Cisco NSO T-SDN Function Pack

User Guide

Version 3.0.0

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**.

Contents

Preface

Abstract

This document serves as the documentation reference and user guide for the NSO Transport SDN Function Pack (T-SDN FP) Bundle version 3.0.0.

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

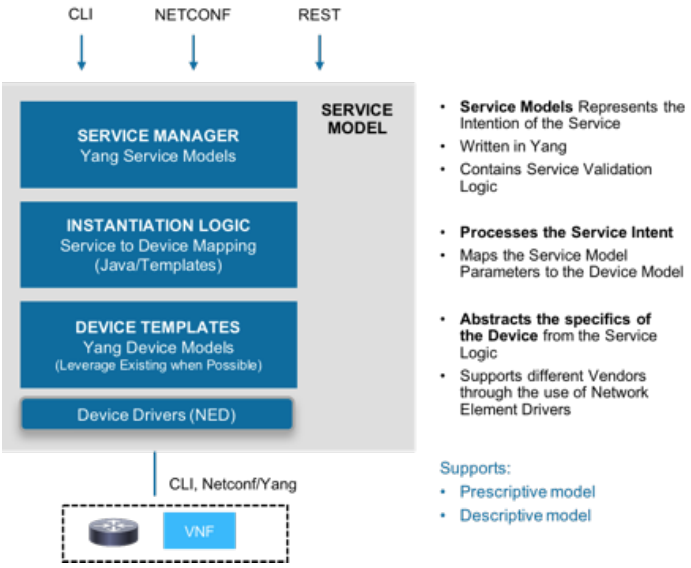
Cisco follows a bias-free documentation policy. According to this policy, Cisco treats all persons with respect—regardless of race, color, ancestry, national origin, age, sex, citizenship, veteran status, marital status, sexual orientation, physical or mental ability, religious creed, or medical condition. 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 by using a network command line interface (CLI). NSO enables service providers to dynamically adopt their orchestration solution with changes in the offered service portfolio. NSO is built on a Model-Driven Architecture which 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 changes to the devices 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 multi-vendor 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 assure desired business outcomes.

The NSO T-SDN FPs form an extension of SDN, which allows 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 the 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. SR-TE Core Function Pack (CFP) – a productized and supported implementation of SR-TE automation
2. The following Example Function Packs are intended to be customized for each customer deployment by Cisco Customer Experience (CX):

- Layer 2 VPN (L2VPN)
- Layer 3 VPN (L3VPN)
- Internet Engineering Task Force (IETF) implementation of L2VPN
- IETF implementation of L3VPN and
- IETF-TE (RSVP-TE)

Without having to modify the SR-TE CFP, it allows you to add additional configurations into a device by using custom-templates.

SR-TE CFP comprises the following modules/services:

- SR-TE on-Demand-Network (ODN)
- SR-TE Policies

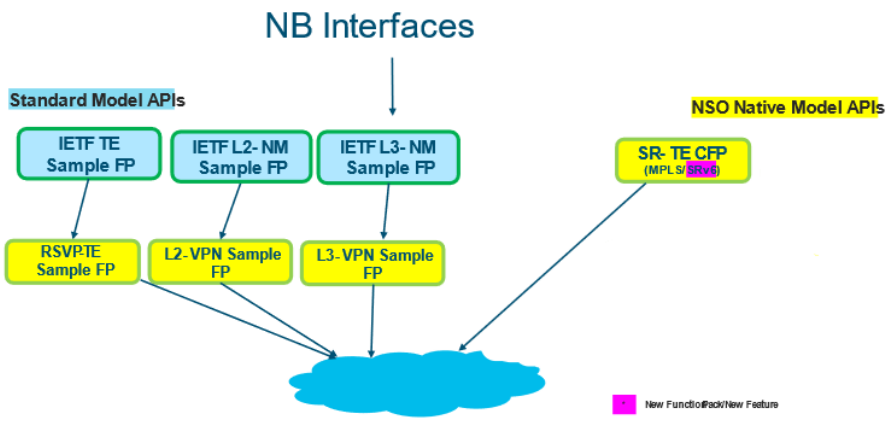
SR policy path computation of SR-TE CFP supports service specific constraints, such as bandwidth, latency, path diversity, and traffic engineering constraints, such as affinity, bandwidth, cost, latency coming from the network.

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

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.

T-SDN Function Packs 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 L2VPN/L2NM, L3VPN/L3NM, and IETF-TE over an existing SR-Policy (ODN or Preferred Path option)

T-SDN FPs are implemented by using the SR-TE CFP and the Example Function Packs –L2VPN, L2NM, L3VPN, L3NM, and IETF-TE.

A router in a 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.

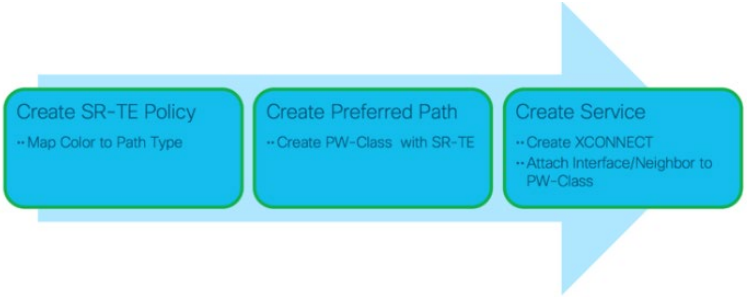
SR-TE CFP provides the SR-TE policies and SR-ODN services. T-SDN FPs configure the SR-TE policies and instantiates services, such as the L2VPN, L3VPN, and IETF-TE function packs that utilize these SR-TE policies.

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

L2VPN allows you to configure point-to-point pseudowire (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 L2VPN 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 and have both source and destination for the network traffic.

Introduction

SR-TE Core Function Pack

Segment Routing prepares networks for business models, where applications can direct network behavior. SR provides the right balance between distributed intelligence and centralized optimization and programming.

SR-TE takes place between a source and destination pair. SR-TE uses the concept of source routing, where 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.

SR-TE requires minimal configuration on the source router. The SR-TE CFP provides the SR-TE policies and SR-TE ODN. This chapter discusses each of these SR-TE CFP services.

For information about yang models and payloads, see **Appendix C - Yang Models** and **Appendix D – Sample Custom-Template Payloads** respectively in this documentation.

SR-TE ODN Services

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

SR-Policy Services

SR-TE uses a policy to steer traffic through the network. 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 SID list is pushed on the packet by the head-end. 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.

At a given head-end, an SR policy is uniquely identified by a tuple (color, endpoint).

There are two types of SR-TE policies:

- **Local Dynamic SR-TE policy:** When you configure local dynamic SR-TE policy, the head-end locally calculates the path to the destination address. 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.

Example Function Packs

L2VPN, L2NM, L3VPN, L3NM, and IETF-TE are Example Function Packs to 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.

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

For information about the yang models and payloads, see **Appendix C - Yang Models** and **Appendix D – Sample Custom-Template Payloads** respectively in this documentation.

L2VPN Services

L2 Virtual Private Network (L2VPN) establishes an end-to-end layer 2 connection. L2VPN allows disparate systems to be connected in a way that it appears as if they are connected by using a single physical connection or are using the same LAN.

L2VPN employs L2 services over MPLS to build a topology of point-to-point connections that connect end customer sites in a VPN.

Note: Association of L2VPN with SRv6 is currently not supported.

L2NM Services

The IETF-L2VPN-NM service provides an IETF model overlay of the Flat L2VPN configuration. It implements the **draft-barguil-opsawg-l2sm-l2nm-02** IETF model. It implements a subset of the IETF YANG model.

Y1731

Configure the Y1731 probes in the service to monitor the service operation for delays, 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 Flat L2VPN/L2NM service.

In L2VPN services, configure and enable the Y1731 probes to monitor the L2VPN end-to-end connectivity, delay, and latency.

L3VPN Services

L3VPN uses virtual routing and forwarding techniques to forward VPN traffic over a network. For every endpoint, e-bgp can be configured to ODN.

The L3VPN service allows you to configure:

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

L3NM Services

The IETF-L3VPN-NM service provides an IETF model overlay of the flat L3VPN configuration. It implements the **draft-ietf-opsawg-l3sm-l3nm-03** IETF model. It implements a subset of the IETF YANG model.

IETF-TE Services

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 and have both source and destination for the network traffic.

This function pack uses the IETF-TE model with some modifications for the T-SDN FP Bundle.

Monitoring Service Assurance with Automated Assurance

Automated Assurance (AA) is an optional feature and is applicable only to Flat L2VPN/L2NM and Flat L3VPN/L3NM Example Function Packs. The AA model, when installed help 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 North Bound system to indicate changes in the device configuration.

Configuration

The topics in this chapter provide information on how to configure the services by using the T-SDN FPs. T-SDN FPs comprise SR-TE CFP. The SR-TE CFP consists of SR-TE ODN module and the SR-TE Policy module.

Segment Routing integrates with multi-service capabilities of MPLS, including Layer 3 VPN (L3VPN), Virtual Private Wire Service (VPWS), Virtual Private LAN Service (VPLS), and Ethernet VPN (EVPN).

L2VPN, L3VPN, and IETF-TE are Example Function Packs used to demonstrate SR-TE CFP. You can choose to install the required packages for these function packs during or after T-SDN FPs installation.

For more information on the list of packages extracted during T-SDN FP installation, 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.

SR-TE CFP comprises the SR-ODN and SR-Policy modules to allow you to configure the SR-TE policy at various stages.

Segment routing can be applied on both MPLS and IPV6 (SRv6) data plane. In a Segment Routing over IPv6 (SRv6) network, an IPv6 address serves as the SID.

Segment Routing (SR) can be applied on both MPLS and IPv6 data planes. In an SR-MPLS enabled network, an MPLS label is used as the 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.

Creating SR-ODN Services

Use SR-TE ODN service to configure an ODN template for a color for which to instantiate the on-demand SR Policy.

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-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>
       <head-end>
         <name>PIOSXR-0</name>
       </head-end>
       <head-end>
```

```
      <name>PIOSXR-1</name>
    </head-end>
    <maximum-sid-depth>6</maximum-sid-depth>
    <color>300</color>
    <bandwidth>200</bandwidth>
    <dynamic>
      <metric-type>hopcount</metric-type>
      <metric-margin>
       <absolute>30</absolute>
      </metric-margin>
      <disjoint-path>
        <type>link</type>
        <group-id>10</group-id>
        <sub-id>5</sub-id>
      </disjoint-path>
      <affinity>
        <rule>
          <action>include-all</action>
          <color>GREEN</color>
          <color>RED</color>
        </rule>
      </affinity>
    </dynamic>
  </odn-template>
  <odn-template>
    <name>SR-XE-CLI-ODN-400</name>
    <head-end>
      <name>XE-CLI-0</name>
    </head-end>
    <color>300</color>
    <bandwidth>200</bandwidth>
    <maximum-sid-depth>6</maximum-sid-depth>
    <dynamic>
      <metric-margin>
       <absolute>30</absolute>
      </metric-margin>
      <flex-alg>200</flex-alg>
      <pce/>
      <disjoint-path>
        <type>node</type>
        <group-id>10</group-id>
        <source>1.1.1.1</source>
      </disjoint-path>
      <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
```

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

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.

```
<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>color100</name>
      <srv6>
        <locator>
          <locator-name>ALG132r7</locator-name>
        </locator>
```

```
      </srv6>
      <head-end>
        <name>XR-SRv6-1</name>
      </head-end>
      <color>100</color>
      <dynamic>
        <metric-type>igp</metric-type>
        <pce/>
      </dynamic>
    </odn-template>
  </odn>
</sr-te>
```

Display the plan to view the status of the service.

Creating SR-Policies

Use SR-Policies to instruct the routers in the network to follow a specified path instead of the shortest path calculated by the IGP. If a packet is steered into an SR-TE policy, the SID list is pushed on the packet by the head-end. The rest of the network executes the instructions embedded in the SID list.

In a Segment Routing over IPv6 (SRv6) network, an IPv6 address serves as the SID. A locator represents an 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.

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** in this documentation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
<policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te-sr-
policies">
      <policy>
        <name>SR-Policy-1</name>
        <head-end>
          <name>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
```

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

To create SR-Policy on SRv6:

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

```
<sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
  <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te-sr-policies">
    <policy>
      <name>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 Example Function Packs Services

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.

For L2VPN service and L3VPN service, be sure of the model you want to use to manage these services. Before creating a L2VPN service or L3VPN service, choose either the native model implementation or the IETF implementation. Do not switch between these implementation models across services.

The IETF-TE implementation uses only the IETF model.

Creating Flat L2VPN Services

You can implement a custom L2VPN Function Pack to meet specific requirements. Use the Example Function Packs as a starting point or for design patterns.

There are two types of example L2VPN services: **L2VPN P2P** and **L2VPN EVPN VPWS**. Each service requires a mandatory **Local Site** and an optional **Remote Site** to create the service.

The L2VPN-P2P service has two types of preferred path policies:

- **RSVP-TE policy:** Specify either the IETF-TE - RSVP-TE association or manually configure the tunnel-te ID.
- **SR-TE policy:** An SR-TE policy initiates a single selected path. This is the preferred valid candidate path.

Note: Association of L2VPN with SRv6 is currently not supported.

For the Flat L2VPN yang model, see **Appendix C - Yang Models** in this documentation.

Any changes to an SR-Policy attached to L2VPN P2P service do not reflect automatically. After updating the SR-TE service, manually redeploy the L2VPN P2P service to push the corresponding updated configuration to the device.

The following topics in this section show how to configure a flat L2VPN-P2P service and a flat L2VPN-P2P service with the preferred path policies.

Creating Flat L2VPN-P2P Services

To create the Flat L2VPN-P2P service:

1. The following is a sample payload to create a Flat L2VPN-P2P service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L2vpn">
    <name>P2P-DOT1Q</name>
    <service-type>p2p</service-type>
    <flat-L2vpn-p2p>
      <pw-id>100</pw-id>
      <local-site>
        <pe>PIOSXR-0</pe>
        <if-type>HundredGigE</if-type>
        <if-id>0/0/0/8</if-id>
        <if-encap>dot1q</if-encap>
        <vlan-id>100</vlan-id>
        <sub-if-id>100</sub-if-id>
        <rewrite>
          <ingress>
            <translate>1-to-1</translate>
            <dot1q>123</dot1q>
            <mode>symmetric</mode>
          </ingress>
        </rewrite>
        <xconnect-group-name>P2P-DOT1Q</xconnect-group-name>
        <xconnect-encapsulation>mpls</xconnect-encapsulation>
        <p2p-name>P2P-DOT1Q</p2p-name>
        <control-word>no</control-word>
        <pw-class>P2P-DOT1Q</pw-class>
```

```
          <xconnect-local-ip>12.0.0.0</xconnect-local-ip>
          <xconnect-remote-ip>14.0.0.0</xconnect-remote-ip>
          <mpls-local-label>101</mpls-local-label>
          <mpls-remote-label>202</mpls-remote-label>
        </local-site>
        <remote-site>
          <pe>PIOSXR-1</pe>
          <if-type>HundredGigE</if-type>
          <if-id>0/0/0/8</if-id>
          <if-encap>dot1q</if-encap>
          <vlan-id>100</vlan-id>
          <sub-if-id>100</sub-if-id>
          <rewrite>
            <ingress>
              <translate>1-to-1</translate>
              <dot1q>123</dot1q>
              <mode>symmetric</mode>
            </ingress>
          </rewrite>
          <xconnect-group-name>P2P-DOT1Q</xconnect-group-name>
          <p2p-name>P2P-DOT1Q</p2p-name>
          <pw-class>P2P-DOT1Q</pw-class>
        </remote-site>
      </flat-L2vpn-p2p>
    </flat-L2vpn>
  </config>
```

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

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

Plan Component	Value	Description
Type	local-site	This represents the Local-site (mandatory site) of the L2VPN service.
	remote-site	This represents the remote-site (optional site) of the L2VPN service.
State	cisco-flat-L2vpn-fp-nano-plan-services:config-apply	This represents the device configuration state in the nano-plan of a service.

```
admin@ncs% run show flat-L2vpn-plan L2vpn-p2p-1 plan
```

```
                                                                          BACK                                                                                               POST ACTION
TYPE                                                        NAME        TRACK GOAL   STATE                                                       STATUS   WHEN                 ref  STATUS
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                        self        false -      init                                                        reached  2020-03-03T22:06:26  -    -
                                                                                   ready                                                       reached  2020-03-03T22:07:01  -    -
local-site                                                  PIOSXR-0    false -      init                                                        reached  2020-03-03T22:06:26  -    -
                                                                                     cisco-flat-L2vpn-fp-nano-plan-services:config-apply        reached  2020-03-03T22:06:30  -    -
                                                                                     ready                                                       reached  2020-03-03T22:06:30  -    -
remote-site                                                 PIOSXR-1    false -      init                                                        reached  2020-03-03T22:06:26  -    -
                                                                                   cisco-flat-L2vpn-fp-nano-plan-services:config-apply        reached  2020-03-03T22:06:30
ready                                                      reached  2020-03-03T22:06:30   -    -
```

