So we had to
  // remove optional number field
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
  service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
  ntw:signaling-option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-
  ntw:evpn-policies/cisco-l2vpn-ntw:vpn-target/cisco-l2vpn-ntw:route-targets/cisco-
  l2vpn-ntw:route-target" {
    deviate replace {
      type string {
        pattern
          '(0:(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
          + '6[0-4][0-9]{3})|'
          + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0):(429496729[0-5]|'
          + '42949672[0-8][0-9]|'
          + '4294967[01][0-9]{2}|429496[0-6][0-9]{3}|'
          + '42949[0-5][0-9]{4}|'
          + '4294[0-8][0-9]{5}|429[0-3][0-9]{6})|'
          + '42[0-8][0-9]{7}|4[01][0-9]{8}|'
          + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0)|'
          + '(1:(((0-9)|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|'
          + '25[0-5])\\.){3}([0-9]|1[0-9][0-9]|'
          + '1[0-9]{2}|2[0-4][0-9]|25[0-5])):(6553[0-5]|'
          + '655[0-2][0-9]|'
          + '65[0-4][0-9]{2}|6[0-4][0-9]{3})|'
          + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0)|'
          + '(2:(429496729[0-5]|42949672[0-8][0-9]|'
          + '4294967[01][0-9]{2}|'
          + '429496[0-6][0-9]{3}|42949[0-5][0-9]{4}|'
          + '4294[0-8][0-9]{5}|'
          + '429[0-3][0-9]{6}|42[0-8][0-9]{7}|4[01][0-9]{8}|'
          + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0):'
          + '(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
          + '6[0-4][0-9]{3}|'
          + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0))';
        tailf:info "0:2-octet-asn:4-octet-number
          1:4-octet-ipv4addr:2-octet-number
          2:4-octet-asn:2-octet-number";
      }
    }
  }

  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
  service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
  ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-
  l2tp/l2vpn-ntw:ldp/l2vpn-ntw:pw-peer-list" {
    deviate add {
      max-elements 1;
    }
  }

  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
  service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
  accesses/l2vpn-ntw:vpn-network-access" {
    deviate add {

```

```

        min-elements 1;
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:mtu" {
    deviate replace {
        type uint16 {
            range "64..65535";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:interface-id" {
    deviate replace {
        type string {
            pattern "(Bundle-
Ether|FiftyGigE|FortyGigE|FourHundredGigE|HundredGigE|TenGigE|TwentyFiveGigE|TwoHun-
dredGigE|GigabitEthernet|Ethernet|CEM) [0-9]+(/[0-9]+)*";
        }
    }
    deviate add {
        mandatory true;
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service" {
    deviate add {
        must "count(l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/md-name)=2 or
"+
        "count(l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/md-name)=0" {
            error-message "Unable to enable ethernet-service-oam for a single node. " +
                "It must either be enabled or disabled for both nodes.";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes" {
    deviate add {
        must "(not(..l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn') and count(l2vpn-
ntw:vpn-node) = 2)
or (..l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn' and count(l2vpn-
ntw:vpn-node) > 0)" {
            error-message "2 vpn-nodes must be configured unless if vpn-type 'vpn-
common:mpls-evpn' is used in which case a minimum of 1 vpn-node can be configured";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-accesses" {
    deviate add {
        must "..l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn' or count(l2vpn-
ntw:vpn-network-access) = 1" {
            error-message "Only a single vpn-network-access may be defined when not

```



```

using vpn-type 'mpls-evpn';
}
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:cvlan-id" {
  deviate add {
    must "../l2vpn-ntw:encap-type='vpn-common:dot1q'"{
      error-message "cvlan-id can only be set when encapsulation-type is 'vpn-
common:dot1q'";
    }
    mandatory true;
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q" {
  deviate add {
    must "../l2vpn-ntw:encap-type='vpn-common:dot1q' and l2vpn-ntw:cvlan-id"{
      error-message "cvlan-id is required when encapsulation-type is 'vpn-
common:dot1q'";
    }
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations/l2vpn-ntw:op-
choice/l2vpn-ntw:pop/l2vpn-ntw:pop" {
  deviate replace {
    type enumeration {
      enum "1" {
        tailf:info "Remove outer tag only";
      }
      enum "2" {
        tailf:info "Remove two outermost tags";
      }
    }
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations/l2vpn-ntw:op-
choice/l2vpn-ntw:translate/l2vpn-ntw:translate" {
  deviate replace {
    type enumeration {
      enum "1-to-1" {
        tailf:info "Replace the outermost tag with another tag";
      }
    }
  }
}

```

```

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731" {
  deviate add {
    max-elements 1;
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731/l2vpn-ntw:message-period" {
  deviate replace {
    type string {
      pattern '(3\\.3ms|10ms|100ms|1s|10s|1m|10m)';
    }
    default '1s';
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:md-
level" {
  deviate replace {
    type uint8 {
      range "0..7";
    }
  }
  deviate add {
    must "not(..../..../..../..../l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-
ntw:vpn-network-accesses/l2vpn-ntw:vpn-network-access[l2vpn-ntw:ethernet-service-
oam/l2vpn-ntw:md-level!=current()])" {
      error-message "md-level must be same among all vpn-nodes";
    }
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:md-
name" {
  deviate add {
    must "not(..../..../..../..../l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-
ntw:vpn-network-accesses/l2vpn-ntw:vpn-network-access[l2vpn-ntw:ethernet-service-
oam/l2vpn-ntw:md-name!=current()])" {
      error-message "md-name must be same among all vpn-nodes";
    }
  }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731/l2vpn-ntw:mep-id" {
  deviate replace {
    type uint16 {
      range "1..8191";
    }
  }
}

```

```

    deviate add {
        mandatory true;
        must "count(..../..../..../..../l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-
node/l2vpn-ntw:vpn-network-accesses/l2vpn-ntw:vpn-network-access/l2vpn-
ntw:ethernet-service-oam/l2vpn-ntw:y-1731[l2vpn-ntw:mep-id=current()]=1" {
            error-message "mep-id must be unique among all vpn-nodes";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731/l2vpn-ntw:maid" {
    deviate replace {
        type string {
            length "1..max";
        }
    }
    deviate add {
        must "not(..../..../..../..../l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-
ntw:vpn-network-accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-
oam/l2vpn-ntw:y-1731[l2vpn-ntw:maid!=current()])" {
            error-message "maid must be same among all vpn-nodes";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam" {
    deviate add {
        must "not(./md-name or ./md-level or ./y-1731) or (./md-name and ./y-1731 and
./md-level)" {
            error-message "md-name, md-level and y-1731 are required when configuring
ethernet service OAM";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:md-
level" {
    deviate replace {
        type uint8 {
            range "0..7";
        }
    }
    deviate add {
        must "not(..../..../..../..../l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-
ntw:vpn-network-accesses/l2vpn-ntw:vpn-network-access[l2vpn-ntw:ethernet-service-
oam/l2vpn-ntw:md-level!=current()])" {
            error-message "md-level must be same among all vpn-nodes";
        }
    }
}

deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:md-

```

```

name" {
  deviate add {
    must "not(..../..../..../..../l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-
ntw:vpn-network-accesses/l2vpn-ntw:vpn-network-access[l2vpn-ntw:ethernet-service-
oam/l2vpn-ntw:md-name!=current()]" {
      error-message "md-name must be same among all vpn-nodes";
    }
  }
}
}

```

IETF-L2VPN-NM Augmentations

```

module cisco-l2vpn-ntw {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-ntw";
  prefix cisco-l2vpn-ntw;

  import ietf-l2vpn-ntw { prefix l2vpn-ntw; }
  import ietf-inet-types { prefix inet; }
  import cisco-tdsn-core-fp-common { prefix tsdn-core-fp-common; }
  import tailf-common { prefix tailf; }
  import tailf-ncs { prefix ncs; }
  import lsa-utils { prefix lsa-utils; }
  import resource-allocator { prefix ralloc; }
  import id-allocator { prefix idalloc; }
  import custom-template-hook { prefix ct-hook; }
  import tailf-kicker { prefix kicker; }
  import ietf-vpn-common { prefix vpn-common; }
  import ietf-routing-types { prefix rt-types; }
  import ieee802-dot1q-types { prefix dot1q-types; }

  description "IETF L2VPN NTW Cisco NSO Modifications";

  ...

  typedef asn-ip-type {
    type string {
      tailf:info "ASN:nn or IPV4-address:nn,VPN Route Distinguisher";
      pattern '((([0-9]+)|((([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5])'
+ '\\.'){3}([0-9]|[1-9][0-9]|1[0-9][0-9]|2[0-4][0-9]|25[0-5]))):[0-9]+';
    }
  }

  grouping allocation-data {
    container rt-allocation-data {
      leaf hub-rt {
        tailf:info "hub-rt allocated by resource-manager id-allocator";
        description "hub-rt allocated by resource-manager id-allocator";
        type string;
      }
      leaf spoke-rt {
        tailf:info "spoke-rt allocated by resource-manager id-allocator";
        description "spoke-rt allocated by resource-manager id-allocator";
        type string;
      }
    }
    tailf:info "rt resource allocations";
    description "rt resource allocations";
  }
}

```

```

}
container evi-allocation-data {
  leaf evi-id {
    tailf:info "evi-id allocated by resource-manager id-allocator";
    description "evi-id allocated by resource-manager id-allocator";
    type string;
  }
  leaf evi-source {
    tailf:info "evi-source allocated by resource-manager id-allocator";
    description "evi-source allocated by resource-manager id-allocator";
    type string;
  }
  leaf evi-target {
    tailf:info "evi-target allocated by resource-manager id-allocator";
    description "evi-target allocated by resource-manager id-allocator";
    type string;
  }
  tailf:info "EVI resource allocations";
  description "EVI resource allocations";
}
}

grouping y-1731-profile-schedule {
  container schedule {
    description
      "Schedule Parameters";
    tailf:info
      "Schedule Parameters";

    leaf interval {
      description
        "<1|2|3|4|5|6|8|9|10|12|15|16|18|20|24|30|32|36|40|45|48|60|80|90>
        Interval between operations expressed in minute. Must be a factor of
1440.
        Only applicable for Cisco XR devices.";
      tailf:info
        "<1|2|3|4|5|6|8|9|10|12|15|16|18|20|24|30|32|36|40|45|48|60|80|90>
        Interval between operations expressed in minutes. Must be a factor of
1440.
        Only applicable for Cisco XR devices.";
      type uint8 {
        range "1|2|3|4|5|6|8|9|10|12|15|16|18|20|24|30|32|36|40|45|48|60|80|90";
      }
    }
    leaf duration {
      tailf:display-when "/l2vpn-ntw:l2vpn-ntw/y-1731-
profile[name=current()/.../name]/probe/measurement-interval != 'once'";

      description
        "<1-1440> Duration of operations expressed in minutes.
        The 'forever' option will try to schedule infinitely if supported;
otherwise,
        maximum number of allowed minutes will be used";
      tailf:info
        "<1-1440> Duration of operations expressed in minutes.
        The 'forever' option will try to schedule infinitely if supported;
otherwise,
        maximum number of allowed minutes will be used";
      type union {
        type uint16 {
          range "1..1440";
        }
      }
    }
  }
}

```

```

    }
    type enumeration {
      enum forever;
    }
  }
}
}
}

augment "/l2vpn-ntw:l2vpn-ntw" {
  container cfp-configurations {
    list dynamic-device-mapping {
      tailf:info "Dynamic Device Mapping";
      description "Dynamic Device Mapping";
      key "ned-id";
      leaf ned-id {
        tailf:info "Device NED ID, eg ned:netconf";
        description "Device NED ID, eg ned:netconf";
        type string;
        tailf:cli-allow-range;
      }
      leaf python-impl-class-name {
        tailf:info "Device Python implementation class, eg module.class";
        description "Device Python implementation class, eg module.class";
        type string;
        mandatory true;
      }
    }

    leaf l2nm-validation-enabled {
      tailf:hidden tsdn;
      tailf:info "If set to true, following validations are done for l2nm
service:
      1. All the site interfaces are available on the site.";
      description "If set to true, following validations are done for l2nm
service:
      1. All the site interfaces are available on the site.";
      type boolean;
      default true;
    }

    leaf iosxr-default-ned-id {
      tailf:info "NED ID of the default IOSXR Device.";
      description "NED ID of the default IOSXR Device.";
      tailf:hidden tsdn;
      type string;
      default "cisco-iosxr-cli-7.52:cisco-iosxr-cli-7.52";
    }
  }

  container id-pools {
    presence "true";
    leaf evi-id-pool-name {
      type leafref {
        path "/ralloc:resource-pools/idalloc:id-pool/idalloc:name";
      }
      tailf:info "Select a Resource pool from which EVI-ID will be auto assigned
";
      description "Select a Resource pool from which EVI-ID will be auto assigned
";
    }
  }
}

```

```

leaf evi-source-target-pool-name {
  type leafref {
    path "/ralloc:resource-pools/idalloc:id-pool/idalloc:name";
  }
  tailf:info "Select a resource pool that will be used to auto-assign EVI
source target identifier.";
  description "Select a resource pool that will be used to auto-assign EVI
source target identifier.";
}
leaf rt-pool-name {
  type leafref {
    path "/ralloc:resource-pools/idalloc:id-pool/idalloc:name";
  }
  tailf:info "Select a Resource pool from which rt-value will be auto
assigned ";
  description "Select a Resource pool from which rt-value will be auto
assigned ";
}
tailf:info "id-pools are used for evi-id, evi-source and evi-target
configs.";
description "id-pools are used for evi-id, evi-source and evi-target
configs.";
}

list y-1731-profile {
  description "L2NM Y-1731 profile.
Can be standard profile or customized profile.";
  tailf:info "L2NM Y-1731 profile.
Can be standard profile or customized profile.";

uses y-1731-profile-schedule {
  refine schedule/interval {
    default 60;
  }
  refine schedule/duration {
    default 60;
  }
}

key name;
leaf name {
  type string;
  description
  "Unique name for y-1731-profile";
  tailf:info
  "Unique name for y-1731-profile";
}
leaf type {
  type enumeration {
    enum 'delay';
    enum 'loss';
    enum 'synthetic-loss';
  }
  mandatory true;
  description
  "Performance monitor types";
  tailf:info "Performance monitor types";
}
container probe {
  description
  "SLA Profile Probe. Only applicable to Cisco XR devices.";
}

```

```

tailf:info
  "SLA Profile Probe. Only applicable to Cisco XR devices.";

leaf type {
  description "Probe Type";
  tailf:info "Probe Type";
  type enumeration {
    enum burst;
    enum packet;
  }
  default burst;

  must ". = 'burst' or (. = 'packet' and ../../type != 'loss')" {
    error-message "probe type cannot be set to packet for loss profile";
    tailf:dependency "../../type";
  }
}

container burst {
  when "../../type = 'burst'";

  description "Burst Parameters";
  tailf:info "Burst Parameters";

  leaf message-count {
    description
      "<2-1200> Defines number of OAM messages sent per interval";
    tailf:info
      "<2-1200> Defines number of OAM messages sent per interval";
    type uint16 {
      range "2..1200";
    }
    default 60;
  }

  leaf message-period {
    type uint32 {
      range "50..30000";
    }
    default 1000;
    description
      "<50-30000> Defines the interval between OAM messages.
      The message period is expressed in milliseconds";
    tailf:info
      "<50-30000> Defines the interval between OAM messages.
      The message period is expressed in milliseconds";
  }

  must "(message-period * message-count) <= (../measurement-interval *
1000)" {
    error-message "The measurement time (message-period * message-count /
1000)"
      + " must be less than or equal to the measurement-
interval time.";
  }
}

leaf measurement-interval {
  type union {
    type uint32 {
      range "1..3600";
    }
    type enumeration {
      enum once;
    }
  }
}

```



```

    }
    default 60;
    description
      "<1-3600> Specifies the measurement interval for statistics. The
      measurement interval is expressed in seconds";
    tailf:info "<1-3600> Specifies the measurement interval for statistics.
The
      measurement interval is expressed in seconds";
    must "(. = 'once' and ../type = 'packet') or (. != 'once')" {
      error-message "measurement-interval 'once' is only supported with probe
type 'packet'";
      tailf:dependency "../type";
    }
    must "not(../type = 'synthetic-loss' and . = 'once')" {
      error-message "measurement-interval 'once' not supported with
synthetic-loss profile type";
      tailf:dependency "../type";
    }
  }
}
leaf frame-size {
  when "../type != 'loss'";
  type uint16 {
    range "1..9000";
  }
  default 1000;
  description "<1-9000> Frame size";
  tailf:info "<1-9000> Frame size";
}
leaf priority {
  description "<0-7> Specify the priority to use when sending OAM
messages";
  tailf:info "<0-7> Specify the priority to use when sending OAM messages";
  type uint8 {
    range "0..7";
  }
}
}
}
container delay-params {
  when "../type = 'delay'";
  description
    "Delay Parameters";
  tailf:info
    "Delay Parameters";
  list statistic {
    description
      "Statistic";
    tailf:info
      "Statistic";
    key type;
    leaf type {
      description
        "Statistics Type. Jitter type only applicable for Cisco XR devices.";
      tailf:info
        "Statistics Type. Jitter type only applicable for Cisco XR devices.";
      type enumeration {
        enum 'delay-two-way';
        enum 'delay-sd';
        enum 'delay-ds';
        enum 'jitter-two-way';
        enum 'jitter-sd';
        enum 'jitter-ds';
      }
    }
  }
}

```

```

    }
  }
  // leaf cos {
  //   description
  //     "<0..7>;Class of service. Only applicable for Cisco XE devices.";
  //   tailf:info
  //     "<0..7>;Class of service. Only applicable for Cisco XE devices.";
  //   type uint8 {
  //     range "0..7";
  //   }
  //   default 0;
  // }
}
leaf version {
  description
    "Delay Version";
  tailf:info
    "Delay Version";
  type enumeration {
    enum 0;
    enum 1;
  }
  default 1;
}
// leaf delay-type {
//   description
//     "Delay type. Only applicable for Cisco XE devices.";
//   tailf:info
//     "Delay type. Only applicable for Cisco XE devices.";
//   type enumeration {
//     enum 'delay';
//     enum 'delay-variation';
//     enum 'both';
//   }
//   default "delay";
// }
}
container loss-params {
  when "../type = 'loss' or ../type = 'synthetic-loss'";
  description
    "Loss Parameters";
  tailf:info
    "Loss Parameters";
  list statistic {
    description
      "Statistic";
    tailf:info
      "Statistic";
    key type;
    leaf type {
      description
        "Statistics Type";
      tailf:info
        "Statistics Type";
      type enumeration {
        enum 'loss-sd';
        enum 'loss-ds';
      }
    }
  }
}
// leaf cos {
//   description

```

```

//      "<0..7>;Class of service. Only applicable for Cisco XE devices.";
//      tailf:info
//      "<0..7>;Class of service. Only applicable for Cisco XE devices.";
//      type uint8 {
//          range "0..7";
//      }
//      default 0;
//  }
}
container bucket-details {
    description
        "Configuration for buckets in which statistics are collected. Only
applicable for Cisco XR devices.";
    leaf bucket-size {
        description
            "<1-100> Specifies the size of each bucket.
The number of probes that each buckets may contain. Only applicable
for Cisco XR devices.";
        tailf:info
            "<1-100> Specifies the size of each bucket.
The number of probes that each buckets may contain. Only applicable
for Cisco XR devices.";
        type uint8 {
            range "1..100";
        }
        default 1;
    }
    leaf bucket-archive {
        description "<1-100> Number of buckets to store. Only applicable for
Cisco XR devices.";
        tailf:info "<1-100> Number of buckets to store. Only applicable for Cisco
XR devices.";
        type uint8 {
            range "1..100";
        }
        default 100;
    }
}
}
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services" {
    list vpn-service-plan {
        config false;
        tailf:cdb-oper {
            tailf:persistent true;
        }
        key "vpn-id";
        leaf vpn-id {
            type string;
        }
    }

    leaf-list local-sites {
        tailf:hidden full;
        type string;
    }
    uses ncs:nano-plan-data {
        augment "plan" {
            uses tsdn-core-fp-common:status-code-plan-augmentation;
            uses allocation-data;
        }
    }
}

```

```

    }
    augment "plan/component" {
      uses tsdn-core-fp-common:status-code-component-augmentation;
      uses tsdn-core-fp-common:static-config-redeploy-indicator-component-
augmentation;
      leaf pe {
        tailf:hidden full;
        type string;
      }
    }
  }
}
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-
ntw:underlay-transport" {
  container discovered-underlay-transport {
    tailf:info "HCO service to underlay transport mapping";
    description "HCO service to underlay transport mapping";
    config false;

    list te-tunnel-ref {
      tailf:info "List of rsvp-te tunnels";
      description "List of rsvp-te tunnels";

      key "tunnel-id source";
      leaf tunnel-id {
        tailf:info "TE tunnel Identifier";
        description "TE tunnel Identifier";
        type string;
      }
      leaf source {
        tailf:info "TE tunnel Source IP";
        description "TE tunnel Source IP";
        type inet:ip-address;
      }
      leaf destination {
        tailf:info "TE tunnel Destination IP";
        description "TE tunnel Destination IP";
        type inet:ip-address;
      }
    }
  }

  list sr-policy-ref {
    tailf:info "List of sr-policies (static or dynamically created)";
    description "List of sr-policies (static or dynamically created)";

    key "headend color endpoint";
    leaf headend {
      tailf:info "SR policy Headend Device ID";
      description "SR policy Headend Device ID";
      type string;
    }
    leaf color {
      tailf:info "<1-4294967295> - SR policy color";
      description "<1-4294967295> - SR policy color";
      type uint32 {
        range "1..4294967295";
      }
    }
    leaf endpoint {

```

```

        tailf:info "SR policy Endpoint IP";
        description "SR policy Endopint IP";
        type inet:ip-address;
    }
}
}
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service" {
    uses ct-hook:template-hook;

    choice evi-id-choice {
        when "not(l2vpn-ntw:vpn-type = 'vpn-common:vpws')";
        default "auto-evi-id";
        leaf auto-evi-id {
            type empty;
            tailf:info "Auto allocate Local EVI Id";
            description "Auto allocate Local EVI Id";
        }
        leaf evi-id {
            tailf:info "<1-65534> - EVI Id";
            description "<1-65534> - EVI Id";
            type uint16 {
                tailf:info "1-65534;;Ethernet VPN ID to set";
                range "1..65534";
            }
            mandatory true;
        }
    }

    choice evi-source-choice {
        when "l2vpn-ntw:vpn-type = 'vpn-common:vpws-evpn'";
        default "auto-evi-source";
        leaf auto-evi-source {
            type empty;
            tailf:info "Auto allocate Local ACI Id";
            description "Auto allocate Local ACI Id";
        }
        leaf evi-source {
            tailf:info "<1-16777215> - Local ACI Id";
            description "<1-16777215> - Local ACI Id";
            type uint32 {
                tailf:info "1-16777215;;Source ac-id (hex or decimal format)";
                range "1..16777215";
            }
            mandatory true;
        }
    }

    choice evi-target-choice {
        when "l2vpn-ntw:vpn-type = 'vpn-common:vpws-evpn'";
        default "auto-evi-target";
        leaf auto-evi-target {
            type empty;
            tailf:info "Auto allocate Remote ACI Id";
            description "Auto allocate Remote ACI Id";
        }
        leaf evi-target {
            tailf:info "<1-16777215> - Remote ACI Id";
            description "<1-16777215> - Remote ACI Id";
            type uint32 {

```

```

        tailf:info "1-16777215;;Remote ac-id (hex or decimal format)";
        range "1..16777215";
    }
    mandatory true;
}
}

tailf:action self-test {
    tailf:actionpoint ietf-l2vpn-nm-self-test-actionpoint;
    input {
    }
    output {
        leaf status {
            type string;
        }
        leaf message {
            type string;
        }
    }
}

leaf control-word {
    when "../l2vpn-ntw:vpn-type = 'vpn-common:vpws'";
    tailf:info "Control Word";
    description "Control Word";
    type enumeration {
        enum yes;
        enum no;
    }
    default no;
}

container vpn-target {
    presence true;
    when "../l2vpn-ntw:vpn-service-topology = 'vpn-common:any-to-any' or
../l2vpn-ntw:vpn-service-topology = 'vpn-common:hub-spoke'";
    tailf:info "VPN Target";
    description "VPN Target";

    choice hub-rt-choice {
        leaf auto-hub-rt {
            tailf:display-when "/l2vpn-ntw:l2vpn-ntw/id-pools/rt-pool-name";
            type empty;
            tailf:info "Auto allocate hub rt";
            description "Auto allocate hub rt";
        }
        leaf hub-rt-value {
            tailf:info "Hub Route-Target Value";
            description "Hub Route-Target Value";
            type rt-types:route-target;
        }
    }
    mandatory true;
}

choice spoke-rt-choice {
    when "../l2vpn-ntw:vpn-service-topology = 'vpn-common:hub-spoke'";
    leaf auto-spoke-rt {
        tailf:display-when "/l2vpn-ntw:l2vpn-ntw/id-pools/rt-pool-name";
        type empty;
        tailf:info "Auto allocate spoke rt";
        description "Auto allocate spoke rt";
    }
}
}

```

```

    leaf spoke-rt-value {
      tailf:info "Spoke Route-Target Value";
      description "Spoke Route-Target Value";
      type rt-types:route-target;
    }
    mandatory true;
  }
}

uses tsdn-core-fp-common:service-assurance-grouping {
  when "/tsdn-core-fp-common:enable-service-assurance = 'true'";
}

leaf bridge-group {
  when "../l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn'";
  tailf:info "Bridge-group is mandatory if vpn-type = mpls-evpn";
  description "Bridge-group is mandatory if vpn-type = mpls-evpn";
  type string {
    pattern "[a-zA-Z0-9\\-\\_]+";
    length "1..17";
  }
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node" {
  uses ct-hook:template-hook;
  tailf:action error-recovery {
    tailf:actionpoint ietf-l2vpn-nm-vpn-node-error-recovery;
    tailf:info "Action to recover this device during create/update from a
transient failures
device
description "Action to recover this device during create/update from a
transient
reachable again.
device
input {
  leaf sync-direction {
    tailf:info "sync device with sync-from or sync-to";
    description "sync device with sync-from or sync-to";
    type enumeration {
      enum sync-from;
      enum sync-to;
    }
    mandatory true;
  }
}
output {
  leaf success {
    type boolean;
    mandatory true;
  }
  leaf detail {
    type string;
  }
}

```

```

    }
    tailf:confirm-text "#####\n" +
        "#          Warning          #\n" +
        "#####\n" +
        "You are about to recover a T-SDN service.\n" +
        "This will issue a sync on the device.\n" +
        "Are you sure you want to proceed?" {
        tailf:confirm-default false;
    }
}

container multi-home {
    when "not(..../l2vpn-ntw:vpn-type = 'vpn-common:vpws')";
    presence "true";

    leaf spoke-multi-homed {
        when "..../l2vpn-ntw:role = 'vpn-common:spoke-role'";
        tailf:info "Enable multi-homing for spoke vpn-node";
        description "Enable multi-homing for spoke vpn-node";
        type empty;
    }

    leaf ethernet-segment-identifier {
        tailf:info "<0-ff> 9-octet ESI value. 00.00.00.00.00.00.00.00.00 is not
allowed";
        description "<0-ff> 9-octet ESI value. 00.00.00.00.00.00.00.00.00 is not
allowed";
        type string {
            tailf:info "0-ff;;9-octet ESI value. 00.00.00.00.00.00.00.00.00 is not
allowed";
            pattern '[0-9a-f]{2}(\.[0-9a-f]{2}){8}';
            pattern '00\.\00\.\00\.\00\.\00\.\00\.\00\.\00\.\00\.\00\.\00\.\00\.\00' {
                modifier invert-match;
            }
        }
        mandatory true;
    }
}

container te-service-mapping {
    container srv6 {
        presence "true";
        tailf:info "SRv6 Options";
        description "SRv6 Options";
        when "..../l2vpn-ntw:vpn-type = 'vpn-common:vpws-evpn'";
        tailf:display-when "not(contains(/ncs:devices/l2vpn-ntw:lsa-
utils:dispatch-map/l2vpn-ntw:device[lsa-utils:name=current()/..../l2vpn-ntw:vpn-
node-id]/lsa-utils:ned-id,'cisco-ios-cli-'))";

        leaf locator {
            tailf:info "SRv6 locator name (1-58 chars) at individual evi service
level,
                leave it empty if evi or global level locator is used.
Special characters are not allowed.";
            description "SRv6 locator name (1-58 chars) at individual evi service
level,
                leave it empty if evi or global level locator is used.
Special characters are not allowed.";
            type string {
                pattern '[0-9a-zA-Z_]*';
                length "1..58";
            }
        }
    }
}

```