Creating Flat L2VPN – P2P Service with RSVP-TE Association

1. The following is a sample payload to create a Flat L2VPN-P2P service with the RSVP-TE association.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L2vpn">
    <name>L2vpn01</name>
    <service-type>p2p</service-type>
    <flat-L2vpn-p2p>
      <local-site>
        <rsvp-te>
          <preferred-path>
            <te-tunnel-id>54321</te-tunnel-id>
            <fallback>disable</fallback>
          </preferred-path>
        </rsvp-te>
      </local-site>
      <remote-site>
        <rsvp-te>
          <preferred-path>
            <!--There must exist an IETF-TE service named "IETF-RSVP-
TE"-->
            <ietf-te-service>IETF-RSVP-TE</ietf-te-service>
          </preferred-path>
        </rsvp-te>
      </remote-site>
    </flat-L2vpn-p2p>
  </flat-L2vpn>
</config>
```

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

Creating Flat L2VPN – P2P Service with SR-TE Policy

1. The following is a sample payload to create a Flat L2VPN-P2P service with the SR-TE policy.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
      <name>L2vpn01</name>
      <service-type>p2p</service-type>
      <flat-L2vpn-p2p>
        <local-site>
          <sr-te>
            <preferred-path>
              <policy>SR-CLI-DYNAMIC-P2P-PIOSXR-0</policy>
            </preferred-path>
          </sr-te>
        </local-site>
        <remote-site>
          <sr-te>
            <preferred-path>
              <policy>SR-CLI-DYNAMIC-P2P-PIOSXR-1</policy>
            </preferred-path>
          </sr-te>
        </remote-site>
      </flat-L2vpn-p2p>
  </flat-L2vpn>
</config>
```

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

Creating Flat L2VPN-P2P Service with Automated Assurance

1. The following is a sample payload to create a Flat L2VPN-P2P service with AA service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>P2P-DOT1Q</name>
    <service-type>p2p</service-type>
    <service-assurance>
      <monitoring-state>pause</monitoring-state>
      <profile-name>profile-A custom</profile-name>
      <rule-name>rule-A custom</rule-name>
    </service-assurance>
  </flat-L2vpn>
</config>
```

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

Creating Flat L2VPN-P2P Service with Y1731

1. The following is a sample payload to create a Flat L2VPN-P2P service with Y1731.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <y-1731-profile xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L2vpn">
    <name>delay</name>
    <type>delay</type>
    <probe>
      <burst>
        <message-count>30</message-count>
        <message-period>100</message-period>
      </burst>
      <measurement-interval>60</measurement-interval>
      <frame-size>1000</frame-size>
      <priority>5</priority>
    </probe>
    <delay-params>
      <statistic>
        <type>delay-two-way</type>
      </statistic>
      <statistic>
        <type>delay-sd</type>
      </statistic>
      <statistic>
        <type>delay-ds</type>
      </statistic>
      <statistic>
        <type>jitter-two-way</type>
      </statistic>
      <statistic>
        <type>jitter-sd</type>
      </statistic>
      <statistic>
        <type>jitter-ds</type>
      </statistic>
    </delay-params>
    <schedule>
      <interval>5</interval>
      <duration>5</duration>
    </schedule>
  </y-1731-profile>
```

```
  <y-1731-profile xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L2vpn">
    <name>loss</name>
    <type>synthetic-loss</type>
    <probe>
      <burst>
        <message-count>30</message-count>
        <message-period>100</message-period>
      </burst>
      <measurement-interval>60</measurement-interval>
      <frame-size>1000</frame-size>
      <priority>5</priority>
    </probe>
    <loss-params>
      <statistic>
        <type>loss-sd</type>
      </statistic>
      <statistic>
        <type>loss-ds</type>
      </statistic>
    </loss-params>
    <schedule>
      <interval>1</interval>
      <duration>1</duration>
    </schedule>
  </y-1731-profile>
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>L2VPN-p2p-y-1731-AA</name>
    <service-type>p2p</service-type>
    <flat-L2vpn-p2p>
      <pw-id>1001</pw-id>
      <local-site>
        <pe>PIOSXR-0</pe>
        <if-type>GigabitEthernet</if-type>
        <if-id>0/0/0/1</if-id>
        <if-encap>dot1q</if-encap>
        <vlan-id>601</vlan-id>
        <sub-if-id>601</sub-if-id>
        <rewrite>
          <ingress>
            <push/>
            <dot1q>123</dot1q>
            <mode>symmetric</mode>
          </ingress>
```

```
      </rewrite>
      <mtu>65</mtu>
      <xconnect-group-name>l2vpn-p2p-y-1731-AA</xconnect-group-name>
      <p2p-name>l2vpn-p2p-y-1731-AA-l2vpn-p2p-ac1</p2p-name>
      <control-word>yes</control-word>
      <pw-class>pw-class-l2vpn-p2p-y-1731-AA</pw-class>
      <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>icc-based</id-type>
          <message-period>1s</message-period>
          <y-1731-profile>
            <name>delay</name>
          </y-1731-profile>
          <y-1731-profile>
            <name>loss</name>
          </y-1731-profile>
        </y-1731>
      </ethernet-service-oam>
      <xconnect-local-ip>198.18.1.4</xconnect-local-ip>
      <xconnect-remote-ip>198.18.1.5</xconnect-remote-ip>
      <mpls-local-label>101</mpls-local-label>
      <mpls-remote-label>102</mpls-remote-label>
    </local-site>
    <remote-site>
      <pe>PIOSXR-1</pe>
      <if-type>GigabitEthernet</if-type>
      <if-id>0/0/0/1</if-id>
      <if-encap>dot1q</if-encap>
      <vlan-id>601</vlan-id>
      <sub-if-id>601</sub-if-id>
      <rewrite>
        <ingress>
          <push/>
          <dot1q>234</dot1q>
          <mode>symmetric</mode>
        </ingress>
      </rewrite>
      <mtu>64</mtu>
      <xconnect-group-name>l2vpn-p2p-y-1731-AA</xconnect-group-name>
      <p2p-name>l2vpn-p2p-y-1731-AA-l2vpn-p2p-ac1</p2p-name>
```

```
            <control-word>yes</control-word>
            <pw-class>pw-class-l2vpn-p2p-y-1731-AA</pw-class>
            <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>icc-based</id-type>
                <message-period>1s</message-period>
                <y-1731-profile>
                  <name>delay</name>
                </y-1731-profile>
                <y-1731-profile>
                  <name>loss</name>
                </y-1731-profile>
              </y-1731>
            </ethernet-service-oam>
          </remote-site>
        </flat-L2vpn-p2p>
        <service-assurance>
          <monitoring-state>pause</monitoring-state>
          <profile-name>profile-A custom</profile-name>
          <rule-name>rule-A custom</rule-name>
        </service-assurance>
      </flat-L2vpn>
    </config>
```

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

Creating Flat L2VPN – EVPN Services

Virtual Private Wire Service is the EVPN service provisioned by L2VPN. For every site (local/remote), you can configure BGP for ODN in EVPN service. SR-ODN allows a service head-end router to automatically instantiate an SR-policy to a BGP next-hop, when required, for the EVPN service.

Note: You must create the L2VPN-EVPN service to create the L2VPN SR-TE preferred path service and the L2VPN SR-TE-ODN service.

To create the Flat L2VPN –EVPN-VPWS service:

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

```
  <config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
```

```
    <name>L2vpn-dynamic-02</name>
    <service-type>evpn-vpws</service-type>
    <flat-L2vpn-evpn-vpws>
      <evi-id>1000</evi-id>
      <local-site>
        <pe>PIOSXR-0</pe>
        <if-type>HundredGigE</if-type>
        <if-id>0/0/1/0</if-id>
        <if-description>L2VPN-Dynamic-02</if-description>
        <if-encap>untagged</if-encap>
        <sub-if-id>23</sub-if-id>
        <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
        <p2p-name>EVPN-PIOSXR-0</p2p-name>
        <evi-source>1</evi-source>
        <evi-target>2</evi-target>
      </local-site>
      <remote-site>
        <pe>PIOSXR-1</pe>
        <if-type>TenGigE</if-type>
        <if-id>0/0/0/35</if-id>
        <if-description>L2VPN-Dynamic-02</if-description>
        <if-encap>untagged</if-encap>
        <sub-if-id>40</sub-if-id>
        <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
        <p2p-name>EVPN-PIOSXR-1</p2p-name>
      </remote-site>
    </flat-L2vpn-evpn-vpws>
  </flat-L2vpn>
</config>
```

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

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

Plan Component	Value	Description
Type	local-site	This represents the Local-site (mandatory site) of the L2VPN service.
	remote-site	This represents the remote-site (optional site) of the L2VPN service.
State	cisco-flat-L2vpn-fp-nano-plan-services:config-apply	This represents the device configuration state in the nano-plan of a service.

```
admin@ncs% run show flat-L2vpn-plan L2vpn-dynamic-02 plan component

                                BACK                                                                                                                                                                  POST ACTION
TYPE                                                    NAME      TRACK  GOAL  STATE                                                                                  STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                    self      false  -     init                                                                                    reached  2020-01-15T00:53:46  -    -
                                                                               ready                                                                                   reached  2020-01-15T00:53:51  -    -
local-site                                              PIOSXR-0  false  -     init                                                                                    reached  2020-01-15T00:53:46  -    -
                                                                               cisco-flat-L2vpn-fp-nano-plan-services:config-apply                                     reached  2020-01-15T00:53:50  -    -
                                                                               ready                                                                                   reached  2020-01-15T00:53:50  -    -
remote-site                                             PIOSXR-1  false  -     init                                                                                    reached  2020-01-15T00:53:46  -    -
                                                                               cisco-flat-L2vpn-fp-nano-plan-services:config-apply                                     reached  2020-01-15T00:53:50  -    -
                                                                               ready                                                                                   reached  2020-01-15T00:53:50  -    -
```

Creating Flat L2VPN SR-TE ODN Service

On-Demand Next Hop (ODN) for L2VPN creates a SR-TE auto-tunnel and uses the auto-tunnel for pseudowire data plane. You must create the L2VPN route policy to use the SR-TE ODN 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.

To modify the **parent-rr-route-policy** after associating the L2 EVPN VPWS SR-TE ODN, you must update the original value. For more information, see **Appendix B: Updating the Original Definition of the parent-route-policy.**

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

To create flat L2VPN SR-TE ODN service:

1. The following is a sample payload to create the route-policy.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l2vpn-route-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
tsdn-flat-L2vpn">
    <name>L2-RP-local</name>
    <color>
      <id>100</id>
      <ipv4>
        <rd>1.1.1.1:1</rd>
        <rd>1.1.1.2:1</rd>
      </ipv4>
    </color>
    <color>
      <id>101</id>
      <ipv4>
        <rd>2.1.1.1:1</rd>
        <rd>2.1.1.2:1</rd>
      </ipv4>
    </color>
  </l2vpn-route-policy>
```

```
  <l2vpn-route-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
tsdn-flat-L2vpn">
    <name>L2-RP-remote</name>
    <color>
      <id>100</id>
      <ipv4>
        <rd>3.1.1.1:1</rd>
        <rd>3.1.1.2:1</rd>
      </ipv4>
    </color>
    <color>
      <id>101</id>
      <ipv4>
        <rd>4.1.1.1:1</rd>
        <rd>4.1.1.2:1</rd>
      </ipv4>
    </color>
  </l2vpn-route-policy>
</config>
```

The following is a sample payload to create the L2VPN service with SR-TE ODN.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>L2vpn-dynamic-02</name>
    <service-type>evpn-vpws</service-type>
    <flat-L2vpn-evpn-vpws>
      <evi-id>1000</evi-id>
      <local-site>
        <pe>PIOSXR-0</pe>
        <if-type>HundredGigE</if-type>
        <if-id>0/0/1/0</if-id>
        <if-description>L2VPN-Dynamic-02</if-description>
        <if-encap>untagged</if-encap>
        <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
        <p2p-name>EVPN-PIOSXR-0</p2p-name>
        <sr-te>
          <odn>
            <route-policy>L2-RP-local</route-policy>
            <attach-point>
              <parent-rr-route-policy>L2-ATTACH</parent-rr-route-policy>
            </attach-point>
          </odn>
        </sr-te>
```

```
        <evi-source>1</evi-source>
        <evi-target>2</evi-target>
      </local-site>
      <remote-site>
        <pe>PIOSXR-1</pe>
        <if-type>TenGigE</if-type>
        <if-id>0/0/0/35</if-id>
        <if-description>L2VPN-Dynamic-02</if-description>
        <if-encap>untagged</if-encap>
        <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
        <p2p-name>EVPN-PIOSXR-1</p2p-name>
        <sr-te>
          <odn>
            <route-policy>L2-RP-remote</route-policy>
            <attach-point>
              <parent-rr-route-policy>L2-ATTACH</parent-rr-route-policy>
            </attach-point>
          </odn>
        </sr-te>
      </remote-site>
    </flat-L2vpn-evpn-vpws>
  </flat-L2vpn>
</config>
```

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

Creating Flat L2VPN SR-TE Preferred Path Service

A candidate path has a preference. If two policies have same {color, endpoint} but different preferences, the policy with the highest preference is selected. An SR-TE policy initiates a single (selected) path, which is the preferred valid candidate path.

You must create the L2VPN SR-TE preferred path service on top of the L2VPN-EVPN-VPWS service. For more information, see section **Creating Flat L2VPN – EVPN Services** in this documentation.

To create flat L2VPN SR-TE preferred path service:

1. The following is a sample payload to create the SR-TE preferred path.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>L2vpn-dynamic-02</name>
    <service-type>evpn-vpws</service-type>
    <flat-L2vpn-evpn-vpws>
      <evi-id>1000</evi-id>
      <local-site>
        <pe>PIOSXR-0</pe>
```

```
          <if-type>HundredGigE</if-type>
          <if-id>0/0/1/0</if-id>
          <if-description>L2VPN-Dynamic-02</if-description>
          <if-encap>untagged</if-encap>
          <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
          <p2p-name>EVPN-PIOSXR-0</p2p-name>
          <sr-te>
            <preferred-path>
              <policy>SR-CLI-ERO-VPWS-PIOSXR-0</policy>
            </preferred-path>
          </sr-te>
          <pw-class>ero-nso</pw-class>
          <evi-source>1</evi-source>
          <evi-target>2</evi-target>
        </local-site>
        <remote-site>
          <pe>PIOSXR-1</pe>
          <if-type>TenGigE</if-type>
          <if-id>0/0/0/35</if-id>
          <if-description>L2VPN-Dynamic-02</if-description>
          <if-encap>untagged</if-encap>
          <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
          <p2p-name>EVPN-PIOSXR-1</p2p-name>
          <sr-te>
            <preferred-path>
              <policy>SR-CLI-ERO-VPWS-PIOSXR-1</policy>
            </preferred-path>
          </sr-te>
          <pw-class>ero-nso</pw-class>
        </remote-site>
      </flat-L2vpn-evpn-vpws>
    </flat-L2vpn>
  </config>
```

2. Verify the configuration on the device by running the `sh run l2vpn` command.

Creating Flat L2VPN – EVPN Service with Automated Assurance

1. The following is a sample payload to create a L2VPN-EVPN service with AA.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>L2vpn-dynamic-02</name>
    <service-type>evpn-vpws</service-type>
    <service-assurance>
```

```
    <monitoring-state>pause</monitoring-state>
    <profile-name>profile-A custom</profile-name>
    <rule-name>rule-A custom</rule-name>
  </service-assurance>
  </flat-L2vpn>
</config>
```

2. Display the plan to verify the configuration.

Creating Flat L2VPN – EVPN Service with Y1731

1. The following is a sample payload to create a L2VPN-EVPN service with Y1731.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <y-1731-profile xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L2vpn">
    <name>delay</name>
    <type>delay</type>
    <probe>
      <burst>
        <message-count>30</message-count>
        <message-period>100</message-period>
      </burst>
      <measurement-interval>10</measurement-interval>
      <frame-size>2000</frame-size>
      <priority>2</priority>
    </probe>
    <delay-params>
      <statistic>
        <type>delay-two-way</type>
      </statistic>
      <statistic>
        <type>delay-sd</type>
      </statistic>
      <statistic>
        <type>delay-ds</type>
      </statistic>
      <statistic>
        <type>jitter-two-way</type>
      </statistic>
      <statistic>
        <type>jitter-sd</type>
      </statistic>
      <statistic>
        <type>jitter-ds</type>
      </statistic>
```

```
    </delay-params>
    <schedule>
      <interval>5</interval>
      <duration>5</duration>
    </schedule>
    <bucket-details>
      <bucket-size>1</bucket-size>
      <bucket-archive>3</bucket-archive>
    </bucket-details>
  </y-1731-profile>
  <y-1731-profile xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L2vpn">
    <name>loss</name>
    <type>synthetic-loss</type>
    <probe>
      <burst>
        <message-count>30</message-count>
        <message-period>100</message-period>
      </burst>
      <measurement-interval>10</measurement-interval>
      <frame-size>2000</frame-size>
      <priority>2</priority>
    </probe>
    <loss-params>
      <statistic>
        <type>loss-sd</type>
      </statistic>
      <statistic>
        <type>loss-ds</type>
      </statistic>
    </loss-params>
    <schedule>
      <interval>1</interval>
      <duration>1</duration>
    </schedule>
    <bucket-details>
      <bucket-size>1</bucket-size>
      <bucket-archive>3</bucket-archive>
    </bucket-details>
  </y-1731-profile>
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>L2VPN-EVPN-UNTAGGED-Y1731</name>
    <service-type>evpn-vpws</service-type>
    <flat-L2vpn-evpn-vpws>
```

```
<evi-id>1</evi-id>
<local-site>
  <pe>PIOSXR-0</pe>
  <if-type>GigabitEthernet</if-type>
  <if-id>0/0/0/1</if-id>
  <if-encap>untagged</if-encap>
  <sub-if-id>501</sub-if-id>
  <rewrite>
    <ingress>
      <push/>
      <dot1q>234</dot1q>
      <mode>symmetric</mode>
    </ingress>
  </rewrite>
  <xconnect-group-name>L2VPN-EVPN-UNTAGGED</xconnect-group-name>
  <p2p-name>L2VPN-EVPN-UNTAGGED-l2vpn-evpn-ac1</p2p-name>
  <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>number</id-type>
      <message-period>1s</message-period>
      <y-1731-profile>
        <name>delay</name>
      </y-1731-profile>
      <y-1731-profile>
        <name>loss</name>
      </y-1731-profile>
    </y-1731>
  </ethernet-service-oam>
  <evi-source>1</evi-source>
  <evi-target>2</evi-target>
</local-site>
<remote-site>
  <pe>PN73-0</pe>
  <if-type>HundredGigE</if-type>
  <if-id>0/0/0/1</if-id>
  <if-encap>untagged</if-encap>
  <multi-home>
    <esi-value>00.01.00.ac.ce.55.00.0a.00</esi-value>
  </multi-home>
  <sub-if-id>500</sub-if-id>
```

```
            <rewrite>
              <ingress>
                <push/>
                <dot1q>123</dot1q>
                <mode>symmetric</mode>
              </ingress>
            </rewrite>
            <xconnect-group-name>L2VPN-EVPN-UNTAGGED</xconnect-group-name>
            <p2p-name>L2VPN-EVPN-UNTAGGED-l2vpn-evpn-ac1</p2p-name>
            <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>number</id-type>
                <message-period>1s</message-period>
                <y-1731-profile>
                  <name>delay</name>
                </y-1731-profile>
                <y-1731-profile>
                  <name>loss</name>
                </y-1731-profile>
              </y-1731>
            </ethernet-service-oam>
          </remote-site>
        </flat-L2vpn-evpn-vpws>
      </flat-L2vpn>
    </config>
```

2. Display the plan to verify the configuration.

Creating IETF-L2VPN-NM Services

The IETF-L2VPN-NM service provides an IETF model overlay of the Flat L2VPN configuration. It implements the **draft-barguil-opsawg-l2sm-l2nm-02** IETF model. It implements a subset of the IETF YANG model.

You must create the Flat L2VPN service before you can create the IETF-L2VPN-NM service. For more information, see **Creating Flat L2VPN Services**.

The IETF-L2VPN-NM has the t-ldp and evpn-bgp service types. Each service can have two vpn-nodes. For the IETF-L2VPN-NM yang model, see **Appendix C - Yang Models** in this documentation.

The following topics show how to configure a flat IETF-L2VPN-NM service.

Creating IETF-L2VPN-NM t-ldp Service

The IETF-L2VPN-NM t-ldp 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.
- **RSVP-TE policy:** You may either specify the IETF-TE - RSVP-TE association or manually configure the tunnel-te ID.

To create the IETF-L2VPN-NM t-ldp service:

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

```
<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>
       <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>
           <name>CT-CLI-banner</name>
           <variable>
             <name>BANNER_TEXT</name>
             <value>Welcome_B</value>
           </variable>
         </custom-template>
         <signaling-options>
           <type>vpn-common:t-ldp</type>
           <t-ldp-pwe>
             <ac-pw-list>
               <peer-addr>198.18.1.5</peer-addr>
               <vc-id>1001</vc-id>
               <mpls-label>101</mpls-label>
             </ac-pw-list>
           </t-ldp-pwe>
         </signaling-options>
         <vpn-network-accesses>
```

```
      <vpn-network-access>
        <id>l2vpn-p2p-ac1</id>
        <Interface-mtu>65</Interface-mtu>
        <connection>
      <encapsulation-type>vpn-common:dot1q</encapsulation-type>
          <dot1q-interface>
            <l2-access-type>vpn-common:dot1q</l2-access-type>
            <dot1q>
            <physical-inf>GigabitEthernet0/0/0/1</physical-inf>
              <c-vlan-id>601</c-vlan-id>
              <rewrite>
                <ingress>
                  <push/>
                  <dot1q>123</dot1q>
                  <mode>symmetric</mode>
                </ingress>
              </rewrite>
            </dot1q>
          </dot1q-interface>
        </connection>
      </vpn-network-access>
    </vpn-network-accesses>
  </vpn-node>
  <vpn-node>
    <vpn-node-id>PIOSXR-1</vpn-node-id>
    <ne-id>PIOSXR-1</ne-id>
    <signaling-options>
      <type>vpn-common:t-ldp</type>
      <t-ldp-pwe>
        <ac-pw-list>
          <peer-addr>198.18.1.4</peer-addr>
          <vc-id>1001</vc-id>
          <mpls-label>102</mpls-label>
        </ac-pw-list>
      </t-ldp-pwe>
    </signaling-options>
    <vpn-network-accesses>
      <vpn-network-access>
        <id>l2vpn-p2p-ac1</id>
        <Interface-mtu>64</Interface-mtu>
        <connection>
       <encapsulation-type>vpn-common:dot1q</encapsulation-type>
          <dot1q-interface>
            <l2-access-type>vpn-common:dot1q</l2-access-type>
```

```
                <dot1q>
                 <physical-inf>GigabitEthernet0/0/0/1</physical-inf>
                  <c-vlan-id>601</c-vlan-id>
                  <rewrite>
                    <ingress>
                      <push/>
                      <dot1q>234</dot1q>
                      <mode>symmetric</mode>
                    </ingress>
                  </rewrite>
                </dot1q>
              </dot1q-interface>
            </connection>
          </vpn-network-access>
        </vpn-network-accesses>
      </vpn-node>
    </vpn-nodes>
  </vpn-service>
 </vpn-services>
</l2vpn-ntw>
```

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

Creating IETF-L2VPN-NM t-ldp Service with RSVP-TE Association

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 IETF-L2VPN-NM t-ldp service with RSVP-TE association:

1. The following is a sample payload to create the IETF-L2VPN-NM t-ldp service with RSVP-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>
          <ne-id>PIOSXR-0</ne-id>
          <te-service-mapping>
            <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>
              <ne-id>PIOSXR-1</ne-id>
              <te-service-mapping>
                <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 t-ldp Service with SR-TE Policy

To create IETF-L2VPN-NM t-ldp service with SR-TE policy

1. The following is a sample payload to create the IETF- L2VPN-NM t-ldp service with 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>
           <ne-id>PIOSXR-0</ne-id>
           <te-service-mapping>
             <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>
              <ne-id>PIOSXR-1</ne-id>
              <te-service-mapping>
                <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 t-ldp Service with Automated Assurance

Do the following:

1. The following is a sample payload to create the IETF- L2VPN-NM t-ldp 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/ietf-
l2vpn-ntw-cisco-augmentations">
        <monitoring-state>pause</monitoring-state>
        <profile-name>profile-A custom</profile-name>
        <rule-name>rule-A custom</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 t-ldp Service with Y1731

Y1731 association is currently supported only for XR devices.

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

```
<l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
  <y-1731-profile xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-
ntw-cisco-augmentations">
    <name>delay</name>
    <type>delay</type>
    <probe>
      <burst>
        <message-count>30</message-count>
        <message-period>100</message-period>
      </burst>
      <measurement-interval>60</measurement-interval>
      <frame-size>1000</frame-size>
      <priority>5</priority>
    </probe>
    <delay-params>
      <statistic>
        <type>delay-two-way</type>
      </statistic>
      <statistic>
        <type>delay-sd</type>
      </statistic>
      <statistic>
        <type>delay-ds</type>
      </statistic>
      <statistic>
        <type>jitter-two-way</type>
      </statistic>
      <statistic>
        <type>jitter-sd</type>
      </statistic>
      <statistic>
        <type>jitter-ds</type>
      </statistic>
    </delay-params>
    <schedule>
      <interval>5</interval>
      <duration>5</duration>
    </schedule>
    <bucket-details>
      <bucket-size>1</bucket-size>
      <bucket-archive>3</bucket-archive>
    </bucket-details>
  </y-1731-profile>
```

```
  <y-1731-profile xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-
ntw-cisco-augmentations">
    <name>loss</name>
    <type>synthetic-loss</type>
    <probe>
      <burst>
        <message-count>30</message-count>
        <message-period>100</message-period>
      </burst>
      <measurement-interval>60</measurement-interval>
      <frame-size>1000</frame-size>
      <priority>5</priority>
    </probe>
    <loss-params>
      <statistic>
        <type>loss-sd</type>
      </statistic>
      <statistic>
        <type>loss-ds</type>
      </statistic>
    </loss-params>
    <schedule>
      <interval>1</interval>
      <duration>1</duration>
    </schedule>
    <bucket-details>
      <bucket-size>1</bucket-size>
      <bucket-archive>3</bucket-archive>
    </bucket-details>
  </y-1731-profile>
  <vpn-services>
    <vpn-service>
      <vpn-id>l2nm-p2p-y-1731</vpn-id>
      <control-word>yes</control-word>
      <vpn-svc-type>vpn-common:t-ldp</vpn-svc-type>
      <vpn-nodes>
        <vpn-node>
          <vpn-node-id>PIOSXR-0</vpn-node-id>
          <ne-id>PIOSXR-0</ne-id>
          <signaling-options>
            <type>vpn-common:t-ldp</type>
            <t-ldp-pwe>
              <ac-pw-list>
                <peer-addr>198.18.1.5</peer-addr>
```

```
                <vc-id>1001</vc-id>
                <mpls-label>101</mpls-label>
              </ac-pw-list>
            </t-ldp-pwe>
          </signaling-options>
          <vpn-network-accesses>
            <vpn-network-access>
              <id>l2vpn-p2p-ac1</id>
              <Interface-mtu>65</Interface-mtu>
              <connection>
                <encapsulation-type>vpn-common:dot1q</encapsulation-
type>
                <dot1q-interface>
                  <l2-access-type>vpn-common:dot1q</l2-access-type>
                  <dot1q>
                    <physical-inf>GigabitEthernet0/0/0/1</physical-inf>
                    <c-vlan-id>601</c-vlan-id>
                    <rewrite>
                      <ingress>
                        <push/>
                        <dot1q>123</dot1q>
                        <mode>symmetric</mode>
                      </ingress>
                    </rewrite>
                  </dot1q>
                </dot1q-interface>
              </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/ietf-l2vpn-ntw-cisco-
augmentations">number</id-type>
                  <message-period>1s</message-period>
                  <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-ntw-cisco-
augmentations">
                    <name>delay</name>
                  </y-1731-profile>
                  <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-ntw-cisco-
augmentations">
                    <name>loss</name>
```

```
                  </y-1731-profile>
                </y-1731>
              </ethernet-service-oam>
            </vpn-network-access>
          </vpn-network-accesses>
        </vpn-node>
        <vpn-node>
          <vpn-node-id>PIOSXR-1</vpn-node-id>
          <ne-id>PIOSXR-1</ne-id>
          <signaling-options>
            <type>vpn-common:t-ldp</type>
            <t-ldp-pwe>
              <ac-pw-list>
                <peer-addr>198.18.1.4</peer-addr>
                <vc-id>1001</vc-id>
                <mpls-label>102</mpls-label>
              </ac-pw-list>
            </t-ldp-pwe>
          </signaling-options>
          <vpn-network-accesses>
            <vpn-network-access>
              <id>l2vpn-p2p-ac1</id>
              <Interface-mtu>64</Interface-mtu>
              <connection>
                <encapsulation-type>vpn-common:dot1q</encapsulation-
type>
                <dot1q-interface>
                  <l2-access-type>vpn-common:dot1q</l2-access-type>
                  <dot1q>
                    <physical-inf>GigabitEthernet0/0/0/1</physical-inf>
                    <c-vlan-id>601</c-vlan-id>
                    <rewrite>
                      <ingress>
                        <push/>
                        <dot1q>234</dot1q>
                        <mode>symmetric</mode>
                      </ingress>
                    </rewrite>
                  </dot1q>
                </dot1q-interface>
              </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/ietf-l2vpn-ntw-cisco-
augmentations">number</id-type>
                  <message-period>1s</message-period>
                  <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-ntw-cisco-
augmentations">
                    <name>delay</name>
                  </y-1731-profile>
                  <y-1731-profile
xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-ntw-cisco-
augmentations">
                    <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>
```

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

Creating IETF-L2VPN-NM EVPN-BGP Service

To create EVPN-BGP 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.

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>
  </resource-pools>
  <l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
    <id-pools>
      <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>
    </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>123</evi-id>
        <evi-source>124</evi-source>
        <evi-target>125</evi-target>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
</config>
```

To create the IETF-L2VPN-NM EVPN-BGP 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-svc-type>vpn-common:evpn-bgp</vpn-svc-type>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <ne-id>PIOSXR-0</ne-id>
            <multi-home>
              <esi-value>00.01.00.ac.ce.55.00.0a.00</esi-value>
```

```
            </multi-home>
            <signaling-options>
              <type>vpn-common:evpn-bgp</type>
              <evpn-bgp>
                <type>evpn-vpws</type>
              </evpn-bgp>
            </signaling-options>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>l2vpn-evpn-ac1</id>
                <connection>
                  <encapsulation-type>vpn-common:dot1q</encapsulation-
type>
                  <dot1q-interface>
                    <l2-access-type>vpn-common:dot1q</l2-access-type>
                    <dot1q>
                      <physical-inf>GigabitEthernet0/0/0/1</physical-inf>
                      <c-vlan-id>601</c-vlan-id>
                      <rewrite>
                        <ingress>
                          <push/>
                          <dot1q>123</dot1q>
                          <mode>symmetric</mode>
                        </ingress>
                      </rewrite>
                    </dot1q>
                  </dot1q-interface>
                </connection>
              </vpn-network-access>
            </vpn-network-accesses>
          </vpn-node>
          <vpn-node>
            <vpn-node-id>PIOSXR-1</vpn-node-id>
            <ne-id>PIOSXR-1</ne-id>
            <signaling-options>
              <type>vpn-common:evpn-bgp</type>
              <evpn-bgp>
                <type>evpn-vpws</type>
              </evpn-bgp>
            </signaling-options>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>l2vpn-evpn-ac1</id>
                <connection>
```

```
                    <encapsulation-type>vpn-common:dot1q</encapsulation-
type>
                    <dot1q-interface>
                      <l2-access-type>vpn-common:dot1q</l2-access-type>
                      <dot1q>
                        <physical-inf>GigabitEthernet0/0/0/1</physical-inf>
                        <c-vlan-id>601</c-vlan-id>
                        <rewrite>
                          <ingress>
                            <push/>
                            <dot1q>234</dot1q>
                            <mode>symmetric</mode>
                          </ingress>
                        </rewrite>
                      </dot1q>
                    </dot1q-interface>
                  </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 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
                                                                    POST
                                BACK     STATUS                     ACTION
```

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

Creating the IETF-L2VPN-NM Service with SR-TE Association

For every vpn-site, you can configure the preferred-path. 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 with SR-TE association:

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-nodes>
          <vpn-node>
            <vpn-node-id>PIOSXR-0</vpn-node-id>
            <ne-id>PIOSXR-0</ne-id>
            <te-service-mapping>
              <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>
            <ne-id>PIOSXR-1</ne-id>
            <te-service-mapping>
              <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>
```

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

Creating IETF-L2VPN-NM Service with SR-TE ODN

For every vpn-site, you can configure the SR-TE ODN policy for EVPN-BGP 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.

To modify the **parent-rr-route-policy** after associating the L2 EVPN VPWS SR-TE ODN, you must update the original value. For more information, see **Appendix B: Updating the Original Definition of the parent-route-policy.**

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

To create the IETF-L2VPN-NM service with SR-TE ODN:

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

```
 <config xmlns="http://tail-f.com/ns/config/1.0">
  <!-- Parent Route Policy must first be configured on device (L2VPN-
PARENT-RP-CLI.xml) -->
  <!-- L2NM (Uses l2vpn-route-policy definition from L2VPN-RP.xml) -->
  <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>
            <ne-id>PIOSXR-0</ne-id>
            <te-service-mapping>
              <te-mapping>
                <odn>
                  <route-policy>L2-RP-local</route-policy>
                  <attach-point>
                      <parent-rr-route-policy>L2-ATTACH</parent-rr-route-
policy>
                  </attach-point>
                </odn>
              </te-mapping>
            </te-service-mapping>
```

```
            </vpn-node>
            <vpn-node>
              <vpn-node-id>PIOSXR-1</vpn-node-id>
              <ne-id>PIOSXR-1</ne-id>
              <te-service-mapping>
                <te-mapping>
                  <odn>
                    <route-policy>L2-RP-remote</route-policy>
                    <attach-point>
                         <parent-rr-route-policy>L2-ATTACH</parent-rr-route-
policy>
                    </attach-point>
                  </odn>
                </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 Flat L3VPN Services

You can implement a custom L3VPN function pack to meet specific requirements. Use the Example Function Packs as a starting point or for design patterns.

The example Flat L3VPN service allows you to configure VPN, interface, BGP neighbor, and associate SR-TE policy.

To create Flat L3VPN service:

1. The following is a sample payload to create the Flat L3VPN service. For the yang model, see **Appendix C - Yang Models** in this documentation.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L3vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-
flat-L3vpn">
    <name>L3</name>
    <endpoint>
      <endpoint-name>cli-0</endpoint-name>
      <access-pe>PIOSXR-0</access-pe>
      <if-type>Loopback</if-type>
      <if-id>3</if-id>
      <pe-ip-addr>169.1.1.1/32</pe-ip-addr>
      <as-no>65001</as-no>
      <ce-pe-prot>
```

```
      <e-bgp>
        <neighbor-ipv4>169.1.1.2</neighbor-ipv4>
        <remote-as-ipv4>65002</remote-as-ipv4>
        <ebgp-multihop>
          <ttl-value>40</ttl-value>
          <mpls-deactivation>true</mpls-deactivation>
        </ebgp-multihop>
        <update-source>
          <sub-if-id>100</sub-if-id>
          <if-type>GigabitEthernet</if-type>
          <if-id>0/0/1/3</if-id>
        </update-source>
      </e-bgp>
    </ce-pe-prot>
    <vrf>
      <vrf-definition>L3VPN</vrf-definition>
      <route-distinguisher>1:1</route-distinguisher>
      <address-family>
        <address-family>ipv4</address-family>
        <redistribute-connected/>
        <metric>12</metric>
        <vpn-target>
          <rt-value>100:100</rt-value>
          <rt-type>both</rt-type>
        </vpn-target>
        <vpn-target>
          <rt-value>100:101</rt-value>
          <rt-type>export</rt-type>
        </vpn-target>
        <vpn-target>
          <rt-value>100:102</rt-value>
          <rt-type>import</rt-type>
        </vpn-target>
        <srv6>
          <locator-name>locv4</locator-name>
        </srv6>
      </address-family>
      <address-family>
        <address-family>ipv6</address-family>
        <redistribute-connected/>
        <metric>6</metric>
      </address-family>
    </vrf>
  </endpoint>
```

```
  </flat-L3vpn>
</config>
```

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

   The following table describes the L3VPN plan components.