```

    }
  }
}
container te-mapping {
  choice te {
    container sr-policy {
      when "not(..../..../..../l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn')";
      must "not(..../..../..../l2vpn-ntw:vpn-type = 'vpn-common:vpws' and
policy-type = 'cs-policy')" {
        error-message "policy-type cs-policy is only available for vpn-type
vpws-evpn";
      }
      presence true;
      leaf policy-type {
        type enumeration {
          enum policy;
          enum cs-policy;
        }
        default policy;
      }

      leaf policy {
        tailf:info "Name of SR TE policy";
        description "Name of SR TE policy";
        type string {
          length "1..max";
        }
        mandatory true;
      }
      leaf fallback {
        tailf:info "Fallback option for preferred path";
        description "Fallback option for preferred path";
        type enumeration {
          enum disable {
            tailf:info "Disable fallback for preferred path";
          }
        }
      }
      tailf:info "SR Policy for TE mapping.
L2VPN association with SRv6-TE (SR Policy and ODN) is not
supported for this release";
      description "SR Policy for TE mapping.
L2VPN association with SRv6-TE (SR Policy and ODN) is not
supported for this release";
    } //case sr-policy

    container te-tunnel-list {
      when "..../..../..../l2vpn-ntw:vpn-type = 'vpn-common:vpws'";
      presence true;
      choice tunnel-te-id-source {
        mandatory true;
        leaf te-tunnel-id {
          tailf:info "Specify tunnel number for preferred path <0-65535>";
          description "Specify tunnel number for preferred path <0-65535>";
          type uint16;
        }
        leaf ietf-te-service {
          tailf:info "Specify IETF-TE service name from which tunnel-te id
will be extracted";
          description "Specify IETF-TE service name from which tunnel-te id
will be extracted";
        }
      }
    }
  }
}

```

```

        type string {
            length "1..max";
        }
    }
}
leaf fallback {
    tailf:info "Fallback option for preferred path";
    description "Fallback option for preferred path";
    type enumeration {
        enum disable {
            tailf:info "Disable fallback for preferred path";
        }
    }
}
}
} //case te-tunnel-list

container odn {
    when "not(..../..../..../l2vpn-ntw:vpn-type = 'vpn-common:vpws')";
    presence true;
    leaf-list route-policy {
        tailf:info "Route policy definition";
        description "Route policy definition";
        type leafref {
            path "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-profiles/l2vpn-ntw:valid-
provider-identifiers/l2vpn-ntw:routing-profile-identifier/l2vpn-ntw:id";
        }
        min-elements 1;
    }
    container attach-point {
        choice parent-rr-route-policy-choice {
            leaf parent-rr-route-policy {
                type string {
                    length "1..max";
                    pattern 'PASS ALL' {
                        modifier invert-match;
                        error-message "PASS_ALL is not allowed as parent route policy
name";
                    }
                }
            }
        }
        tailf:info "Parent RR Route Policy to apply local route
policies";
        description "Parent RR Route Policy to apply local route
policies";
    }
    }
    tailf:info "Name of attach point where policy is attached";
    description "Name of attach point where policy is attached";
}
}
must "not(route-policy) or attach-point/parent-rr-route-policy or
..../..../..../l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn'" {
    error-message "If route-policy is defined, parent-rr-route-policy-
choice " +
        "must be selected";
}
}
must "attach-point/parent-rr-route-policy or count(route-policy) = 1 "
{
    error-message "only 1 route-policy can be defined when no parent-rr-
route-policy attached";
}
}
tailf:info "ODN SR Policy.
L2VPN association with SRv6-TE (SR Policy and ODN) is not

```

```

supported for this release";
    description "ODN SR Policy.
        L2VPN association with SRv6-TE (SR Policy and ODN) is not
supported for this release";
    }//case odn
}
    tailf:info "TE Service Mapping";
    description "TE Service Mapping";
}
    tailf:info "Mapping of this service to an optional Traffic Engineered path";
    description "Mapping of this service to an optional Traffic Engineered path";
}

leaf control-word-disable {
    when "../.../l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn'";
    tailf:info "Disabling control-word";
    description "Disabling control-word";
    type boolean;
    default "true";
}
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-
ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-accesses/l2vpn-ntw:vpn-
network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-1731" {
    leaf id-type {
        tailf:info "SMAN ID Type";
        description "SMAN ID Type";
        default icc-based;
        type enumeration {
            enum icc-based;
            enum number;
        }
    }
    leaf sman-id {
        tailf:hidden full;
        type string;
    }
    container sman-id-allocation-data {
        config false;
        tailf:cdb-oper {
            tailf:persistent true;
        }
        leaf icc-based-id {
            tailf:info "icc-based SMAN ID allocated by resource-manager id-
allocator";
            description "icc-based SMAN ID allocated by resource-manager id-
allocator";
            type string;
        }
        leaf number-id {
            tailf:info "number SMAN ID allocated by resource-manager id-allocator";
            description "number SMAN ID allocated by resource-manager id-allocator";
            type string;
        }
    }
    tailf:info "SMAN ID resource allocations";
    description "SMAN ID resource allocations";
}

list y-1731-profile {
    description "L2NM Y-1731 profile.";
}

```

```

tailf:info "L2NM Y-1731 profile.";

uses y-1731-profile-schedule;

key name;
leaf name {
  type leafref {
    path "/l2vpn-ntw:l2vpn-ntw/y-1731-profile/name";
  }
  description
    "References L2NM y-1731-profile definition";
  tailf:info
    "References L2NM y-1731-profile definition";
}
// list statistic {
//   description
//     "Statistic";
//   tailf:info
//     "Statistic";
//   config false;
//   tailf:cdb-oper {
//     tailf:persistent true;
//   }
//   key type;
//   leaf type {
//     type string;
//   }
//   leaf statistic-id {
//     tailf:info "id allocated by resource-manager id-allocator";
//     description "id allocated by resource-manager id-allocator";
//     type uint32 {
//       range "1..2147483647";
//     }
//   }
// }
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-
ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-accesses/l2vpn-ntw:vpn-
network-access/l2vpn-ntw:connection/l2vpn-ntw:lag-interface/l2vpn-ntw:split-
horizon" {
  leaf enable {
    tailf:info "Enable split horizon";
    description "Enable split horizon";
    type boolean;
    default "true";
  }
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations" {
  leaf mode {
    type enumeration {
      enum symmetric {
        tailf:info "All rewrites must be symmetric";
      }
    }
  }
}

```

```

}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-
ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-accesses/l2vpn-ntw:vpn-
network-access/l2vpn-ntw:connection/l2vpn-ntw:encapsulation" {
  container untagged {
    description "Container for untagged Interface";
    tailf:info "Container for untagged Interface";

    when "derived-from-or-self(..l2vpn-ntw:encap-type, 'vpn-common:untagged-
int')" {
      description "Only applies when the type is untagged interface";
    }

    container tag-operations {
      description
        "Sets the tag manipulation policy for this
        VPN network access. It defines a set of
        tag manipulations that allow for the
        insertion, removal, or rewriting
        of 802.1Q VLAN tags. These operations are
        indicated for the CE-PE direction.
        By default, tag operations are symmetric.
        As such, the reverse tag operation is
        assumed on the PE-CE direction.";
      tailf:info
        "Sets the tag manipulation policy for this
        VPN network access. It defines a set of
        tag manipulations that allow for the
        insertion, removal, or rewriting
        of 802.1Q VLAN tags. These operations are
        indicated for the CE-PE direction.
        By default, tag operations are symmetric.
        As such, the reverse tag operation is
        assumed on the PE-CE direction.";
      choice op-choice {
        description
          "Selects the tag rewriting policy for a
          VPN network access.";
        leaf push {
          type empty;
          description
            "Push one tag defined by the
            tag-1 leaf. It is
            assumed that, absent any policy, the
            default value of 0 will be used for
            PCP setting.";
          tailf:info
            "Push one tag defined by the
            tag-1 leaf. It is
            assumed that, absent any policy, the
            default value of 0 will be used for
            PCP setting.";
        }
        leaf translate {
          type enumeration {
            enum "1-to-1" {
              tailf:info "Replace the outermost tag with another tag";
            }
          }
          description

```

```

        "Translate the outer tag to one
        tag. PCP bits are preserved.";
    tailf:info
        "Translate the outer tag to one or tw
        tag. PCP bits are preserved.";
    }
}

leaf tag-1 {
    type dot1q-types:vlanid;
    description
        "A first tag to be used for push or
        translate operations. This tag will be
        used as the outermost tag as a result
        of the tag operation.";
    tailf:info
        "A first tag to be used for push or
        translate operations. This tag will be
        used as the outermost tag as a result
        of the tag operation.";
}

leaf mode {
    type enumeration {
        enum symmetric {
            tailf:info "All rewrites must be symmetric";
        }
    }
}
}
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-
ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-ntw:signaling-
option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-ntw:evpn-policies"
{
    uses vpn-common:vpn-route-targets {
        refine vpn-policies {
            must "(not(import-policy) and not(export-policy)) or not(..../te-
service-mapping/te-mapping/odn/attach-point/parent-rr-route-policy)" {
                error-message "Cannot import/export policy if parent-rr-route-policy is
defined";
            }
        }
    }
}

augment "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-service/l2vpn-
ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-ntw:signaling-
option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-
ntw:ldp/l2vpn-ntw:pw-peer-list" {
    leaf mpls-label {
        tailf:info "<16-1048575> - MPLS Label.
        Use a value within the global range defined for the chosen device.";
        description "<16-1048575> - MPLS Label
        Use a value within the global range defined for the chosen device.";
        type uint32 {
            range "16..1048575";
        }
    }
}
}

```

```

}

augment "/l2vpn-ntw:l2vpn-ntw" {
  container l2nm-actions {
    tailf:action cleanup {
      tailf:actionpoint ietf-l2vpn-nm-cleanup;
      tailf:info "Clean up L2NM service configuration and operational data";
      description "Clean up L2NM service configuration and operational data";
      input {
        leaf service {
          type string;
          mandatory true;
        }
        leaf no-networking {
          tailf:info "Do not send any data to the devices.";
          description "Do not send any data to the devices.";
          type boolean;
          mandatory true;
        }
        leaf vpn-node {
          tailf:info "Clean up l2nm service configuration for given node only";
          description "Clean up l2nm service configuration for given node only";
          type leafref {
            tailf:no-leafref-check;
            path "/ncs:devices/l2nm-ntw:l2nm-ntw:lsa-
utils:device/l2nm-ntw:lsa-
utils:name";
          }
        }
        leaf vpn-network-access-id {
          when "../vpn-node";
          tailf:info "Clean up this vpn-node associated with vpn-network-access-
id" +
          "under given service.";
          description "Clean up this vpn-node associated with vpn-network-access-
id" +
          "under given service.";
          type string;
        }
      }
      output {
        leaf success {
          type boolean;
          mandatory true;
        }
        leaf detail {
          type string;
        }
      }
      tailf:confirm-text "#####\n" +
        "# Warning #\n" +
        "#####\n" +
        "You are about to forcefully cleanup a T-SDN service.\n" +
        "This will affect the deploying service and leave network device(s) & NSO
out-of-sync (for no-networking=true).\n" +
        "Are you sure you want to proceed?" {
          tailf:confirm-default false;
        }
      }
    }

    tailf:action internal-plan-change-handler {
      tailf:hidden full;
    }
  }
}

```

```

tailf:actionpoint ietf-l2vpn-nm-internal-plan-change-handler;
input {
  uses kicker:action-input-params;
}
output {
}
}

tailf:action update-internal-cfp-configurations {
  tailf:hidden tsdn;
  tailf:actionpoint update-l2nm-internal-cfp-configurations;
  tailf:info "Copy over changes in external cfp-configurations to internal
cfp-configurations";
  description "Copy over changes in external cfp-configurations to internal
cfp-configurations";
  input {
  }
  output {
  }
}

tailf:action error-recovery {
  tailf:actionpoint ietf-l2vpn-nm-fp-error-recovery;
  tailf:info "Action to recover a service during create/update/delete from a
transient failures
again.
device
like device connection issues once the device is reachable
For create/update failures, this action issues a sync on the
device
& redeploys the service with reconcile option.
For delete failures, this action issues a sync on the device &
redeploys zombie services.";
  description "Action to recover a service during create/update/delete from a
transient failures
again.
device
like device connection issues once the device is reachable
For create/update failures, this action issues a sync on the
device
& redeploys the service with reconcile option.
For delete failures, this action issues a sync on the device &
redeploys zombie services.";
  input {
    leaf service {
      tailf:info "Service to recover";
      description "Service to recover";
      type string;
      mandatory true;
    }
    leaf vpn-node {
      tailf:info "Recover this vpn-node under given service.";
      description "Recover this vpn-node under given service.";
      type string;
    }
    leaf sync-direction {
      tailf:info "sync device with sync-from or sync-to";
      description "sync device with sync-from or sync-to";
      type enumeration {
        enum sync-from;
        enum sync-to;
      }
      mandatory true;
    }
  }
}

```



```

    }
  }
  output {
    leaf success {
      type boolean;
      mandatory true;
    }
    leaf detail {
      type string;
    }
  }
  tailf:confirm-text "#####\n" +
    "#      Warning      #\n" +
    "#####\n" +
    "You are about to recover a T-SDN service.\n" +
    "This will issue a sync on the device.\n" +
    "Are you sure you want to proceed?" {
    tailf:confirm-default false;
  }
}
}
}

// Annotations
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:role" {
  when "../.../l2vpn-ntw:vpn-service-topology != 'vpn-common:any-to-any'";
}

tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-
ntw:evpn-policies" {
  when "../.../l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn'";
}

tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-service-topology" {
  when "../l2vpn-ntw:vpn-type = 'vpn-common:mpls-evpn'";
}

tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:bgp" {
  when "not(.../vpn-type = 'vpn-common:vpws')";
}

tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp" {
  when "../.../vpn-type = 'vpn-common:vpws'";
  tailf:info "Container for LDP or L2TP-signaled PWs choice.";
}

tailf:annotate "/l2vpn-ntw:l2vpn-ntw" {
  tailf:info "Container for the L2NM.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-profiles" {
  tailf:info "Container for VPN profiles.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services" {

```

```

    tailf:info "Container for L2VPN services.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service" {
    tailf:info "Container of a VPN service.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-id" {
    tailf:info "A VPN identifier that uniquely identifies a VPN.
    This identifier has a local meaning, e.g., within
    a service provider network.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-type" {
    tailf:info "Service type.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-service-topology" {
    tailf:info "Defining service topology, supported topologies are any-to-any,
hub-spoke and custom.
    Point-to-point VPN service topology is not supported.";
    description "Defining service topology, supported topologies are any-to-any,
hub-spoke and custom.
    Point-to-point VPN service topology is not supported.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes" {
    tailf:info "Set of VPN nodes that are involved in the L2NM.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node" {
    tailf:info "Container of the VPN nodes.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-node-id" {
    tailf:info "Sets the identifier of the VPN node.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:role" {
    tailf:info "Role of the VPN node in the VPN.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option" {
    tailf:info "Container for the L2VPN signaling.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-
ntw:evpn-policies" {
    tailf:info "Includes a set of EVPN policies such
    as those related to handling MAC
    addresses.";
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:bgp/l2vpn-ntw:bgp-type/l2vpn-ntw:evpn-bgp/l2vpn-
ntw:evpn-policies/l2vpn-ntw:mac-learning-mode" {
    tailf:info "Indicates through which plane MAC
    addresses are advertised.";
  }
  }
  tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-

```

```

service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp" {
    tailf:info "Choice of LDP or L2TP-sigaled PWs.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-
l2tp/l2vpn-ntw:ldp/l2vpn-ntw:pw-peer-list" {
    tailf:info "List of AC and PW bindings.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-
l2tp/l2vpn-ntw:ldp/l2vpn-ntw:pw-peer-list/l2vpn-ntw:peer-addr" {
    tailf:info "Indicates the peer's IP address.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-option/l2vpn-
ntw:signaling-option/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-l2tp/l2vpn-ntw:ldp-or-
l2tp/l2vpn-ntw:ldp/l2vpn-ntw:pw-peer-list/l2vpn-ntw:vc-id" {
    tailf:info "VC label used to identify a PW.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-accesses" {
    tailf:info "Main container for VPN network accesses.";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access" {
    tailf:info "List of VPN network accesses.";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:id" {
    tailf:info "Identifier of the network access";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:interface-id" {
    tailf:info "Refers to a physical or logical interface.";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection" {
    tailf:info "Container for the bearer and AC.";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-ntw:encapsulation"
{
    tailf:info "Container for Layer 2 encapsulation.";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:encap-type" {
    description "Tagged interface type";
    tailf:info "Tagged interface type";
}
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-

```

```

accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q" {
    tailf:info "Tagged interface.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:cvlan-id" {
    tailf:info "VLAN identifier.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations" {
    tailf:info "Sets the tag manipulation policy for this
VPN network access. It defines a set of
tag manipulations that allow for the
insertion, removal, or rewriting
of 802.1Q VLAN tags. These operations are
indicated for the CE-PE direction.
By default, tag operations are symmetric.
As such, the reverse tag operation is
assumed on the PE-CE direction.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations/l2vpn-ntw:op-
choice/l2vpn-ntw:pop/l2vpn-ntw:pop" {
    tailf:info "Pop the outer tag.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations/l2vpn-ntw:op-
choice/l2vpn-ntw:push/l2vpn-ntw:push" {
    description "Push one tag defined by the tag-1 leaf.";
    tailf:info "Push one tag defined by the tag-1 leaf.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations/l2vpn-ntw:op-
choice/l2vpn-ntw:translate/l2vpn-ntw:translate" {
    tailf:info "Translate the outer tag to one or two tags. PCP bits are
preserved.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-
ntw:encapsulation/l2vpn-ntw:dot1q/l2vpn-ntw:tag-operations/l2vpn-ntw:tag-1" {
    tailf:info
    "A first tag to be used for push or
    translate operations. This tag will be
    used as the outermost tag as a result
    of the tag operation.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-ntw:lag-interface"
{

```

```

tailf:info
  "Container of LAG interface attributes
  configuration.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-ntw:lag-
interface/l2vpn-ntw:split-horizon" {
  when "../../../../../../../role = 'vpn-common:spoke-role'";
  tailf:info
    "Configuration with split horizon enabled.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam" {
  tailf:info
    "Container for Ethernet service OAM.";
  tailf:display-when "not(contains(/ncs:devices/l2vpn-ntw:lsa-utils:lsa-utils:dispatch-
map/l2vpn-ntw:device[l2vpn-ntw:lsa-utils:name=current()/../../../../vpn-node-id]/l2vpn-ntw:ned-
id,'cisco-ios-cli-')) and not../../../../vpn-type = 'vpn-common:mpls-evpn')";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:md-
name" {
  tailf:info
    "Maintenance domain name.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:md-
level" {
  tailf:info
    "Maintenance domain level.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731" {
  tailf:info
    "List of configured Y-1731 instances.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731/l2vpn-ntw:maid" {
  tailf:info "<1-43> - Specify the Maintenance Association (MA)ID for y-1731
instance, maid must be same among all vpn-nodes";
  description "<1-43> - Specify the Maintenance Association (MA)ID for y-1731
instance, maid must be same among all vpn-nodes";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731/l2vpn-ntw:mep-id" {
  tailf:info "<1-8191> - Specify the Local Maintenance Entity Group End Point
(MEP) ID for the y-1731 instance. mep-id must be unique among all vpn-nodes.";
  description "<1-8191> - Specify the Local Maintenance Entity Group End Point
(MEP) ID for the y-1731 instance. mep-id must be unique among all vpn-nodes.";
}
tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-

```

```

service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:ethernet-service-oam/l2vpn-ntw:y-
1731/l2vpn-ntw:message-period" {
    tailf:info
        "Defines the interval between OAM messages.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service" {
    tailf:info
        "Container for service";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:qos" {
    tailf:info
        "QoS configuration.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:qos/l2vpn-
ntw:qos-profile" {
    tailf:info
        "QoS profile configuration.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:qos/l2vpn-
ntw:qos-profile/l2vpn-ntw:qos-profile" {
    tailf:info
        "QoS profile. Can be standard profile or customized profile.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:qos/l2vpn-
ntw:qos-profile/l2vpn-ntw:qos-profile/l2vpn-ntw:profile" {
    tailf:info
        "QoS profile to be used.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:qos/l2vpn-
ntw:qos-profile/l2vpn-ntw:qos-profile/l2vpn-ntw:direction" {
    tailf:info
        "The direction to which the QoS profile is applied.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:service/l2vpn-ntw:mtu" {
    tailf:info
        "Layer 2 MTU, it is also known as the maximum
        transmission unit or maximum frame size.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-profiles/l2vpn-ntw:valid-
provider-identifiers/l2vpn-ntw:routing-profile-identifier/l2vpn-ntw:id" {
    description "Valid VPN profile to be referenced in the VPN service. Routing
    Policy needs to be created and mapped here.";
    tailf:info "Valid VPN profile to be referenced in the VPN service. Routing
    Policy needs to be created and mapped here.";
}
    tailf:annotate "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-

```

```

service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-network-
accesses/l2vpn-ntw:vpn-network-access/l2vpn-ntw:connection/l2vpn-ntw:l2-
termination-point" {
    description "Specify the Name of Layer 2 interface / Attachment Circuit";
    tailf:info "Specify the Name of Layer 2 interface / Attachment Circuit";
}
}

```

IETF-L2VPN-NM Route Policy Model

```

actions.yang cisco-l2vpn-routing-policy.yang
module: cisco-l2vpn-routing-policy
  +--rw l2vpn-routing-policy
    +--rw defined-sets
      | +--rw rd-sets
      | | +--rw rd-set* [name]
      | |   +--rw name      string
      | |   +--rw rd*      asn-ip-type
      | +--rw evpn-route-type-sets
      | | +--rw evpn-route-type-set* [name]
      | |   +--rw name      string
      | |   +--rw evpn-route-type* uint8
      | +--rw tag-sets
      |   +--rw tag-set* [name]
      |     +--rw name      string
      |     +--rw tag-value* uint32
    +--rw policy-definitions
      +--rw policy-definition* [name]
        | +--rw name      string
        | +--rw statements
        |   +--rw statement* [name]
        |     +--rw name      string
        |     +--rw conditions
        |     | +--rw match-rd-set!
        |     | | +--rw rd-set  -> ../../../../../../../defined-sets/rd-
sets/rd-set/name
        |     | | +--rw match-evpn-route-type-set!
        |     | | | +--rw evpn-route-type-set  ->
../../../../../defined-sets/evpn-route-type-sets/evpn-route-type-set/name
        |     | | +--rw match-manual?          string
        |     +--rw actions
        |       +--rw bgp-actions!
        |         +--rw set-ext-community!
        |           +--rw (method)?
        |             +--:(reference)
        |             +--rw ext-community-set-ref?  ->
../../../../../defined-sets/tag-sets/tag-set/name
        +--x internal-defined-sets-change-handler
          +---w input
            +---w kicker-id?  string
            +---w path?       tailf:node-instance-identifier
            +---w tid?         uint32

```

IETF-L3VPN-NM Yang Models

This section includes:

- IETF-L3VPN-NM Service Model

- IETF-L3VPN-NM Service Plan Model
- IETF-L3VPN-NM Deviation Model
- IETF-L3VPN-NM Augmentations Model
- IETF-L3VPN-NM MVPN Augmentations Model
- IETF-L3VPN-NM Routing Policy Model

IETF-L3VPN-NM Service Model

```

module: ietf-l3vpn-ntw
  +--rw l3vpn-ntw
    +--rw vpn-profiles
      | +--rw valid-provider-identifiers
      | | +--rw qos-profile-identifier* [id]
      | | | +--rw id string
      | | +--rw routing-profile-identifier* [id]
      | | | +--rw id -> /cisco-l3vpn-routing-policy:l3vpn-routing-
policy/policy-definitions/policy-definition/name
    +--rw vpn-services
      | +--rw vpn-service* [vpn-id]
      | | +--rw vpn-id string
      | | +--rw vpn-service-topology? identityref
      | | +--rw vpn-instance-profiles
      | | | +--rw vpn-instance-profile* [profile-id]
      | | | | +--rw profile-id string
      | | | | +--rw role? identityref
      | | | | +--rw local-as? union {vpn-common:rtg-bgp}?
      | | | | +--rw (rd-choice)?
      | | | | | +--:(directly-assigned)
      | | | | | | +--rw rd? string
      | | | | +--rw address-family* [address-family]
      | | | | | +--rw address-family identityref
      | | | | +--rw vpn-targets
      | | | | | +--rw vpn-target* [id]
      | | | | | | +--rw id uint8
      | | | | | | +--rw route-targets* [route-target]
      | | | | | | | +--rw route-target string
      | | | | | | +--rw route-target-type rt-types:route-target-type
      | | | | +--rw vpn-policies
      | | | | | +--rw import-policy? -> /l3vpn-ntw/vpn-profiles/valid-
provider-identifiers/routing-profile-identifier/id
      | | | | | +--rw export-policy? -> /l3vpn-ntw/vpn-profiles/valid-
provider-identifiers/routing-profile-identifier/id
      | | | | +--rw cisco-l3nm:srv6!
      | | | | | +--rw cisco-l3nm:address-family* [name]
      | | | | | | +--rw cisco-l3nm:name? identityref
      | | | | | | +--rw cisco-l3nm:locator-name? string
      | | +--rw underlay-transport
      | | | +--ro cisco-l3nm:discovered-underlay-transport
      | | | | +--ro cisco-l3nm:te-tunnel-ref* [tunnel-id source]
      | | | | | +--ro cisco-l3nm:tunnel-id? string
      | | | | | +--ro cisco-l3nm:source? inet:ip-address
      | | | | | +--ro cisco-l3nm:destination? inet:ip-address
      | | | | +--ro cisco-l3nm:sr-policy-ref* [headend color endpoint]
      | | | | | +--ro cisco-l3nm:headend? string
      | | | | | +--ro cisco-l3nm:color? uint32
      | | | | | +--ro cisco-l3nm:endpoint? inet:ip-address

```



```

| | +--rw vpn-nodes
| | | +--rw vpn-node* [vpn-node-id]
| | | +--rw vpn-node-id
utils:lsa/dispatch-map/device/name -> /ncs:devices/lsa-
| | | +--rw local-as? union {vpn-common:rtg-bgp}?
| | | +--rw active-vpn-instance-profiles
| | | | +--rw vpn-instance-profile* [profile-id]
| | | | +--rw profile-id -> ../../../../vpn-instance-
profiles/vpn-instance-profile/profile-id
| | | | +--rw local-as? union {vpn-common:rtg-bgp}?
| | | | +--rw (rd-choice)?
| | | | | +--:(directly-assigned)
| | | | | +--rw rd? string
| | | | +--rw address-family* [address-family]
| | | | +--rw address-family identityref
| | | | +--rw vpn-targets
| | | | | +--rw vpn-target* [id]
| | | | | | +--rw id uint8
| | | | | | +--rw route-targets* [route-target]
| | | | | | | +--rw route-target string
| | | | | | +--rw route-target-type rt-types:route-target-
type
| | | | +--rw vpn-policies
| | | | +--rw import-policy? -> /l3vpn-ntw/vpn-
profiles/valid-provider-identifiers/routing-profile-identifier/id
| | | | +--rw export-policy? -> /l3vpn-ntw/vpn-
profiles/valid-provider-identifiers/routing-profile-identifier/id
| | | | +--rw cisco-l3nm:srv6!
| | | | | +--rw cisco-l3nm:address-family* [name]
| | | | | +--rw cisco-l3nm:name? identityref
| | | | | +--rw cisco-l3nm:locator-name? string
| | | | +--rw vpn-network-accesses
| | | | | +--rw vpn-network-access* [id]
| | | | | +--rw id vpn-common:vpn-id
| | | | | +--rw interface-id string
| | | | | +--rw connection
| | | | | | +--rw encapsulation
| | | | | | | +--rw type? identityref
| | | | | | | +--rw dot1q
| | | | | | | | +--rw cvlan-id uint16
| | | | | | | | +--rw cisco-l3nm:BDI? uint16
| | | | | +--rw ip-connection
| | | | | | +--rw ipv4 {vpn-common:ipv4}?
| | | | | | | +--rw local-address? inet:ipv4-address
| | | | | | | | +--rw prefix-length? uint8
| | | | | | +--rw ipv6 {vpn-common:ipv6}?
| | | | | | | +--rw local-address? inet:ipv6-address
| | | | | | | +--rw prefix-length? uint8
| | | | | +--rw routing-protocols
| | | | | | +--rw routing-protocol* [id]
| | | | | | | +--rw id string
| | | | | | | +--rw type identityref
| | | | | | | +--rw bgp
| | | | | | | +--rw peer-as inet:as-
number
| | | | | | +--rw address-family? enumeration
| | | | | | +--rw neighbor* inet:ip-
address
| | | | | | +--rw multihop? uint8
| | | | | | +--rw redistribute-connected* [address-family]
| | | | | | | +--rw address-family identityref

```



```

| | | | +--rw cisco-mvpn:switch-threshold? uint32
| | | | +--rw cisco-mvpn:per-item-tunnel-limit uint32
| | | | +--rw cisco-mvpn:switch-wildcard-mode enumeration
| | | | +--rw cisco-mvpn:group-acl-ipv4? string
| | | | +--rw cisco-mvpn:route-policy? -> /l3vpn-ntw/vpn-
profiles/valid-provider-identifiers/routing-profile-identifier/id
| | | | +--rw cisco-mvpn:immediate-switch? boolean
| | | | +--rw cisco-l3nm:static-sr-mpls-p2mp* [policy-name]
| | | | | +--rw cisco-l3nm:policy-name? string
| | | | | +--rw cisco-l3nm:sr-p2mp-policy!
| | | | | +--rw cisco-l3nm:group-address* [address]
| | | | | +--rw cisco-l3nm:address? inet:ipv4-address-no-
zone
| | | | | +--rw cisco-l3nm:source-address* [address]
| | | | | | +--rw cisco-l3nm:address? inet:ipv4-address-no-zone
| | | | | | +--rw cisco-l3nm:inc-mask!
| | | | | | +--rw cisco-l3nm:address? inet:ipv4-address-no-
zone
| | | | | | +--rw cisco-l3nm:count uint32
| | | | | | +--rw cisco-l3nm:inc-mask!
| | | | | | +--rw cisco-l3nm:address? inet:ipv4-address-no-zone
| | | | | | +--rw cisco-l3nm:count uint32
| | | | | +--rw cisco-l3nm:custom-template* [name]
| | | | | +--rw cisco-l3nm:name? -> /ct-info:custom-template-
info/template-name
| | | | | +--rw cisco-l3nm:variable* [name]
| | | | | +--rw cisco-l3nm:name? -> deref(..../name)/../ct-
info:variables
| | | | | +--rw cisco-l3nm:value string
| | | | | +--rw cisco-l3nm:iteration* [number]
| | | | | +--rw cisco-l3nm:number? uint16
| | | | | +--rw cisco-l3nm:variable* [name]
| | | | | +--rw cisco-l3nm:name? -> deref(..../name)/../ct-
info:variables
| | | | | +--rw cisco-l3nm:value string
| | | | | +--rw cisco-l3nm:service-assurance!
| | | | | +--rw cisco-l3nm:monitoring-state? aa-monitoring-state
| | | | | +--rw cisco-l3nm:preservation? historical-data-options
| | | | | +--rw cisco-l3nm:profile-name string
| | | | | +--rw cisco-l3nm:rule-name string
| | | | | +--rw cisco-l3nm:probes!
| | | | | +--rw cisco-l3nm:endpoint* [id]
| | | | | +--rw cisco-l3nm:id? string
| | | | | +--rw cisco-l3nm:vpn-node -> ../../../../vpn-nodes/vpn-
node/vpn-node-id
| | | | | +--rw cisco-l3nm:vpn-network-access -> ../../../../vpn-
nodes/l3nmvpn-node[l3nmvpn-node-id=current()/../vpn-node]/vpn-network-accesses/vpn-
network-access/id
| | | | | +--rw cisco-l3nm:interface-id -> ../../../../vpn-
nodes/l3nmvpn-node[l3nmvpn-node-id=current()/../vpn-node]/vpn-network-
accesses/l3nmvpn-network-access[l3nmid=current()/../vpn-network-access]/interface-
id
| | | | | +--rw cisco-l3nm:agent-ipv4? inet:ipv4-address
| | | | | +--rw cisco-l3nm:agent-ipv6? inet:ipv6-address
| | | | | +--rw cisco-l3nm:agent-vlan uint16
| | | | | +--rw (connection-type)?
| | | | | | +--:(connection-point-to-point)
| | | | | | | +--rw cisco-l3nm:point-to-point
| | | | | | | | +--rw cisco-l3nm:connection* [source destination]
| | | | | | | | +--rw cisco-l3nm:source -> ../../../../endpoint/id
| | | | | | | | +--rw cisco-l3nm:destination -> ../../../../endpoint/id

```

```

| | | +---:(connection-hub-and-spoke)
| | | | +---rw cisco-l3nm:hub-and-spoke* [id]
| | | | | +---rw cisco-l3nm:id? uint16
| | | | | +---rw cisco-l3nm:hub* -> ../../endpoint/id
| | | | | +---rw cisco-l3nm:spoke* -> ../../endpoint/id
| | | +---:(connection-mesh)
| | | | +---rw cisco-l3nm:mesh? empty
+---rw cisco-l3nm:vpn-target
| | +---rw (hub-rt-choice)?
| | | +---:(auto-hub-rt)
| | | | +---rw cisco-l3nm:auto-hub-rt? empty
| | | +---:(hub-rt-value)
| | | | +---rw cisco-l3nm:hub-rt-value? rt-types:route-target
+---rw (spoke-rt-choice)?
| | +---:(auto-spoke-rt)
| | | +---rw cisco-l3nm:auto-spoke-rt? empty
| | | +---:(spoke-rt-value)
| | | | +---rw cisco-l3nm:spoke-rt-value? rt-types:route-target
+---x cisco-l3nm:self-test
| | +---ro cisco-l3nm:output
| | | +---ro cisco-l3nm:status? string
| | | +---ro cisco-l3nm:message? string
+---x cisco-l3nm:error-recovery
| | +---w cisco-l3nm:input
| | | +---w cisco-l3nm:sync-direction enumeration
| | +---ro cisco-l3nm:output
| | | +---ro cisco-l3nm:success boolean
| | | +---ro cisco-l3nm:detail? string
+---x cisco-l3nm:vpn-service-change-handler
| | +---w cisco-l3nm:input
| | | +---w cisco-l3nm:kicker-id? string
| | | +---w cisco-l3nm:path? tailf:node-instance-identifier
| | | +---w cisco-l3nm:tid? uint32
+---x cisco-l3nm:touch-vpn-nodes
| | +---ro cisco-l3nm:output
| | | +---ro cisco-l3nm:detail? string
+---x cisco-l3nm:re-deploy
| | +---w cisco-l3nm:input
| | | +---w cisco-l3nm:no-networking? boolean
| | | +---w cisco-l3nm:reconcile!
| | | | +---w (c-non-service-config)?
| | | | | +---:(keep-non-service-config)
| | | | | | +---w cisco-l3nm:keep-non-service-config? empty
| | | | | +---:(discard-non-service-config)
| | | | | | +---w cisco-l3nm:discard-non-service-config? empty
| | | +---ro cisco-l3nm:output
| | | +---ro cisco-l3nm:success boolean
| | | +---ro cisco-l3nm:detail? string
+---x cisco-l3nm:reactive-re-deploy
| | +---ro cisco-l3nm:output
| | | +---ro cisco-l3nm:success boolean
| | | +---ro cisco-l3nm:detail? string

```

IETF-L3VPN-NM Service Plan Model

L3NM service can now configure multiple vpn-network-access per vpn-node. Each vpn-network-access has its own component in the plan in the format <VPN_NODE NE_ID>_<VPN_NETWORK_ACCESS ID>.

```

| +---ro cisco-l3nm:vpn-service-plan* [vpn-id]
|   | +---ro cisco-l3nm:vpn-id?                               string
|   | +---ro cisco-l3nm:plan
|   | | +---ro cisco-l3nm:component* [type name]
|   | | | +---ro cisco-l3nm:name?                             string
|   | | | +---ro cisco-l3nm:type                             plan-component-type-t
|   | | | +---ro cisco-l3nm:state* [name]
|   | | | | +---ro cisco-l3nm:name?                           plan-state-name-t
|   | | | | +---ro cisco-l3nm:status?                         plan-state-status-t
|   | | | | +---ro cisco-l3nm:when?                           yang:date-and-
|   | +---ro cisco-l3nm:commit-queue!
|   | | +---ro cisco-l3nm:queue-item* [id]
|   | | | +---ro cisco-l3nm:id?                               uint64
|   | +---ro cisco-l3nm:failed?                               empty
|   | +---ro cisco-l3nm:error-info!
|   | | +---ro cisco-l3nm:message?                           string
|   | | +---ro cisco-l3nm:log-entry?                         instance-identifier
|   | +---ro cisco-l3nm:deleting?                             empty
|   | +---ro cisco-l3nm:service-location?                    instance-identifier
|   | +---ro cisco-l3nm:rt-allocation-data
|   | | +---ro cisco-l3nm:hub-rt?                             string
|   | | +---ro cisco-l3nm:spoke-rt?                          string
|   | +---ro cisco-l3nm:failed-vpn-nodes?                    string
| +---ro cisco-l3nm:globally-updated-vpn-node* [vpn-node-id]
|   | +---ro cisco-l3nm:vpn-node-id?                          string
| +---x cisco-l3nm:purge-globally-updated-vpn-node
|   | +---w cisco-l3nm:input
|   | | +---w (vpn-nodes-choice)
|   | | | +---: (all)
|   | | | | +---w cisco-l3nm:all?                             empty
|   | | | +---: (vpn-nodes)
|   | | | | +---w cisco-l3nm:vpn-nodes*                       string
|   | +---ro cisco-l3nm:output
|   | | +---ro cisco-l3nm:success                            boolean
|   | | +---ro cisco-l3nm:detail?                            string
| +---ro cisco-l3nm:vpn-nodes
|   | +---ro cisco-l3nm:vpn-node-plan* [vpn-node-id]
|   | | +---ro cisco-l3nm:vpn-node-id?                       string
|   | | +---ro cisco-l3nm:plan
|   | | | +---ro cisco-l3nm:component* [type name]
|   | | | | +---ro cisco-l3nm:name?                           string
|   | | | | +---ro cisco-l3nm:type                             plan-
component-type-t
|   | | | | +---ro cisco-l3nm:state* [name]
|   | | | | | +---ro cisco-l3nm:name?                         plan-state-name-t
|   | | | | | +---ro cisco-l3nm:status?                       plan-state-
status-t
|   | | | | | +---ro cisco-l3nm:when?                           yang:date-and-
time
|   | | +---ro cisco-l3nm:commit-queue!
|   | | | +---ro cisco-l3nm:queue-item* [id]
|   | | | | +---ro cisco-l3nm:id?                             uint64
|   | | +---ro cisco-l3nm:failed?                             empty
|   | | +---ro cisco-l3nm:error-info!

```

```

|         | | +---ro cisco-l3nm:message?      string
|         | | +---ro cisco-l3nm:log-entry?    instance-identifier
|         | | +---ro cisco-l3nm:deleting?   empty
|         | | +---ro cisco-l3nm:service-location? instance-identifier
|         | | +---ro cisco-l3nm:rt-allocation-data
|         | | | +---ro cisco-l3nm:hub-rt?    string
|         | | | +---ro cisco-l3nm:spoke-rt?  string
|         | | +---ro cisco-l3nm:status-code-detail* [type name]
|         | | +---ro cisco-l3nm:type?       ncs:plan-component-
type-t
|         | | +---ro cisco-l3nm:name?       string
|         | | +---ro cisco-l3nm:code?       string
|         | | +---ro cisco-l3nm:context* [context-name]
|         | | | +---ro cisco-l3nm:context-name? string
|         | | | +---ro cisco-l3nm:context-msg? string
|         | | +---ro cisco-l3nm:severity?   enumeration
|         | | +---ro cisco-l3nm:recommended-action? string
|         | | +---ro cisco-l3nm:impacted-device? string
|         | +---ro cisco-l3nm:internal
|         | +---ro cisco-l3nm:endpoint-names* string

```

IETF-L3VPN-NM Deviation Model

```

module ietf-l3vpn-ntw-deviations {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/ietf-l3vpn-ntw-deviations";
  prefix l3vpn-ntw-deviations;

  import ietf-l3vpn-ntw { prefix l3nm; }
  import ietf-mvpn { prefix mvpn; }
  import cisco-mvpn { prefix cisco-mvpn; }
  import cisco-l3vpn-routing-policy { prefix cisco-l3vpn-routing-policy; }
  import tailf-common { prefix tailf; }
  import tailf-ncs { prefix ncs; }
  import ietf-inet-types { prefix inet; }
  import tailf-kicker { prefix kicker; }
  import ietf-vpn-common { prefix vpn-common; }
  import ieee802-dot1q-types { prefix dot1q-types; }
  import cisco-tdsn-core-fp-common { prefix tsdn-core-fp-common; }
  import lsa-utils { prefix lsa-utils; }
  import resource-allocator { prefix ralloc; }
  import id-allocator { prefix idalloc; }

  ...

  // ===== routing-protocol =====
  deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:routing-
protocols/l3nm:routing-protocol" {
    deviate add {
      max-elements 1;
    }
  }
  deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:routing-
protocols/l3nm:routing-protocol/l3nm:type" {
    deviate add {
      mandatory true;
    }
  }

```

```

}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:routing-
protocols/l3nm:routing-protocol/l3nm:bgp/l3nm:address-family" {
    deviate replace {
        type enumeration {
            enum ipv4;
            enum ipv6;
            enum dual-stack;
        }
    }
}

// ===== routing-profile-identifier =====
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-profiles/l3nm:valid-provider-
identifiers/l3nm:routing-profile-identifier/l3nm:id" {
    deviate replace {
        type leafref {
            path "/cisco-l3vpn-routing-policy:l3vpn-routing-policy/cisco-l3vpn-routing-
policy:policy-definitions/cisco-l3vpn-routing-policy:policy-definition/cisco-l3vpn-
routing-policy:name";
        }
    }
}

// ===== vpn-service =====
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-id" {
    deviate replace {
        // vpn-id is used as a key for internal service. This means we must match the
pattern
        // restrictions imposed by internal service.
        type string {
            pattern '[a-zA-Z0-9\-\_]+';
        }
    }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-service-
topology" {
    deviate replace {
        default "vpn-common:custom";
    }
    deviate add {
        must ". = 'vpn-common:any-to-any' or . = 'vpn-common:hub-spoke' or . = 'vpn-
common:custom'" {
            error-message "vpn-service-topology must be one of the following: vpn-
common:any-to-any, "
                + " vpn-common:hub-spoke, vpn-common:custom";
        }
    }
}

// ===== ip-connection =====
// ipv4 prefix-length mandatory when provider-address is defined
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:ip-
connection/l3nm:ipv4" {
    deviate add {
        must "./local-address and ./prefix-length or not(./local-address and
./prefix-length)" {
            error-message "local-address and prefix-length are both required for the IP
address.";
        }
    }
}

```

```

    }
  }
}
// ipv6 prefix-length mandatory when provider-address is defined
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:ip-
connection/l3nm:ipv6" {
  deviate add {
    must "./local-address and ./prefix-length or not(./local-address and
./prefix-length)" {
      error-message "local-address and prefix-length are both required for the IP
address.";
    }
  }
}
// At least one primary-address must be defined
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:ip-
connection" {
  deviate add {
    must "ipv4/local-address or ipv6/local-address" {
      error-message "At least one ip-connection local-address must be defined";
    }
  }
}

// ===== connection =====
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:connection/l3nm:encapsulation/l3nm:type" {
  deviate add {
    must ". = 'vpn-common:dot1q'" {
      error-message "Connection encapsulation type must be vpn-common:dot1q";
    }
  }
  deviate delete {
    default "vpn-common:priority-tagged";
  }
}

deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:connection/l3nm:encapsulation/l3nm:dot1q/l3nm:cvlan-id" {
  deviate add {
    mandatory true;
  }
}

// ===== vpn-instance-profiles =====
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-
policies/l3nm:import-policy" {
  deviate replace {
    type leafref {
      path "/l3nm:l3vpn-ntw/l3nm:vpn-profiles/l3nm:valid-provider-
identifiers/l3nm:routing-profile-identifier/l3nm:id";
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-

```



```

policies/l3nm:export-policy" {
  deviate replace {
    type leafref {
      path "/l3nm:l3vpn-ntw/l3nm:vpn-profiles/l3nm:valid-provider-
identifiers/l3nm:routing-profile-identifier/l3nm:id";
    }
  }
}
// L3VPN route-distinguisher supports RD and RT type 0,1,2. So we have to remove
other L3NM supported types
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile/l3nm:rd-choice/l3nm:directly-assigned/l3nm:rd" {
  deviate replace {
    type string {
      pattern
        '(0:(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0):(429496729[0-5]|'
        + '42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2}|429496[0-6][0-9]{3})|'
        + '42949[0-5][0-9]{4})|'
        + '4294[0-8][0-9]{5}|429[0-3][0-9]{6})|'
        + '42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0)|'
        + '(1:(((0-9)|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|'
        + '25[0-5])\\.){3}([0-9]|1[0-9][0-9]|'
        + '1[0-9]{2}|2[0-4][0-9]|25[0-5])):(6553[0-5]|'
        + '655[0-2][0-9]|'
        + '65[0-4][0-9]{2}|6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0)|'
        + '(2:(429496729[0-5]|42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2})|'
        + '429496[0-6][0-9]{3}|42949[0-5][0-9]{4})|'
        + '4294[0-8][0-9]{5}|'
        + '429[0-3][0-9]{6}|42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0):'
        + '(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0)';
      tailf:info "0:2-octet-asn:4-octet-number
                1:4-octet-ipv4addr:2-octet-number
                2:4-octet-asn:2-octet-number";
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-
target/l3nm:route-targets/l3nm:route-target" {
  deviate replace {
    type string {
      pattern
        '(0:(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0):(429496729[0-5]|'
        + '42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2}|429496[0-6][0-9]{3})|'
        + '42949[0-5][0-9]{4})|'
        + '4294[0-8][0-9]{5}|429[0-3][0-9]{6})|'
        + '42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0)|'
        + '(1:(((0-9)|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9]|'

```

```

+      '25[0-5])\.) {3} ([0-9] | [1-9] [0-9] | '
+      '1[0-9]{2} | 2[0-4] [0-9] | 25[0-5])) : (6553[0-5] | '
+      '655[0-2] [0-9] | '
+      '65[0-4] [0-9] {2} | 6[0-4] [0-9] {3} | '
+      '[1-5] [0-9] {4} | [1-9] [0-9] {0,3} | 0) | '
+ ' (2: (429496729[0-5] | 42949672 [0-8] [0-9] | '
+   '4294967 [01] [0-9] {2} | '
+   '429496 [0-6] [0-9] {3} | 42949 [0-5] [0-9] {4} | '
+   '4294 [0-8] [0-9] {5} | '
+   '429 [0-3] [0-9] {6} | 42 [0-8] [0-9] {7} | 4 [01] [0-9] {8} | '
+   '[1-3] [0-9] {9} | [1-9] [0-9] {0,8} | 0) : '
+   '(6553[0-5] | 655[0-2] [0-9] | 65[0-4] [0-9] {2} | '
+   '6[0-4] [0-9] {3} | '
+   '[1-5] [0-9] {4} | [1-9] [0-9] {0,3} | 0) )';
tailf:info "0:2-octet-asn:4-octet-number
          1:4-octet-ipv4addr:2-octet-number
          2:4-octet-asn:2-octet-number";
    }
  }
}

// ===== vpn-node =====
// Make vpn-node ne-id to point to dispatch-map instead of device tree
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-node-id" {
  deviate replace {
    type leafref {
      path "/ncs:devices/l3nm:utils:lsa/l3nm:utils:dispatch-map/l3nm:
utils:device/l3nm:utils:name";
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:profile-id" {
  deviate replace {
    type leafref {
      path "../../../../../../../vpn-instance-profiles/vpn-instance-profile/profile-
id";
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-policies/l3nm:import-policy"
{
  deviate replace {
    type leafref {
      path "/l3nm:l3vpn-ntw/l3nm:vpn-profiles/l3nm:valid-provider-
identifiers/l3nm:routing-profile-identifier/l3nm:id";
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-policies/l3nm:export-policy"
{
  deviate replace {
    type leafref {
      path "/l3nm:l3vpn-ntw/l3nm:vpn-profiles/l3nm:valid-provider-
identifiers/l3nm:routing-profile-identifier/l3nm:id";
    }
  }
}

```

```

    }
  }
}
// L3VPN route-distinguisher supports RD and RT type 0,1,2. So we have to remove
other L3NM supported types
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:rd-choice/l3nm:directly-assigned/l3nm:rd" {
  deviate replace {
    type string {
      pattern
        '(0:(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0):(429496729[0-5]|'
        + '42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2}|429496[0-6][0-9]{3})|'
        + '42949[0-5][0-9]{4})|'
        + '4294[0-8][0-9]{5}|429[0-3][0-9]{6})|'
        + '42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0))|'
        + '(1:((([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9])|'
        + '25[0-5])\\.){3}([0-9]|[1-9][0-9]|'
        + '1[0-9]{2}|2[0-4][0-9]|25[0-5])):(6553[0-5]|'
        + '655[0-2][0-9]|'
        + '65[0-4][0-9]{2}|6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0))|'
        + '(2:(429496729[0-5]|42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2}|'
        + '429496[0-6][0-9]{3}|42949[0-5][0-9]{4})|'
        + '4294[0-8][0-9]{5}|'
        + '429[0-3][0-9]{6}|42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0):'
        + '(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0))';
      tailf:info "0:2-octet-asn:4-octet-number
        1:4-octet-ipv4addr:2-octet-number
        2:4-octet-asn:2-octet-number";
    }
  }
}
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-target/l3nm:route-
targets/l3nm:route-target" {
  deviate replace {
    type string {
      pattern
        '(0:(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0):(429496729[0-5]|'
        + '42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2}|429496[0-6][0-9]{3})|'
        + '42949[0-5][0-9]{4})|'
        + '4294[0-8][0-9]{5}|429[0-3][0-9]{6})|'
        + '42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0))|'
        + '(1:((([0-9]|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9])|'
        + '25[0-5])\\.){3}([0-9]|[1-9][0-9]|'
        + '1[0-9]{2}|2[0-4][0-9]|25[0-5])):(6553[0-5]|'
        + '655[0-2][0-9]|'

```

```

+       '65[0-4][0-9]{2}|6[0-4][0-9]{3}|'
+       '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0))|'
+ ' (2:(429496729[0-5]|42949672[0-8][0-9]|'
+   '4294967[01][0-9]{2}|'
+   '429496[0-6][0-9]{3}|42949[0-5][0-9]{4}|'
+   '4294[0-8][0-9]{5}|'
+   '429[0-3][0-9]{6}|42[0-8][0-9]{7}|4[01][0-9]{8}|'
+   '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0):'
+   '(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2}|'
+   '6[0-4][0-9]{3}|'
+   '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0))';
tailf:info "0:2-octet-asn:4-octet-number
           1:4-octet-ipv4addr:2-octet-number
           2:4-octet-asn:2-octet-number";
    }
  }
}
// L3VPN as-no has type union (as defined below), L3NM local-as type uint32
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:local-as" {
  deviate replace {
    type union {
      type uint32 {
        tailf:info "1-4294967295;;Autonomous system number";
        range "1..4294967295";
      }
      type string {
        tailf:info "1.0-XX.YY;;Autonomous system number";
        pattern '[0-9]+\.[0-9]+';
      }
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:local-as" {
  deviate replace {
    type union {
      type uint32 {
        tailf:info "1-4294967295;;Autonomous system number";
        range "1..4294967295";
      }
      type string {
        tailf:info "1.0-XX.YY;;Autonomous system number";
        pattern '[0-9]+\.[0-9]+';
      }
    }
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile/l3nm:local-as" {
  deviate replace {
    type union {
      type uint32 {
        tailf:info "1-4294967295;;Autonomous system number";
        range "1..4294967295";
      }
      type string {
        tailf:info "1.0-XX.YY;;Autonomous system number";
        pattern '[0-9]+\.[0-9]+';
      }
    }
  }
}

```

```

    }
  }
}

deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile/l3nm:role" {
  deviate add {
    default "vpn-common:hub-role";
  }
}

// ===== vpn-network-access =====
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access" {
  deviate add {
    // interface-id maps to Flat L3VPN flat-L3vpn -> endpoint -> if-type which
    // is mandatory which means vpn-network-access list is mandatory
    min-elements 1;
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:interface-id" {
  deviate add {
    // interface-id maps to Flat L3VPN flat-L3vpn -> endpoint -> if-type which
    // is mandatory
    mandatory true;
  }
}
deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:interface-id" {
  deviate replace {
    type string {
      pattern "(Bundle-
Ether|BVI|FiftyGigE|FortyGigE|FourHundredGigE|HundredGigE|Loopback|TenGigE|TwentyFi
veGigE|TwoHundredGigE|GigabitEthernet|Ethernet) [0-9]+(/[0-9]+)*";
    }
  }
}

deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:service/l3nm:mtu" {
  deviate replace {
    type uint16 {
      range "64..65535";
    }
  }
}

deviation "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access/l3nm:routing-
protocols/l3nm:routing-protocol/l3nm:bgp/l3nm:redistribute-connected/l3nm:enable" {
  deviate add {
    default "false";
  }
}
}

```

IETF-L3VPN-NM Augmentations Model

```

module cisco-l3vpn-ntw {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-ntw";
  prefix cisco-l3nm;

  import ietf-yang-types { prefix yang; }
  import ietf-l3vpn-ntw { prefix l3nm; }
  import cisco-mvpn { prefix cisco-mvpn; }
  import tailf-common { prefix tailf; }
  import tailf-ncs { prefix ncs; }
  import ietf-inet-types { prefix inet; }
  import tailf-kicker { prefix kicker; }
  import ietf-vpn-common { prefix vpn-common; }
  import cisco-tdsn-core-fp-common { prefix tsdn-core-fp-common; }
  import lsa-utils { prefix lsa-utils; }
  import resource-allocator { prefix ralloc; }
  import id-allocator { prefix idalloc; }
  import custom-template-hook { prefix ct-hook; }
  import ietf-routing-types {
    prefix rt-types;
    reference
      "RFC 8294: Common YANG Data Types for the Routing Area";
  }

  description "IETF L3VPN NTW Cisco NSO Modifications";

  ...

  typedef service-name {
    type string {
      pattern "[a-zA-Z0-9\\-_]+";
    }
  }

  grouping static-inc-mask-count {
    container inc-mask {
      presence "true";
      description "Increment mask for group or source address";
      tailf:info "Increment mask for group or source address";
      leaf address {
        description "group address to join";
        tailf:info "group address to join";
        type inet:ipv4-address-no-zone;
      }

      leaf count {
        description "No. of group addresses to join";
        tailf:info "No. of group addresses to join";
        type uint32 {
          range "1..512";
        }
        mandatory true;
      }
    }
  }

  grouping srv6-grouping {
    container srv6 {

```

```

presence true;

tailf:info "Associate SRv6 Policy";
description "Associate SRv6 Policy";

list address-family {
  min-elements 1;

  key name;
  leaf name {
    tailf:info
      "Address Family";
    description
      "Address Family";
    type identityref {
      base vpn-common:address-family;
    }
  }
  leaf locator-name {
    tailf:info
      "SRv6 locator name, should match locators configured at a node-global
level on each router";
    description
      "SRv6 locator name, should match locators configured at a node-global
level on each router";
    type string {
      length 1..64;
    }
  }
}
}
}
}
}
grouping rt-allocation-data {
  container rt-allocation-data {
    leaf hub-rt {
      type string;
    }
    leaf spoke-rt {
      type string;
    }
  }
}
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/cisco-
mvpn:multicast/cisco-mvpn:ipv4" {
  list static-sr-mpls-p2mp {
    when "../cisco-mvpn:mvpn-ipmsi-tunnel-ipv4/cisco-mvpn:tunnel-type = 'static-
sr-mpls-p2mp'";
    key policy-name;
    leaf policy-name {
      tailf:info "Name of the SR P2MP Policy";
      description "Name of the SR P2MP Policy";
      type string;
    }
  }
  container sr-p2mp-policy {
    presence "true";
    tailf:info "Segment-routing P2MP policy";
    description "Segment-routing P2MP policy";

    list group-address {
      tailf:info "IPv4 static multicast group";

```

```

description "IPv4 static multicast group";
min-elements 1;
key address;
leaf address {
    tailf:info "IPv4 static multicast group";
    description "IPv4 static multicast group";
    type inet:ipv4-address-no-zone;
}

list source-address {
    tailf:info "Source address";
    description "Source address";
    max-elements 1;
    key address;
    leaf address {
        tailf:info "Source address";
        description "Source address";
        type inet:ipv4-address-no-zone;
    }
    uses static-inc-mask-count;
}
uses static-inc-mask-count;
}
}
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node" {
    container multicast {
        tailf:info
            "Multicast parameters for the vpn node.";
        description
            "Multicast parameters for the vpn node.";
        leaf is-receiver-site-only {
            when "../../../cisco-mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-
ipmsi-tunnel-ipv4/cisco-mvpn:tunnel-type = 'sr-mpls-p2mp' or (../../../cisco-
mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-spmsi-tunnels-ipv4/cisco-mvpn:mvpn-
spmsi-tunnel-ipv4[cisco-mvpn:tunnel-type='sr-mpls-p2mp'] [cisco-mvpn:switch-
wildcard-mode='star-star'])";
            type boolean;
            default false;
            description "Current PE has only receiver sites.";
        }
        leaf enable-inter-as {
            when "../../../cisco-mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-
ipmsi-tunnel-ipv4/cisco-mvpn:tunnel-type = 'sr-mpls-p2mp' or (../../../cisco-
mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-spmsi-tunnels-ipv4/cisco-mvpn:mvpn-
spmsi-tunnel-ipv4[cisco-mvpn:tunnel-type='sr-mpls-p2mp'] [cisco-mvpn:switch-
wildcard-mode='star-star'])";
            type empty;
            description "Enable Inter-AS on the vpn-node";
        }
        list static-sr-mpls-p2mp {
            when "../../../cisco-mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-
ipmsi-tunnel-ipv4/cisco-mvpn:tunnel-type = 'static-sr-mpls-p2mp'";
            key policy-name;
            leaf policy-name {
                tailf:info "Name of the SR P2MP Policy";
                description "Name of the SR P2MP Policy";
                type leafref {

```