Plan Component	Value	Description
Type	endpoint	This component represents one of the endpoints of L3VPN service.
State	cisco-flat-L3vpn-fp-nano-plan-services:config-apply	This represents the device configuration state in the nano-plan of a service.

```
admin@ncs% run show flat-L3vpn-plan

                                                                                                                                  POST
                BACK         STATUS                                                                                            ACTION
TYPE      NAME  TRACK  GOAL  CODE    STATE                                                  STATUS   WHEN                 ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------
self      self  false  -     -       init                                                   reached  2020-05-02T00:45:30  -    -
                                     ready                                                  reached  2020-05-02T00:45:33  -    -
endpoint  nc-0  false  -     -       init                                                   reached  2020-05-02T00:45:30  -    -
                                     cisco-flat-L3vpn-fp-nano-plan-services:config-apply    reached  2020-05-02T00:45:33  -    -
                                     ready                                                  reached  2020-05-02T00:45:33  -    -
```

Creating Flat L3VPN Service with Automated Assurance

Do the following:

1. The following is a sample payload to create the Flat L3VPN service with AA.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L3vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-L3vpn">
    <name>L3</name>
    <service-assurance>
        <monitoring-state>pause</monitoring-state>
        <profile-name>test custom</profile-name>
        <rule-name>test custom</rule-name>
      </service-assurance>
  </flat-L3vpn>
</config>
```

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

Creating L3VPN Extra Route Policy

Create a service custom template to create user specific route policies and then apply these policies to the route policy in the service. The custom template is created as part of day 0.

The extra policy name is applied to the main L3 route policy. These extra route policies can be created by normal custom templates or by using a pre-existing policy on the device.

1. The following is a sample payload to create the Flat L3VPN extra route policy.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <!-- L3VPN ROUTE POLICY WITH EXTRA ROUTE POLICY -->
  <l3vpn-route-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
tsdn-flat-L3vpn">
    <name>L3-RP-EP1</name>
    <color>
      <id>100</id>
      <ipv4>
        <prefix>1.1.1.1/32</prefix>
        <prefix>1.1.1.2/32</prefix>
      </ipv4>
    </color>
    <color>
      <id>101</id>
      <ipv4>
        <prefix>2.1.1.1/32</prefix>
        <prefix>2.1.1.2/32</prefix>
      </ipv4>
      <exclusive>true</exclusive>
    </color>
    <extra-policy>
      <name>RP-FOR-COMMUNITY</name>
      <operation>prepend</operation>
    </extra-policy>
    <extra-policy>
      <name>EXTRA-1</name>
      <operation>prepend</operation>
    </extra-policy>
    <extra-policy>
      <name>EXTRA-2</name>
      <operation>append</operation>
    </extra-policy>
  </l3vpn-route-policy>

    <endpoint>
      <endpoint-name>cli-0</endpoint-name>
      <access-pe>PIOSXR-0</access-pe>
      <if-type>Loopback</if-type>
      <if-id>3</if-id>
      <pe-ip-addr>169.1.1.1/32</pe-ip-addr>
```

```
      <as-no>65001</as-no>
      <ce-pe-prot>
        <e-bgp>
          <neighbor-ipv4>169.1.1.2</neighbor-ipv4>
          <remote-as-ipv4>65002</remote-as-ipv4>
          <ebgp-multihop>
            <ttl-value>40</ttl-value>
            <mpls-deactivation>true</mpls-deactivation>
          </ebgp-multihop>
          <update-source>
            <sub-if-id>100</sub-if-id>
            <if-type>GigabitEthernet</if-type>
            <if-id>0/0/1/3</if-id>
          </update-source>
        </e-bgp>
      </ce-pe-prot>
      <vrf>
        <vrf-definition>L3VPN</vrf-definition>
        <route-distinguisher>1:1</route-distinguisher>
        <address-family>
          <address-family>ipv4</address-family>
          <redistribute-connected/>
          <metric>12</metric>
          <vpn-target>
            <rt-value>100:100</rt-value>
            <rt-type>both</rt-type>
          </vpn-target>
          <vpn-target>
            <rt-value>100:101</rt-value>
            <rt-type>export</rt-type>
          </vpn-target>
          <vpn-target>
            <rt-value>100:102</rt-value>
            <rt-type>import</rt-type>
          </vpn-target>
        </address-family>
        <address-family>
          <address-family>ipv6</address-family>
          <redistribute-connected/>
          <metric>6</metric>
        </address-family>
      </vrf>
    </endpoint>
  </flat-L3vpn>
```

```
</config>
```

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

Configuring the Route Distinguisher for L3VPN Service

Define a unique identifier or a prefix to a route to distinguish to which VPN the route belongs to. The route distinguisher can either be a 16-bit AS number or a 32-bit IP address.

Enable or disable the **global-rd** flag to instruct NSO to apply the route distinguisher either under global VRF definition or under BGP → VRF respectively.

The following is an example to apply the route distinguisher under the global VRF.

```
admin@ncs% set cisco-flat-L3vpn-fp:cfp-configurations global-rd-enabled
[ok]
admin@ncs% commit
Commit complete.
```

Display the plan to view the status of the service.

```
admin@ncs% run show flat-L3vpn-plan
```

```
                                                        BACK                                                                                    POST ACTION
TYPE                                                    NAME      TRACK  GOAL  STATE                                                   STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                    self      false  -     init                                                    reached  2020-02-20T03:15:34  -    -
                                                                               ready                                                   reached  2020-02-20T03:15:38  -    -
endpoint                                                nc-0      false  -     init                                                    reached  2020-02-20T03:15:34  -    -
                                                                               cisco-flat-L3vpn-fp-nano-plan-services:config-apply     reached  2020-02-20T03:15:36  -    -
                                                                               ready                                                   reached  2020-02-20T03:15:36  -    -
endpoint                                                nc-1      false  -     init                                                    reached  2020-02-20T03:15:34  -    -
                                                                               cisco-flat-L3vpn-fp-nano-plan-services:config-apply     reached  2020-02-20T03:15:36  -    -
                                                                               ready                                                   reached  2020-02-20T03:15:36  -    -

[ok]
```

The following show examples of device configuration outputs when the global-rd flag is enabled and disabled.

Device configuration without global-rd-enabled flag

```
admin@ncs% show devices device P-0 config ipv4-bgp-cfg:bgp instance default
instance-as 0 four-byte-as 65001 vrfs vrf L3VPN vrf-global route-distinguisher

type      as;
as-xx     0;
as        1;
as-index 2;
[ok]
```

Device config with global-rd-enabled flag

```
admin@ncs% show devices device P-0 config infra-rsi-cfg:vrfs vrf L3VPN bgp-
global
```

```
route-distinguisher {
    type     as;
    as-xx    0;
    as       1;
    as-index 2;
}
[ok]
```

Creating IETF-L3VPN-NM Services

The IETF-L3VPN-NM service provides an IETF model overlay of the flat L3VPN configuration. It implements the **draft-ietf-opsawg-l3sm-l3nm-03** IETF 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 NE_ID>_<VPN_NETWORK_ACCESS ID> format.

You must create the Flat L3VPN service before you can create the IETF-L3VPN-NM service. For more information, see **Creating Flat L3VPN Services**.

To create IETF-L3VPN-NM service:

1. Create the L3vpn-route-policy and commit it in a separate transaction before you create the L3vpn-ntw service.

   When a L3vpn-route-policy is created, the corresponding routing-profile-identifier is automatically populated. Therefore, you do not require to manually configure the /l3vpn-ntw/vpn-profiles/valid-provider-identifiers/routing-profile-identifier.

   The following is a sample L3vpn-route-policy payload.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <l3vpn-route-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
tsdn-flat-L3vpn">
    <name>TEST_POLICY</name>
    <color>
      <id>100</id>
      <ipv4>
        <prefix>1.1.1.1/32</prefix>
        <prefix>1.1.1.2/32</prefix>
      </ipv4>
    </color>
    <color>
      <id>101</id>
      <ipv4>
        <prefix>2.1.1.1/32</prefix>
        <prefix>2.1.1.2/32</prefix>
      </ipv4>
    </color>
```

```
    </l3vpn-route-policy>
  </config>
```

2. The following is a sample payload to create the IETF-L3VPN-NM service. For the yang model, see **Appendix C - Yang Models** in this documentation.

```
  <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>
          <ie-profiles>
            <ie-profile>
              <ie-profile-id>ie_00</ie-profile-id>
              <rd>0:65100:87400024</rd>
              <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>
                  <export-policy>TEST_POLICY</export-policy>
                </vpn-policies>
              </vpn-targets>
            </ie-profile>
```

```
        </ie-profiles>
        <vpn-nodes>
          <vpn-node>
            <ne-id>PIOSXR-1</ne-id>
            <local-autonomous-system>65001</local-autonomous-system>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>23</id>
                <port-id>GigabitEthernet1/1/1/1</port-id>
                <connection>
                  <encapsulation-type>tagged-int</encapsulation-type>
                  <tagged-interface>
                    <type>dot1q</type>
                    <dot1q-vlan-tagged>
                      <cvlan-id>1234</cvlan-id>
                    </dot1q-vlan-tagged>
                  </tagged-interface>
                </connection>
                <ip-connection>
                  <ipv4>
                    <address-allocation-type xmlns:l3vpn-
svc="urn:ietf:params:xml:ns:yang:ietf-l3vpn-svc">l3vpn-svc:static-
address</address-allocation-type>
                    <static-addresses>
                      <primary-address>test-ipv4-address</primary-
address>
                      <address>
                        <address-id>test-ipv4-address</address-id>
                        <provider-address>10.1.1.1</provider-address>
                        <prefix-length>24</prefix-length>
                      </address>
                    </static-addresses>
                  </ipv4>
                </ip-connection>
                <routing-protocols>
                  <routing-protocol>
                    <id>TEST_PROTO</id>
                    <type>bgp</type>
                    <bgp>
                      <address-family>ipv4</address-family>
                      <neighbor>10.1.1.1</neighbor>
                      <peer-autonomous-system>65003</peer-autonomous-
system>
                      <multihop>11</multihop>
                      <mpls-deactivation>true</mpls-deactivation>
```

```
                        <update-source>
                          <if-type>GigabitEthernet</if-type>
                          <if-id>3</if-id>
                          <sub-if-id>200</sub-if-id>
                        </update-source>
                        <srv6
xmlns="http://cisco.com/ns/nso/fp/examples/ietf-l3vpn-ntw-cisco-
augmentations">
                          <address-family>
                            <name>ipv4</name>
                            <locator-name>locv4</locator-name>
                          </address-family>
                        </srv6>
                      </bgp>
                    </routing-protocol>
                  </routing-protocols>
                </vpn-network-access>
              </vpn-network-accesses>
              <node-ie-profile>ie_00</node-ie-profile>
            </vpn-node>
            <vpn-node>
              <ne-id>PIOSXR-0</ne-id>
              <local-autonomous-system>65001</local-autonomous-system>
              <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>
              <vpn-network-accesses>
                <vpn-network-access>
                  <id>23</id>
                  <port-id>GigabitEthernet1/1/1/1</port-id>
                  <connection>
                    <encapsulation-type>tagged-int</encapsulation-type>
                    <tagged-interface>
                      <type>dot1q</type>
                      <dot1q-vlan-tagged>
                        <cvlan-id>1234</cvlan-id>
```

```
                    </dot1q-vlan-tagged>
                  </tagged-interface>
                </connection>
                <ip-connection>
                  <ipv6>
                    <address-allocation-type xmlns:l3vpn-
svc="urn:ietf:params:xml:ns:yang:ietf-l3vpn-svc">l3vpn-svc:static-
address</address-allocation-type>
                    <static-addresses>
                      <primary-address>test-ipv6-address</primary-
address>
                      <address>
                        <address-id>test-ipv6-address</address-id>
                        <provider-address>2001:db8::1</provider-address>
                        <prefix-length>32</prefix-length>
                      </address>
                    </static-addresses>
                  </ipv6>
                </ip-connection>
                <routing-protocols>
                  <routing-protocol>
                    <id>TEST_PROTO</id>
                    <type>bgp</type>
                    <bgp>
                      <address-family>ipv6</address-family>
                      <neighbor>2001:db8::2</neighbor>
                      <peer-autonomous-system>65003</peer-autonomous-
system>
                      <multihop>12</multihop>
                      <mpls-deactivation>false</mpls-deactivation>
                      <update-source>
                        <if-type>GigabitEthernet</if-type>
                        <if-id>4</if-id>
                        <sub-if-id>400</sub-if-id>
                      </update-source>
                    </bgp>
                  </routing-protocol>
                </routing-protocols>
              </vpn-network-access>
            </vpn-network-accesses>
            <node-ie-profile>ie_00</node-ie-profile>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
```

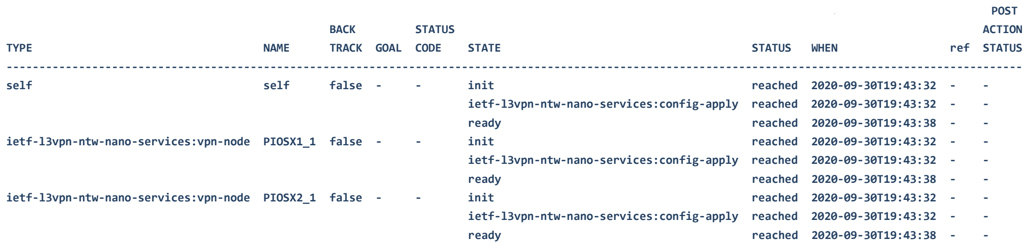
```
  </l3vpn-ntw>
</config>
```

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

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

Plan Component	Value	Description
Type	ietf-l3vpn-ntw-nano-services:vpn-node	This component represents the node of IETF-L3VPN-NM service.
State	ietf-l3vpn-ntw-nano-services:config-apply	This represents acceptance of service intent and respective config application on device in NSO CDB.

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



```
                                                                                                                                            POST
                                               BACK        STATUS                                                                          ACTION
TYPE                                  NAME     TRACK GOAL  CODE   STATE                                     STATUS  WHEN                ref STATUS
-------------------------------------------------------------------------------------------------------------------------------------------------
self                                  self     false -     -      init                                      reached 2020-09-30T19:43:32 -   -
                                                                  ietf-l3vpn-ntw-nano-services:config-apply reached 2020-09-30T19:43:32 -   -
                                                                  ready                                     reached 2020-09-30T19:43:38 -   -
ietf-l3vpn-ntw-nano-services:vpn-node PIOSX1_1 false -     -      init                                      reached 2020-09-30T19:43:32 -   -
                                                                  ietf-l3vpn-ntw-nano-services:config-apply reached 2020-09-30T19:43:32 -   -
                                                                  ready                                     reached 2020-09-30T19:43:38 -   -
ietf-l3vpn-ntw-nano-services:vpn-node PIOSX2_1 false -     -      init                                      reached 2020-09-30T19:43:32 -   -
                                                                  ietf-l3vpn-ntw-nano-services:config-apply reached 2020-09-30T19:43:32 -   -
                                                                  ready                                     reached 2020-09-30T19:43:38 -   -
```

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>test custom</profile-name>
          <rule-name>test 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-TE Services

Create the IETF-TE service to push the RSVP-TE configurations to devices. The RSVP-TE configuration allows you to set up a bidirectional tunnel by configuring both source and destination devices.

The unnumbered IPV4 Loopback0 → Loopback interface configuration is considered day -1 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>PIOSXR-0</head-end>
        <destination>222.2.2.2</destination>
        <tail-end>PIOSXR-1</tail-end>
        <bidirectional>true</bidirectional>
        <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>
        <p2p-primary-paths>
          <p2p-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>
          </p2p-primary-path>
          <p2p-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>
          </p2p-primary-path>
          <p2p-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>
          </p2p-primary-path>
          <p2p-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>
          </p2p-primary-path>
        </p2p-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

                                                                                                                                                POST
                                BACK        STATUS                                                                                          ACTION
TYPE         NAME      TRACK  GOAL  CODE    STATE                                                STATUS   WHEN                 ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------
self         self      false  -     -       init                                                 reached  2020-08-24T23:45:50  -    -
                                             ready                                                reached  2020-08-24T23:45:53  -    -
source       111.1.1.1 false  -     -       init                                                 reached  2020-08-24T23:45:50  -    -
                                             ietf-te-fp-tunnel-nano-plan-services:config-apply    reached  2020-08-24T23:45:50  -    -
                                             ready                                                reached  2020-08-24T23:45:53  -    -
destination  222.2.2.2 false  -     -       init                                                 reached  2020-08-24T23:45:50  -    -
                                             ietf-te-fp-tunnel-nano-plan-services:config-apply    reached  2020-08-24T23:45:50  -    -
                                             ready                                                reached  2020-08-24T23:45:53  -    -
```

3. Verify the configuration is pushed on the devices.

Working with Action Commands

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 entries of the service and the plan may display the 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.