```

        path "../../../../../../cisco-mvpn:multicast/cisco-mvpn:ipv4/static-sr-
mpls-p2mp/policy-name";
    }
    leaf role {
        type enumeration {
            enum source;
            enum receiver;
        }
        mandatory true;
    }
}
must "not(../../../cisco-mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-
ipmsi-tunnel-ipv4/cisco-mvpn:tunnel-type = 'static-sr-mpls-p2mp') or
count(static-sr-mpls-p2mp)>=1" {
    error-message "SR P2MP Policy must be added if tunnel-type is static-sr-
mpls-p2mp";
}
}
}

augment "/l3nm:l3vpn-ntw" {
    container l3nm-actions {
        tailf:action cleanup {
            tailf:actionpoint ietf-l3vpn-nm-cleanup;
            tailf:info "Clean up L3NM service configuration and operational data";
            description "Clean up L3NM service configuration and operational data";
            input {
                leaf service {
                    type string;
                    mandatory true;
                }
                leaf vpn-node {
                    tailf:info "Clean up L3NM service configuration for a given vpn-node";
                    description "Clean up L3NM service configuration for a given vpn-node";
                    type string;
                }
                leaf vpn-network-access-id {
                    tailf:info "Clean up this vpn-node associated with vpn-network-access-
id" +
                        "under given service.";
                    description "Clean up this vpn-node associated with vpn-network-access-
id" +
                        "under given service.";
                    type string;
                    when "../vpn-node";
                }
                leaf no-networking {
                    type boolean;
                    mandatory true;
                }
            }
        }
        output {
            leaf success {
                type boolean;
                mandatory true;
            }
            leaf detail {
                type string;
            }
        }
    }
}

```

```

tailf:confirm-text "#####\n" +
  "#      Warning      #\n" +
  "#####\n" +
  "You are about to forcefully cleanup a T-SDN service.\n" +
  "This will affect the deploying service and leave network device(s) & NSO
out-of-sync (for no-networking=true).\n" +
  "Are you sure you want to proceed?" {
    tailf:confirm-default false;
  }
}
tailf:action internal-plan-change-handler {
  tailf:hidden full;
  tailf:actionpoint ietf-l3vpn-nm-internal-plan-change-handler;
  input {
    uses kicker:action-input-params;
  }
  output {
  }
}
tailf:action vpn-node-plan-change-handler {
  tailf:hidden full;
  tailf:actionpoint ietf-l3vpn-nm-vpn-node-plan-change-handler;
  input {
    uses kicker:action-input-params;
  }
  output {
  }
}
tailf:action error-recovery {
  tailf:actionpoint ietf-l3vpn-nm-fp-error-recovery;
  tailf:info "Action to recover a service during create/update/delete from a
transient failures
again.
device
like device connection issues once the device is reachable
For create/update failures, this action issues a sync on the
device
& redeploys the service with reconcile option.
For delete failures, this action issues a sync on the device &
redeploys zombie services.";
  description "Action to recover a service during create/update/delete from a
transient failures
again.
device
like device connection issues once the device is reachable
For create/update failures, this action issues a sync on the
device
& redeploys the service with reconcile option.
For delete failures, this action issues a sync on the device &
redeploys zombie services.";
  input {
    leaf service {
      tailf:info "Service to recover";
      description "Service to recover";
      type string;
      mandatory true;
    }
    leaf vpn-node {
      tailf:info "Recover this vpn-node under given service.";
      description "Recover this vpn-node under given service.";
      type string;
    }
    leaf vpn-network-access-id {

```

```

id" +
    tailf:info "Recover this vpn-node associated with vpn-network-access-
                "under given service.";
id" +
    description "Recover this vpn-node associated with vpn-network-access-
                "under given service.";
    type string;
    when "../vpn-node";
}
leaf sync-direction {
    tailf:info "sync device with sync-from or sync-to";
    description "sync device with sync-from or sync-to";
    type enumeration {
        enum sync-from;
        enum sync-to;
    }
    mandatory true;
}
}
output {
    leaf success {
        type boolean;
        mandatory true;
    }
    leaf detail {
        type string;
    }
}
}
tailf:confirm-text "#####\n" +
    "#          Warning          #\n" +
    "#####\n" +
    "You are about to recover a T-SDN service.\n" +
    "This will issue a sync on the device.\n" +
    "Are you sure you want to proceed?" {
    tailf:confirm-default false;
}
}
}
tailf:action update-internal-cfp-configurations {
    tailf:hidden tsdn;
    tailf:actionpoint update-l3nm-internal-cfp-configurations;
    tailf:info "Copy over changes in external cfp-configurations to internal
cfp-configurations";
    description "Copy over changes in external cfp-configurations to internal
cfp-configurations";
    input {
    }
    output {
    }
}
}
tailf:action vpn-service-delete-handler {
    tailf:hidden full;
    tailf:actionpoint ietf-l3vpn-nm-vpn-service-delete-handler;
    input {
        uses kicker:action-input-params;
    }
}
}
tailf:action check-sync {
    tailf:hidden tsdn;
    tailf:actionpoint ietf-l3vpn-nm-check-sync;
    tailf:info "Action to check whether all vpn-nodes that belong to a service
are in sync.";
}

```

```

        description "Action to check whether all vpn-nodes that belong to a service
are in sync.";
        input {
            leaf service {
                type leafref {
                    path "../..../..../l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-
service/l3nm:vpn-id";
                }
                mandatory true;
            }
        }
        output {
            leaf success {
                type boolean;
                mandatory true;
            }
            leaf detail {
                type string;
            }
        }
    }
    tailf:action deep-check-sync {
        tailf:hidden tsdn;
        tailf:actionpoint ietf-l3vpn-nm-deep-check-sync;
        tailf:info "Action to check whether all vpn-nodes that belong to a service
are in sync." +
            "Otherwise performs a sync-from those vpn-nodes.";
        description "Action to check whether all vpn-nodes that belong to a service
are in sync." +
            "Otherwise performs a sync-from those vpn-nodes.";
        input {
            leaf service {
                type leafref {
                    path "../..../..../l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-
service/l3nm:vpn-id";
                }
                mandatory true;
            }
        }
        output {
            leaf success {
                type boolean;
                mandatory true;
            }
            leaf detail {
                type string;
            }
        }
        tailf:confirm-text "#####\n" +
            "#      Warning      #\n" +
            "#####\n" +
            "You are about to perform a deep-check-sync on all the vpn-nodes.\n" +
            "This may cause severe lag or scale issues momentarily.\n" +
            "Are you sure you want to proceed?" {
            tailf:confirm-default false;
        }
    }
}
container cfp-configurations {
    container resource-pools {
        leaf rt-pool-name {

```

```

    tailf:info "Route-Target RT pool";
    description "Route-Target RT pool";
    type leafref {
      path "/ralloc:resource-pools/idalloc:id-pool/idalloc:name";
    }
  }
}
list dynamic-device-mapping {
  tailf:info "Dynamic Device Mapping";
  description "Dynamic Device Mapping";
  key "ned-id";
  leaf ned-id {
    tailf:info "Device NED ID, eg ned:netconf";
    description "Device NED ID, eg ned:netconf";
    type string;
    tailf:cli-allow-range;
  }
  leaf python-impl-class-name {
    tailf:info "Device Python implementation class, eg module.class";
    description "Device Python implementation class, eg module.class";
    type string;
    mandatory true;
  }
}

leaf l3vpn-validation-enabled {
  tailf:hidden tsdn;
  tailf:info "If set to true, following validations are done for l3vpn
service:
    1. All the endpoint interfaces are available on the endpoint.";
  description "If set to true, following validations are done for l3vpn
service:
    1. All the endpoint interfaces are available on the endpoint.";
  type boolean;
  default true;
}

leaf l3vpn-device-error-validation-enabled {
  tailf:info "If set to true, following validations are done for l3vpn
service:
    1. Check if RD change will be accepted by endpoint.";
  description "If set to true, following validations are done for l3vpn
service:
    1. Check if RD change will be accepted by endpoint.";
  type boolean;
  default true;
}

leaf iosxr-default-ned-id {
  tailf:info "NED ID of the default IOSXR Device.";
  description "NED ID of the default IOSXR Device.";
  tailf:hidden tsdn;
  type string;
  default "cisco-iosxr-cli-7.52:cisco-iosxr-cli-7.52";
}

container global-rd-enabled {
  tailf:info "Set this flag to apply RD under global VRF definition.";
  description "Set this flag to apply RD under global VRF definition.";
  presence "true";
  uses ncs:service-data;
}

```

```

        ncs:servicepoint "ietf-l3nm-cfp-config";
    }
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services" {
    list vpn-service-plan {
        config false;
        tailf:cdb-oper {
            tailf:persistent true;
        }
        key "vpn-id";
        leaf vpn-id {
            type string;
            description
                "VPN identifier. This identifier has a local meaning.";
            tailf:info
                "VPN identifier. This identifier has a local meaning.";
        }
        uses ncs:nano-plan-data {
            augment "plan" {
                uses rt-allocation-data;
                leaf failed-vpn-nodes {
                    type string;
                    description
                        "VPN Nodes that are in failed state.";
                }
            }
            augment "plan/component/state" {
                leaf last-when {
                    tailf:hidden full;
                    type yang:date-and-time;
                    tailf:cli-value-display-template "$(.|datetime)";
                    when '../status != "not-reached"';
                    description
                        "Timestamp for last notification sent. This field is exclusively
updated
                        by cisco-tdsn-core-fp-common external plan handler.";
                }
            }
        }
        list globally-updated-vpn-node {
            tailf:hidden tsdn;
            description "This list tracks any vpn-nodes that are affected due to global
vpn-service
                        level changes. Item in this list is added by
L3NMVpnServiceUpdateHandler and
                        will be removed in the IETFL3vpnNMVPNNodeValidator after the
vpn-node service
                        successfully completes.";
            key vpn-node-id;
            leaf vpn-node-id {
                type string;
            }
        }
        tailf:action purge-globally-updated-vpn-node {
            tailf:hidden full;
            tailf:actionpoint ietf-l3vpn-nm-purge-global-nodes-actionpoint;
            input {
                choice vpn-nodes-choice {
                    leaf all {

```

```

        type empty;
    }
    leaf-list vpn-nodes {
        type string;
    }
    mandatory true;
}
}
output {
    leaf success {
        type boolean;
        mandatory true;
    }
    leaf detail {
        type string;
    }
}
}
}

container vpn-nodes {
    list vpn-node-plan {
        key "vpn-node-id";
        leaf vpn-node-id {
            type string;
        }

        uses ncs:nano-plan-data {
            augment "plan" {
                uses rt-allocation-data;
                uses tsdn-core-fp-common:status-code-plan-augmentation;
            }
            augment "plan/component" {
                uses tsdn-core-fp-common:status-code-component-augmentation;
                uses tsdn-core-fp-common:static-config-redeploy-indicator-component-
augmentation;
            }
        }
        container internal {
            tailf:hidden full;
            leaf-list endpoint-names {
                type string;
            }
        }
    }
}
}
}

augment "/13nm:l3vpn-ntw/13nm:vpn-services/13nm:vpn-service/13nm:underlay-
transport" {
    container discovered-underlay-transport {
        tailf:info "HCO service to underlay transport mapping";
        description "HCO service to underlay transport mapping";
        config false;

        list te-tunnel-ref {
            tailf:info "List of rsvp-te tunnels";
            description "List of rsvp-te tunnels";

            key "tunnel-id source";

```

```

leaf tunnel-id {
  tailf:info "TE tunnel Identifier";
  description "TE tunnel Identifier";
  type string;
}
leaf source {
  tailf:info "TE tunnel Source IP";
  description "TE tunnel Source IP";
  type inet:ip-address;
}
leaf destination {
  tailf:info "TE tunnel Destination IP";
  description "TE tunnel Destination IP";
  type inet:ip-address;
}
}

list sr-policy-ref {
  tailf:info "List of sr-policies (static or dynamically created)";
  description "List of sr-policies (static or dynamically created)";

  key "headend color endpoint";
  leaf headend {
    tailf:info "SR policy Headend Device ID";
    description "SR policy Headend Device ID";
    type string;
  }
  leaf color {
    tailf:info "<1-4294967295> - SR policy color";
    description "<1-4294967295> - SR policy color";
    type uint32 {
      range "1..4294967295";
    }
  }
  leaf endpoint {
    tailf:info "SR policy Endpoint IP";
    description "SR policy Endopint IP";
    type inet:ip-address;
  }
}
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service" {
  uses ct-hook:template-hook;
  uses tsdn-core-fp-common:service-assurance-grouping {
    when "/tsdn-core-fp-common:enable-service-assurance = 'true'";
  }
}

container probes {
  tailf:info "Probes container";
  description "Probes container";
  presence "true";

  list endpoint {
    key "id";
    tailf:info "List of endpoint";
    description "List of endpoint";
    min-elements 2;

    leaf id {

```



```

    tailf:info "Endpoint identifier";
    description "Endpoint identifier";
    type string;
}

leaf vpn-node {
    tailf:info "A vpn node";
    description "A vpn node";
    type leafref {
        path "../..../l3nm:vpn-nodes/l3nm:vpn-node/l3nm:vpn-node-id";
    }
    mandatory true;
}

leaf vpn-network-access {
    tailf:info "VPN network access";
    description "VPN network access";
    type leafref {
        path "../..../l3nm:vpn-nodes/l3nm:vpn-node[l3nm:vpn-node-
id=current()../vpn-node]/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:id";
    }
    mandatory true;
}

leaf interface-id {
    tailf:info "VPN network access interface-id";
    description "VPN network access interface-id";
    type leafref {
        path "../..../l3nm:vpn-nodes/l3nm:vpn-node[l3nm:vpn-node-
id=current()../vpn-node]/l3nm:vpn-network-accesses/l3nm:vpn-network-
access[l3nm:id=current()../vpn-network-access]/l3nm:interface-id";
    }
    mandatory true;
}

leaf agent-ipv4 {
    tailf:info "Agent's ipv4 address";
    description "Agent's ipv4 address";
    type inet:ipv4-address;
}

leaf agent-ipv6 {
    tailf:info "Agent's ipv6 address";
    description "Agent's ipv6 address";
    type inet:ipv6-address;
}

leaf agent-vlan {
    tailf:info "Agent's vlan";
    description "Agent's vlan";
    type uint16 {
        range "1..4094";
    }
    mandatory true;
}

must "agent-ipv4 or agent-ipv6" {
    error-message "Either agent-ipv4 or agent-ipv6 must be present";
}
}

```

```

choice connection-type {
  case connection-point-to-point {

    container point-to-point {
      tailf:info "Includes point to point based connections";
      description "Includes point to point based connections";

      list connection {
        tailf:info "List of point to point based connections";
        description "List of point to point based connections";
        key "source destination";
        min-elements 1;

        leaf source {
          tailf:info "Source identifier";
          description "Source identifier";
          type leafref {
            path "../..../endpoint/id";
          }
          mandatory true;
        }

        leaf destination {
          tailf:info "Destination identifier";
          description "Destination identifier";
          type leafref {
            path "../..../endpoint/id";
          }
          mandatory true;
        }
      }
    }
  }
}

case connection-hub-and-spoke {
  list hub-and-spoke {
    tailf:info "List of connections based on a hub and spoke topology";
    description "List of connections based on a hub and spoke topology";
    key "id";
    min-elements 1;

    leaf id {
      tailf:info "Hub and spoke identifier; Range can be 0 to 65535";
      description "Hub and spoke identifier; Range can be 0 to 65535";
      type uint16;
    }

    leaf-list hub {
      tailf:info "Hub endpoint";
      description "Hub endpoint";
      type leafref {
        path "../..../endpoint/id";
      }
      min-elements 1;
    }

    leaf-list spoke {
      tailf:info "Spoke endpoint";
      description "Spoke endpoint";
      type leafref {

```

```

        path "../../endpoint/id";
    }
    min-elements 1;
}
}

case connection-mesh {
    leaf mesh {
        tailf:info "Connection based on mesh topology";
        description "Connection based on mesh topology";
        type empty;
    }
}

must "point-to-point or hub-and-spoke or mesh" {
    error-message "At least one connection-type must be present";
}

}

container vpn-target {
    when "../../l3nm:vpn-service-topology = 'vpn-common:any-to-any' or ../../l3nm:vpn-
service-topology = 'vpn-common:hub-spoke'";
    tailf:info "VPN Target";
    description "VPN Target";

    choice hub-rt-choice {
        default "auto-hub-rt";
        leaf auto-hub-rt {
            tailf:info "Auto Hub Route-Target";
            description "Auto Hub Route-Target";
            type empty;
        }
        leaf hub-rt-value {
            tailf:info "Hub Route-Target Value";
            description "Hub Route-Target Value";
            type rt-types:route-target;
        }
    }

    choice spoke-rt-choice {
        when "../../l3nm:vpn-service-topology = 'vpn-common:hub-spoke'";
        default "auto-spoke-rt";
        leaf auto-spoke-rt {
            tailf:info "Auto Spoke Route-Target";
            description "Auto Spoke Route-Target";
            type empty;
        }
        leaf spoke-rt-value {
            tailf:info "Spoke Route-Target Value";
            description "Spoke Route-Target Value";
            type rt-types:route-target;
        }
    }
}

tailf:action self-test {
    tailf:actionpoint ietf-l3vpn-nm-self-test-actionpoint;
    input {
    }
}

```

```

output {
  leaf status {
    type string;
  }
  leaf message {
    type string;
  }
}
}
tailf:action error-recovery {
  tailf:actionpoint ietf-l3vpn-nm-service-error-recovery;
  tailf:info "Action to recover a service during create/update from a transient
failures
          like device connection issues once the device is reachable again.
          For create/update failures, this action issues a sync on the
device
          & redeploys the service with reconcile option.";
  description "Action to recover a service during create/update from a
transient failures
          like device connection issues once the device is reachable again.
          For create/update failures, this action issues a sync on the
device
          & redeploys the service with reconcile option.";
  input {
    leaf sync-direction {
      tailf:info "sync device with sync-from or sync-to";
      description "sync device with sync-from or sync-to";
      type enumeration {
        enum sync-from;
        enum sync-to;
      }
      mandatory true;
    }
  }
  output {
    leaf success {
      type boolean;
      mandatory true;
    }
    leaf detail {
      type string;
    }
  }
  tailf:confirm-text "#####\n" +
    "#          Warning          #\n" +
    "#####\n" +
    "You are about to recover a T-SDN service.\n" +
    "This will issue a sync on the device.\n" +
    "Are you sure you want to proceed?" {
    tailf:confirm-default false;
  }
}
tailf:action vpn-service-change-handler {
  tailf:hidden full;
  tailf:actionpoint ietf-l3vpn-nm-vpn-service-change-handler;
  input {
    uses kicker:action-input-params;
  }
}
tailf:action touch-vpn-nodes {
  tailf:actionpoint ietf-l3vpn-nm-touch-vpn-service-handler;

```

```

    tailf:info "Action to touch vpn nodes alongside commit dry-run on vpn-service
to
    see the device configuration which will be sent to the vpn
nodes";
    description "Action to touch vpn nodes alongside commit dry-run on vpn-
service to
    see the device configuration which will be sent to the vpn
nodes";
    input {
    }
    output {
    leaf detail {
        type string;
    }
    }
}
tailf:action re-deploy {
tailf:actionpoint ietf-l3vpn-nm-re-deploy-vpn-service-handler;
tailf:info "Action to re-deploy vpn service";
description "Action to re-deploy vpn service";
input {
    leaf no-networking {
        type boolean;
        default false;
    }
    container reconcile {
        tailf:info
            "Reconcile the service data. All data which existed before
            the service was created will now be owned by the service.
            The reference count will indicate how many services control
            the data. When the service is removed that data will also be
            removed. In technical terms the reference count will be
            decreased by one for everything which existed prior to the
            service.

            If manually configured data exists below in the configuration
            tree that data is kept unless the option
            'discard-non-service-config' is used.";
        description
            "Reconcile the service data. All data which existed before
            the service was created will now be owned by the service.
            The reference count will indicate how many services control
            the data. When the service is removed that data will also be
            removed. In technical terms the reference count will be
            decreased by one for everything which existed prior to the
            service.

            If manually configured data exists below in the configuration
            tree that data is kept unless the option
            'discard-non-service-config' is used.";
        presence "";
        choice c-non-service-config {
            leaf keep-non-service-config {
                type empty;
            }
            leaf discard-non-service-config {
                type empty;
            }
        }
        default keep-non-service-config;
    }
}
}

```

```

    }
    output {
      leaf success {
        type boolean;
        mandatory true;
      }
      leaf detail {
        type string;
      }
    }
  }
}
tailf:action reactive-re-deploy {
  tailf:actionpoint ietf-l3vpn-nm-reactive-re-deploy-vpn-service-handler;
  tailf:info "Action to reactive-re-deploy vpn service";
  description "Action to reactive-re-deploy vpn service";
  input {
  }
  output {
    leaf success {
      type boolean;
      mandatory true;
    }
    leaf detail {
      type string;
    }
  }
}
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-nodes"
  + "/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access"
  + "/l3nm:routing-protocols/l3nm:routing-protocol/l3nm:bgp" {
  container update-source {
    presence "true";
    tailf:info "Source of routing updates";
    description "Source of routing updates";
    leaf if-type {
      tailf:display-when "not(contains(/ncs:devices/l3nm:lsas/l3nm:
lsas:dispatch-map/l3nm:lsas:device
      [l3nm:lsas:name=current()/../../../../../../../../l3nm:vpn-node-id]/l3nm:
lsas:ned-id,
      'cisco-ios-cli-'))
      or
      not(../../../../../../../../l3nm:connection/l3nm:encapsulation/l3nm:dot1q/cisco-
l3nm:BDI)";
      tailf:info "Interface Type";
      description "Interface Type";
      type enumeration {
        enum Bundle-Ether;
        enum Ethernet;
        enum FiftyGigE;
        enum FortyGigE;
        enum FourHundredGigE;
        enum GigabitEthernet;
        enum HundredGigE;
        enum Loopback;
        enum TenGigE;
        enum TwentyFiveGigE;
        enum TwoHundredGigE;
      }
    }
  }
}
}

```

```

    leaf if-id {
        tailf:display-when "not(contains(/ncs:devices/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-nodes/
utils:dispatch-map/l3nm:vpn-network-accesses/l3nm:vpn-network-access
[lsa-utils:name=current()/../../../../../../../../l3nm:vpn-node-id]/lsa-
utils:ned-id,
    'cisco-ios-cli-'))
        or
        not(../../../../../../../../l3nm:connection/l3nm:encapsulation/l3nm:dot1q/cisco-
l3nm:BDI)";
        tailf:info "Interface Id";
        description "Interface Id";
        type string {
            pattern '[0-9]+(/[0-9]+)*';
        }
    }

    leaf sub-if-id {
        tailf:display-when "not(contains(/ncs:devices/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-nodes/
utils:dispatch-map/l3nm:vpn-network-accesses/l3nm:vpn-network-access
[lsa-utils:name=current()/../../../../../../../../l3nm:vpn-node-id]/lsa-
utils:ned-id,
    'cisco-ios-cli-')) or ../if-type != 'Loopback'";
        tailf:info "Sub Interface Id";
        description "Sub Interface Id";
        type int32;
    }

    must "not(if-type) or (((if-type!='Loopback' and if-type!='Bundle-Ether') or
" +
        " ((if-type='Loopback' or if-type='Bundle-Ether')
        and (string(number(if-id))!='NaN'))))" {
        error-message "Id must be a number";
    }
    must "not(if-type) or (((if-type!='Bundle-Ether') or (if-type='Bundle-Ether')
        and (string(number(if-id))>'0')
        and (string(number(if-id))<'65536')))" {
        error-message "if-id must be in the range 1-65535";
    }
    must "not(if-type) or (((if-type!='Loopback') or (if-type='Loopback')
        and (string(number(if-id))>='0')
        and (string(number(if-id))<='2147483647')))" {
        error-message "if-id must be in the range 0-2147483647";
    }
}

leaf mpls-deactivation {
    tailf:info "Set to true to disable BGP MPLS forwarding.";
    description "Set to true to disable BGP MPLS forwarding.";
    type boolean;
    default "false";
}

}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-nodes"
+ "/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access"
+ "/l3nm:routing-protocols/l3nm:routing-
protocol/l3nm:bgp/l3nm:redistribute-connected" {
    leaf metric {
        tailf:info "Default metric <0..4294967295>";
    }
}

```

```

        description "Default metric <0..4294967295>";
        type uint32 {
            range "0..4294967295";
        }
    }
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-instance-
profiles/l3nm:vpn-instance-profile" {
    uses srv6-grouping;
}
augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-profile" {
    uses srv6-grouping {
        refine srv6 {
            tailf:display-when "not(contains(/ncs:devices/l3nm:vpn-service/l3nm:vpn-
instance-profiles/l3nm:vpn-instance-profile/l3nm:vpn-node-id]/l3nm:vpn-node-id, 'cisco-ios-cli-'))";
        }
    }
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node" {
    uses ct-hook:template-hook;
    tailf:action error-recovery {
        tailf:actionpoint ietf-l3vpn-nm-vpn-node-error-recovery;
        tailf:info "Action to recover this device during create/update from a
transient failures
like device connection issues once the device is reachable again.
For create/update failures, this action issues a sync on the
device
& redeploys the service with reconcile option.";
        description "Action to recover this device during create/update from a
transient
failures like device connection issues once the device is
reachable again.
For create/update failures, this action issues a sync on the
device
& redeploys the service with reconcile option.";
        input {
            leaf vpn-network-access-id {
                tailf:info "Recover this vpn-node associated with vpn-network-access-id"
+
                "under given service.";
                description "Recover this vpn-node associated with vpn-network-access-id"
+
                "under given service.";
                type string;
            }
            leaf sync-direction {
                tailf:info "sync device with sync-from or sync-to";
                description "sync device with sync-from or sync-to";
                type enumeration {
                    enum sync-from;
                    enum sync-to;
                }
                mandatory true;
            }
        }
    }
}
output {

```



```

        leaf success {
            type boolean;
            mandatory true;
        }
        leaf detail {
            type string;
        }
    }
}
tailf:confirm-text "#####\n" +
    "#          Warning          #\n" +
    "#####\n" +
    "You are about to recover a T-SDN service.\n" +
    "This will issue a sync on the device.\n" +
    "Are you sure you want to proceed?" {
    tailf:confirm-default false;
}
}
}

augment "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-nodes"
+ "/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access"
+ "/l3nm:connection/l3nm:encapsulation/l3nm:dot1q" {
    leaf BDI {
        tailf:info
            "<1-4095> BDI interface number. BDI must be unique for each sub-interface
on the device.";
        description
            "<1-4095> BDI interface number. BDI must be unique for each sub-interface
on the device.";
        tailf:display-when "../l3nm:cvlan-id and contains(/ncs:devices/l3nm:
utils:lsa/l3nm:lsa-utils:dispatch-map/l3nm:lsa-utils:device
[l3nm:lsa-utils:name=current()/../..../..../..../..../l3nm:vpn-node-id]/l3nm:
utils:ned-id,
'cisco-ios-cli-')";
        type uint16 {
            range "1..4095";
        }
    }
}

/**

Annotations

**/

tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/service-
assurance" {
    tailf:validate "ietf-l3vpn-nm-service-assurance-validation" {
        tailf:dependency ".";
    }
}

tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/probes" {
    tailf:validate "ietf-l3vpn-nm-probes-validation" {
        tailf:dependency ".";
    }
}

tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
instance-profiles" {

```

```

    tailf:info "Contains VPN instance profiles for a node. Use the profiles created
here to map to active-vpn-instance-profiles.";
    description "Contains VPN instance profiles for a node. Use the profiles
created here to map to active-vpn-instance-profiles.";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
instance-profiles/l3nm:vpn-instance-profile" {
    tailf:info "This is the list of VPN instance profiles for a node. Use the
profiles created here to map to active-vpn-instance-profiles.";
    description "This is the list of VPN instance profiles for a node. Use the
profiles created here to map to active-vpn-instance-profiles.";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
instance-profiles/l3nm:vpn-instance-profile/l3nm:profile-id" {
    tailf:info "This is the VPN instance profile of the node. It is mandatory to
create a vpn-instance-profile under vpn-service. Values added under profile
override any corresponding global values.";
    description "This is the VPN instance profile of the node. It is mandatory to
create a vpn-instance-profile under vpn-service. Values added under profile
override any corresponding global values.";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
instance-profiles/l3nm:vpn-instance-profile/l3nm:role" {
    when "../../../l3nm:vpn-service-topology = 'vpn-common:hub-spoke'";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
service-topology" {
    description "Defining service topology, supported topologies are any-to-any,
hub-spoke and custom.
        Point-to-point VPN service topology is not supported.";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes"
    + "/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-access"
    + "/l3nm:connection/l3nm:encapsulation/l3nm:dot1q/l3nm:cvlan-id" {
    when "not(contains(.../interface-id, 'Loopback')) and
not(contains(.../interface-id, 'BVI'))";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:service/l3nm:mtu" {
    when "not(contains(.../l3nm:interface-id, 'Loopback'))";
    tailf:info "<64-65535> - Set the MTU on an interface";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:vpn-network-accesses/l3nm:vpn-network-
access/l3nm:connection/l3nm:encapsulation/l3nm:type" {
    tailf:info "Encapsulation type. Dot1q is the only supported tagged interface.";
    description "Encapsulation type. Dot1q is the only supported tagged
interface.";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-

```