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

Arguments	Description
service-type	This is the type of the service, such as SR-TE ODN.
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 you do not provide the device name, it cleans up the entire service.
local-site-only	This can point to an unreachable local-site that is replaced by another reachable local site device in the service.
remote-site-only	This can point to an unreachable remote-site that is removed from the service.
endpoint	This is the endpoint name in the service. If you do not provide any other parameter, it cleans up the entire service.
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.
vpn-node	Name of the vpn-node to clean up. If specified, the clean up is performed only for the vpn-node. This is an optional parameter.
vpn-network-access-id	Name of the vpn-network-access ID for input vpn-node to clean up. This is a mandatory parameter if the vpn-node parameter is specified.

SR-TE CFP

This topic discusses how to execute the service cleanup action on the SR-ODN service and the SR-policy service.

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

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
```

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.

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
```

Example Function Packs

This topic discusses how to execute the service Cleanup action on the Example services.

L2VPN Example Services

You can clean up the service only for a remote site. The local site is a mandatory entity for the existence of L2VPN service and hence does not support the Cleanup action.

Use the Cleanup action to clean up a specific remote site in the service or to clean up the entire service. Cleaning up the entire service cleans up all the remote sites and local sites in the service.

To clean up the service for a specific remote site

```
admin@ncs% request flat-L2vpn-actions cleanup service P2P-DOT1Q remote-site-
only PIOSXR-1
Value for 'no-networking' [false,true]: 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 service: P2P-DOT1Q
 Cleaning up L2vpn internal remote-site service: P2P-DOT1Q PIOSXR-1
 Removed all plan components
 Removing side-effect queue: /ncs:side-effect-queue/side-effect{75}
 Removed side-effects
 Removed kickers
 Cleanup Successful for L2vpn internal remote-site service: P2P-DOT1Q  PIOSXR-1

 Removed all external plan components
 Removed side-effects
 Cleanup Successful
```

To clean up the entire L2VPN service

The following command shows how to clean up the service.

```
admin@ncs% request flat-L2vpn-actions cleanup service P2P-DOT1Q 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 service: P2P-DOT1Q


Cleaning up L2vpn internal local-site service: P2P-DOT1Q PIOSXR-0
 Removed all plan components
 Removing side-effect queue: /ncs:side-effect-queue/side-effect{58}
 Removing side-effect queue: /ncs:side-effect-queue/side-effect{7}
 Removed side-effects
 Removed kickers
 Cleanup Successful for L2vpn internal local-site service: P2P-DOT1Q PIOSXR-0


 Removed all external plan components
 Removing zombie service: /ncs:zombies/ncs:service{"/flat-L2vpn[name='P2P-
DOT1Q']"}
 Removed zombie service
 Removed side-effects
 Removed kickers
 Removing plan path: /cisco-flat-L2vpn-fp:flat-L2vpn-plan{P2P-DOT1Q}
 Removed plan path
 Cleanup Successful
```

L2NM Example Service

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

```
admin@ncs% request 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
```

L3VPN Example Services

Perform the Cleanup action to clean up a specific endpoint or the entire service. Cleaning up the entire service cleans up all the endpoints in the service.

To clean up the service per device

```
admin@ncs% request flat-L3vpn-actions cleanup service L3 endpoint cli-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 L3vpn service: L3Cleaning up L3VPN Internal Services: L3
 Removed all internal plan components
 Removing side-effect queue: /ncs:side-effect-queue/side-effect{17}
 Removed side-effects
 Removed kickers
 Cleanup Successful for L3VPN Internal Services
 Removed all plan components
 Removing zombie service: /ncs:zombies/ncs:service{"/cisco-flat-L3vpn-fp:flat-
L3vpn[name='L3']"}
 Removed zombie service
 Removed side-effectsCustom Template Per NODE
 Removed kickers
 Cleanup Successful
```

To clean up the entire service

```
admin@ncs% request flat-L3vpn-actions cleanup service L3 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 L3vpn service: L3
 Removed all plan components
 Removed side-effects
 Removed kickers
 Cleanup Successful
```

L3NM Example Services

Use the Cleanup action to clean up the L3NM service as follows:

```
admin@ncs% request 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
 Removing service /cisco-flat-L3vpn-fp:flat-L3vpn{L3NM-0-65008740-internal}
 Removed service /cisco-flat-L3vpn-fp:flat-L3vpn{L3NM-0-65008740-
internal}Cleaning up L3VPN Internal Services: 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
```

IETF-TE Example Services

Use the Cleanup action to clean the IETF-TE service as follows:

```
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
```

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 (such as SR-ODN) to be recovered.
service	Name of the service to be recovered.
local-site-only / remote-site-only	Name of the local-site/remote-site to recover. If specified, only the local-site/remote-site recovery is performed. This is an optional parameter.
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.
vpn-network-access-id	Name of the vpn-network-access ID for input vpn-node to be recovered. This is a mandatory parameter if the vpn-node parameter is specified.
device	Name of the device under a given service that needs to be recovered. If no device name is provided, 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.

SR-TE CFP

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.

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

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

```
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 a successful recovery of the service.

```
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
```

SR-Policy Services

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

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

                                                                                                                                                  POST
                                                                                                                                                  ACTION
TYPE                                                  NAME     BACK TRACK  GOAL  STATUS CODE  STATE                                          STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                  self     false  -     -            init                                           reached  2020-08-17T19:41:17  -    -
                                                                                          ready                                          failed   2020-08-17T19:41:51  -    -
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end  PIOSXR-0  false  -     TSDN-SR-301  init                                           reached  2020-08-17T19:41:17  -    -
                                                                                          cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply  reached  2020-08-17T19:41:17  -    -
                                                                                          ready                                          failed   2020-08-17T19:41:51  -    -
```

```
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."
```

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 a successful recovery of the service.

```
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
```

Example Function Packs

This topic discusses how to execute the error recovery action on the example services.

L2VPN Example Services

The following is an example of a failed plan for both local site and remote site.

```
admin@ncs% run show flat-L2vpn-plan P2P-DOT1Q plan

                                                                                                                                                  POST
                                                                                                                                                  ACTION
                    BACK                                                                                                                
TYPE        NAME    TRACK  GOAL  STATUS CODE    STATE                                                STATUS   WHEN                ref  STATUS
-------------------------------------------------------------------------------------------------------------------------------------------------
self        self    false  -     -             init                                                 reached  2020-08-18T19:12:04  -    -
                                                ready                                                failed   2020-08-18T19:12:08  -    -
local-site  PIOSXR-0 false  -     TSDN-L2VPN-301 init                                                reached  2020-08-18T19:12:04  -    -
                                                cisco-flat-L2vpn-fp-nano-plan-services:config-apply  reached  2020-08-18T19:12:04  -    -
                                                ready                                                failed   2020-08-18T19:12:08  -    -
remote-site PIOSXR-1 false  -     TSDN-L2VPN-301 init                                                reached  2020-08-18T19:12:04  -    -
                                                cisco-flat-L2vpn-fp-nano-plan-services:config-apply  reached  2020-08-18T19:12:04  -    -
                                                ready                                                failed   2020-08-18T19:12:08  -    -

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
```

Local-site and remote-site devices come back up and you can request error recovery on service. The plan is successful after the service is recovered successfully.

The following shows error recovery for a service.

```
admin@ncs% request flat-L2vpn-actions error-recovery service P2P-DOT1Q 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 L2vpn service: P2P-DOT1Q
Recovered create failure on PIOSXR-0
Recovery Complete for L2VPN Internal Service
Recovered create failure on PIOSXR-1
Recovery Complete for L2VPN Internal Service
Recovery Complete
```

Error recovery for a local site

The following is an example of a failed plan for a local-site.

```
admin@ncs% run show flat-L2vpn-plan P2P-DOT1Q plan
```

```
                                                                                                                  POST
                       BACK                                                                                                ACTION
  TYPE         NAME      TRACK  GOAL  STATUS CODE     STATE                                         STATUS   WHEN                 ref  STATUS
  ------------------------------------------------------------------------------------------------------------------------------------------
  self         self      false  -     -               init                                          reached  2020-08-18T18:42:13  -    -
                                                      ready                                         failed   2020-08-18T18:42:18  -    -
  local-site   PIOSXR-0  false  -     TSDN-L2VPN-301  init                                          reached  2020-08-18T18:42:13  -    -
                                                      cisco-flat-L2vpn-fp-nano-plan-services:config-apply  reached  2020-08-18T18:42:13  -    -
                                                      ready                                         failed   2020-08-18T18:42:18  -    -
  remote-site  PIOSXR-1  false  -     -               init                                          reached  2020-08-18T18:42:13  -    -
                                                      cisco-flat-L2vpn-fp-nano-plan-services:config-apply  reached  2020-08-18T18:42:13  -    -
                                                      ready                                         reached  2020-08-18T18:42:18  -    -
```

```
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
```

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

```
admin@ncs% request flat-L2vpn-actions error-recovery service P2P-DOT1Q local-
site-only 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 L2vpn service: P2P-DOT1Q
Recovered create failure on PIOSXR-0
Recovery Complete for L2VPN Internal Service
Recovery Complete
```

Error recovery action for a remote-site

Error recovery action for L2vpn P2P service

The following is a sample failed plan for a remote site.

```
admin@ncs% run show flat-L2vpn-plan P2P-DOT1Q plan

                                                                                                                                                  POST
                     BACK                                                                                                                       ACTION
TYPE        NAME     TRACK  GOAL  STATUS CODE    STATE                                                  STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------
self        self     false  -     -             init                                                   reached  2020-08-18T19:18:05  -    -
                                                 ready                                                  failed   2020-08-18T19:18:11  -    -
local-site  PIOSXR-0 false  -     -             init                                                   reached  2020-08-18T19:18:05  -    -
                                                 cisco-flat-L2vpn-fp-nano-plan-services:config-apply    reached  2020-08-18T19:18:05  -    -
                                                 ready                                                  reached  2020-08-18T19:18:11  -    -
remote-site PIOSXR-1 false  -     TSDN-L2VPN-301 init                                                   reached  2020-08-18T19:18:05  -    -
                                                 cisco-flat-L2vpn-fp-nano-plan-services:config-apply    reached  2020-08-18T19:18:05  -    -
                                                 ready                                                  failed   2020-08-18T19:18:11  -    -

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 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 error recovery action for the l2vpn-p2p remote site is as follows:

```
admin@ncs% request flat-L2vpn P2P-DOT1Q flat-L2vpn-p2p remote-site action
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 L2vpn service: P2P-DOT1Q
Recovered create failure on PIOSXR-1
Recovery Complete for L2VPN Internal Service
Recovery Complete
```

Similarly, you can perform error recovery on L2vpn P2P local-site.

```
request flat-L2vpn P2P-DOT1Q flat-L2vpn-p2p local-site action error-recovery
```

Error recovery action for L2vpn EVPN service

The following is a sample failed plan for a remote site.

```
admin@ncs% run show flat-L2vpn-plan COLT-L2vpn-dynamic plan

                                                                                                                                  POST
                    BACK                                                                                                           ACTION
TYPE        NAME    TRACK  GOAL  STATUS CODE    STATE                                                STATUS   WHEN                ref  STATUS
-------------------------------------------------------------------------------------------------------------------------------------------
self        self    false  -     -             init                                                 reached  2020-08-18T19:27:50  -    -
                                                ready                                                failed   2020-08-18T19:27:55  -    -
local-site  PIOSXR-0 false  -     -             init                                                 reached  2020-08-18T19:27:50  -    -
                                                cisco-flat-L2vpn-fp-nano-plan-services:config-apply  reached  2020-08-18T19:27:50  -    -
                                                ready                                                reached  2020-08-18T19:27:55  -    -
remote-site PIOSXR-1 false  -     TSDN-L2VPN-301 init                                                reached  2020-08-18T19:27:50  -    -
                                                cisco-flat-L2vpn-fp-nano-plan-services:config-apply  reached  2020-08-18T19:27:50  -    -
                                                ready                                                failed   2020-08-18T19:27:55  -    -


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 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 error recovery action for the l2vpn-evpn remote site is as follows:

```
admin@ncs% request flat-L2vpn COLT-L2vpn-dynamic flat-L2vpn-evpn-vpws remote-
site action 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 L2vpn service: COLT-L2vpn-dynamic
Recovered create failure on PIOSXR-1
Recovery Complete for L2VPN Internal Service
Recovery Complete
```

Similarly, you can perform error recovery on L2vpn EVPN local-site.

```
request flat-L2vpn COLT-L2vpn-dynamic flat-L2vpn-evpn-vpws local-site action
error-recovery
```

L2VPN-NM Example 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

                                                                                                                                                  POST
                                                BACK                                                                                            ACTION
TYPE                                  NAME      TRACK  GOAL  STATUS CODE     STATE                                   STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------
self                                  self      false  -     -               init                                    reached  2020-09-22T19:09:56  -    -
                                                                             ready                                   failed   2020-09-22T19:10:01  -    -
ietf-l2vpn-ntw-nano-services:vpn-node PIOSXR-0  false  -     TSDN-L2VPN-301  init                                    reached  2020-09-22T19:09:56  -    -
                                                                             ietf-l2vpn-ntw-nano-services:config-apply reached  2020-09-22T19:09:56  -    -
                                                                             ready                                   failed   2020-09-22T19:10:01  -    -
ietf-l2vpn-ntw-nano-services:vpn-node PIOSXR-1  false  -     TSDN-L2VPN-301  init                                    reached  2020-09-22T19:09:56  -    -
                                                                             ietf-l2vpn-ntw-nano-services:config-apply reached  2020-09-22T19:09:56  -    -
                                                                             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 is recovered successfully.

The following shows error recovery for the service.

```
admin@ncs% request 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 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

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

L3VPN Example Services

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

```
admin@ncs% run show flat-L3vpn-plan

                                                                                                                  POST
                  BACK                                                                                                                  ACTION
TYPE      NAME    TRACK  GOAL  STATUS CODE     STATE                                                    STATUS   WHEN                 ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------
self      self    false  -     -              init                                                     reached  2020-08-19T17:52:15  -    -
                                              ready                                                    failed   2020-08-19T18:14:34  -    -
endpoint  cli-0   false  -     TSDN-L3VPN-301 init                                                     reached  2020-08-19T17:52:15  -    -
                                              cisco-flat-L3vpn-fp-nano-plan-services:config-apply      reached  2020-08-19T17:52:15  -    -
                                              ready                                                    failed   2020-08-19T18:14:32  -    -
endpoint  nc-0    false  -     -              init                                                     reached  2020-08-19T17:52:15  -    -
                                              cisco-flat-L3vpn-fp-nano-plan-services:config-apply      reached  2020-08-19T17:52:15  -    -
                                              ready                                                    reached  2020-08-19T17:52:23  -    -

 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 endpoint cli-0
  code                TSDN-L3VPN-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 flat-L3vpn-actions error-recovery service L3 endpoint cli-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: L3
Recovered create failure on PIOSXR-0
Recovery Complete for L3VPN Internal Services
Recovery Complete
```

Error recovery action under a service - Create failure

```
admin@ncs% request flat-L3vpn L3 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: L3
Recovered create failure on PIOSXR-0
Recovery Complete for L3VPN Internal Services
Recovery Complete
```

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

Error recovery action under a service node - Create failure

```
admin@ncs% request flat-L3vpn L3 endpoint cli-0 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: L3
Recovered create failure on PIOSXR-0
Recovery Complete for L3VPN Internal Services
Recovery Complete
```

L3VPN-NM Example Services

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

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

                                                                                POST
                                BACK                                                                    ACTION
```

```
TYPE                                NAME      TRACK  GOAL  STATUS CODE     STATE                                    STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------
self                                self      false  -     -               init                                     reached  2020-09-30T19:44:45  -    -
                                                                          ietf-l3vpn-ntw-nano-services:config-apply reached  2020-09-30T19:44:45  -    -
                                                                          ready                                    failed   2020-09-30T19:44:50  -    -
ietf-l3vpn-ntw-nano-services:vpn-node  PIOSX0_1  false  -     TSDN-L3VPN-301  init                                     reached  2020-09-30T19:44:45  -    -
                                                                          ietf-l3vpn-ntw-nano-services:config-apply reached  2020-09-30T19:44:45  -    -
                                                                          ready                                    failed   2020-09-30T19:44:50  -    -
ietf-l3vpn-ntw-nano-services:vpn-node  PIOSX0_2  false  -     -               init                                     reached  2020-09-30T19:44:45  -    -
                                                                          ietf-l3vpn-ntw-nano-services:config-apply reached  2020-09-30T19:44:45  -    -
                                                                          ready                                    reached  2020-09-30T19:44:50  -    -
```

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

```
TYPE      NAME      CODE           SEVERITY  RECOMMENDED ACTION                                          CONTEXT NAME        CONTEXT MSG
-----------------------------------------------------------------------------------------------------------------------------------------
endpoint  PIOSX0  TSDN-L3VPN-301    ERROR     Check device connectivity from NSO and perform recovery steps.  Device unreachable  Failed to
connect to device PIOSX0: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state
```

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 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.

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

                                                                                                                                                  POST
                                                  BACK                                                                                          ACTION
  TYPE                                  NAME      TRACK  GOAL  STATUS CODE     STATE                                    STATUS   WHEN                ref  STATUS
  ------------------------------------------------------------------------------------------------------------------------------------------------------------------
  self                                  self      false  -     -              init                                     reached  2020-09-30T19:44:45  -    -
                                                                              ietf-l3vpn-ntw-nano-services:config-apply reached  2020-09-30T19:44:45  -    -
                                                                              ready                                    failed   2020-09-30T19:51:24  -    -
  ietf-l3vpn-ntw-nano-services:vpn-node  PIOSX0_0  false  -     TSDN-L3VPN-301 init                                     reached  2020-09-30T19:44:45  -    -
                                                                              ietf-l3vpn-ntw-nano-services:config-apply reached  2020-09-30T19:44:45  -    -
                                                                              ready                                    failed   2020-09-30T19:51:24  -    -
  ietf-l3vpn-ntw-nano-services:vpn-node  PIOSX0_1  false  -     -              init                                     reached  2020-09-30T19:44:45  -    -
                                                                              ietf-l3vpn-ntw-nano-services:config-apply reached  2020-09-30T19:44:45  -    -
                                                                              ready                                    reached  2020-09-30T19:44:50  -    -


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

TYPE      NAME      CODE          SEVERITY  RECOMMENDED ACTION                                                    CONTEXT NAME        CONTEXT MSG
-----------------------------------------------------------------------------------------------------------------------------------------------------
endpoint  PIOSX0  TSDN-L3VPN-301  ERROR     Check device connectivity from NSO and perform recovery steps.  Device unreachable  Failed to
connect to device PIOSX0: connection refused: NEDCOM CONNECT: Connection refused (Connection refused) in new state
```

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 PIOSX0 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
```

IETF-TE Example Services

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

```
admin@ncs> show te tunnels tunnel-plan

                                                                                                                                                POST
                                BACK                                                                                                                                                ACTION
TYPE         NAME      TRACK  GOAL  STATUS CODE      STATE                                                STATUS   WHEN                ref  STATUS
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self         self      false  -     -               init                                                 reached  2020-08-25T00:11:24  -    -
                                                     ready                                                failed   2020-08-25T00:11:26  -    -
source       111.1.1.1 false  -     TSDN-IETF-TE-301 init                                                 reached  2020-08-25T00:11:24  -    -
                                                     ietf-te-fp-tunnel-nano-plan-services:config-apply  reached  2020-08-25T00:11:24  -    -
                                                     ready                                                failed   2020-08-25T00:11:26  -    -
destination  222.2.2.2 false  -     -               init                                                 reached  2020-08-25T00:11:24  -    -
                                                     ietf-te-fp-tunnel-nano-plan-services:config-apply  reached  2020-08-25T00:11:24  -    -
                                                     ready                                                reached  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
```

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 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***.

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

SR-TE CFP

This section discusses how to execute the get-modifications action on the SR-TE CFP services.

SR-ODN Services

```
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;
+                                           }
+                                       }
+                                       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;
+                               }
+                           }
+                       }
+                   }
                }
            }
        }

    }
}
```

SR-Policy Services

get-modifications action for service

```
admin@ncs> request sr-te policies policy SR-Policy-1 get-modifications
cli {
    local-node {
        data  devices {
                  device PIOSXR-0 {
                      config {
+                         segment-routing {
```

```
+                 traffic-eng {
+                     segment-list mysidlist {
+                         index 1 {
+                             mpls {
+                                 label 17001;
+                             }
+                         }
+                     }
+                     policy srte_c_100_ep_7.7.7.7 {
+                         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;
```

```
                +                                                    }
                +                                              }
                +                                        }
                +                                    }
                +                                }
                +                            }
                +                        }
                +                  }
                +             }
                          }
                     }
                 }

    }
}
```

Example Function Packs

L2VPN Example Services

In L2VPN, perform the get-modifications action at the service level.

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

```
admin@ncs> request flat-L2vpn L2vpn01-nc get-modifications
cli {
    local-node {
        data  devices {
                   device PN73-0 {
                       config {
                           interface-configurations {
                +                interface-configuration act TenGigE0/0/0/6 {
                +                    description "T-SDN Interface";
                +                }
                +                interface-configuration act TenGigE0/0/0/6.100 {
                +                    interface-mode-non-physical l2-transport;
                +                    description l2vpn-static-01;
                +                    ethernet-service {
                +                        encapsulation {
                +                            outer-tag-type match-dot1q;
                +                            outer-range1-low 100;
                +                        }
                +                    }
                +                }
                           }
                           l2vpn {
                               database {
```

```
                                xconnect-groups {
              +                     xconnect-group p2p-nso {
              +                         p2p-xconnects {
              +                             p2p-xconnect STATIC-PN73-0 {
              +                                 pseudowire-neighbor-
pseudowire-ids {
              +                                     pseudowire-neighbor-
pseudowire-id 14.0.0.0 100 {
              +                                         mpls-static-labels {
              +                                             local-static-
label 101;
              +                                             remote-static-
label 202;
              +                                         }
              +                                         class mpls-nso;
              +                                     }
              +                                 }
              +                                 attachment-circuits {
              +                                     attachment-circuit
TenGigE0/0/0/6.100 {
              +                                         enable;
              +                                     }
              +                                 }
              +                             }
              +                         }
              +                     }
                                }
                                pseudowire-classes {
              +                     pseudowire-class mpls-nso {
              +                         mpls-encapsulation {
              +                             enable;
              +                         }
              +                         enable;
              +                     }
                                }
                            }
              +             enable;
                        }
                    }
                }
                device PN73-1 {
                    config {
                        interface-configurations {
              +             interface-configuration act TenGigE0/0/0/7 {
              +                 description "T-SDN Interface";
```

```
+                 }
+                 interface-configuration act TenGigE0/0/0/7.200 {
+                     interface-mode-non-physical l2-transport;
+                     description l2vpn-static-01;
+                     ethernet-service {
+                         encapsulation {
+                             outer-tag-type match-dot1q;
+                             outer-range1-low 200;
+                         }
+                     }
+                 }
                 }
                 l2vpn {
                     database {
                         xconnect-groups {
+                         xconnect-group p2p-nso {
+                             p2p-xconnects {
+                                 p2p-xconnect STATIC-PN73-1 {
+                                     pseudowire-neighbor-
pseudowire-ids {
+                                         pseudowire-neighbor-
pseudowire-id 12.0.0.0 100 {
+                                             mpls-static-labels {
+                                                 local-static-
label 202;
+                                                 remote-static-
label 101;
+                                             }
+                                             class mpls-nso;
+                                         }
+                                     }
+                                     attachment-circuits {
+                                         attachment-circuit
TenGigE0/0/0/7.200 {
+                                             enable;
+                                         }
+                                     }
+                                 }
+                             }
+                         }
                         pseudowire-classes {
+                         pseudowire-class mpls-nso {
+                             mpls-encapsulation {
+                                 enable;
```

```
+                                }
+                                enable;
+                            }
                        }
                    }
+                   enable;
                }
            }
        }
    }

    }
}
```

L2VPN-NM Example 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 reverse
cli {
    local-node {
        data -flat-L2vpn L2NM-l2nm-p2p-internal {
              -    service-type p2p;
              -    flat-L2vpn-p2p {
              -        pw-id 1001;
              -        local-site {
              -            pe PN73-0;
              -            if-type GigabitEthernet;
              -            if-id 0/0/0/1;
              -            if-encap dot1q;
              -            vlan-id 601;
              -            sub-if-id 601;
              -            rewrite {
              -                ingress {
              -                    push;
              -                    dot1q 123;
              -                    mode symmetric;
              -                }
              -            }
              -            mtu 65;
              -            xconnect-group-name l2nm-p2p;
              -            p2p-name l2nm-p2p;
              -            control-word yes;
```

```
-            pw-class l2nm-p2p;
-            xconnect-local-ip 198.18.1.4;
-            xconnect-remote-ip 198.18.1.5;
-            mpls-local-label 101;
-            mpls-remote-label 102;
-        }
-        remote-site {
-            pe PN73-1;
-            if-type GigabitEthernet;
-            if-id 0/0/0/1;
-            if-encap dot1q;
-            vlan-id 601;
-            sub-if-id 601;
-            rewrite {
-                ingress {
-                    push;
-                    dot1q 234;
-                    mode symmetric;
-                }
-            }
-            mtu 64;
-            xconnect-group-name l2nm-p2p;
-            p2p-name l2nm-p2p;
-            control-word yes;
-            pw-class l2nm-p2p;
-        }
-    }
-}
 devices {
     device PN73-0 {
         config {
             interface-configurations {
-                interface-configuration act
GigabitEthernet0/0/0/1 {
-                    description "T-SDN Interface";
-                }
-                interface-configuration act
GigabitEthernet0/0/0/1.601 {
-                    mtus {
-                        mtu sub_vlan {
-                            mtu 65;
-                        }
-                    }
-                    interface-mode-non-physical l2-transport;
-                    description "T-SDN Interface";
```

```
-                    ethernet-service {
-                        encapsulation {
-                            outer-tag-type match-dot1q;
-                            outer-range1-low 601;
-                        }
-                        rewrite {
-                            rewrite-type push1;
-                            outer-tag-type match-dot1q;
-                            outer-tag-value 123;
-                        }
-                    }
-                }
                }
                l2vpn {
                    database {
                        xconnect-groups {
-                        xconnect-group l2nm-p2p {
-                            p2p-xconnects {
-                                p2p-xconnect l2nm-p2p {
-                                    pseudowire-neighbor-
pseudowire-ids {
-                                        pseudowire-neighbor-
pseudowire-id 198.18.1.5 1001 {
-                                            mpls-static-labels {
-                                                local-static-
label 101;
-                                                remote-static-
label 102;
-                                            }
-                                            class l2nm-p2p;
-                                        }
-                                    }
-                                    attachment-circuits {
-                                        attachment-circuit
GigabitEthernet0/0/0/1.601 {
-                                            enable;
-                                        }
-                                    }
-                                }
-                            }
-                        }
                        }
                        pseudowire-classes {
-                        pseudowire-class l2nm-p2p {
-                            mpls-encapsulation {
```

```
-                                   enable;
-                                   control-word enable;
-                               }
-                               enable;
-                           }
                        }
                    }
-                   enable;
                }
            }
        }
        device PN73-1 {
            config {
                interface-configurations {
-                   interface-configuration act
GigabitEthernet0/0/0/1 {
-                       description "T-SDN Interface";
-                   }
-                   interface-configuration act
GigabitEthernet0/0/0/1.601 {
-                       mtus {
-                           mtu sub_vlan {
-                               mtu 64;
-                           }
-                       }
-                       interface-mode-non-physical l2-transport;
-                       description "T-SDN Interface";
-                       ethernet-service {
-                           encapsulation {
-                               outer-tag-type match-dot1q;
-                               outer-range1-low 601;
-                           }
-                           rewrite {
-                               rewrite-type push1;
-                               outer-tag-type match-dot1q;
-                               outer-tag-value 234;
-                           }
-                       }
-                   }
                }
                l2vpn {
                    database {
                        xconnect-groups {
-                           xconnect-group l2nm-p2p {
-                               p2p-xconnects {
```

```
                -                                   p2p-xconnect l2nm-p2p {
                -                                       pseudowire-neighbor-
pseudowire-ids {
                -                                           pseudowire-neighbor-
pseudowire-id 198.18.1.4 1001 {
                -                                               mpls-static-labels {
                -                                                   local-static-
label 102;
                -                                                   remote-static-
label 101;
                -                                               }
                -                                               class l2nm-p2p;
                -                                           }
                -                                       }
                -                                       attachment-circuits {
                -                                           attachment-circuit
GigabitEthernet0/0/0/1.601 {
                -                                               enable;
                -                                           }
                -                                       }
                -                                   }
                -                               }
                -                           }
                                        }
                                        pseudowire-classes {
                -                           pseudowire-class l2nm-p2p {
                -                               mpls-encapsulation {
                -                                   enable;
                -                                   control-word enable;
                -                               }
                -                               enable;
                -                           }
                                        }
                                    }
                -                   enable;
                                }
                            }
                        }
                    }

    }
}
```

L3VPN Example Services

Perform the get-modifications action at the service level.

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

```
admin@ncs> request flat-L3vpn L3 get-modifications
cli {
    local-node {
        data  devices {
                    device P-0 {
                        config {
                            interface-configurations {
              +                 interface-configuration act Loopback3 {
              +                     interface-virtual;
              +                     description "T-SDN Interface";
              +                     vrf L3VPN;
              +                     ipv4-network {
              +                         addresses {
              +                             primary {
              +                                 address 169.1.1.1;
              +                                netmask 255.255.255.255;
              +                             }
              +                         }
              +                     }
              +                 }
                            }
                            vrfs {
              +                 vrf L3VPN {
              +                     vpn-id {
              +                         vpn-oui 18;
              +                         vpn-index 20;
              +                     }
              +                     create;
              +                     afs {
              +                         af ipv4 unicast default {
              +                             create;
              +                             bgp {
              +                                 import-route-targets {
              +                                     route-targets {
              +                                       route-target as {
              +                                           as-or-four-byte-as 0 100 101 0;
              +                                           as-or-four-byte-as 0 100 102 0;
              +                                           as-or-four-byte-as 0 200 100 0;
              +                                         }
              +                                     }
              +                                 }
              +                                 export-route-targets {
```

```
+                                   route-targets {
+                                       route-target as {
+                                        as-or-four-byte-as 0 200 100 0;
+                                       }
+                                   }
+                               }
+                           }
+                       }
+                       af ipv6 unicast default {
+                           create;
+                       }
+                   }
+                   bgp-global {
+                       route-distinguisher {
+                           type as;
+                           as-xx 0;
+                           as 1;
+                           as-index 2;
+                       }
+                   }
+               }
            }
            bgp {
+               instance default {
+                   instance-as 0 {
+                       four-byte-as 65001 {
+                           vrfs {
+                               vrf L3VPN {
+                                   vrf-global {
+                                       vrf-global-afs {
+                                           vrf-global-af ipv4-unicast {
+                                               enable;
+                                               connected-routes {
+                                                   default-metric 10;
+                                               }
+                                           }
+                                           vrf-global-af ipv6-unicast {
+                                               enable;
+                                               connected-routes {
+                                                   default-metric 5;
+                                               }
+                                           }
+                                       }
+                                       exists;
```

```
+                                    }
+                                    vrf-neighbors {
+                                        vrf-neighbor 169.1.1.2 {
+                                         vrf-neighbor-afs {
+                                          vrf-neighbor-af ipv4-unicast {
+                                                    activate;
+                                               route-policy-in PASS_ALL;
+                                              route-policy-out PASS_ALL;
+                                                }
+                                            }
+                                            ebgp-multihop {
+                                                max-hop-count 29;
+                                                mpls-deactivation true;
+                                            }
+                                            remote-as {
+                                                as-xx 0;
+                                                as-yy 65002;
+                                            }
+                                            update-source-interface
GigabitEthernet0/0/2/4.300;
+                                        }
+                                    }
+                                }
+                            }
+                            bgp-running;
+                        }
+                    }
+                }
                }
                routing-policy {
                    route-policies {
+                       route-policy PASS_ALL {
+                           rpl-route-policy "route-policy PASS_ALL
                   pass
                   end-policy
                 ";
+                      }
                    }
                }
            }
        }
        device PIOSXR-0 {
            config {
                vrf {
```

```
+                vrf-list L3VPN {
+                    vpn {
+                        id 11:13;
+                    }
+                    address-family {
+                        ipv4 {
+                            unicast {
+                                import {
+                                    route-target {
+                                        address-list 100:100;
+                                        address-list 100:102;
+                                    }
+                                }
+                                export {
+                                    route-target {
+                                        address-list 100:100;
+                                        address-list 100:101;
+                                    }
+                                }
+                            }
+                        }
+                        ipv6 {
+                            unicast {
+                            }
+                        }
+                    }
+                }
             }
             interface {
+                Loopback 3 {
+                    description "T-SDN Interface";
+                    vrf L3VPN;
+                    ipv4 {
+                        address {
+                            ip 169.1.1.1;
+                            mask 255.255.255.255;
+                        }
+                    }
+                }
+                Loopback 4 {
+                    description "T-SDN Interface";
+                    vrf L3VPN;
+                    ipv4 {
+                        address {
```

```
+                           ip 170.1.1.1;
+                           mask 255.255.255.255;
+                       }
+                   }
+               }
            }
+           route-policy PASS_ALL {
+           }
            router {
                bgp {
+                   bgp-no-instance 65001 {
+                       vrf L3VPN {
+                           address-family {
+                               ipv4 {
+                                   unicast {
+                                       redistribute {
+                                           connected {
+                                               metric 12;
+                                           }
+                                       }
+                                   }
+                               }
+                               ipv6 {
+                                   unicast {
+                                       redistribute {
+                                           connected {
+                                               metric 6;
+                                           }
+                                       }
+                                   }
+                               }
+                           }
+                           neighbor 169.1.1.2 {
+                               remote-as 65002;
+                               ebgp-multihop {
+                                   ttl-value 40;
+                                   mpls;
+                               }
+                               update-source {
+                                   GigabitEthernet-subinterface {
+                                       GigabitEthernet 0/0/1/3.100;
+                                   }
+                               }
+                               address-family {
```