```

profile/l3nm:profile-id" {
    description "Node's active VPN instance profile. Creating vpn-instance-profile
under vpn-service is mandatory and needs to be mapped here. Values added under
profile override any corresponding global values.";
    tailf:info "Node's active VPN instance profile. Creating vpn-instance-profile
under vpn-service is mandatory and needs to be mapped here. Values added under
profile override any corresponding global values.";
}

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-profiles/l3nm:valid-provider-
identifiers/l3nm:routing-profile-identifier/l3nm:id" {
        description "Valid VPN profile to be referenced in the VPN service. Routing
Policy needs to be created and mapped here.";
        tailf:info "Valid VPN profile to be referenced in the VPN service. Routing
Policy needs to be created and mapped here.";
    }

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
instance-profiles/l3nm:vpn-instance-profile/l3nm:rd-choice/l3nm:directly-
assigned/l3nm:rd" {
        description "Indicates an RD value that is explicitly assigned. Examples -
0:65100:87400024, 1:255.255.255.255:65100, 2:87400024:65100";
        tailf:info "Indicates an RD value that is explicitly assigned. Examples -
0:65100:87400024, 1:255.255.255.255:65100, 2:87400024:65100";
    }

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
instance-profiles/l3nm:vpn-instance-profile/l3nm:address-family/l3nm:vpn-
targets/l3nm:vpn-target/l3nm:route-targets/l3nm:route-target" {
        description "Conveys an RT value. Examples - 0:65100:87400024,
1:255.255.255.255:65100, 2:87400024:65100";
        tailf:info "Conveys an RT value. Examples - 0:65100:87400024,
1:255.255.255.255:65100, 2:87400024:65100";
    }

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:rd-choice/l3nm:directly-assigned/l3nm:rd" {
        description "Indicates an RD value that is explicitly assigned. Examples -
0:65100:87400024, 1:255.255.255.255:65100, 2:87400024:65100";
        tailf:info "Indicates an RD value that is explicitly assigned. Examples -
0:65100:87400024, 1:255.255.255.255:65100, 2:87400024:65100";
    }

    tailf:annotate "/l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/l3nm:vpn-
nodes/l3nm:vpn-node/l3nm:active-vpn-instance-profiles/l3nm:vpn-instance-
profile/l3nm:address-family/l3nm:vpn-targets/l3nm:vpn-target/l3nm:route-
targets/l3nm:route-target" {
        description "Conveys an RT value. Examples - 0:65100:87400024,
1:255.255.255.255:65100, 2:87400024:65100";
        tailf:info "Conveys an RT value. Examples - 0:65100:87400024,
1:255.255.255.255:65100, 2:87400024:65100";
    }
}

```

IETF-L3VPN-NM MVPN Augmentations Model

```

module: cisco-mvpn

augment /l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service:
  +--rw multicast
    +--rw ipv4!
      +--rw mvpn-ipmsi-tunnel-ipv4
        | +--rw tunnel-type? p-tunnel
        | +--rw (ipmsi-tunnel-attribute)?
      +--rw mvpn-spmsi-tunnels-ipv4
        +--rw mvpn-spmsi-tunnel-ipv4* [tunnel-type switch-wildcard-mode]
          +--rw tunnel-type? p-tunnel
          +--rw (spmsi-tunnel-attribute)?
            +--rw switch-threshold? uint32
            +--rw per-item-tunnel-limit uint32
            +--rw switch-wildcard-mode enumeration
            +--rw group-acl-ipv4? string

augment /l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/cisco-
mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-spmsi-tunnels-ipv4/cisco-mvpn:mvpn-
spmsi-tunnel-ipv4:
  +---- route-policy? -> /l3nm:l3vpn-ntw/vpn-profiles/valid-provider-
identifiers/routing-profile-identifier/id
  +---- immediate-switch? boolean

augment /l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/cisco-
mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-ipmsi-tunnel-ipv4/cisco-mvpn:ipmsi-
tunnel-attribute:
  +--:(sr-mpls-p2mp)
    +---- color? uint32
    +---- fast-reroute!
    +---- lfa!

augment /l3nm:l3vpn-ntw/l3nm:vpn-services/l3nm:vpn-service/cisco-
mvpn:multicast/cisco-mvpn:ipv4/cisco-mvpn:mvpn-spmsi-tunnels-ipv4/cisco-mvpn:mvpn-
spmsi-tunnel-ipv4/cisco-mvpn:spmsi-tunnel-attribute:
  +--:(sr-mpls-p2mp)
    +---- color? uint32
    +---- fast-reroute!
    +---- lfa!

```

IETF-L3VPN-NM Routing Policy Model

```

module: cisco-l3vpn-routing-policy
  +--rw l3vpn-routing-policy
    +--rw defined-sets
      | +--rw prefix-sets
      | | +--rw source-prefix-set* [name]
      | | | +--rw name string
      | | | +--rw mode? enumeration
      | | | +--rw prefixes
      | | |   +--rw prefix-list* [ip-prefix]
      | | |   +--rw ip-prefix inet:ip-prefix
      | | +--rw dest-prefix-set* [name]
      | | | +--rw name string
      | | | +--rw mode? enumeration
      | | | +--rw prefixes
      | | |   +--rw prefix-list* [ip-prefix]
      | | |   +--rw ip-prefix inet:ip-prefix
      | +--rw tag-sets
      | | +--rw tag-set* [name]
      | | | +--rw name string

```

```

|         +--rw tag-value*   uint32
+--rw policy-definitions
  +--rw policy-definition* [name]
    |   +--rw name           string
    |   +--rw statements
    |     +--rw statement* [name]
    |       +--rw name       string
    |       +--rw conditions
    |       | +--rw match-source-prefix-set!
    |       | | +--rw prefix-set   -> ../../../../../../../../defined-
sets/prefix-sets/source-prefix-set/name
    |       | | +--rw match-dest-prefix-set!
    |       | | +--rw prefix-set   -> ../../../../../../../../defined-
sets/prefix-sets/dest-prefix-set/name
    |       +--rw actions
    |         +--rw policy-result?   rt-pol:policy-result-type
    |         +--rw bgp-actions!
    |         | +--rw set-ext-community!
    |         | | +--rw (method)?
    |         | | +--:(reference)
    |         | | +--rw ext-community-set-ref?   ->
../../../../../../../../../../defined-sets/tag-sets/tag-set/name
    |         +--rw multicast-actions!
    |         | +--rw set-fast-reroute!
    |         | | +--rw set-lfa!
    |         | | +--rw set-data-mdt?           string
    |         | | +--rw set-on-demand-color   ->
../../../../../../../../../../defined-sets/tag-sets/tag-set/name
    |         +--rw apply-policy!
    |         +--rw extra-policy*   string

```

Example Function Packs

This topic discusses the yang models for the example services.

- IETF-TE Yang Models
- PM Yang Models
- NSS Yang Models

IETF-TE Yang Models

This yang model is a derivative of the standardized IETF model with the following modifications:

- **Tunnel name:** The string type is modified to string with pattern.
- **source, destination:** This is the tunnel source and destination. The devices' head-end and tail-end leaf elements are used as source and destination.

This section includes:

- IETF-TE Service Model
- IETF-TE Service Plan Model

- IETF-TE Deviation Model
- IETF-TE Augmentations Model

IETF-TE Service Model

```

module: ietf-te
  +--rw te
    +--rw tunnels
      | +--rw tunnel* [name]
      | | +--rw name string
      | | +--rw identifier uint16
      | | +--rw description? string
      | | +--rw source te-types:te-node-id
      | | +--rw destination te-types:te-node-id
      | | +--rw bidirectional? boolean
      | | +--rw te-bandwidth
      | | | +--rw (technology)?
      | | | +--:(generic)
      | | | | +--rw generic? te-bandwidth
      | | +--rw setup-priority? uint8
      | | +--rw hold-priority? uint8
      | | +--rw signaling-type? enumeration
      | | +--rw primary-paths
      | | | +--rw primary-path* [name]
      | | | | +--rw name string
      | | | | +--rw path-computation-method? identityref
      | | | | +--rw preference? uint8
      | | | | +--rw optimizations
      | | | | | +--rw (algorithm)?
      | | | | | +--:(metric) {path-optimization-metric}?
      | | | | | +--rw optimization-metric* [metric-type]
      | | | | | | +--rw metric-type enumeration
      | | | +--rw explicit-route-objects-always
      | | | | +--rw route-object-include-exclude* [index]
      | | | | +--rw explicit-route-usage? identityref
      | | | | +--rw index uint32
      | | | | +--rw (type)?
      | | | | | +--:(numbered-node-hop)
      | | | | | | +--rw numbered-node-hop
      | | | | | | | +--rw node-id te-node-id
      | | | | | | | +--rw hop-type? te-hop-type
      | | | | | +--:(label)
      | | | | | | +--rw label-hop
      | | | | | | | +--rw te-label
      | | | | | | | +--rw (technology)?
      | | | | | | | +--:(generic)
      | | | | | | | +--rw generic? rt-types:generalized-
label
      | | | | | | |
      | | | | +--rw cisco-te:custom-template* [name]
      | | | | | +--rw cisco-te:name? -> /ct-info:custom-template-
info/template-name
      | | | | | +--rw cisco-te:variable* [name]
      | | | | | | +--rw cisco-te:name? -> deref(..../name)/../ct-info:variables
      | | | | | | +--rw cisco-te:value string
      | | | | +--rw cisco-te:iteration* [number]
      | | | | | +--rw cisco-te:number? uint16
      | | | | | +--rw cisco-te:variable* [name]
      | | | | | | +--rw cisco-te:name? -> deref(..../name)/../ct-

```

```

info:variables
| | | +--rw cisco-te:value string
| | | +--rw cisco-te:fast-reroute? empty
| | | +--rw cisco-te:backup!
| | | | +--rw cisco-te:interfaces* [if-type if-id]
| | | | | +--rw cisco-te:if-type? enumeration
| | | | | +--rw cisco-te:if-id? string
| | | | +--rw cisco-te:performance-measurement!
| | | | +--rw cisco-te:head-end? -> /ncs:devices/lisa-
utils:lisa/dispatch-map/device/name
| | | +--rw cisco-te:tail-end? -> /ncs:devices/lisa-
utils:lisa/dispatch-map/device/name
| | | +--rw cisco-te:traffic-steering
| | | | +--rw (steering-choice)?
| | | | | +--:(autoroute)
| | | | | | +--rw cisco-te:autoroute
| | | | | | | +--rw cisco-te:announce
| | | | | | | | +--rw cisco-te:enable? boolean
| | | | | | | | +--rw cisco-te:metric!
| | | | | | | | | +--rw (metric-choice)?
| | | | | | | | | | +--:(constant)
| | | | | | | | | | | +--rw cisco-te:constant? uint32
| | | | | | | | | | | +--:(absolute)
| | | | | | | | | | | | +--rw cisco-te:absolute? uint32
| | | | | | | | | | | | +--:(relative)
| | | | | | | | | | | | | +--rw cisco-te:relative? int8
| | | | | | | | | | | +--rw cisco-te:destination* [address]
| | | | | | | | | | | +--rw cisco-te:address? inet:ipv4-address
| | | | | +--:(forwarding-adjacency)
| | | | | | +--rw cisco-te:forwarding-adjacency!
| | | | | | | +--rw cisco-te:holdtime? uint16
| | | | | | | +--rw cisco-te:include-ipv6? empty
| | | +----x cisco-te:error-recovery
| | | | +---w cisco-te:input
| | | | | +---w cisco-te:sync-direction enumeration
| | | | | | +---w (node-type)?
| | | | | | | +---:(source)
| | | | | | | | +---w cisco-te:source? te-types:te-node-id
| | | | | | | | +---:(destination)
| | | | | | | | | +---w cisco-te:destination? te-types:te-node-id
| | | | | +---ro cisco-te:output
| | | | | | +---ro cisco-te:success boolean
| | | | | | +---ro cisco-te:detail? string
| | | | +--rw cisco-te:action
| | | | | +----x cisco-te:self-test
| | | | | | +---ro cisco-te:output
| | | | | | | +---ro cisco-te:status? string
| | | | | | | +---ro cisco-te:message? string
+--rw cisco-te:cfp-configurations
| +--rw cisco-te:dynamic-device-mapping* [ned-id]
| | +--rw cisco-te:ned-id? string
| | +--rw cisco-te:python-impl-class-name string
+--rw cisco-te:iosxr-default-ned-id? string

```

IETF-TE Service Plan Model

```

module: ietf-te
+--rw te!
| +--rw tunnels
| | | +--ro cisco-te:tunnel-plan* [name]

```

```

| | | +--ro cisco-te:name?  string
| | | +--ro cisco-te:plan
| | |   +--ro cisco-te:component* [type name]
| | |     | +--ro cisco-te:name?          string
| | |     | +--ro cisco-te:type          plan-component-type-t
| | |     | +--ro cisco-te:state* [name]
| | |     | | +--ro cisco-te:name?          plan-state-name-t
| | |     | | +--ro cisco-te:status?       plan-state-status-t
| | |     | | +--ro cisco-te:when?        yang:date-and-time
| | |   +--ro cisco-te:failed?           empty
| | |   +--ro cisco-te:error-info!
| | |     | +--ro cisco-te:message?       string
| | |     | +--ro cisco-te:log-entry?     instance-identifier
| | |   +--ro cisco-te:deleting?         empty
| | |   +--ro cisco-te:service-location? instance-identifier
| | |   +--ro cisco-te:status-code-detail* [type name]
| | |     +--ro cisco-te:type?           ncs:plan-component-type-t
| | |     +--ro cisco-te:name?          string
| | |     +--ro cisco-te:code?          string
| | |     +--ro cisco-te:context* [context-name]
| | |       | +--ro cisco-te:context-name? string
| | |       | +--ro cisco-te:context-msg?  string
| | |     +--ro cisco-te:severity?       enumeration
| | |     +--ro cisco-te:recommended-action? string
| | |     +--ro cisco-te:impacted-device? String

```

IETF-TE Deviation Model

```

module ietf-te-deviations {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/ietf-te-deviations";
  prefix ietf-te-deviations;

  import tailf-common { prefix tailf; }
  import tailf-ncs { prefix ncs; }
  import custom-template-hook { prefix ct-hook; }
  import tailf-kicker { prefix kicker; }
  import ietf-inet-types {
    prefix inet;
  }
  /* Import TE base model */
  import ietf-te {
    prefix te;
    reference "draft-ietf-teas-yang-te: A YANG Data Model for Traffic
              Engineering Tunnels and Interfaces";
  }

  import ietf-te-types {
    prefix te-types;
    reference "draft-ietf-teas-yang-te-types: Common YANG Data Types for Traffic
              Engineering";
  }

  description "IETF TE Cisco NSO Deviations";

  ...

  deviation "/te:te/te:tunnels/te:tunnel/te:name" {
    deviate replace {

```



```

    type string {
      pattern '[\w\-\.\:,\_@#%$\+=\|;]+';
    }
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:identifier" {
  deviate replace {
    type uint16 {
      range "1..65535";
    }
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:identifier" {
  deviate add {
    mandatory true;
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:source" {
  deviate add {
    mandatory true;
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:destination" {
  deviate add {
    mandatory true;
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:signaling-type" {
  deviate replace {
    type enumeration {
      enum "te-types:path-setup-rsvp";
    }
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:optimizations/"
  + "te:algorithm/te:metric/te:optimization-metric/te:metric-type" {
  deviate replace {
    type enumeration {
      enum "te-types:path-metric-te";
      enum "te-types:path-metric-igp";
      enum "te-types:path-metric-delay-minimum";
      enum "te-types:path-metric-delay-average";
    }
  }
}

deviation "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-path/"
  + "te:explicit-route-objects-always/te:route-object-include-
exclude/te:index" {
  deviate replace {
    type uint32 {
      range "1..65535";
    }
  }
}

```

```

}
}

```

IETF-TE Augmentations Model

```

module cisco-te {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/cisco-te";
  prefix cisco-te;

  import lsa-utils { prefix lsa-utils; }
  import cisco-tdsn-core-fp-common { prefix tsdn-core-fp-common; }
  import tailf-common { prefix tailf; }
  import tailf-ncs { prefix ncs; }
  import custom-template-hook { prefix ct-hook; }
  import tailf-kicker { prefix kicker; }
  import ietf-inet-types {
    prefix inet;
  }
  /* Import TE base model */
  import ietf-te {
    prefix te;
    reference "draft-ietf-teas-yang-te: A YANG Data Model for Traffic
      Engineering Tunnels and Interfaces";
  }

  import ietf-te-types {
    prefix te-types;
    reference "draft-ietf-teas-yang-te-types: Common YANG Data Types for Traffic
      Engineering";
  }

  description "IETF TE Cisco NSO Augmentations";

  ...

  grouping redundancy-support {
    leaf fast-reroute {
      when "not(..../backup)";
      tailf:info "Specify MPLS tunnel can be fast-rerouted";
      description "Specify MPLS tunnel can be fast-rerouted";
      type empty;
    }
    container backup {
      tailf:info "Enable tunnel as a backup tunnel";
      description "Enable tunnel as a backup tunnel";
      presence true;
      list interfaces {
        tailf:info "List backup interfaces";
        description "List backup interfaces";
        key "if-type if-id";
        leaf if-type {
          tailf:info "Interface Type";
          description "Interface Type";
          type enumeration {
            enum Bundle-Ether;
            enum Ethernet;
            enum FiftyGigE;
          }
        }
      }
    }
  }
}

```

```

        enum FortyGigE;
        enum FourHundredGigE;
        enum GigabitEthernet;
        enum HundredGigE;
        enum TenGigE;
        enum TwentyFiveGigE;
        enum TwoHundredGigE;
    }
}
leaf if-id {
    tailf:info "Interface Id";
    description "Interface Id";
    type string {
        pattern '[0-9]+(/[0-9]+)*';
    }
}
}
}
}

grouping performance-measurement-grp {
    container performance-measurement {
        tailf:info "Enable default delay measurement profile";
        description "Enable default delay measurement profile";
        presence "true";
    }
}

augment "/te:te/te:tunnels" {
    list tunnel-plan {
        config false;
        tailf:cdb-oper {
            tailf:persistent true;
        }
        key "name";
        leaf name {
            type string;
        }
        uses ncs:nano-plan-data {
            augment "plan" {
                uses tsdn-core-fp-common:status-code-plan-augmentation;
            }
            augment "plan/component" {
                uses tsdn-core-fp-common:status-code-component-augmentation;
            }
        }
    }
}

container actions {
    tailf:action cleanup {
        tailf:actionpoint ietf-te-fp-cleanup;
        tailf:info "Clean up RSVP-TE service configuration and operational data";
        description "Clean up RSVP-TE service configuration and operational data";
        input {
            leaf service {
                type string;
                mandatory true;
            }
            leaf no-networking {
                tailf:info "Do not send any data to the devices.";
                description "Do not send any data to the devices.";
            }
        }
    }
}

```

```

        type boolean;
        mandatory true;
    }
}
output {
    leaf success {
        type boolean;
        mandatory true;
    }
    leaf detail {
        type string;
    }
}
}
tailf:confirm-text "#####\n" +
    "#          Warning          #\n" +
    "#####\n" +
    "You are about to forcefully cleanup a T-SDN service.\n" +
    "This will affect the deploying service and leave network device(s) " +
    " & NSO out-of-sync (for no-networking=true).\n" +
    "Are you sure you want to proceed?" {
    tailf:confirm-default false;
}
}
}

tailf:action error-recovery {
    tailf:actionpoint cisco-ietf-te-fp-error-recovery;
    tailf:info "Action to recover a service during create/update/delete from a
transient failures
again.
device
like device connection issues once the device is reachable
For create/update failures, this action issues a sync on the
& redeploys the service with reconcile option.
For delete failures, this action issues a sync on the device &
redeploys zombie services.";
    description "Action to recover a service during create/update/delete from a
transient failures
again.
device
like device connection issues once the device is reachable
For create/update failures, this action issues a sync on the
& redeploys the service with reconcile option.
For delete failures, this action issues a sync on the device &
redeploys zombie services.";
}
input {
    leaf service {
        tailf:info "Service to recover";
        description "Service to recover";
        type string;
        mandatory true;
    }
    leaf sync-direction {
        tailf:info "sync device with sync-from or sync-to";
        description "sync device with sync-from or sync-to";
        type enumeration {
            enum sync-from;
            enum sync-to;
        }
        mandatory true;
    }
}
choice node-type {

```

```

        leaf source {
            tailf:info "Trigger error recovery for source";
            description "Trigger error recovery for source";
            type te-types:te-node-id;
        }
        leaf destination {
            tailf:info "Trigger error recovery for destination";
            description "Trigger error recovery for destination";
            type te-types:te-node-id;
        }
    }
}
output {
    leaf success {
        type boolean;
        mandatory true;
    }
    leaf detail {
        type string;
    }
}
tailf:confirm-text "#####\n" +
    "#          Warning          #\n" +
    "#####\n" +
    "You are about to recover a T-SDN service.\n" +
    "This will issue a sync on the device.\n" +
    "Are you sure you want to proceed?" {
    tailf:confirm-default false;
}
}
}
}

augment "/te:te/te:tunnels/te:tunnel" {
    uses ct-hook:template-hook;
    uses redundancy-support;
    uses performance-measurement-grp;
    leaf head-end {
        tailf:info "Source device name; This field is mandatory and a commit will
fail without entering head-end information.";
        description "Source device name; This field is mandatory and a commit will
fail without entering head-end information.";
        type leafref {
            path "/ncs:devices/lsa-utils:lsa/lsa-utils:dispatch-map/lsa-
utils:device/lsa-utils:name";
        }
    }
    leaf tail-end {
        tailf:info "Destination device name.";
        description "Destination device name.";
        type leafref {
            path "/ncs:devices/lsa-utils:lsa/lsa-utils:dispatch-map/lsa-
utils:device/lsa-utils:name";
        }
    }
}
uses tunnel-steering-grouping;

tailf:action error-recovery {
    tailf:actionpoint cisco-ietf-te-fp-service-error-recovery;
    tailf:info "Action to recover a service during create/update/delete from a
transient failures

```

```

        like device connection issues once the device is reachable again.
        For create/update failures, this action issues a sync on the
device
        & redeploys the service with reconcile option.
        For delete failures, this action issues a sync on the device &
        redeploys zombie services.";
        description "Action to recover a service during create/update/delete from a
transient failures
        like device connection issues once the device is reachable again.
device
        For create/update failures, this action issues a sync on the

        & redeploys the service with reconcile option.
        For delete failures, this action issues a sync on the device &
        redeploys zombie services.";
input {
  leaf sync-direction {
    tailf:info "sync device with sync-from or sync-to";
    description "sync device with sync-from or sync-to";
    type enumeration {
      enum sync-from;
      enum sync-to;
    }
    mandatory true;
  }
  choice node-type {
    leaf source {
      tailf:info "Trigger error recovery for source";
      description "Trigger error recovery for source";
      type te-types:te-node-id;
    }
    leaf destination {
      tailf:info "Trigger error recovery for destination";
      description "Trigger error recovery for destination";
      type te-types:te-node-id;
    }
  }
}
output {
  leaf success {
    type boolean;
    mandatory true;
  }
  leaf detail {
    type string;
  }
}
tailf:confirm-text "#####\n" +
"#          Warning          #\n" +
"#####\n" +
"You are about to recover a T-SDN service.\n" +
"This will issue a sync on the device.\n" +
"Are you sure you want to proceed?" {
  tailf:confirm-default false;
}
}
container action {
  tailf:action self-test {
    tailf:actionpoint ietf-te-self-test-actionpoint;
    input {
    }
    output {

```

```

        leaf status {
            type string;
        }
        leaf message {
            type string;
        }
    }
}

augment "/te:te" {
    container ietf-te-actions {
        tailf:hidden tsdn;
        tailf:action internal-plan-change-handler {
            tailf:hidden full;
            tailf:actionpoint ietf-te-internal-plan-change-handler;
            input {
                uses kicker:action-input-params;
            }
            output {
            }
        }
        tailf:action update-internal-fp-configurations {
            tailf:hidden tsdn;
            tailf:actionpoint ietf-te-internal-fp-configurations;
            tailf:info "Copy over changes in external fp-configurations to internal fp-
configurations";
            description "Copy over changes in external fp-configurations to internal
fp-configurations";
            input {
            }
            output {
            }
        }
    }

    container cfp-configurations {
        list dynamic-device-mapping {
            tailf:info "Dynamic Device Mapping";
            description "Dynamic Device Mapping";
            key "ned-id";
            leaf ned-id {
                tailf:info "Device NED ID, eg ned:netconf";
                description "Device NED ID, eg ned:netconf";
                type string;
                tailf:cli-allow-range;
            }
            leaf python-impl-class-name {
                tailf:info "Device Python implementation class, eg module.class";
                description "Device Python implementation class, eg module.class";
                type string;
                mandatory true;
            }
        }

        leaf iosxr-default-ned-id {
            tailf:info "NED ID of the default IOSXR Device.";
            description "NED ID of the default IOSXR Device.";
            tailf:hidden tsdn;
            type string;
        }
    }
}

```

```

        default "cisco-iosxr-cli-7.52:cisco-iosxr-cli-7.52";
    }
}

grouping tunnel-steering-grouping {
    container traffic-steering {
        tailf:info "Steer traffic traffic through the tunnel";
        description "Steer traffic traffic through the tunnel";
        choice steering-choice {
            // interface tunnel-te* / autoroute
            default autoroute;
            case autoroute {
                container autoroute {
                    tailf:info "Parameters for IGP routing over tunnel";
                    description "Parameters for IGP routing over tunnel";

                    // interface tunnel-te* / autoroute announce
                    container announce {
                        tailf:info "Announce tunnel to IGP";
                        description "Announce tunnel to IGP";

                        leaf enable{
                            tailf:info "Enable autoroute announce";
                            description "Enable autoroute announce";
                            type boolean;
                            default 'true';
                        }
                    }
                    // interface tunnel-te* / autoroute announce / metric
                    container metric {
                        presence true;
                        tailf:info "Specify MPLS tunnel metric";
                        description "Specify MPLS tunnel metric";
                        choice metric-choice {
                            // interface tunnel-te* / autoroute announce / metric constant
                            leaf constant {
                                tailf:info "<1-2147483647> - Set constant metric";
                                description "<1-2147483647> - Set constant metric";
                                type uint32 {
                                    range "1..2147483647";
                                }
                            }
                            // interface tunnel-te* / autoroute announce / metric absolute
                            leaf absolute {
                                tailf:info "<1-2147483647> - Set absolute metric";
                                description "<1-2147483647> - Set absolute metric";
                                type uint32 {
                                    range "1..2147483647";
                                }
                            }
                            // interface tunnel-te* / autoroute announce / metric relative
                            leaf relative {
                                tailf:info "<-10,+10> - Set relative metric";
                                description "<-10,+10> - Set relative metric";
                                type int8 {
                                    range "-10..10";
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```



```

// interface tunnel-te* / autoroute destination *
list destination {
  tailf:info "Adds static route address A.B.C.D to destination";
  description "Adds static route address A.B.C.D to destination";
  key address;
  leaf address {
    tailf:info "A.B.C.D - IPv4 Address of static route to add";
    description "A.B.C.D - IPv4 Address of static route to add";
    type inet:ipv4-address;
  }
}
}
}
// interface tunnel-te* / forwarding-adjacency
case forwarding-adjacency {
  container forwarding-adjacency {
    presence "true";
    tailf:info "Enable tunnel as a forwarding-adjacency";
    description "Enable tunnel as a forwarding-adjacency";

    // interface tunnel-te* / forwarding-adjacency / holdtime
    leaf holdtime {
      tailf:info "<0-20000> - Hold time in ms";
      description "<0-20000> - Hold time in ms";
      type uint16 {
        range "0..20000";
      }
    }

    // interface tunnel-te* / forwarding-adjacency / include-ipv6
    leaf include-ipv6 {
      tailf:info "Announce tunnel as an IPv6 FA also";
      description "Announce tunnel as an IPv6 FA also";
      type empty;
    }
  }
}
}
}
}

// tailf info annotations
tailf:annotate "/te:te" {
  tailf:info "TE global container.";
}
tailf:annotate "/te:te/te:tunnels" {
  tailf:info "Tunnels TE configuration data container.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel" {
  tailf:info "The list of TE tunnels.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:name" {
  tailf:info "TE tunnel name.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:identifier" {
  tailf:info "TE tunnel Identifier.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:description" {
  tailf:info "Textual description for this TE tunnel.";
}
}

```

```

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths" {
  tailf:info "The set of primary paths.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:source" {
  tailf:info "TE tunnel source node ID.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:destination" {
  tailf:info "TE tunnel destination node identifier.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:bidirectional" {
  tailf:info "Indicates a bidirectional co-routed LSP.";
}
tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-path" {
  tailf:info "List of primary paths for this tunnel.";
}
  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:name" {
    tailf:info "TE path name.";
  }
  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:path-computation-method" {
    tailf:info "The method used for computing the path, either locally computed,
queried from a server or not computed at all (explicitly configured).";
  }
  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:preference" {
    tailf:info "Specifies a preference for this path. The lower the number higher
the preference.";
  }
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:setup-priority" {
  tailf:info "<0-7> - TE LSP requested setup priority";
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:hold-priority" {
  tailf:info "<0-7> - TE LSP requested hold priority";
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:signaling-type" {
  tailf:info "path-setup-rsvp;; TE tunnel path signaling type.";
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:te-bandwidth" {
  tailf:info "Container that specifies TE bandwidth. The choices can be augmented
for specific dataplane technologies.";
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:te-
bandwidth/te:technology/te:generic/te:generic" {
  tailf:info "Bandwidth specified in a generic format. It is a string containing
a list of numbers separated by commas, where each of these numbers can be non-
negative decimal, hex integer, or hex float. Example - 0x1p10, '0,2,3,1'.";
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always" {
  tailf:info "Container for the exclude route object list";
}

tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-exclude" {

```

```

    tailf:info "List of route objects to include or exclude in path computation";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-exclude/te:explicit-
route-usage" {
    tailf:info "Include or exclude usage. Default is to include";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-exclude/te:index" {
    tailf:info "Route object include-exclude index Range <1-65535>. The index is
used to identify an entry in the list. The order of entries is defined by the user
without relying on key values";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-
exclude/te:type/te:numbered-node-hop/te:numbered-node-hop" {
    tailf:info "Numbered node route hop";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-
exclude/te:type/te:numbered-node-hop/te:numbered-node-hop/te:node-id" {
    tailf:info "The identifier of a node in the TE topology.";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-
exclude/te:type/te:numbered-node-hop/te:numbered-node-hop/te:hop-type" {
    tailf:info "Strict or loose hop.";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:optimizations" {
    tailf:info "The objective function container that includes attributes to impose
when computing a TE path";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:optimizations/te:algorithm/te:metric/te:optimization-metric" {
    tailf:info "TE path metric type.";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:optimizations/te:algorithm/te:metric/te:optimization-metric/te:metric-type"
{
    tailf:info "Identifies an entry in the list of metric-types to optimize the TE
path for.";
  }

  tailf:annotate "/te:te/te:tunnels/te:tunnel/te:primary-paths/te:primary-
path/te:explicit-route-objects-always/te:route-object-include-
exclude/te:type/te:label/te:label-hop" {
    tailf:info "Label hop type.";
  }
}

```

PM Yang Models

This section includes:

- PM Service Model
- PM Service Plan Model

PM Service Model

```

module: cisco-pm-fp
  +--rw pm
  |   +--rw pm-profiles
  |   |   +--rw protocol
  |   |   |   +--rw profile* [name]
  |   |   |   |   +--rw name          string
  |   |   |   |   +--rw twamp-light!
  |   |   |   |   +--rw measurement!
  |   |   |   |   +--rw delay!
  |   |   |   |   +--rw unauthenticated!
  |   |   |   |   |   +--rw querier-dst-port?  uint32
  |   |   |   |   |   +--rw querier-src-port?  uint32
  |   |   |   |   |   +--rw ipv4!
  |   |   |   |   |   |   +--rw timestamp1!
  |   |   |   |   |   |   |   +--rw label?    uint32
  |   |   |   |   |   |   |   +--rw timestamp2!
  |   |   |   |   |   |   |   |   +--rw label?    uint32
  |   |   |   |   |   +--rw ipv6!
  |   |   |   |   |   |   +--rw timestamp1!
  |   |   |   |   |   |   |   +--rw label?    uint32
  |   |   |   |   |   |   +--rw timestamp2!
  |   |   |   |   |   |   |   +--rw label?    uint32
  |   |   +--rw delay-profile
  |   |   |   +--rw interface
  |   |   |   |   +--rw profile* [name]
  |   |   |   |   |   +--rw name          string
  |   |   |   |   |   +--rw probe!
  |   |   |   |   |   |   +--rw computation-interval?  uint16
  |   |   |   |   |   |   +--rw tx-interval?          uint32
  |   |   |   |   |   |   +--rw tos!
  |   |   |   |   |   |   |   +--rw dscp?            uint32
  |   |   |   |   |   |   |   +--rw traffic-class?    uint32
  |   |   |   |   |   |   |   +--rw protocol?        enumeration
  |   |   |   |   |   |   |   +--rw measurement-mode?  enumeration
  |   |   |   |   |   +--rw advertisement!