```
+                                   ipv4 {
+                                       unicast {
+                                           route-policy in {
+                                               name PASS_ALL;
+                                           }
+                                           route-policy out {
+                                               name PASS_ALL;
+                                           }
+                                       }
+                                   }
+                               }
+                           }
+                           neighbor 170.1.1.2 {
+                               remote-as 65004;
+                               address-family {
+                                   ipv4 {
+                                       unicast {
+                                           route-policy in {
+                                               name PASS_ALL;
+                                           }
+                                           route-policy out {
+                                               name PASS_ALL;
+                                           }
+                                       }
+                                   }
+                               }
+                           }
+                       }
+                   }
                   }
               }
           }
       }
    }
}
```

L3VPN-NM Example Services

The following is a sample command to execute the get-modifications action on the L3VPN-NM example 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;
+                               }
+                           }
+                       }
                    }
                }
                extcommunity-set {
+                   opaque COLOR_100 {
+                       set 100;
+                   }
+                   opaque COLOR_101 {
+                       set 101;
+                   }
                }
+               route-policy PASS_ALL {
+                   value pass;
+               }
+               route-policy SET_COLORv4_TEST_POLICY {
+                   value "  if destination in (1.1.1.1/32,
1.1.1.2/32) then\r
                  set extcommunity color COLOR_100\r
                endif\r
                if destination in (2.1.1.1/32, 2.1.1.2/32) then\r
                  set extcommunity color COLOR_101\r
                endif\r
              ";
+               }
                router {
                    bgp {
+                       bgp-no-instance 65001 {
+                           vrf 0-65008740 {
+                               rd 65100:87400024;
+                               address-family {
+                                   ipv6 {
+                                       unicast {
+                                       }
+                                   }
+                               }
+                               neighbor 2001:db8::2 {
+                                   remote-as 65003;
+                                   ebgp-multihop {
+                                       ttl-value 12;
+                                   }
+                                   address-family {
```

```
+                                       ipv6 {
+                                           unicast {
+                                               route-policy in {
+                                                   name PASS_ALL;
+                                               }
+                                               route-policy out {
+                                                   name PASS_ALL;
+                                               }
+                                           }
+                                       }
+                                   }
+                               }
+                           }
+                       }
                       }
                   }
               }
           }
           device PIOSXR-1 {
               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;
+                                          }
```

```
+                                   }
+                               }
+                           }
+                       }
+                   }
                }
                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;
+                       ipv4 {
+                           address {
+                               ip 10.1.1.1;
+                               mask 255.255.255.0;
+                           }
+                       }
+                   }
                }
                }
                extcommunity-set {
+               opaque COLOR_100 {
+                   set 100;
+               }
+               opaque COLOR_101 {
+                   set 101;
+               }
                }
+               route-policy PASS_ALL {
+               value pass;
+               }
+               route-policy SET_COLORv4_TEST_POLICY {
+               value "  if destination in (1.1.1.1/32,
1.1.1.2/32) then\r
                  set extcommunity color COLOR_100\r
                endif\r
```

```
  if destination in (2.1.1.1/32, 2.1.1.2/32) then\r
    set extcommunity color COLOR_101\r
  endif\r
";
+            }
             router {
                 bgp {
+                    bgp-no-instance 65002 {
+                        vrf 0-65008740 {
+                            rd 65100:87400024;
+                            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;
+                                            }
+                                        }
+                                    }
+                                }
+                            }
+                        }
+                    }
                 }
             }
         }
     }
 }
+cisco-flat-L3vpn-fp:flat-L3vpn L3NM-0-65008740-internal {
+    endpoint PIOSXR-0 {
+        access-pe PIOSXR-0;
```

```
+        if-type GigabitEthernet;
+        if-id 1/1/1/1;
+        pe-ipv6-addr 2001:db8::1/32;
+        as-no 65001;
+        vlan-id 1234;
+        ce-pe-prot {
+            e-bgp {
+                neighbor-ipv6 2001:db8::2;
+                remote-as-ipv6 65003;
+                ebgp-multihop {
+                    ttl-value 12;
+                }
+            }
+        }
+        vrf {
+            vrf-definition 0-65008740;
+            route-distinguisher 65100:87400024;
+            address-family ipv6 {
+                vpn-target 65010:17405 {
+                    rt-type both;
+                }
+                vpn-target 65010:17406 {
+                    rt-type both;
+                }
+            }
+        }
+        sr-te {
+            route-policy TEST_POLICY;
+        }
+    }
+    endpoint PIOSXR-1 {
+        access-pe PIOSXR-1;
+        custom-template CT-CLI-banner {
+            variable BANNER_TEXT {
+                value Welcome_B;
+            }
+        }
+        if-type GigabitEthernet;
+        if-id 1/1/1/1;
+        pe-ip-addr 10.1.1.1/24;
+        as-no 65002;
+        vlan-id 1234;
+        ce-pe-prot {
+            e-bgp {
```

```
+                neighbor-ipv4 10.1.1.1;
+                remote-as-ipv4 65003;
+                ebgp-multihop {
+                    ttl-value 11;
+                }
+            }
+        }
+        vrf {
+            vrf-definition 0-65008740;
+            route-distinguisher 65100:87400024;
+            address-family ipv4 {
+                vpn-target 65010:17401 {
+                    rt-type both;
+                }
+                vpn-target 65010:17402 {
+                    rt-type both;
+                }
+                vpn-target 65010:17403 {
+                    rt-type import;
+                }
+                vpn-target 65010:17404 {
+                    rt-type export;
+                }
+            }
+        }
+        sr-te {
+            route-policy TEST_POLICY;
+        }
+    }
+}

    }
}
```

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;
+                        }
+                    }
+                }
                }
            }
        }
    }
}
```

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***.

SR-TE CFP

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.

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 a SR-policy service.

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
```

Example Function Packs

This section discusses how to execute the re-deploy action on the example services.

L2VPN Example Services

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

Re-deploy

```
admin@ncs> request flat-L2vpn L2vpn01-nc re-deploy
```

Re-deploy reconcile

```
admin@ncs> request flat-L2vpn L2vpn01-nc re-deploy reconcile
```

L2VPN-NM Example 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
```

L3VPN Example Services

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

Re-deploy

```
admin@ncs> request flat-L3vpn L3 re-deploy
```

Re-deploy reconcile

```
admin@ncs> request flat-L3vpn L3 re-deploy reconcile
```

L3VPN-NM Example Services

The following sample commands show how to execute the service re-deploy actions on 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
```

IETF-TE Example Services

The following sample commands show how to execute the service re-deploy actions on 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
```

Using Custom-Templates

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

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 section **Appendix D – Sample Custom-Template Payloads** 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-Iosxr-syslog** 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
```

Applying custom-template at 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/off at any time and custom 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
```

Multi-Vendor 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 Multi-Vendors Support Services (MVSS) thus allows you to provide services to the devices or NED packages you already have.

T-SDN FP Bundle supports multi-vendors 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**.

You must configure the dynamic-device mapping for these NEDs. To create multi-vendor 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 the Cisco IOSXR device. 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'}]
```

SR-TE CFP

Use the information in this section to support multi-vendors services for SR-ODN service and the SR-Policy service with IOSXR Netconf NED or the IOSXE CLI NED.

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

Note: Make sure the **nso-<NSO_release_version>-tsdn-<TSDN_release_version>/core-fp-packages/ncs-<NSO_release_version>-sr-te-multi-vendors- xxx.tar.gz** package is installed on NSO. For more information, see the **Cisco T-SDN FP Bundle Installation Guide**.

Supporting Multi-Vendors for IOSXR Netconf Services for SR-TE CFP

This section discusses the support for IOSXR Netconf NED for the SR-ODN service and the SR-Policy service.

To support multi-vendors 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.

```
<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_netconf-7.3:cisco-iosxr_netconf-7.3</ned-id>
      <python-impl-class-name>sr_te_multi_vendors.NativeXR</python-impl-class-
name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting Multi-Vendors for IOSXE CLI Services

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

To support multi-vendors 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-6.74:cisco-ios-cli-6.74</ned-id>
      <python-impl-class-name>sr_te_multi_vendors.IosXE</python-impl-class-
name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Example Function Packs

Use the information in this section to support multi-vendors services for SR-ODN service and the SR-Policy service with IOSXR Netconf NED or the IOSXE CLI NED.

To create the multi-vendors services, you must copy the multi-vendors package to the TSDN FP Bundle installation folder, establish the dynamic-device mapping, and create the SR-TE service on the Netconf device.

Note: Make sure the **nso-<NSO_release_version>-tsdn-<TSDN_release_version>/core-fp-packages/ncs-<NSO_release_version>-sr-te-multi-vendors-xxx.tar.gz** package is installed on your NSO. For more information, see the **Cisco T-SDN FP Bundle Installation Guide**.

Supporting Multi-Vendors for IOSXR Netconf Services for Example Services

This section discusses the support for IOSXR Netconf NED for the example services.

Supporting Multi-Vendors for L2VPN/L2NM IOSXR Netconf Services

To support multi-vendors for L2VPN/L2NM IOSXR Netconf services, configure the **cisco-flat-l2vpn-fp** dynamic-device mapping for IOSXR Netconf as follows. Make sure to provide the correct NED-ID.

Note: Since L2NM is dependent on the L2VPN cfp-configuration dynamic-device mapping, the procedure to support multi-vendors is the same for both these services.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-L2vpn">
    <dynamic-device-mapping>
      <ned-id>cisco-iosxr_netconf-7.3:cisco-iosxr_netconf-7.3</ned-id>
<python-impl-class-name>flat_l2vpn_multi_vendors.NativeXR</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting Multi-Vendors for L3VPN/L3NM Netconf Services

To support multi-vendors for L3VPN IOSXR Netconf services, configure **the cisco-flat-l3vpn-fp** dynamic-device mapping for IOSXR Netconf 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 multi-vendors is the same for both these services.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-L3vpn">
    <dynamic-device-mapping>
      <ned-id>cisco-iosxr_netconf-7.3:cisco-iosxr_netconf-7.3</ned-id>
      <python-impl-class-name>flat_l3vpn_multi_vendors.NativeXR</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting Multi-Vendors for IETF-TE Netconf Services

To support multi-vendors 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_netconf-7.3:cisco-iosxr_netconf-7.3</ned-id>
 <python-impl-class-name>rsvp_te_multi_vendors.NativeXR</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting Multi-Vendors for IOSXE CLI Services

This section discusses the multi-vendors support for IOSXE CLI NED for the example services.

Supporting Multi-Vendors for L2VPN/L2NM IOSXE CLI Services

To support multi-vendors for L2VPN IOSXE CLI services, configure the **cisco-flat-l2vpn-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 multi-vendors is the same for both these services.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-L2vpn">
    <dynamic-device-mapping>
      <ned-id>cisco-ios-cli-6.74:cisco-ios-cli-6.74</ned-id>
      <python-impl-class-name>flat_l2vpn_multi_vendors.IosXE</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

Supporting Multi-Vendors for L3VPN/L3NM IOSXE CLI Services

To support multi-vendors 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 multi-vendors is the same for both these services.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <cfp-configurations xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-L3vpn">
    <dynamic-device-mapping>
      <ned-id>cisco-ios-cli-6.74:cisco-ios-cli-6.74</ned-id>
      <python-impl-class-name>flat_l3vpn_multi_vendors.IosXE</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

In the L3VPN service, you can configure Maximum Transmission unit (MTU) and apply it to an interface with or without the Bridge Domain Interface (BDI).

When applying MTU to an interface without BDI, the MTU is attached directly under the main interface as shown below:

```
interface GigabitEthernet2
 description l2vpn-static-01
 mtu 1600
 no ip address
 negotiation auto
 no mop enabled
```

```
 no mop sysid
!
```

When applying mtu coupled with a BDI interface, the MTU configuration is attached to the BDI interfaces shown below:

```
interface GigabitEthernet0/0/4
 no ip address
 media-type auto-select
 negotiation auto
 service instance 200 ethernet
  encapsulation dot1q 200
  bridge-domain 100
 !
!
interface BDI100
 vrf forwarding L3VPN
 ip address 169.1.1.1 255.255.255.240
 ip mtu 1600
!
```

Note: For CSR8kv devices, set the **interface BDI → mtu** to highest MTU range as a day-0 to extend the range for IP MTU under BDI.

Supporting Multi-Vendors for IETF-TE IOSXE CLI Services

To support multi-vendors 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.

```
<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-6.74:cisco-ios-cli-6.74</ned-id>
 <python-impl-class-name>rsvp_te_multi_vendors.IosXE</python-impl-class-name>
    </dynamic-device-mapping>
  </cfp-configurations>
</config>
```

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***.

Deleting Services

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

Deleting SR-TE CFP 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
```

Deleting Example Services

Delete all L2VPN services

```
admin@ncs>delete flat-L2vpn
```

Delete all L3VPN services

```
admin@ncs>delete flat-L3vpn
```

Delete all IETF-TE services

```
admin@ncs>delete te tunnels tunnel
```

Delete all L2VPN-NM services

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

Delete all L3VPN-NM services

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

Notifications

SR-TE CFP generates the following notifications that communicate important information about the various events. Use this information to modify and examine the payload content for a specific event.

SR-TE CFP

This topic discusses the notifications generated for the policy service and the ODN service in SR-TE CFP.

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
    }
}
```

Example Function Packs

Flat L2VPN Notifications

This topic includes the create and delete notifications for L2VPN P2P and L2VPN VPWS.

The Flat L2VPN P2P notifications and Flat L2VPN VPWS notifications are similar, except for the service **/flat-L2vpn[name=]** parameter.

The notifications for Flat L2VPN P2P are shown below:

Create Notification

```
notification {
    eventTime 2020-08-27T17:33:18.229859+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component PIOSXR-0
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-08-27T17:33:18.23001+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component PIOSXR-1
        state init
        operation created
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:33:18.23005+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component PIOSXR-1
        state cisco-flat-L2vpn-fp-nano-plan-services:config-apply
        operation created
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:33:18.230089+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component PIOSXR-1
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-08-27T17:33:21.621153+00:00
```

```
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component self
        state ready
        operation modified
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:33:21.621266+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component PIOSXR-0
        state ready
        operation modified
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:33:21.621404+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        component PIOSXR-1
        state ready
        operation modified
        status reached
    }
}
```

Delete Notifications

```
notification {
    eventTime 2020-08-27T10:32:08.322619+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-DOT1Q']
        operation deleted
    }
}
```

The following notification displays if the delete operation fails.

```
 notification {
    eventTime 2020-08-27T10:36:32.85181+00:00
    plan-state-change {
        service /flat-L2vpn[name='P2P-UNTAG-SUB-IF']
```

```
        operation deleted
        status failed
    }
}
```

Automated Assurance Notification

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 2021-05-04T22:18:10.341957+00:00
    service-assurance-config-change {
        service /cisco-flat-L2vpn-fp:flat-L2vpn[name=flat-L2vpn-service]
        operation modified
    }
}
```

Flat L2VPN-NM Notifications

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

Since L2NM uses Flat L2VPN FP as an internal stacked service, Flat L2VPN notifications are also displayed.

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.

```
admin@ncs% run show notification stream service-aa-changes
notification {
    eventTime 2021-05-04T22:18:10.341957+00:00
    service-assurance-config-change {
        service /cisco-flat-L2vpn-fp:flat-L2vpn[name=L2NM-l2nm-service-
internal]
        operation modified
    }
}
notification {
    eventTime 2021-05-04T22:18:10.352061+00:00
    service-assurance-config-change {
        service /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service[vpn-id=l2nm-
service]
        operation modified
    }
}
```

Flat L3VPN Notifications

This topic includes the create and delete notifications for L3VPN service.