```

```

| | | | +---rw logging!
| | | | | +---rw delay-exceeded!
| | | | +---rw periodic!
| | | | | +---rw disabled!
| | | | | +---rw interval? uint32
| | | | | +---rw threshold? uint32
| | | | | +---rw minimum-change? uint32
| | | | +---rw accelerated!
| | | | | +---rw threshold? uint32
| | | | | +---rw minimum-change? uint32
| | | | +---rw anomaly-check
| | | | | +---rw upper-bound? uint32
| | | | | +---rw lower-bound? uint32
| | | +---rw rsvp-te
| | | | +---rw profile* [name]
| | | | | +---rw name string
| | | | | +---rw probe!
| | | | | | +---rw computation-interval? uint16
| | | | | | +---rw tx-interval? uint32
| | | | | | +---rw tos!
| | | | | | | +---rw dscp? uint32
| | | | | | | +---rw traffic-class? uint32
| | | | | | +---rw protocol? enumeration
| | | | | | +---rw measurement-mode? enumeration
| | | | +---rw advertisement!
| | | | | +---rw logging!
| | | | | | +---rw delay-exceeded!
| | | | | +---rw periodic!
| | | | | | +---rw disabled!
| | | | | | +---rw interval? uint32
| | | | | | +---rw threshold? uint32
| | | | | | +---rw minimum-change? uint32
| | | | | +---rw accelerated!
| | | | | | +---rw threshold? uint32
| | | | | | +---rw minimum-change? uint32
| | | | | +---rw threshold-check!
| | | | | | +---rw average-delay? empty
| | | | | | +---rw maximum-delay? empty
| | | | | | +---rw minimum-delay? empty
| | | +---rw sr-policy
| | | | +---rw profile* [name]
| | | | | +---rw name string
| | | | | +---rw probe!
| | | | | | +---rw computation-interval? uint16

```

```

| | | | | | +--rw tx-interval?          uint32
| | | | | | +--rw tos!
| | | | | | | +--rw dscp?          uint32
| | | | | | | +--rw traffic-class?  uint32
| | | | | | +--rw protocol?        enumeration
| | | | | | +--rw sweep!
| | | | | | | +--rw destination
| | | | | | | | +--rw ipv4?      inet:ipv4-address
| | | | | | | | +--rw range?    uint32
| | | | | | | +--rw measurement-mode?  enumeration
| | | | | +--rw advertisement!
| | | | |   +--rw logging!
| | | | |   | +--rw delay-exceeded!
| | | | |   +--rw periodic!
| | | | |   | +--rw disabled!
| | | | |   | +--rw interval?      uint32
| | | | |   | +--rw threshold?     uint32
| | | | |   | +--rw minimum-change? uint32
| | | | |   +--rw accelerated!
| | | | |   | +--rw threshold?     uint32
| | | | |   | +--rw minimum-change? uint32
| | | | |   +--rw anomaly-check
| | | | |   | +--rw upper-bound?   uint32
| | | | |   | +--rw lower-bound?   uint32
| | | | |   +--rw threshold-check!
| | | | |     +--rw average-delay?  empty
| | | | |     +--rw maximum-delay?  empty
| | | | |     +--rw minimum-delay?  empty
| | | | +--rw endpoint
| | | |   +--rw profile* [name]
| | | |     +--rw name              string
| | | |     +--rw probe!
| | | |       +--rw computation-interval?  uint16
| | | |       +--rw tx-interval?          uint32
| | | |       +--rw tos!
| | | |       | +--rw dscp?      uint32
| | | |       +--rw protocol?    enumeration
| | | |       +--rw sweep!
| | | |       | +--rw destination
| | | |       | | +--rw ipv4?    inet:ipv4-address
| | | |       | | +--rw range?  uint32
| | | |       | +--rw measurement-mode?  enumeration
| | | |       +--rw advertisement!
| | | |       +--rw logging!

```

```

| | | | +---rw delay-exceeded!
| | | | +---rw periodic!
| | | | | +---rw disabled!
| | | | | +---rw interval? uint32
| | | | | +---rw threshold? uint32
| | | | | +---rw minimum-change? uint32
| | | | +---rw accelerated!
| | | | | +---rw threshold? uint32
| | | | | +---rw minimum-change? uint32
| | | | +---rw threshold-check!
| | | | | +---rw average-delay? empty
| | | | | +---rw maximum-delay? empty
| | | | | +---rw minimum-delay? empty
| | +---rw liveness-profile
| | | +---rw sr-policy
| | | | +---rw profile* [name]
| | | | | +---rw name string
| | | | | +---rw probe!
| | | | | | +---rw tx-interval? uint32
| | | | | | +---rw tos!
| | | | | | | +---rw dscp? uint32
| | | | | | +---rw sweep!
| | | | | | +---rw destination
| | | | | | | +---rw ipv4? inet:ipv4-address
| | | | | | | +---rw range? uint32
| | | | | +---rw liveness-detection!
| | | | | | +---rw multiplier? uint32
| | | +---rw endpoint
| | | | +---rw profile* [name]
| | | | | +---rw name string
| | | | | +---rw probe!
| | | | | | +---rw tx-interval? uint32
| | | | | | +---rw tos!
| | | | | | | +---rw dscp? uint32
| | | | | +---rw liveness-detection!
| | | | | | +---rw multiplier? uint32
| | | | | +---rw logging!
| | | | | +---rw state-change!
| | +---rw loss-profile
| | | +---rw interface
| | | | +---rw profile* [name]
| | | | | +---rw name string
| | | | | +---rw probe!
| | | | | | +---rw computation-interval? uint16

```

```

| | | | | | +--rw tx-interval?          uint32
| | | | | | +--rw tos!
| | | | | |   +--rw dscp?            uint32
| | | | | |   +--rw traffic-class?   uint32
| | | | | | +--rw advertisement
| | | | | |   +--rw periodic
| | | | | |     +--rw disabled!
| | | | | |     +--rw interval?      uint16
| | | | | |     +--rw threshold?     uint8
| | | | | |     +--rw minimum-change? uint32
| | | | | |   +--rw accelerated
| | | | | |     +--rw threshold?     uint8
| | | | | |     +--rw minimum-change? uint32
| | | | +--rw sr-policy
| | | | | +--rw profile* [name]
| | | | |   +--rw name                string
| | | | |   +--rw probe!
| | | | | | +--rw computation-interval? uint16
| | | | | | +--rw tx-interval?        uint32
| | | | | | +--rw tos!
| | | | | |   +--rw dscp?            uint32
| | | | | |   +--rw traffic-class?   uint32
| | | | | | +--rw advertisement
| | | | | |   +--rw periodic
| | | | | |     +--rw disabled!
| | | | | |     +--rw interval?      uint16
| | | | | |     +--rw threshold?     uint8
| | | | | |     +--rw minimum-change? uint32
| | | | | |   +--rw accelerated
| | | | | |     +--rw threshold?     uint8
| | | | | |     +--rw minimum-change? uint32
| | | | +--rw rsvp-te
| | | | | +--rw profile* [name]
| | | | |   +--rw name                string
| | | | |   +--rw probe!
| | | | | | +--rw computation-interval? uint16
| | | | | | +--rw tx-interval?        uint32
| | | | | | +--rw tos!
| | | | | |   +--rw dscp?            uint32
| | | | | |   +--rw traffic-class?   uint32
| | | | | | +--rw advertisement
| | | | | |   +--rw periodic
| | | | | |     +--rw disabled!
| | | | | |     +--rw interval?      uint16

```



```

| | | | +---rw threshold?          uint8
| | | | +---rw minimum-change?    uint32
| | | | +---rw accelerated
| | | | +---rw threshold?          uint8
| | | | +---rw minimum-change?    uint32
| +---rw svc-profiles* [name]
| | +---rw name                    string
| | +---rw performance-measurement!
| | | +---rw protocol-profile
| | | | +---rw name?   -> /pm/pm-profiles/protocol/profile/name
| | | | +---rw delay-profile
| | | | | +---rw interface
| | | | | | +---rw default-profile? -> /pm/pm-profiles/delay-
profile/interface/profile/name
| | | | | | +---rw profile* [name]
| | | | | | +---rw name   -> /pm/pm-profiles/delay-
profile/interface/profile/name
| | | | | +---rw rsvp-te
| | | | | | +---rw default-profile? -> /pm/pm-profiles/delay-profile/rsvp-
te/profile/name
| | | | | +---rw sr-policy
| | | | | | +---rw default-profile? -> /pm/pm-profiles/delay-profile/sr-
policy/profile/name
| | | | | | +---rw profile* [name]
| | | | | | +---rw name   -> /pm/pm-profiles/delay-profile/sr-
policy/profile/name
| | | | | +---rw endpoint
| | | | | | +---rw default-profile? -> /pm/pm-profiles/delay-
profile/endpoint/profile/name
| | | | | | +---rw profile* [name]
| | | | | | +---rw name   -> /pm/pm-profiles/delay-
profile/endpoint/profile/name
| | | | +---rw liveness-profile
| | | | | +---rw sr-policy
| | | | | | +---rw default-profile? -> /pm/pm-profiles/liveness-profile/sr-
policy/profile/name
| | | | | | +---rw profile* [name]
| | | | | | +---rw name   -> /pm/pm-profiles/liveness-profile/sr-
policy/profile/name
| | | | | +---rw endpoint
| | | | | | +---rw default-profile? -> /pm/pm-profiles/liveness-
profile/endpoint/profile/name
| | | | | | +---rw profile* [name]
| | | | | | +---rw name   -> /pm/pm-profiles/liveness-
profile/endpoint/profile/name
| | | | +---rw loss-profile
| | | | +---rw interface

```

```

| | | | +--rw default-profile? -> /pm/pm-profiles/loss-
profile/interface/profile/name
| | | | +--rw profile* [name]
| | | | +--rw name -> /pm/pm-profiles/loss-
profile/interface/profile/name
| | | | +--rw rsvp-te
| | | | +--rw default-profile? -> /pm/pm-profiles/loss-profile/rsvp-
te/profile/name
| | | | +--rw profile* [name]
| | | | +--rw name -> /pm/pm-profiles/loss-profile/rsvp-
te/profile/name
| | | | +--rw sr-policy
| | | | +--rw default-profile? -> /pm/pm-profiles/loss-profile/sr-
policy/profile/name
| | | | +--rw profile* [name]
| | | | +--rw name -> /pm/pm-profiles/loss-profile/sr-
policy/profile/name
| | | +--rw devices* [name]
| | | +--rw name -> /ncs:devices/lsa-
utils:lsa/dispatch-map/device/name
| | | +--rw interface-profile-attachment!
| | | | +--rw interfaces* [if-type if-id]
| | | | +--rw if-type enumeration
| | | | +--rw if-id string
| | | | +--rw (profile-type)?
| | | | | +--:(delay)
| | | | | +--rw delay-measurement!
| | | | | +--rw (delay-profile-name)?
| | | | | | +--:(profile-name)
| | | | | | +--rw profile-name? ->
../../../../../../../../../../../../pm/pm-profiles/delay-profile/interface/profile/name
| | | | | | +--:(explicit-delay-profile)
| | | | | | +--rw explicit-delay-profile? string
| | | | | | +--rw advertise-delay? uint32
| | | | | +--rw (nexthop-address)?
| | | | | +--:(next-hop-ip-addr)
| | | | | | +--rw next-hop-ip-addr? inet:ipv4-address
| | | | | | +--:(next-hop-ipv6-addr)
| | | | | | +--rw next-hop-ipv6-addr? inet:ipv6-address
| | | +--rw custom-template* [name]
| | | | +--rw name -> /ct-info:custom-template-info/template-name
| | | | +--rw variable* [name]
| | | | | +--rw name -> deref(..../name)/../ct-info:variables
| | | | | +--rw value string
| | | | +--rw iteration* [number]
| | | | +--rw number uint16

```

```

| | |      +---rw variable* [name]
| | |      +---rw name      -> deref(..../..../name)/../ct-info:variables
| | |      +---rw value     string
| | +---rw custom-template* [name]
| |   +---rw name          -> /ct-info:custom-template-info/template-name
| |   +---rw variable* [name]
| |     | +---rw name      -> deref(..../..../name)/../ct-info:variables
| |     | +---rw value     string
| |   +---rw iteration* [number]
| |     +---rw number      uint16
| |     +---rw variable* [name]
| |       +---rw name      -> deref(..../..../name)/../ct-info:variables
| |       +---rw value     string

```

PM Service Plan Model

```

module: cisco-pm-fp
| +---ro pm-plan* [name]
| | +---ro name     string
| | +---ro plan
| |   +---ro component* [type name]
| |     | +---ro name                string
| |     | +---ro type                plan-component-type-t
| |     | +---ro state* [name]
| |     | | +---ro name              plan-state-name-t
| |     | | +---ro status?          plan-state-status-t
| |     | | +---ro when?            yang:date-and-time
| |     +---ro failed?              empty
| |     +---ro error-info!
| |       | +---ro message?         string
| |       | +---ro log-entry?      instance-identifier
| |     +---ro deleting?           empty
| |     +---ro service-location?   instance-identifier
| |     +---ro status-code-detail* [type name]
| |       +---ro type              ncs:plan-component-type-t
| |       +---ro name              string
| |       +---ro code?             string
| |       +---ro context* [context-name]
| |         | +---ro context-name    string
| |         | +---ro context-msg?   string
| |       +---ro severity?         enumeration
| |       +---ro recommended-action? string
| |       +---ro impacted-device?  string

```

NSS Yang Models

This section includes:

- NSS Service Model
- NSS Service Plan Model
- NSS Deviations Model
- NSS Augmentations Model

NSS Service Model

```

module: ietf-network-slice-service
  +--rw network-slice-services
    +--rw slo-sle-templates
      | +--rw slo-sle-template* [id]
      | | +--rw id
      | | |
      | | | cisco-sr-te-cfp-sr-types:Cisco-ios-
xr-string
      | | | +--rw template-description?
      | | | | string
      | | | +--ro cisco-nss:status
      | | | | +--ro cisco-nss:color-allocation-data
      | | | | | +--ro cisco-nss:color?
      | | | | | | uint32
      | | | | +--rw cisco-nss:qos-policy!
      | | | | | +--rw cisco-nss:L2!
      | | | | | | +--rw cisco-nss:input-policy?
      | | | | | | | -> /network-slice-services/slo-
sle-templates/qos-catalog/L2/input-qos-policy/name
      | | | | | | | +--rw cisco-nss:output-policy?
      | | | | | | | -> /network-slice-services/slo-
sle-templates/qos-catalog/L2/output-qos-policy/name
      | | | | | | | +--rw cisco-nss:L3!
      | | | | | | | +--rw cisco-nss:input-policy?
      | | | | | | | -> /network-slice-services/slo-
sle-templates/qos-catalog/L3/input-qos-policy/name
      | | | | | | | +--rw cisco-nss:output-policy?
      | | | | | | | -> /network-slice-services/slo-
sle-templates/qos-catalog/L3/output-qos-policy/name
      | | | | +--rw (forwarding)?
      | | | | | +--:(odn)
      | | | | | | +--rw cisco-nss:odn
      | | | | | | | +--rw cisco-nss:forwarding-plane-policy?
      | | | | | | | -> /cisco-sr-
te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-template/name
      | | | | | | | +--rw cisco-nss:forwarding-plane-policy-type?
      | | | | | | | | enumeration
      | | | | | +--rw cisco-nss:custom?
      | | | | | | empty
      | | | | +--rw cisco-nss:service-assurance
      | | | | | +--rw cisco-nss:heuristics!
      | | | | | | +--rw cisco-nss:monitoring-state?
      | | | | | | | aa-monitoring-state
      | | | | | | | +--rw cisco-nss:preservation?
      | | | | | | | | historical-data-options
      | | | | | | | +--rw cisco-nss:L2!
      | | | | | | | | +--rw cisco-nss:point-to-point!
      | | | | | | | | | +--rw cisco-nss:profile-name
      | | | | | | | | | | string
      | | | | | | | | | | +--rw cisco-nss:rule-name
      | | | | | | | | | | | string
      | | | | | | | | | | +--rw cisco-nss:multipoint!
      | | | | | | | | | | | +--rw cisco-nss:profile-name
      | | | | | | | | | | | | string
      | | | | | | | | | | | | +--rw cisco-nss:rule-name
      | | | | | | | | | | | | | string
      | | | | | | | +--rw cisco-nss:L3!
      | | | | | | | | +--rw cisco-nss:profile-name
      | | | | | | | | | string
      | | | | | | | | +--rw cisco-nss:rule-name
      | | | | | | | | | string
      | | | | | +--rw cisco-nss:ethernet-service-oam!
      | | | | | | +--rw cisco-nss:md-name
      | | | | | | | string
      | | | | | | | +--rw cisco-nss:md-level
      | | | | | | | | uint8

```

```

| | | +--rw cisco-nss:y-1731
| | | +--rw cisco-nss:id-type? enumeration
| | | +--rw cisco-nss:message-period? string
| | | +--rw cisco-nss:profile-delay? -> /l2vpn-ntw:l2vpn-
ntw/cisco-l2vpn-ntw:y-1731-profile/name
| | | +--rw cisco-nss:profile-loss? -> /l2vpn-ntw:l2vpn-
ntw/cisco-l2vpn-ntw:y-1731-profile/name
| | | +--rw cisco-nss:profile-synthetic-loss? -> /l2vpn-ntw:l2vpn-
ntw/cisco-l2vpn-ntw:y-1731-profile/name
| | +--rw cisco-nss:performance-measurement!
| | +--rw cisco-nss:sr-te
| | +--rw cisco-nss:pm-svc-profile -> /cisco-pm-
fp:pm/svc-profiles/name
| | +--rw (profile-type)?
| | +--:(delay)
| | | +--rw cisco-nss:delay-measurement!
| | | +--rw cisco-nss:profile? -> /cisco-pm-fp:pm/cisco-
pm-fpsvc-profiles[cisco-pm-fpname=current()/../../pm-svc-profile]/performance-
measurement/delay-profile/sr-policy/profile/name
| | +--:(liveliness)
| | +--rw cisco-nss:liveness-detection!
| | +--rw cisco-nss:profile? -> /cisco-pm-fp:pm/cisco-
pm-fpsvc-profiles[cisco-pm-fpname=current()/../../pm-svc-profile]/performance-
measurement/liveness-profile/sr-policy/profile/name
| +--rw cisco-nss:qos-catalog
| +--rw cisco-nss:L2
| | +--rw cisco-nss:output-qos-policy* [name]
| | | +--rw cisco-nss:name? string
| | | +--rw cisco-nss:description? string
| | +--rw cisco-nss:input-qos-policy* [name]
| | | +--rw cisco-nss:name? string
| | | +--rw cisco-nss:description? string
| +--rw cisco-nss:L3
| | +--rw cisco-nss:output-qos-policy* [name]
| | | +--rw cisco-nss:name? string
| | | +--rw cisco-nss:description? string
| | +--rw cisco-nss:input-qos-policy* [name]
| | | +--rw cisco-nss:name? string
| | | +--rw cisco-nss:description? string
+--rw slice-service* [service-id]
| +--rw service-id string
| +--rw service-description? string
| +--rw service-tags
| | +--rw tag-type* [tag-type]
| | | +--rw tag-type identityref
| | | +--rw value* union
| | +--rw tag-opaque* [tag-name]
| | | +--rw tag-name string
| | | +--rw value* string
| +--rw (slo-sle-policy)
| | +--:(standard)
| | | +--rw slo-sle-template? -> /network-slice-services/slo-sle-
templates/slo-sle-template/id
| | | +--rw cisco-nss:isolation? identityref
| | +--:(custom)
| | +--rw service-slo-sle-policy
| +--rw status
| | +--ro oper-status
| | | +--ro status? identityref
| | | +--ro last-change? yang:date-and-time
| +--rw sdps

```

```

| | +--rw sdp* [sdp-id]
| |   +--rw sdp-id string
| |   +--rw sdp-description? string
| |   +--rw location
| |     | +--rw altitude? int64
| |     | +--rw latitude? decimal64
| |     | +--rw longitude? decimal64
| |   +--rw node-id -> /ncs:devices/lisa-
utils:lisa/dispatch-map/device/name
| |   +--rw sdp-ip? inet:ip-address
| |   +--rw service-match-criteria
| |     | +--rw match-criterion* [index]
| |     |   +--rw index uint32
| |     |   +--rw target-connection-group-id ->
../.../.../.../connection-groups/connection-group/connection-group-id
| |     +--rw connection-group-sdp-role? identityref
| |   +--rw sdp-peering
| |     | +--rw protocol* [protocol-type]
| |     |   +--rw protocol-type identityref
| |     |   +--rw cisco-nss:bgp-attributes
| |     |     +--rw (as-type)?
| |     |       +--:(as-no-from-device)
| |     |         +--rw cisco-nss:as-no-from-device? empty
| |   +--rw attachment-circuits
| |     | +--rw attachment-circuit* [ac-id]
| |     |   +--rw ac-id string
| |     |   +--rw ac-description? string
| |     |   +--rw ac-tp-id string
| |     |   +--rw ac-ip-address inet:ip-address
| |     |   +--rw ac-ip-prefix-length uint8
| |     |   +--rw ac-tags
| |     |     | +--rw ac-tags* [ac-tag-type]
| |     |     |   +--rw ac-tag-type identityref
| |     |     |   +--rw value* union
| |     |   +--rw sdp-peering
| |     |     +--rw protocol* [protocol-type]
| |     |     +--rw protocol-type identityref
| |     |     +--rw cisco-nss:bgp-attributes!
| |     |       +--rw cisco-nss:neighbor* inet:ip-address
| |     |       +--rw cisco-nss:remote-as inet:as-number
| |     |       +--rw cisco-nss:update-source!
| |     |         | +--rw cisco-nss:if-type enumeration
| |     |         | +--rw cisco-nss:if-id string
| |     |         | +--rw cisco-nss:sub-if-id? int32
| |     |         +--rw cisco-nss:mpls-deactivation? boolean
| |     |         +--rw cisco-nss:multihop? uint8
| |     +--rw cisco-nss:custom-template* [name]
| |       +--rw cisco-nss:name? -> /ct-info:custom-template-
info/template-name
| |     | +--rw cisco-nss:variable* [name]
| |     |   +--rw cisco-nss:name? -> deref(..../name)/../ct-
info:variables
| |     |   | +--rw cisco-nss:value string
| |     |   +--rw cisco-nss:iteration* [number]
| |     |     +--rw cisco-nss:number? uint16
| |     |     +--rw cisco-nss:variable* [name]
| |     |     +--rw cisco-nss:name? -> deref(..../name)/../ct-
info:variables
| |     |     +--rw cisco-nss:value string
| |     +--rw cisco-nss:vpn-targets
| |       +--rw cisco-nss:vpn-target* [id]

```

```

| | | +--rw cisco-nss:id? string
| | | +--rw cisco-nss:route-targets* [route-target]
| | | | +--rw cisco-nss:route-target? route-distinguisher
| | | | +--rw cisco-nss:route-target-type rt-types:route-target-type
| | | +--rw cisco-nss:rd? route-distinguisher
| +--rw connection-groups
| | +--rw connection-group* [connection-group-id]
| | | +--rw connection-group-id string
| | | +--rw connectivity-type? identityref
| | | +--rw (slo-sle-policy)?
| | | | +--:(standard)
| | | | | +--rw slo-sle-template? -> /network-slice-services/slo-
sle-templates/slo-sle-template/id
| | | | +--:(custom)
| | | | | +--rw service-slo-sle-policy
| | | | | | +--rw cisco-nss:bandwidth? uint32
| | | | +--rw cisco-nss:vpn-target
| | | | | +--rw (hub-rt-choice)?
| | | | | | +--:(auto-hub-rt)
| | | | | | | +--rw cisco-nss:auto-hub-rt? empty
| | | | | | +--:(hub-rt)
| | | | | | | +--rw cisco-nss:hub-rt? route-distinguisher
| | | | | +--rw (spoke-rt-choice)?
| | | | | | +--:(auto-spoke-rt)
| | | | | | | +--rw cisco-nss:auto-spoke-rt? empty
| | | | | | +--:(spoke-rt)
| | | | | | | +--rw cisco-nss:spoke-rt? route-distinguisher
| +--rw cisco-nss:custom-template* [name]
| | +--rw cisco-nss:name? -> /ct-info:custom-template-info/template-
name
| | | +--rw cisco-nss:variable* [name]
| | | | +--rw cisco-nss:name? -> deref(..../name)/../ct-info:variables
| | | | +--rw cisco-nss:value string
| | | +--rw cisco-nss:iteration* [number]
| | | | +--rw cisco-nss:number? uint16
| | | | +--rw cisco-nss:variable* [name]
| | | | +--rw cisco-nss:name? -> deref(..../..../name)/../ct-
info:variables
| | | +--rw cisco-nss:value string
| | +--rw cisco-nss:rd? route-distinguisher
| +--rw cisco-nss:shared
| | +--rw cisco-nss:slice* [service-id]
| | | +--rw cisco-nss:service-id? -> /network-slice-services/slice-
service/service-id
| | | +--rw cisco-nss:single-sided-control? boolean
| +--rw cisco-nss:service-assurance
| | +--rw cisco-nss:heuristics!
| | | +--rw cisco-nss:monitoring-state? aa-monitoring-state
+--rw cisco-nss:actions
| +---x cisco-nss:internal-plan-change-handler
| | +---w cisco-nss:input
| | | +---w cisco-nss:kicker-id? string
| | | +---w cisco-nss:path? tailf:node-instance-identifier
| | | +---w cisco-nss:tid? uint32
| +---x cisco-nss:cleanup
| | +---w cisco-nss:input
| | | +---w cisco-nss:service -> /network-slice-services/cisco-
nss:slice-service-plan/service-id
| | | | +---w cisco-nss:sdp* -> /network-slice-services/cisco-
nss:slice-service-plan/plan/component/name
| | | | +---w cisco-nss:no-networking boolean

```

```

| | +--ro cisco-nss:output
| |   +--ro cisco-nss:success    boolean
| |   +--ro cisco-nss:detail?   string
| +---x cisco-nss:error-recovery
|   +---w cisco-nss:input
|     | +---w cisco-nss:service      -> /network-slice-services/cisco-
nss:slice-service-plan/service-id
|     | +---w cisco-nss:sync-direction  enumeration
|     +--ro cisco-nss:output
|       +--ro cisco-nss:success    boolean
|       +--ro cisco-nss:detail?   string
+--rw cisco-nss:cfp-configurations
| +--rw cisco-nss:mep-id-pool-name?  -> /ralloc:resource-pools/idalloc:id-
pool/name
| +--rw cisco-nss:color-pool-name?  -> /ralloc:resource-pools/idalloc:id-
pool/name
| +--rw cisco-nss:validation
|   +--rw cisco-nss:slo-sle-template-enabled?  boolean
|   +--rw cisco-nss:nss-enabled?              boolean
+--rw cisco-nss:global-settings
| +--rw cisco-nss:parent-rr-route-policy?  string

```

NSS Service Plan Model

```

module: ietf-network-slice-service
+--rw network-slice-services
+--ro cisco-nss:slice-service-plan* [service-id]
| +--ro cisco-nss:service-id?  string
| +--ro cisco-nss:plan
|   +--ro cisco-nss:component* [type name]
|   | +--ro cisco-nss:name?      string
|   | +--ro cisco-nss:type      plan-component-type-t
|   | +--ro cisco-nss:state* [name]
|   | | +--ro cisco-nss:name?    plan-state-name-t
|   | | +--ro cisco-nss:status?  plan-state-status-t
|   | | +--ro cisco-nss:when?    yang:date-and-time
|   | +--ro cisco-nss:status-code?  string
|   | +--ro cisco-nss:node?        string
|   | +--ro cisco-nss:attachment-circuit*  string
+--ro cisco-nss:commit-queue!
| +--ro cisco-nss:queue-item* [id]
|   +--ro cisco-nss:id?  uint64
+--ro cisco-nss:failed?      empty
+--ro cisco-nss:error-info!
| +--ro cisco-nss:message?  string
| +--ro cisco-nss:log-entry?  instance-identifier
+--ro cisco-nss:deleting?    empty
+--ro cisco-nss:service-location?  instance-identifier
+--ro cisco-nss:status
| +--ro cisco-nss:color-allocation-data
| | +--ro cisco-nss:color?  uint32
| +--ro cisco-nss:service-tag-service*  string
| +--ro cisco-nss:forwarding-plane-policy?  string
| +--ro cisco-nss:rt-allocation-data
| | +--ro cisco-nss:hub-rt?  string
| | +--ro cisco-nss:spoke-rt?  string
| +--ro cisco-nss:evi-allocation-data
| | +--ro cisco-nss:evi-id?  string
| | +--ro cisco-nss:evi-source?  string
| | +--ro cisco-nss:evi-target?  string

```



```

| | +--ro cisco-nss:mep-id-allocation-data
| | | +--ro cisco-nss:sdp* [sdp-id]
| | | | +--ro cisco-nss:sdp-id? string
| | | | +--ro cisco-nss:mep-id? uint16
| | +--ro cisco-nss:sman-id-allocation-data
| | | +--ro cisco-nss:sman-id? string
| | +--ro cisco-nss:dedicated-slice* [service-id]
| | | +--ro cisco-nss:service-id? string
| | +--ro cisco-nss:internal
| | | +--ro cisco-nss:sr-te-odn-name? string
| | | +--ro cisco-nss:pm-svc-profile-name? string
| +--ro cisco-nss:status-code-detail* [type name]
| | +--ro cisco-nss:type? ncs:plan-component-type-t
| | +--ro cisco-nss:name? string
| | +--ro cisco-nss:code? string
| | +--ro cisco-nss:context* [context-name]
| | | +--ro cisco-nss:context-name? string
| | | +--ro cisco-nss:context-msg? string
| | +--ro cisco-nss:severity? enumeration
| | +--ro cisco-nss:recommended-action? string
| | +--ro cisco-nss:impacted-device? string

```

NSS Deviations Model

```

module ietf-network-slice-service-deviations {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/ietf-network-slice-service-
  deviationsR";
  prefix ietf-nss-deviations;

  import ietf-inet-types { prefix inet; }
  import ietf-routing-types { prefix rt-types; }
  import ietf-network-slice-service { prefix ietf-nss; }
  import tailf-common { prefix tailf; }
  import tailf-kicker { prefix kicker; }
  import tailf-ncs { prefix ncs; }
  import resource-allocator { prefix ralloc; }
  import id-allocator { prefix idalloc; }
  import lsa-utils { prefix lsa-utils; }
  import custom-template-hook { prefix ct-hook; }
  import ietf-l2vpn-ntw { prefix l2vpn-ntw; }
  import cisco-sr-te-cfp-sr-types { prefix cisco-sr-te-cfp-sr-types; }

  ...
  deviation "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates/ietf-
  nss:slo-sle-template/ietf-nss:id" {
    deviate replace {
      type cisco-sr-te-cfp-sr-types:Cisco-ios-xr-string;
    }
  }
  deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service" {
    deviate add {
      must "(ietf-nss:connection-groups/ietf-nss:connection-group/ietf-
      nss:connectivity-type != 'ietf-nss:point-to-point') or count(ietf-nss:sdps/ietf-
      nss:sdp) = 2" {
        error-message "Only 2 SDPs may be defined for point-to-point";
      }
    }
  }
}

```

```

    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-id" {
        deviate replace {
            type string {
                pattern '[a-zA-Z0-9\-\_]+';
                length "1..17";
            }
        }
    }
    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-nss:slo-
sle-policy" {
        deviate add {
            mandatory true;
        }
    }
    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-tags" {
        deviate add {
            must "ietf-nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-service']"
{
                error-message "A 'service-tag-service' must be defined";
            }
        }
    }
    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups/ietf-nss:connection-group" {
        deviate add {
            min-elements 1;
            max-elements 1;
        }
    }
    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups/ietf-nss:connection-group/ietf-nss:connectivity-type" {
        deviate add {
            must "not(. = 'ietf-nss:point-to-point' and ../../../../ietf-nss:service-
tags/ietf-nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-service']/ietf-
nss:value = 'L3')" {
                error-message "connectivity-type 'point-to-point' is not supported with
service-type 'L3'";
            }
            must ". = 'vpn-common:any-to-any' or . = 'vpn-common:hub-spoke' or . = 'ietf-
nss:point-to-point'" {
                error-message "Supported topologies are: 'any-to-any', 'hub-spoke', or
'point-to-point'";
            }
        }
    }
    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-tags/ietf-nss:tag-type" {
        deviate add {
            min-elements 1;
        }
    }
    deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-tags/ietf-nss:tag-type/ietf-nss:value" {
        deviate replace {
            type union {
                type enumeration {
                    enum L3;
                    enum L2;
                }
            }
        }
    }

```

```

        type string;
    }
}
deviate add {
    min-elements 1;
    max-elements 1;

    must "(../ietf-nss:tag-type = 'ietf-nss:service-tag-service' and . = 'L3')
    or (../ietf-nss:tag-type = 'ietf-nss:service-tag-service' and . = 'L2')
    or ../ietf-nss:tag-type = 'ietf-nss:service-tag-customer'
    or ../ietf-nss:tag-type = 'ietf-nss:service-tag-opaque'" {
        error-message "'service-tag-service' must be either 'L3' or 'L2'";
    }
}
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp" {
    deviate add {
        min-elements 1;
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:node-id" {
    deviate add {
        mandatory true;
    }
    deviate replace {
        type leafref {
            path "/ncs:devices/lsa-utils:lsa/lsa-utils:dispatch-map/lsa-
utils:device/lsa-utils:name";
        }
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:service-match-criteria/ietf-nss:match-criterion" {
    deviate add {
        min-elements 1;
        max-elements 1;
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:service-match-criteria/ietf-nss:match-
criterion/ietf-nss:target-connection-group-id" {
    deviate add {
        mandatory true;
    }
    deviate replace {
        type leafref {
            path "../../../../../../../../../ietf-nss:connection-groups/ietf-nss:connection-
group/ietf-nss:connection-group-id";
        }
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:service-match-criteria/ietf-nss:match-
criterion/ietf-nss:connection-group-sdp-role" {
    deviate replace {
        default "vpn-common:hub-role";
    }
    deviate add {
        must ". = 'vpn-common:hub-role' or . = 'vpn-common:spoke-role'" {

```

```

        error-message "Valid roles are 'hub-role' or 'spoke-role'";
    }
}
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-circuit" {
    deviate add {
        min-elements 1;
        max-elements 1;
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-tp-id" {
    deviate add {
        mandatory true;
        must "not(starts-with(., 'Loopback') and (../../../../../../../../ietf-nss:service-
tags/ietf-nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-service']/ietf-
nss:value = 'L2'))" {
            error-message "ac-tp-id 'Loopback' is not supported with service-tag-
service 'L2'";
        }
    }
    deviate replace {
        type string {
            pattern "(Bundle-
Ether|Loopback|FiftyGigE|FortyGigE|FourHundredGigE|HundredGigE|TenGigE|TwentyFiveGi
gE|TwoHundredGigE|GigabitEthernet|Ethernet) [0-9]+(/[0-9]+)*";
        }
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-ip-address" {
    deviate add {
        mandatory true;
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-ip-prefix-length" {
    deviate add {
        mandatory true;
    }
    deviate replace {
        type uint8 {
            range "1..128";
        }
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-tags/ietf-nss:ac-tags" {
    deviate add {
        max-elements 1;
    }
}
deviation "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-tags/ietf-nss:ac-tags/ietf-nss:value" {
    deviate add {

```

```

    max-elements 1;
  }
  deviate replace {
    type union {
      type uint16 {
        range "1..4094";
      }
      type tailf:ip-address-and-prefix-length;
    }
  }
}
}
}

```

NSS Augmentations Model

```

module cisco-network-slice-service {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/cisco-network-slice-service";
  prefix cisco-nss;

  import ietf-inet-types { prefix inet; }
  import ietf-routing-types { prefix rt-types; }
  import ietf-network-slice-service { prefix ietf-nss; }
  import tailf-common { prefix tailf; }
  import tailf-kicker { prefix kicker; }
  import tailf-ncs { prefix ncs; }
  import resource-allocator { prefix ralloc; }
  import id-allocator { prefix idalloc; }
  import lsa-utils { prefix lsa-utils; }
  import custom-template-hook { prefix ct-hook; }
  import cisco-tdsn-core-fp-common { prefix tsdn-core-fp-common; }
  import cisco-sr-te-cfp-sr-odn { prefix cisco-sr-te-cfp-sr-odn; }
  import cisco-sr-te-cfp { prefix cisco-sr-te-cfp; }
  import ietf-l2vpn-ntw { prefix l2vpn-ntw; }
  import cisco-l2vpn-ntw { prefix cisco-l2vpn-ntw; }
  import cisco-pm-fp { prefix cisco-pm-fp; }

  ...

  /**
  Typedefs
  **/
  typedef route-distinguisher {
    type string {
      pattern
        '(0:(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2})|'
        + '6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0):(429496729[0-5]|'
        + '42949672[0-8][0-9]|'
        + '4294967[01][0-9]{2}|429496[0-6][0-9]{3})|'
        + '42949[0-5][0-9]{4}|'
        + '4294[0-8][0-9]{5}|429[0-3][0-9]{6})|'
        + '42[0-8][0-9]{7}|4[01][0-9]{8})|'
        + '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0))|'
        + '(1:(((0-9)|[1-9][0-9]|1[0-9]{2}|2[0-4][0-9])|'
        + '25[0-5])\.) {3}([0-9]|1-9)[0-9]|'
        + '1[0-9]{2}|2[0-4][0-9]|25[0-5])):(6553[0-5]|'
        + '655[0-2][0-9]|'
        + '65[0-4][0-9]{2}|6[0-4][0-9]{3})|'
        + '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0))|'

```

```

+ '(2:(429496729[0-5]|42949672[0-8][0-9]|'
+   '4294967[01][0-9]{2}|'
+   '429496[0-6][0-9]{3}|42949[0-5][0-9]{4}|'
+   '4294[0-8][0-9]{5}|'
+   '429[0-3][0-9]{6}|42[0-8][0-9]{7}|4[01][0-9]{8}|'
+   '[1-3][0-9]{9}|[1-9][0-9]{0,8}|0):'
+   '(6553[0-5]|655[0-2][0-9]|65[0-4][0-9]{2}|'
+   '6[0-4][0-9]{3}|'
+   '[1-5][0-9]{4}|[1-9][0-9]{0,3}|0)>';
tailf:info "0:2-octet-asn:4-octet-number
           1:4-octet-ipv4addr:2-octet-number
           2:4-octet-asn:2-octet-number";
}
}

/**

Groupings

**/
grouping vpn-route-targets {
  description
    "A grouping that specifies Route Target import-export rules
    used in a BGP-enabled VPN.";
  container vpn-targets {
    list vpn-target {
      key "id";
      leaf id {
        type string;
        description
          "Identifies each VPN Target";
        tailf:info
          "Identifies each VPN Target";
      }
      list route-targets {
        key "route-target";
        leaf route-target {
          type route-distinguisher;
          description
            "Route Target value. Example- 0:65100:87400024.
            Refer to IETF route-target model";
        }
        description
          "List of Route Targets.";
        tailf:info
          "List of Route Targets.";
      }
      leaf route-target-type {
        type rt-types:route-target-type;
        mandatory true;
        description
          "Import/export type of the Route Target.";
        tailf:info
          "Import/export type of the Route Target.";
      }
      description
        "VPN Target Configuration";
      tailf:info
        "VPN Target Configuration";
    }
  }
}

```

```

}
grouping color-allocation-data {
  container color-allocation-data {
    leaf color {
      tailf:info "color allocated by resource-manager id-allocator";
      description "color allocated by resource-manager id-allocator";
      type uint32;
    }
    tailf:info "color resource allocations";
    description "color resource allocations";
  }
}
grouping slice-status {
  container status {
    uses color-allocation-data;

    leaf-list service-tag-service {
      tailf:info "Service type for deployed network slice";
      description "Service type for deployed network slice";
      type string;
    }
    leaf forwarding-plane-policy {
      tailf:info "Forwarding plane policy used by network slice";
      description "Forwarding plane policy used by network slice";
      type string;
    }
  }
  container rt-allocation-data {
    leaf hub-rt {
      tailf:info "hub-rt allocated by resource-manager id-allocator";
      description "hub-rt allocated by resource-manager id-allocator";
      type string;
    }
    leaf spoke-rt {
      tailf:info "spoke-rt allocated by resource-manager id-allocator";
      description "spoke-rt allocated by resource-manager id-allocator";
      type string;
    }
    tailf:info "rt resource allocations";
    description "rt resource allocations";
  }
  container evi-allocation-data {
    leaf evi-id {
      tailf:info "evi-id allocated by resource-manager id-allocator";
      description "evi-id allocated by resource-manager id-allocator";
      type string;
    }
    leaf evi-source {
      tailf:info "evi-source allocated by resource-manager id-allocator";
      description "evi-source allocated by resource-manager id-allocator";
      type string;
    }
    leaf evi-target {
      tailf:info "evi-target allocated by resource-manager id-allocator";
      description "evi-target allocated by resource-manager id-allocator";
      type string;
    }
    tailf:info "EVI resource allocations";
    description "EVI resource allocations";
  }
  container mep-id-allocation-data {
    list sdp {

```

```

    key "sdp-id";

    leaf sdp-id {
      tailf:info "Unique identifier for the referred slice service SDP.";
      description "Unique identifier for the referred slice service SDP.";
      type string;
    }
    leaf mep-id {
      tailf:info "mep-id used by Y.1731 allocated by resource-manager id-
allocator";
      description "mep-id used by Y.1731 allocated by resource-manager id-
allocator";
      type uint16 {
        range "1..8191";
      }
    }
  }
}
container sman-id-allocation-data {
  tailf:hidden full;
  leaf sman-id {
    type string;
  }
}
list dedicated-slice {
  key "service-id";

  leaf service-id {
    type string;
  }
}
container internal {
  tailf:hidden full;
  leaf sr-te-odn-name {
    type string;
  }
  leaf pm-svc-profile-name {
    type string;
  }
}
}
}
grouping qos-catalog {
  list output-qos-policy {
    tailf:info "Egress QoS Policy";
    description "Egress QoS Policy";
    key name;

    leaf name {
      tailf:info "Policy Name";
      description "Policy Name";
      type string;
    }

    leaf description {
      tailf:info "Policy Description";
      description "Policy Description";
      type string;
    }
  }
}
}

```



```

list input-qos-policy {
  tailf:info "Ingress QoS Policy";
  description "Ingress QoS Policy";
  key name;

  leaf name {
    tailf:info "Policy Name";
    description "Policy Name";
    type string;
  }

  leaf description {
    tailf:info "Policy Description";
    description "Policy Description";
    type string;
  }
}
}
grouping L2-qos-policy {
  container L2 {
    presence true;

    leaf input-policy {
      tailf:info "Ingress Policy";
      description "Ingress Policy";
      type leafref {
        path "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates/qos-
catalog/L2/input-qos-policy/name";
      }
    }
    leaf output-policy {
      tailf:info "Egress Policy";
      description "Egress Policy";
      type leafref {
        path "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates/qos-
catalog/L2/output-qos-policy/name";
      }
    }
  }
}
grouping L3-qos-policy {
  container L3 {
    presence true;

    leaf input-policy {
      tailf:info "Ingress Policy";
      description "Ingress Policy";
      type leafref {
        path "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates/qos-
catalog/L3/input-qos-policy/name";
      }
    }
    leaf output-policy {
      tailf:info "Egress Policy";
      description "Egress Policy";
      type leafref {
        path "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates/qos-
catalog/L3/output-qos-policy/name";
      }
    }
  }
}
}

```

```

}
grouping L2-service-assurance-profile-and-rule {
  uses tsdn-core-fp-common:service-assurance-profile-and-rule {
    refine profile-name {
      description "Please input profile name in following format: '<profile-name>
(custom|system)'"
      The possible profile names for system are:
Gold_L2VPN_ConfigProfile, Silver_L2VPN_ConfigProfile.";
      tailf:info "Please input profile name in following format: '<profile-name>
(custom|system)'"
      The possible profile names for system are:
Gold_L2VPN_ConfigProfile, Silver_L2VPN_ConfigProfile.";
    }
    refine rule-name {
      description "Please input rule name in following format: '<rule-name>
(custom|system)'"
      The possible rule names for system are: Rule-L2VPN-NM, Rule-
L2VPN-NM-Basic, Rule-L2VPN-NM-P2P, Rule-L2VPN-NM-P2P-Basic.";
      tailf:info "Please input rule name in following format: '<rule-name>
(custom|system)'"
      The possible rule names for system are: Rule-L2VPN-NM, Rule-
L2VPN-NM-Basic, Rule-L2VPN-NM-P2P, Rule-L2VPN-NM-P2P-Basic.";
    }
  }
}
}
/**

Augmentations

**/
augment "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates" {
  container qos-catalog {
    container L2 {
      uses qos-catalog;
    }
    container L3 {
      uses qos-catalog;
    }
  }
}
augment "/ietf-nss:network-slice-services/ietf-nss:slo-sle-templates/ietf-
nss:slo-sle-template" {
  container status {
    config false;
    tailf:cdb-oper {
      tailf:persistent true;
    }
    uses color-allocation-data;
  }
  container qos-policy {
    presence true;

    tailf:info
      "QoS Configuration";
    description
      "QoS Configuration";

    uses L2-qos-policy;
    uses L3-qos-policy;
  }
}

```

```

choice forwarding {
  case odn {
    container odn {
      tailf:info "SR-TE ODN Template";
      description "SR-TE ODN Template";

      leaf forwarding-plane-policy {
        tailf:info "SR-TE ODN Template Reference";
        description "SR-TE ODN Template Reference";
        type leafref {
          path "/cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/cisco-sr-te-
cfp-sr-odn:odn-template/cisco-sr-te-cfp-sr-odn:name";
        }
      }
      leaf forwarding-plane-policy-type {
        when "../forwarding-plane-policy";
        tailf:info "Create Custom Forwarding Plane Policy Per SLO SLE
Template";
        description "Create Custom Forwarding Plane Policy Per SLO SLE
Template";
        type enumeration {
          enum "as-is";
          enum "as-blueprint";
        }
        default "as-is";
      }
    }
  }
  default odn;
}

leaf custom {
  when "../odn/forwarding-plane-policy-type = 'as-blueprint'";
  tailf:info "Create Custom Forwarding Plane Policy Per Slice";
  description "Create Custom Forwarding Plane Policy Per Slice";
  type empty;
}

container service-assurance {
  container heuristics {
    when "/tsdn-core-fp-common:enable-service-assurance = 'true'";
    presence true;

    uses tsdn-core-fp-common:service-assurance-monitoring;
    uses tsdn-core-fp-common:service-assurance-preservation;

    container L2 {
      tailf:info "Heuristic Configuration for L2VPN's";
      description "Heuristic Configuration for L2VPN's";

      presence true;

      container point-to-point {
        presence true;
        uses L2-service-assurance-profile-and-rule;
      }
      container multipoint {
        presence true;
        uses L2-service-assurance-profile-and-rule;
      }
      must "point-to-point or multipoint" {

```

```

        error-message "Either 'point-to-point', 'multipoint', or both must be
configured";
    }
}
container L3 {
    tailf:info "Heuristic Configuration for L3VPN's";
    description "Heuristic Configuration for L3VPN's";

    presence true;
    uses tsdn-core-fp-common:service-assurance-profile-and-rule {
        refine profile-name {
            description "Please input profile name in following format:
'<profile-name> (custom|system)'"
            The possible profile names for system are:
Gold L3VPN ConfigProfile, Silver L3VPN ConfigProfile.";
            tailf:info "Please input profile name in following format: '<profile-
name> (custom|system)'"
            The possible profile names for system are:
Gold L3VPN ConfigProfile, Silver_L3VPN_ConfigProfile.";
        }
        refine rule-name {
            description "Please input rule name in following format: '<rule-name>
(custom|system)'"
            The possible rule names for system are: Rule-L3VPN-NM,
Rule-L3VPN-NM-Basic.";
            tailf:info "Please input rule name in following format: '<rule-name>
(custom|system)'"
            The possible rule names for system are: Rule-L3VPN-NM,
Rule-L3VPN-NM-Basic.";
        }
    }
}
}
must "L2 or L3" {
    error-message "L2 or L3 must be defined when heuristics is enabled";
}
}
container ethernet-service-oam {
    presence true;

    leaf md-name {
        type string;
        description "Maintenance domain name";
        tailf:info "Maintenance domain name";
        mandatory true;
    }
    leaf md-level {
        type uint8;
        description "Maintenance domain level";
        tailf:info "Maintenance domain level";
        mandatory true;
    }
}
container y-1731 {
    leaf id-type {
        type enumeration {
            enum icc-based;
            enum number;
        }
        default icc-based;
        tailf:info "SMAN ID Type";
        description "SMAN ID Type";
    }
}

```

```

leaf message-period {
  type string {
    pattern '(3\.3ms|10ms|100ms|1s|10s|1m|10m)';
  }
  default "1s";
  description
    "Defines the interval between OAM messages. The message
    period is expressed in milliseconds";
  tailf:info "Defines the interval between OAM messages. The message
    period is expressed in 3.3ms|10ms|100ms|1s|10s|1m|10m";
}
leaf profile-delay {
  type leafref {
    path "/l2vpn-ntw:l2vpn-ntw/cisco-l2vpn-ntw:y-1731-profile/cisco-
l2vpn-ntw:name";
  }
  description "Reference to delay type y-1731-profile";
  tailf:info "Reference to delay type y-1731-profile";
}
leaf profile-loss {
  type leafref {
    path "/l2vpn-ntw:l2vpn-ntw/cisco-l2vpn-ntw:y-1731-profile/cisco-
l2vpn-ntw:name";
  }
  description "Reference to loss type y-1731-profile";
  tailf:info "Reference to loss type y-1731-profile";
}
leaf profile-synthetic-loss {
  type leafref {
    path "/l2vpn-ntw:l2vpn-ntw/cisco-l2vpn-ntw:y-1731-profile/cisco-
l2vpn-ntw:name";
  }
  description "Reference to synthetic-loss type y-1731-profile";
  tailf:info "Reference to synthetic-loss type y-1731-profile";
}
}
}
container performance-measurement {
  when "../odn/forwarding-plane-policy-type = 'as-blueprint'";
  presence true;

  container sr-te {
    leaf pm-svc-profile {
      tailf:info "PM Service Name";
      description "PM Service Name";
      type leafref {
        path "/cisco-pm-fp:pm/cisco-pm-fp:svc-profiles/cisco-pm-fp:name";
      }
      mandatory true;
    }
    choice profile-type {
      case delay {
        container delay-measurement {
          presence true;
          leaf profile {
            type leafref {
              path "/cisco-pm-fp:pm/cisco-pm-fp:svc-profiles[cisco-pm-
fp:name=current()/../pm-svc-profile]/cisco-pm-fp:performance-measurement/cisco-
pm-fp:delay-profile/cisco-pm-fp:sr-policy/cisco-pm-fp:profile/cisco-pm-fp:name";
            }
            description "Reference to delay type SR PM Profile";
          }
        }
      }
    }
  }
}

```



```

}
tailf:action cleanup {
  tailf:actionpoint ietf-nss-cleanup;
  tailf:info "Clean up IETF NSS service configuration and operational data";
  description "Clean up IETF NSS service configuration and operational data";
  input {
    leaf service {
      tailf:info "Service to recover";
      description "Service to recover";
      type leafref {
        path "/ietf-nss:network-slice-services/cisco-nss:slice-service-
plan/cisco-nss:service-id";
      }
      mandatory true;
    }
    leaf-list sdp {
      tailf:info "Specific SDPs to recover";
      description "Specific SDPs to recover";
      type leafref {
        path "/ietf-nss:network-slice-services/cisco-nss:slice-service-
plan/cisco-nss:plan/cisco-nss:component/cisco-nss:name";
      }
      must "/ietf-nss:network-slice-services/cisco-nss:slice-service-
plan[service-id=current()/../service]/cisco-nss:plan/cisco-
nss:component[name=current()]/cisco-nss:type = 'ietf-nss-nano:sdp'" {
        error-message "Only SDPs can be chosen";
      }
    }
    leaf no-networking {
      tailf:info "Do not send any data to the devices.";
      description "Do not send any data to the devices.";
      type boolean;
      mandatory true;
    }
  }
  output {
    leaf success {
      type boolean;
      mandatory true;
    }
    leaf detail {
      type string;
    }
  }
  tailf:confirm-text "#####\n" +
"#          Warning          #\n" +
"#####\n" +
"You are about to forcefully cleanup a T-SDN service.\n" +
"This will affect the deploying service and leave network device(s) & NSO
out-of-sync (for no-networking=true).\n" +
"Are you sure you want to proceed?" {
  tailf:confirm-default false;
}
}
tailf:action error-recovery {
  tailf:actionpoint ietf-nss-fp-error-recovery;
  tailf:info "Action to recover a service during create/update/delete from a
transient failures
like device connection issues once the device is reachable
again.
For create/update failures, this action issues a sync on the

```

```

device
    & redeploys the service with reconcile option.
    For delete failures, this action issues a sync on the device &
    redeploys zombie services.";
    description "Action to recover a service during create/update/delete from a
    transient failures
    again.
    like device connection issues once the device is reachable
    device
    For create/update failures, this action issues a sync on the
    device
    & redeploys the service with reconcile option.
    For delete failures, this action issues a sync on the device &
    redeploys zombie services.";
    input {
        leaf service {
            tailf:info "Service to recover";
            description "Service to recover";
            type leafref {
                path "/ietf-nss:network-slice-services/cisco-nss:slice-service-
                plan/cisco-nss:service-id";
            }
            mandatory true;
        }
        leaf sync-direction {
            tailf:info "sync device with sync-from or sync-to";
            description "sync device with sync-from or sync-to";
            type enumeration {
                enum sync-from;
                enum sync-to;
            }
            mandatory true;
        }
    }
    output {
        leaf success {
            type boolean;
            mandatory true;
        }
        leaf detail {
            type string;
        }
    }
    tailf:confirm-text "#####\n" +
    "#      Warning      #\n" +
    "#####\n" +
    "You are about to recover a T-SDN service.\n" +
    "This will issue a sync on the device.\n" +
    "Are you sure you want to proceed?" {
        tailf:confirm-default false;
    }
}
}
container cfp-configurations {
    leaf mep-id-pool-name {
        tailf:info "MEP ID pool";
        description "MEP ID pool";
        type leafref {
            path "/ralloc:resource-pools/idalloc:id-pool/idalloc:name";
        }
    }
}
leaf color-pool-name {

```



```

    tailf:info "Color pool";
    description "Color pool";
    type leafref {
      path "/ralloc:resource-pools/idalloc:id-pool/idalloc:name";
    }
  }
  container validation {
    tailf:hidden tsdn;
    leaf slo-sle-template-enabled {
      tailf:info "If set to true, following validations are done for slo-sle-
template:
          1. Check if status-codes are loaded properly
          2. Validates referenced Y.1731 profiles types are correct

          Validations regarding restricted service updates will always
run.";

      description "If set to true, following validations are done for slo-sle-
template:
          1. Check if status-codes are loaded properly
          2. Validates referenced Y.1731 profiles type are correct

          Validations regarding restricted service updates will always
run.";

      type boolean;
      default true;
    }
    leaf nss-enabled {
      tailf:info "If set to true, following validations are done for slice-
service:
          1. Check if status-codes are loaded properly
          2. Check if 'parent-rr-route-policy' is configured for L2
point-to-point slices
          3. Check if resource-pools are configured
          4. Validate that 'connection-group-sdp-role' is the same
for all SDPs using the same 'node-id'
          5. Validate that the 'rd' is the same for all SDPs using
the same 'node-id'

          Validations regarding restricted service updates will always
run.";

      description "If set to true, following validations are done for slice-
service:
          1. Check if status-codes are loaded properly
          2. Check if 'parent-rr-route-policy' is configured for L2
point-to-point slices
          3. Check if resource-pools are configured
          4. Validate that 'connection-group-sdp-role' is the same
for all SDPs using the same 'node-id'
          5. Validate that the 'rd' is the same for all SDPs using
the same 'node-id'

          Validations regarding restricted service updates will always
run.";

      type boolean;
      default true;
    }
  }
}
}
container global-settings {

```

```

leaf parent-rr-route-policy {
  type string;
  tailf:info "Parent RR Route Policy to apply local route policies";
  description "Parent RR Route Policy to apply local route policies";
}
}
augment "/ietf-nss:network-slice-services/ietf-nss:slice-service" {
  uses ct-hook:template-hook;

  leaf rd {
    when "boolean(..../ietf-nss:service-tags/ietf-nss:tag-type[ietf-nss:tag-type =
'ietf-nss:service-tag-service']) and ../ietf-nss:service-tags/ietf-nss:tag-
type[ietf-nss:tag-type = 'ietf-nss:service-tag-service']/ietf-nss:value = 'L3'";
    description "Route distinguisher value";
    tailf:info "Route distinguisher value";
    type route-distinguisher;
  }
  container shared {
    when "derived-from-or-self(..../isolation, 'ietf-nss:service-isolation-
dedicated')";

    tailf:info "Shared Slice Participation";
    description "Shared Slice Participation";
    list slice {
      tailf:info "Shared Slice";
      description "Shared Slice";

      key service-id;

      leaf service-id {
        tailf:info "Shared Slice Reference";
        description "Shared Slice Reference";
        type leafref {
          path "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-id";
        }
        must "derived-from-or-self(/ietf-nss:network-slice-services/ietf-
nss:slice-service[ietf-nss:service-id=current()]/isolation, 'ietf-nss:service-
isolation-shared')" {
          error-message "slice-service must be using isolation type 'ietf-
nss:service-isolation-shared'";
        }
      }
    }
  }
  leaf single-sided-control {
    tailf:info "Single-Sided Slice Control Point for import recoloring";
    description "Single-Sided Slice Control Point for import recoloring";
    type boolean;
    default true;
  }
}
container service-assurance {
  container heuristics {
    when "/tsdn-core-fp-common:enable-service-assurance = 'true'";
    presence true;

    uses tsdn-core-fp-common:service-assurance-monitoring;
  }
}
}
}

```

```

    augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-nss:slo-
sle-policy/ietf-nss:standard" {
    leaf isolation {
        when "../ietf-nss:connection-groups/ietf-nss:connection-group/ietf-
nss:connectivity-type != 'ietf-nss:point-to-point'";
        type identityref {
            base ietf-nss:service-isolation-type;
        }
        default "ietf-nss:service-isolation-dedicated";
        tailf:info "The IETF Network Slice isolation SLE requirement.";
        description "The IETF Network Slice isolation SLE requirement.";

        must "derived-from-or-self(., 'ietf-nss:service-isolation-shared') and
../ietf-nss:connection-groups/ietf-nss:connection-group[1]/ietf-nss:connectivity-
type = 'vpn-common:any-to-any'
        or
        derived-from-or-self(., 'ietf-nss:service-isolation-dedicated')" {
            tailf:dependency "../ietf-nss:connection-groups/ietf-nss:connection-
group/ietf-nss:connectivity-type";
            error-message "Connectivity-type, 'any-to-any', is only supported with
'service-isolation-shared' isolation";
        }
    }
}
augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups/ietf-nss:connection-group" {
    container vpn-target {
        when "../ietf-nss:connectivity-type != 'ietf-nss:point-to-point'";

        choice hub-rt-choice {
            default "auto-hub-rt";
            leaf auto-hub-rt {
                tailf:info "Auto assigned Route Targets";
                description "Auto assigned Route Targets";
                type empty;
            }
            leaf hub-rt {
                tailf:info "Manual assigned Route Targets";
                description "Manual assigned Route Targets";
                type route-distinguisher;
            }
        }
        choice spoke-rt-choice {
            when "../ietf-nss:connectivity-type = 'vpn-common:hub-spoke'";
            default "auto-spoke-rt";
            leaf auto-spoke-rt {
                tailf:info "Auto assigned Route Targets";
                description "Auto assigned Route Targets";
                type empty;
            }
            leaf spoke-rt {
                tailf:info "Manual assigned Route Targets";
                description "Manual assigned Route Targets";
                type route-distinguisher;
            }
        }
    }
}
augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups/ietf-nss:connection-group/ietf-nss:slo-sle-policy/ietf-
nss:custom/ietf-nss:service-slo-sle-policy" {

```

```

leaf bandwidth {
  when "deref(..../..../ietf-nss:slo-sle-template)/../cisco-nss:custom" {
    tailf:dependency "..../..../ietf-nss:slo-sle-template";
  }
  tailf:info "<1-4294967295> Requested Bandwidth value in kbps. Cannot be
configured if slo-sle-template policy-type is as-blueprint with customization and
ODN with SRV6";
  description "<1-4294967295> Requested Bandwidth value in kbps. Cannot be
configured if slo-sle-template policy-type is as-blueprint with customization and
ODN with SRV6";
  type uint32 {
    range "1..4294967295";
  }
}
}
augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp" {
  uses ct-hook:template-hook;
  uses vpn-route-targets {
    when "deref(ietf-nss:service-match-criteria/ietf-nss:match-criterion[1]/ietf-
nss:target-connection-group-id)/../ietf-nss:connectivity-type != 'ietf-nss:point-
to-point'" {
      tailf:dependency "../ietf-nss:service-match-criteria/ietf-nss:match-
criterion/ietf-nss:target-connection-group-id";
    }
  }
  leaf rd {
    when "boolean(..../..../ietf-nss:service-tags/ietf-nss:tag-type[ietf-nss:tag-
type = 'ietf-nss:service-tag-service']) and ../..../ietf-nss:service-tags/ietf-
nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-service']/ietf-nss:value =
'L3'";
    description "Route distinguisher value.";
    tailf:info "Route distinguisher value.";
    type route-distinguisher;
    tailf:default-ref "../..../rd";
  }
}
augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:sdp-peering/ietf-nss:protocol" {
  container bgp-attributes {
    when "derived-from-or-self(..../ietf-nss:protocol-type, 'ietf-nss:peering-
protocol-bgp')";

    choice as-type {
      default as-no-from-device;

      leaf as-no-from-device {
        tailf:info "Autonomous Number learned from device";
        description "Autonomous Number learned from device";
        type empty;
      }
    }
  }
}
// TODO: Address with QoS Override
// augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-circuit" {
//   container qos-policy {
//     presence true;

//     tailf:info

```

```

//      "QoS Configuration";
//      description
//      "QoS Configuration";

//      uses L2-qos-policy {
//          when "../../../../../../../ietf-nss:service-tags/ietf-nss:tag-type[ietf-
nss:tag-type = 'ietf-nss:service-tag-service']/ietf-nss:value = 'L2'";
//      }
//      uses L3-qos-policy {
//          when "../../../../../../../ietf-nss:service-tags/ietf-nss:tag-type[ietf-
nss:tag-type = 'ietf-nss:service-tag-service']/ietf-nss:value = 'L3'";
//      }
//  }
// }
augment "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:sdp-peering/ietf-nss:protocol" {
    container bgp-attributes {
        when "derived-from-or-self(../ietf-nss:protocol-type, 'ietf-nss:peering-
protocol-bgp')";
        presence true;

        leaf-list neighbor {
            tailf:info "IP address of the BGP neighbor. An IPv4 or IPv6 neighbor may be
indicated.";
            description "IP address of the BGP neighbor. An IPv4 or IPv6 neighbor may
be indicated.";
            min-elements 1;
            max-elements 1;
            type inet:ip-address;
            must "(contains(., ':') and contains(../../../../ietf-nss:ac-ip-address,
':')) or
                (contains(., '.') and contains(../../../../ietf-nss:ac-ip-address,
'.'))" {
                error-message "Neighbor address type must match attachment-circuit ac-ip-
address type";
            }
        }
        leaf remote-as {
            tailf:info "Customer AS number in case the customer requests BGP routing.";
            description "Customer AS number in case the customer requests BGP
routing.";
            type inet:as-number;
            mandatory true;
        }
        container update-source {
            presence true;
            tailf:info "Source of routing updates";
            description "Source of routing updates";
            leaf if-type {
                when "not(contains(/ncs:devices/lisa-utils:lisa-utils:dispatch-map/lisa-
utils:device
                [lisa-utils:name=current()/../../../../../../../../ietf-nss:node-
id]/lisa-utils:ned-id,'cisco-ios-cli-'))";

                tailf:info "Interface Type";
                description "Interface Type";
                type enumeration {
                    enum Bundle-Ether;
                    enum Ethernet;
                    enum FiftyGigE;

```

```

        enum FortyGigE;
        enum FourHundredGigE;
        enum GigabitEthernet;
        enum HundredGigE;
        enum Loopback;
        enum TenGigE;
        enum TwentyFiveGigE;
        enum TwoHundredGigE;
    }
    mandatory true;
}

leaf if-id {
    when "not(contains(/ncs:devices/lisa-utils:lisa/lisa-utils:dispatch-map/lisa-
utils:device
        [lisa-utils:name=current()/../../../../../../../../../../../../ietf-nss:node-
id]/lisa-utils:ned-id,'cisco-ios-cli-'))";
    tailf:info "Interface Id";
    description "Interface Id";
    type string {
        pattern '[0-9]+(/[0-9]+)*';
    }
    mandatory true;
}

leaf sub-if-id {
    when "not(contains(/ncs:devices/lisa-utils:lisa/lisa-utils:dispatch-map/lisa-
utils:device
        [lisa-utils:name=current()/../../../../../../../../../../../../ietf-nss:node-
id]/lisa-utils:ned-id,'cisco-ios-cli-'))
        or ../if-type != 'Loopback'";
    tailf:info "Sub Interface Id";
    description "Sub Interface Id";
    type int32;
}

must "not(if-type) or (((if-type!='Loopback' and if-type!='Bundle-Ether')
or " +
    " ((if-type='Loopback' or if-type='Bundle-Ether')
    and (string(number(if-id))!='NaN'))))" {
    error-message "Id must be a number";
}
must "not(if-type) or (((if-type!='Bundle-Ether') or (if-type='Bundle-
Ether')
    and (string(number(if-id))>'0')
    and (string(number(if-id))<'65536')))" {
    error-message "if-id must be in the range 1-65535";
}
must "not(if-type) or (((if-type!='Loopback') or (if-type='Loopback')
    and (string(number(if-id))>='0')
    and (string(number(if-id))<='2147483647')))" {
    error-message "if-id must be in the range 0-2147483647";
}
}
leaf mpls-deactivation {
    tailf:info "Set to true to disable BGP MPLS forwarding.";
    description "Set to true to disable BGP MPLS forwarding.";
    type boolean;
    default "false";
}
leaf multihop {

```

```

        type uint8;
        tailf:info "Describes the number of hops allowed between a given BGP
neighbor and the PE router.";
        description "Describes the number of hops allowed between a given BGP
neighbor and the PE router.";
    }
}
}

/**

Annotations

**/
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:slo-sle-policy/ietf-nss:custom/ietf-nss:service-slo-sle-policy" {
    when "'supported' = 'false'";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups/ietf-nss:connection-group/ietf-nss:slo-sle-policy/ietf-
nss:standard/ietf-nss:slo-sle-template" {
    when "'supported' = 'false'";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:sdp-peering" {
    when "boolean(..../..../ietf-nss:service-tags/ietf-nss:tag-type[ietf-nss:tag-
type = 'ietf-nss:service-tag-service']) and ..../..../ietf-nss:service-tags/ietf-
nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-service']/ietf-nss:value =
'L3'";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:sdp-peering" {
    when "boolean(..../..../..../ietf-nss:service-tags/ietf-nss:tag-type[ietf-
nss:tag-type = 'ietf-nss:service-tag-service']) and ..../..../..../ietf-
nss:service-tags/ietf-nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-
service']/ietf-nss:value = 'L3'";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-ip-prefix-length" {
    when "boolean(..../..../..../..../ietf-nss:service-tags/ietf-nss:tag-type[ietf-
nss:tag-type = 'ietf-nss:service-tag-service']) and ..../..../..../..../ietf-
nss:service-tags/ietf-nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-
service']/ietf-nss:value = 'L3'";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp/ietf-nss:attachment-circuits/ietf-nss:attachment-
circuit/ietf-nss:ac-ip-address" {
    when "boolean(..../..../..../..../..../ietf-nss:service-tags/ietf-nss:tag-type[ietf-
nss:tag-type = 'ietf-nss:service-tag-service']) and ..../..../..../..../..../ietf-
nss:service-tags/ietf-nss:tag-type[ietf-nss:tag-type = 'ietf-nss:service-tag-
service']/ietf-nss:value = 'L3'";
}

tailf:annotate-module "ietf-network-slice-service" {
    tailf:annotate-statement grouping[name='service-match-criteria'] {
        tailf:annotate-statement container[name='service-match-criteria'] {
            tailf:info "Describes the slice service match criteria.";

            tailf:annotate-statement list[name='match-criterion'] {

```

```

tailf:info "List of the slice service traffic match criteria.";

tailf:annotate-statement leaf[name='index'] {
    tailf:info "The entry index.";
}
tailf:annotate-statement leaf[name='target-connection-group-id'] {
    tailf:info "reference to a slice service connection group.";
}
tailf:annotate-statement leaf[name='connection-group-sdp-role'] {
    tailf:info "Indicates the role in the connection group when a slice
service has multiple multipoint-to-multipoint connection groups, e.g., hub-spoke.";

    when "deref(..ietf-nss:target-connection-group-id)/../ietf-
nss:connectivity-type = 'vpn-common:hub-spoke'" {
        tailf:dependency "..ietf-nss:target-connection-group-id";
        tailf:dependency "..../..../..../ietf-nss:connection-groups/ietf-
nss:connection-group/ietf-nss:connection-group-id";
        tailf:dependency "..../..../..../ietf-nss:connection-groups/ietf-
nss:connection-group/ietf-nss:connectivity-type";
    }
}
}
}
}
tailf:annotate-statement grouping[name='sdp-peering'] {
    tailf:annotate-statement container[name='sdp-peering'] {
        tailf:info "Describes SDP peering attributes.";

        tailf:annotate-statement list[name='protocol'] {
            tailf:info "List of the SDP peering protocol.";

            tailf:annotate-statement leaf[name='protocol-type'] {
                tailf:info "Identifies an entry in the list of SDP peering protocol
type.";
            }
        }
    }
}
tailf:annotate-statement grouping[name='sdp-attachment-circuits'] {
    tailf:annotate-statement container[name='attachment-circuits'] {
        tailf:info "List of attachment circuit.";

        tailf:annotate-statement list[name='attachment-circuit'] {
            tailf:info "The IETF Network Slice service SDP attachment circuit related
parameters.";

            tailf:annotate-statement leaf[name='ac-id'] {
                tailf:info "Uniquely identifier a attachment circuit.";
            }
            tailf:annotate-statement leaf[name='ac-description'] {
                tailf:info "The attachment circuit description.";
            }
            tailf:annotate-statement leaf[name='ac-tp-id'] {
                tailf:info "The termination port ID of the attachment circuit.";
            }
            tailf:annotate-statement leaf[name='ac-ip-address'] {
                tailf:info "The IP address of the attachment circuit.";
            }
            tailf:annotate-statement leaf[name='ac-ip-prefix-length'] {
                tailf:info "The subnet prefix length expressed in bits.";
            }
        }
    }
}

```



```

tailf:annotate-statement container[name='ac-tags'] {
  tailf:info "Container for the attachment circuit tags.";

  tailf:annotate-statement list[name='ac-tags'] {
    tailf:info "The attachment circuit tags list.";

    tailf:annotate-statement leaf[name='ac-tag-type'] {
      tailf:info "The attachment circuit tag type.";
    }
    tailf:annotate-statement leaf-list[name='value'] {
      tailf:info "The attachment circuit tag value.";
    }
  }
}
}
}
}
tailf:annotate-statement grouping[name='geolocation-container'] {
  tailf:annotate-statement container[name='location'] {
    tailf:info "A container containing a GPS location.";

    tailf:annotate-statement leaf[name='altitude'] {
      tailf:info "Distance above the sea level.";
    }
    tailf:annotate-statement leaf[name='latitude'] {
      tailf:info "Relative position north or south on the Earth's surface.";
    }
    tailf:annotate-statement leaf[name='longitude'] {
      tailf:info "Angular distance east or west on the Earth's surface.";
    }
  }
}
}
tailf:annotate-statement grouping[name='sdp'] {
  tailf:annotate-statement leaf[name='sdp-id'] {
    tailf:info "Unique identifier for the referred slice service SDP.";
  }
  tailf:annotate-statement leaf[name='sdp-description'] {
    tailf:info "Give more description of the SDP.";
  }
  tailf:annotate-statement leaf[name='node-id'] {
    tailf:info "Uniquely identifies an edge node of the SDP.";
  }
  tailf:annotate-statement leaf[name='sdp-ip'] {
    tailf:info "The IP address of the SDP.";
  }
}
}
tailf:annotate-statement grouping[name='connection-group'] {
  tailf:annotate-statement leaf[name='connection-group-id'] {
    tailf:info "The connection group identifier.";
  }
  tailf:annotate-statement leaf[name='connectivity-type'] {
    tailf:info "Connection group connectivity type.";
  }
}
}
tailf:annotate-statement grouping[name='slice-service-template'] {
  tailf:annotate-statement container[name='slo-sle-templates'] {
    tailf:info "Contains a set of slice service templates.";

    tailf:annotate-statement list[name='slo-sle-template'] {
      tailf:info "List for SLO and SLE template identifiers.";
    }
  }
}
}

```

```

        tailf:annotate-statement leaf[name='id'] {
            tailf:info "Identification of the Service Level Objective (SLO) and
Service Level Expectation (SLE) template to be used. Local administration
meaning.";
        }
        tailf:annotate-statement leaf[name='template-description'] {
            tailf:info "Description of the SLO and SLE policy template.";
        }
    }
}
tailf:annotate-statement grouping[name='service-slo-sle-policy'] {
    tailf:annotate-statement choice[name='slo-sle-policy'] {
        tailf:annotate-statement case[name='standard'] {
            tailf:annotate-statement leaf[name='slo-sle-template'] {
                tailf:info "Standard SLO and SLE template to be used.";
            }
        }
    }
}
tailf:annotate "/ietf-nss:network-slice-services" {
    tailf:info "Contains a list of IETF network slice services";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service" {
    tailf:info "A slice service is identified by a service-id.";
    tailf:validate "ietf-nss-validation" {
        tailf:dependency ".";
    }
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-id" {
    tailf:info "A unique slice service identifier.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-description" {
    tailf:info "Textual description of the slice service.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-tags" {
    tailf:info "Container for the list of service tags.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-tags/ietf-nss:tag-type" {
    tailf:info "Slice service tag list.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:service-tags/ietf-nss:tag-type/ietf-nss:value" {
    tailf:info "Slice service tag value.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps" {
    tailf:info "Slice service SDPs.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:sdps/ietf-nss:sdp" {
    tailf:info "List of SDPs in this slice service.";
}
tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups" {
    tailf:info "Contains connections group.";
}

```

```
    }
    tailf:annotate "/ietf-nss:network-slice-services/ietf-nss:slice-service/ietf-
nss:connection-groups/ietf-nss:connection-group" {
        tailf:info "List of connection groups.";
    }
}
```

Appendix D - Handling Zombies

Zombies are the internal operational data model in NSO to store deleted service data. Zombies are helpful when performing staged deletions and RFM. RFM is the NSO version of eventual consistency.

When a service deletion is triggered, NSO maintains references of the deleted services (zombies) in operational data. The zombies are deleted from CDB when all the configurations for the service are removed from the devices.

Zombies inform the Northbound of the progress of a service deletion. It also informs the stage it is waiting on, which helps to point to the problematic area. For more information on nano-services and zombies, see the ***NSO Developer Guide***.

In T-SDN FPs, you trigger a deletion to clean up the configuration on all the endpoints/nodes in a service. However, depending on the number of nodes or endpoints in a service, deleting the configuration at once may lock down the database until the slowest device removes the configuration. It is, therefore, recommended that the configurations be deleted in separate transactions, such as per device.

Once the configuration is successfully removed from all the devices in the service, continue to update the plan state to communicate the progress of deletion to Northbound. When the last device configuration is removed, remove the plan, zombies, and all the service-related operational data from CDB.

Sometimes, zombies are not deleted in the following scenarios, even after deleting the last device configuration.

1. Device is not reachable during deletion.
2. Device is reachable, but the configuration removal fails on the device for other reasons.

Some failures may require manual intervention to delete configuration references from the devices. In such cases, run the Cleanup action on the device and then run the service cleanup action on T-SDN FPs. For more information on the Cleanup actions, see **Working with Action Commands - CFP Services** in this documentation.

To recreate a service with the same name as a deleted service, wait for the service plan to be deleted. If the Northbound system tries to recreate the service instance before the zombie/delete is fully processed, the following error displays, which indicates the deletion process is still in progress.

"Aborted: Operation failed because Service is still in a zombie state: 'YYY'"

Note: T-SDN FPs do not support zombie resurrect and redeploy options.

The following are some sample zombies for the SR-ODN service.

```
admin@ncs% run show zombies
::::::::::::::::::::::::::::INTERNAL ZOMBIE::::::::::::::::::::::::::::
```

```
zombies service /sr-te/cisco-sr-te-cfp-sr-odn-internal:odn/odn-template[name='SR-ODN'] [head-end='PIOSXR-0']
delete-path /sr-te/cisco-sr-te-cfp-sr-odn-internal:odn/odn-template[name='SR-ODN'] [head-end='PIOSXR-0']
```

TYPE	NAME	BACK			STATE	STATUS	WHEN	ref	POST
		TRACK	GOAL	GOAL					ACTION
self	self	true	-	init	reached	-	-	-	
				cisco-sr-te-cfp-sr-odn-nano-services:config-apply	reached	-	-	-	
				ready	not-reached	-	-	-	

plan failed

```
plan error-info message "Failed to connect to device PIOSXR-0: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state"
```

```
plan error-info log-entry "/zombies/service[service-path=\"/sr-te/cisco-sr-te-cfp-sr-odn-internal:odn/odn-template[name='SR-ODN'] [head-end='PIOSXR-0']\" ]/log/log-entry[when='2020-03-26T23:34:12.992768+00:00']"
```

WHEN	TYPE	LEVEL	MESSAGE
2020-03-26T23:34:12.992768+00:00	service-modified	error	Failed to connect to device PIOSXR-0: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state

.....EXTERNAL ZOMBIE.....

```
zombies service /sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-template[name='SR-ODN']
```

```
delete-path /sr-te/cisco-sr-te-cfp-sr-odn:odn/odn-template[name='SR-ODN']
```

ACTION	TYPE	NAME	BACK			STATE	STATUS	WHEN	ref	POST
			TRACK	GOAL	GOAL					ACTION
self	self	self	true	-	init	not-reached	-	-	-	
					ready	not-reached	-	-	-	
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end	PIOSXR-0	PIOSXR-0	true	-	init	reached	-	-	-	
					cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply	failed	-	-	-	
					ready	not-reached	-	-	-	

plan failed

Appendix E - Supporting MultiVendors for IOSXR Netconf NEDs

This section discusses the support for IOSXR Netconf NED for the example services.

Note: All bootstrap files for multivendor settings for the Netconf NEDs are in the **tsdn-<version>-nso-<version>/netconf_data** folder.

Supporting MultiVendors for IOSXR Netconf Services for SR-TE CFP

This section discusses the support for IOSXR Netconf NED for the SR-ODN service, SR-Policy service, and CS SR-TE policy service.

Note: All bootstrap files for multivendor settings for Netconf NEDs are in the **tsdn-<version>-nso-<version>/netconf_data** folder.

You can download the Netconf NEDs from the Cisco website. The supported Netconf NED versions are listed in the **T-SDN FP Bundle Installation Guide**.

To support multivendors for the IOSXR Netconf services in SR-TE CFP, configure the **cisco-sr-te-cfp** dynamic-device mapping for IOSXR Netconf as follows. Make sure to provide the correct NED-ID.

Note: The **cs-sr-te-cfp** package in SR-TE CFP-IOSXR CLI is supported only on IOSXR CLI 7.46/7.52 NED, IOSXR NC 7.8 (and later) NED.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
    <dynamic-device-mapping>
      <ned-id>cisco-iosxr-nc-7.3:cisco-iosxr-nc-7.3</ned-id>
      <python-impl-class-name>sr_te_multi_vendors.NativeXR</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting MultiVendors for IETF-L2VPN-NM IOSXR Netconf Services

To support multivendors for IETF-L2VPN-NM IOSXR Netconf services, configure the **cisco-flat-l2vpn-ntw** dynamic-device mapping for IOSXR Netconf as follows. Make sure to provide the correct NED-ID.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
```

```

    <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l2vpn-
ntw">
      <dynamic-device-mapping>
        <ned-id>cisco-iosxr-nc-<version>:cisco-iosxr-nc-<version></ned-id>
        <python-impl-class-name>flat_l2vpn_multi_vendors.<Native/UM>XR</python-
impl-class-name>
      </dynamic-device-mapping>
    </cfp-configurations>
  </config>

```

Supporting MultiVendors for IOSXR Netconf Services for IETF-L3VPN-NM

To support multivendors for L3VPN-NM IOSXR Netconf services, configure the **cisco-flat-l3vpn-ntw** dynamic-device mapping for IOSXR Netconf as follows. Make sure to provide the correct NED-ID.

Note: All bootstrap files for multi-vendor settings for Netconf NEDs are in the **tsdn-<version>-nso-<version>/netconf_data** folder. You can download the Netconf NEDs from the Cisco website. The supported Netconf NED versions are listed in the **T-SDN FP Bundle Installation Guide**.

```

<<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-l3vpn-
ntw">
      <dynamic-device-mapping>
        <ned-id>cisco-iosxr-nc-<version>:cisco-iosxr-nc-<version></ned-id>
        <python-impl-class-name>flat_l3vpn_multi_vendors.<Native/UM>XR</python-
impl-class-name>
      </dynamic-device-mapping>
    </cfp-configurations>
  </l3vpn-ntw>
</config>

```

Supporting MultiVendors for IETF-TE Netconf Services

To support multivendors for IETF-TE IOSXR Netconf services, configure the **ietf-te-fp** dynamic device mapping for IOSXR Netconf as follows. Make sure to provide the correct NED-ID.

```

<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
    <dynamic-device-mapping>
      <ned-id>cisco-iosxr-nc-<version>:cisco-iosxr-nc-<version></ned-id>
      <python-impl-class-name>rsvp_te_multi_vendors.<Native/UM>XR</python-impl-
class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>

```

```
</config>
```

Supporting MultiVendors for PM Netconf Services

To support multivendors for PM IOSXR Netconf services, configure the dynamic device mapping for IOSXR Netconf as follows. Make sure to provide the correct NED-ID.

Note: PM services do not support NativeXR.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tdsn-pm">
    <dynamic-device-mapping>
      <ned-id>cisco-iosxr-nc-<version>:cisco-iosxr-nc-<version></ned-id>
      <python-impl-class-name>pm_multi_vendors.UMXR</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```