Create Notification

```
notification {
    eventTime 2020-08-27T17:54:53.170062+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component self
        state init
        operation created
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:54:53.170141+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component self
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-08-27T17:54:53.170309+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component cli-0
        state init
        operation created
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:54:53.17036+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component cli-0
        state cisco-flat-L3vpn-fp-nano-plan-services:config-apply
        operation created
        status reached
    }
```

```
}
notification {
    eventTime 2020-08-27T17:54:53.170465+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component cli-0
        state ready
        operation created
        status not-reached
    }
}
notification {
    eventTime 2020-08-27T17:54:56.380997+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component self
        state ready
        operation modified
        status reached
    }
}
notification {
    eventTime 2020-08-27T17:54:56.381194+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        component cli-0
        state ready
        operation modified
        status reached
    }
}
```

Delete Notifications

```
notification {
   eventTime 2020-08-27T10:55:47.112117+00:00
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        operation deleted
    }
}
```

The following notification displays if the delete operation fails.

```
notification {
  eventTime 2020-08-27T10:56:46.89771+00:00
```

```
    plan-state-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name='L3']
        operation deleted
        status failed
    }
}
```

Automated Assurance Notification

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 2021-05-04T22:18:10.341957+00:00
    service-assurance-config-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name=flat-L3vpn-service]
        operation modified
    }
}
```

Flat L3VPN-NM Notifications

This topic includes the create notifications, delete notifications, and AA notifications for L3VPN-NM service.

Since L3NM uses Flat L3VPN FP as an internal stacked service, Flat L3VPN notifications are also displayed.

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 2021-05-04T22:18:10.341957+00:00
    service-assurance-config-change {
        service /cisco-flat-L3vpn-fp:flat-L3vpn[name=L3NM-l3nm-service-
internal]
        operation modified
```

```
    }
}
notification {
    eventTime 2021-05-04T22:18:10.352061+00:00
    service-assurance-config-change {
        service /l3vpn-ntw:l3vpn-ntw/vpn-services/vpn-service[vpn-id=l3nm-
service]
        operation modified
    }
}
```

IETF-TE Notifications

This topic includes the create and delete 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
    }
}
```

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 following tables shows the packages in the **Core Function Pack** category and the **Example Packages** category.

The Core Function Pack packages are required to install SR-TE CFP. The Example packages comprises the packages for the Example Function Packs (such as L2VPN) you choose to install.

Core Function Pack Packages	
Package Category	**Packages**
T-SDN FP Bundle packages	ncs-5.5.2.9-cisco-sr-te-cfp-3.0.0.tar.gz ncs-5.5.2.9-cisco-sr-te-cfp-internal-3.0.0.tar.gz
T-SDN FP Bundle common packages	ncs-5.5.2.3-core-fp-plan-notif-generator-1.0.6.tar.gz ncs-5.5.2.3-custom-template-utils-2.0.6.tar.gz ncs-5.5.2.7-core-fp-common-1.26.0.tar.gz ncs-5.5.2.9-cisco-tsdn-core-fp-common-3.0.0.tar.gz
NEDs	IOSXR CLI NED: ncs-5.5.2-cisco-iosxr-7.33.12.tar.gz IOSXE CLI NED: ncs-5.5.2-cisco-ios-6.74.8.tar.gz IOSXR NC NED: ncs-5.5.2-cisco-iosxr_netconf-7.3.2.tar.gz ncs-5.5.2-cisco-iosxr_netconf-7.4.1.tar.gz
T-SDN FP Bundle multi-vendor with IOSXR NETCONF NED and IOSXE CLI NED	ncs-5.5.2.9-sr-te-multi-vendors-3.0.0.tar.gz

Example Packages	
Package Category	**Package Name**
Automated assurance package for L2NM,L3NM,flat-L2 and flat-L3	ncs-5.5.2.9-cisco-aa-service-assurance-EXAMPLE-3.0.0.tar.gz

L2VPN example package with IOSXR CLI NED	ncs-5.5.2.9-cisco-flat-L2vpn-fp-EXAMPLE-3.0.0.tar.gz ncs-5.5.2.9-cisco-flat-L2vpn-fp-internal-EXAMPLE-3.0.0.tar.gz
L2VPN multi-vendor example with IOSXR NETCONF NED and IOSXE CLI NED	ncs-5.5.2.9-flat-l2vpn-multi-vendors-EXAMPLE-3.0.0.tar.gz
L3VPN example package with IOSXR CLI NED	ncs-5.5.2.9-cisco-flat-L3vpn-fp-EXAMPLE-3.0.0.tar.gz ncs-5.5.2.9-cisco-flat-L3vpn-fp-internal-EXAMPLE-3.0.0.tar.gz
L3VPN multi-vendor example with IOSXR NETCONF NED and IOSXE CLI NED	ncs-5.5.2.9-flat-l3vpn-multi-vendors-EXAMPLE-3.0.0.tar.gz
Resource Manager for L2NM	ncs-5.5-resource-manager-3.5.4.tar.gz
Shared Delete Handler package for L2/L3 interfaces	ncs-5.5.2.3-core-fp-delete-tag-service-1.0.5.tar.gz
Northbound/External service package for handling T-SDN Flat L2VPN service creation via IETF L2NM interface	ncs-5.5.2.9-ietf-l2vpn-nm-EXAMPLE-3.0.0.tar.gz
Northbound/External service package for handling T-SDN Flat L3VPN service creation via IETF L3NM interface	ncs-5.5.2.9-ietf-l3vpn-nm-EXAMPLE-3.0.0.tar.gz
Example service package for handling IETF-TE service creation with IOSXR CLI NED	ncs-5.5.2.9-ietf-te-fp-EXAMPLE-3.0.0.tar.gz ncs-5.5.2.9-cisco-rsvp-te-fp-EXAMPLE-3.0.0.tar.gz
Package for handling RSVP-TE configurations on non-supported function pack device with IOSXR NETCONF NED and IOSXE CLI	ncs-5.5.2.9-rsvp-te-multi-vendors-EXAMPLE-3.0.0.tar.gz

Appendix B: Updating the Original Definition of the parent-route-policy

In the L2VPN/NM service, the SR-TE ODN local-route-policy is attached to an existing Day0 parent-route-policy on the device. The definition of this Day0 parent-route-policy is independent of the definition of the L2 local-route-policy attachment. You can, however, update this original definition of the Day0 parent-route-policy as described in the following procedure.

To update the original definition of the parent-route-policy:

1. From the NSO CLI config mode, unhide tsdn.

```
admin@ncs% unhide tsdn
[ok][2020-11-17 15:33:22]
```

2. (*Optional*) View the original **rr-parent-route-policy** value. This is the original definition of the parent-route-policy.

```
admin@ncs% show l2vpn-rr-parent-route-policy L2-ATTACH PIOSXR-0 original-rr
original-rr "ORIGINAL RP VALUE HERE";
[ok][2020-11-17 15:42:26]

admin@ncs% show devices device PIOSXR-0 config route-policy L2-ATTACH
value "  LOCAL L2 RP VALUE HERE\r\nORIGINAL RP VALUE HERE";
[ok]
```

3. Update the **rr-parent-route-policy** with the new value.

```
admin@ncs% set l2vpn-rr-parent-route-policy L2-ATTACH PIOSXR-0 original-rr "NEW ORIGINAL RP VALUE HERE"
[ok]
admin@ncs% commit
Commit complete.
[ok]
```

4. Verify the original **rr-parent-route-policy** is updated with the new value.

   Note: The L2 local-route-policy attachment does not change after updating the original rr-parent-route-policy with the new value.

```
admin@ncs% show devices device PIOSXR-0 config route-policy L2-ATTACH
value "  LOCAL L2 RP VALUE HERE\r\nNEW ORIGINAL RP VALUE HERE";
[ok]
```

Appendix C - Yang Models

This topic contains the yang models for the SR-TE CFP services and the Example Function Packs.

SR-TE CFP

This topic discusses the yang models for SR-ODN and SR-Policy services.

- **SR-ODN Yang Models**
- **SR-Policy 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 head-end* [name]
       |  |  +--rw name    -> /core-fp-common:dispatch-map/device
       |  +--rw maximum-sid-depth?   uint8
       |  +--rw color                uint32
       |  +--rw bandwidth?           uint32
       |  +--rw dynamic!
       |  |  +--rw metric-type?     enumeration
       |  |  +--rw metric-margin!
       |  |  |  +--rw (metric-margin)?
       |  |  |     +--:(absolute)
       |  |  |     |  +--rw absolute?   uint32
       |  |  |     +--:(relative)
       |  |  |        +--rw relative?   uint32
       |  |  +--rw pce!
       |  |  +--rw flex-alg?        uint32
       |  |  +--rw disjoint-path!
       |  |  |  +--rw type        enumeration
```

```
      |  |  |  +--rw group-id    uint16
      |  |  |  +--rw sub-id?     uint16
      |  |  |  +--rw source?     inet:ipv4-address
      |  |  +--rw affinity!
      |  |     +--rw rule* [action]
      |  |        +--rw action    enumeration
      |  |        +--rw color*    string
      |  +--rw source-address?      inet:ip-address
```

SR-ODN Service Plan Model

```
module: cisco-sr-te-cfp-sr-odn
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 failed?             empty
          |  |  +--ro error-info!
          |  |  |  +--ro message?     string
          |  |  |  +--ro log-entry?   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
```

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 head-end* [name]
       |  |  +--rw name    -> /core-fp-common:dispatch-map/device
       |  +--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 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 disjoint-path!
       |  |     |        |  +--rw type        enumeration
       |  |     |        |  +--rw group-id    uint16
       |  |     |        |  +--rw sub-id?     uint16
       |  |     |        |  +--rw source?     inet:ipv4-address
       |  |     |        +--rw affinity!
       |  |     |        |  +--rw rule* [action]
       |  |     |        |     +--rw action    enumeration
       |  |     |        |     +--rw color*    string
       |  |     |        +--rw segments!
       |  |     |           +--rw sid-algorithm?   uint16
```

```
|  |     +--:(dynamic-path)
|  |        +--rw dynamic!
|  |           +--rw metric-type?     enumeration
|  |           +--rw metric-margin!
|  |           |  +--rw (metric-margin)?
|  |           |     +--:(absolute)
|  |           |     |  +--rw absolute?   uint32
|  |           |     +--:(relative)
|  |           |        +--rw relative?   uint32
|  |           +--rw pce!
|  |           +--rw constraints
|  |              +--rw sid-limit?       uint32
|  |              +--rw disjoint-path!
|  |              |  +--rw type        enumeration
|  |              |  +--rw group-id    uint16
|  |              |  +--rw sub-id?     uint16
|  |              |  +--rw source?     inet:ipv4-address
|  |              +--rw affinity!
|  |              |  +--rw rule* [action]
|  |              |     +--rw action    enumeration
|  |              |     +--rw color*    string
|  |              +--rw segments!
|  |                 +--rw sid-algorithm?   uint16
|  +--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 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
```

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 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
```

Example Function Packs

This topic discusses the yang models for the L2VPN, L3VPN, and IETF-TE example services.

Flat L2VPN Yang Models

This section includes:

- **Flat L2VPN Service Model**
- **Flat L2VPN Service Plan Model**

Flat L2VPN Service Model

```
module: cisco-flat-L2vpn-fp
  +--rw y-1731-profile* [name]
  |  +--rw schedule
  |  |  +--rw interval?   uint8
  |  |  +--rw duration?   union
  |  +--rw name                    string
  |  +--rw type                    enumeration
  |  +--rw probe
  |  |  +--rw type?                   enumeration
  |  |  +--rw burst
  |  |  |  +--rw message-count?    uint16
  |  |  |  +--rw message-period?   uint32
  |  |  +--rw measurement-interval?   uint32
  |  |  +--rw frame-size?             uint16
  |  |  +--rw priority?               uint8
  |  +--rw delay-params
  |  |  +--rw statistic* [type]
  |  |  |  +--rw type    enumeration
  |  |  +--rw version?      enumeration
  |  +--rw loss-params
  |  |  +--rw statistic* [type]
  |  |     +--rw type    enumeration
  |  +--rw bucket-details
  |     +--rw bucket-size?      uint8
  |     +--rw bucket-archive?   uint8
  +--rw l2vpn-route-policy* [name]
  |  +--rw name     service-name
  |  +--rw color* [id]
  |     +--rw id      uint32
  |     +--rw ipv4!
  |     |  +--rw rd*   asn-ip-type
  |     +--rw ipv6!
  |        +--rw rd*   asn-ip-type
  +--rw flat-L2vpn* [name]
```

```
  |  +--rw name                        service-name
  |  +--rw service-type                enumeration
  |  +--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 service-assurance!
  |  |  +--rw monitoring-state?   aa-monitoring-state
  |  |  +--rw profile-name?       string
  |  |  +--rw rule-name?          string
  |  +--rw status
  |  |  +--ro oper-status
  |  |     +--ro status?      identityref
  |  |     +--ro timestamp?   yang:date-and-time
  |  +--rw flat-L2vpn-p2p
  |  |  +--rw pw-id          uint32
  |  |  +--rw local-site
  |  |  |  +--rw pe                       -> /core-fp-common:dispatch-
map/device
  |  |  |  +--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 if-type                  enumeration
  |  |  |  +--rw if-id                    string
  |  |  |  +--rw if-description?          string
  |  |  |  +--rw if-encap                 l2-serv-encap-type
  |  |  |  +--rw vlan-id                  int32
  |  |  |  +--rw sub-if-id?               int32
  |  |  |  +--rw rewrite!
  |  |  |  |  +--rw ingress!
  |  |  |  |     +--rw (tag-choice)?
```

```
|  |  |  |     |  +--:(pop)
|  |  |  |     |  |  +--rw pop          enumeration
|  |  |  |     |  +--:(push)
|  |  |  |     |  |  +--rw push         empty
|  |  |  |     |  +--:(translate)
|  |  |  |     |     +--rw translate    enumeration
|  |  |  |     +--rw dot1q              uint16
|  |  |  |     +--rw mode?              enumeration
|  |  |  +--rw mtu?                      uint16
|  |  |  +--rw xconnect-group-name       string
|  |  |  +--rw xconnect-encapsulation?   enumeration
|  |  |  +--rw p2p-name                  string
|  |  |  +--rw (policy-type)?
|  |  |  |  +--:(sr-te)
|  |  |  |  |  +--rw sr-te!
|  |  |  |  |     +--rw preferred-path!
|  |  |  |  |        +--rw policy      string
|  |  |  |  |        +--rw fallback?   enumeration
|  |  |  |  +--:(rsvp-te)
|  |  |  |     +--rw rsvp-te!
|  |  |  |        +--rw preferred-path!
|  |  |  |           +--rw (tunnel-te-id-source)
|  |  |  |           |  +--:(te-tunnel-id)
|  |  |  |           |  |  +--rw te-tunnel-id?      uint16
|  |  |  |           |  +--:(ietf-te-service)
|  |  |  |           |     +--rw ietf-te-service?   string
|  |  |  |           +--rw fallback?                enumeration
|  |  |  +--rw control-word?             enumeration
|  |  |  +--rw pw-class                  string
|  |  |  +--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 y-1731-profile* [name]
|  |  |  |        +--rw name         -> /y-1731-profile/name
|  |  |  |        +--ro statistic* [type]
|  |  |  |           +--ro type            string
|  |  |  |           +--ro statistic-id?   uint32
|  |  |  +--rw xconnect-local-ip?        inet:ip-address
|  |  |  +--rw xconnect-remote-ip?       inet:ip-address
|  |  |  +--rw mpls-local-label?         uint32
```

```
  |  |  |  +--rw mpls-remote-label?        uint32
  |  |  |  +--rw action
  |  |  |     +---x error-recovery
  |  |  |        +---w input
  |  |  |        |  +---w sync-direction    enumeration
  |  |  |        +--ro output
  |  |  |           +--ro success    boolean
  |  |  |           +--ro detail?    string
  |  |  +--rw remote-site!
  |  |     +--rw pe                        -> /core-fp-common:dispatch-
map/device
  |  |     +--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 if-type                  enumeration
  |  |     +--rw if-id                    string
  |  |     +--rw if-description?          string
  |  |     +--rw if-encap                 l2-serv-encap-type
  |  |     +--rw vlan-id                  int32
  |  |     +--rw sub-if-id?               int32
  |  |     +--rw rewrite!
  |  |     |  +--rw ingress!
  |  |     |     +--rw (tag-choice)?
  |  |     |     |  +--:(pop)
  |  |     |     |  |  +--rw pop          enumeration
  |  |     |     |  +--:(push)
  |  |     |     |  |  +--rw push         empty
  |  |     |     |  +--:(translate)
  |  |     |     |     +--rw translate    enumeration
  |  |     |     +--rw dot1q              uint16
  |  |     |     +--rw mode?              enumeration
  |  |     +--rw mtu?                     uint16
  |  |     +--rw xconnect-group-name      string
  |  |     +--rw xconnect-encapsulation?  enumeration
  |  |     +--rw p2p-name                 string
  |  |     +--rw (policy-type)?
  |  |     |  +--:(sr-te)
```

```
|  |     |  |  +--rw sr-te!
|  |     |  |     +--rw preferred-path!
|  |     |  |        +--rw policy      string
|  |     |  |        +--rw fallback?   enumeration
|  |     |  +--:(rsvp-te)
|  |     |     +--rw rsvp-te!
|  |     |        +--rw preferred-path!
|  |     |           +--rw (tunnel-te-id-source)
|  |     |           |  +--:(te-tunnel-id)
|  |     |           |  |  +--rw te-tunnel-id?      uint16
|  |     |           |  +--:(ietf-te-service)
|  |     |           |     +--rw ietf-te-service?   string
|  |     |           +--rw fallback?                enumeration
|  |     +--rw control-word?             enumeration
|  |     +--rw pw-class                  string
|  |     +--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 y-1731-profile* [name]
|  |     |        +--rw name         -> /y-1731-profile/name
|  |     |        +--ro statistic* [type]
|  |     |           +--ro type            string
|  |     |           +--ro statistic-id?   uint32
|  |     +--rw action
|  |        +---x error-recovery
|  |           +---w input
|  |           |  +---w sync-direction    enumeration
|  |           +--ro output
|  |              +--ro success    boolean
|  |              +--ro detail?    string
|  +--rw flat-L2vpn-evpn-vpws
|  |  +--rw evi-id         uint16
|  |  +--rw local-site
|  |  |  +--rw pe                      -> /core-fp-common:dispatch-map/device
|  |  |  +--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 if-type                 enumeration
|  |  |  +--rw if-id                   string
|  |  |  +--rw if-description?         string
|  |  |  +--rw if-encap                l2-serv-encap-type
|  |  |  +--rw multi-home!
|  |  |  |  +--rw esi-value    string
|  |  |  +--rw vlan-id                 int32
|  |  |  +--rw sub-if-id?              int32
|  |  |  +--rw rewrite!
|  |  |  |  +--rw ingress!
|  |  |  |     +--rw (tag-choice)?
|  |  |  |     |  +--:(pop)
|  |  |  |     |  |  +--rw pop          enumeration
|  |  |  |     |  +--:(push)
|  |  |  |     |  |  +--rw push         empty
|  |  |  |     |  +--:(translate)
|  |  |  |     |     +--rw translate    enumeration
|  |  |  |     +--rw dot1q              uint16
|  |  |  |     +--rw mode?              enumeration
|  |  |  +--rw mtu?                    uint16
|  |  |  +--rw xconnect-group-name     string
|  |  |  +--rw p2p-name                string
|  |  |  +--rw sr-te!
|  |  |  |  +--rw (type)?
|  |  |  |     +--:(odn)
|  |  |  |     |  +--rw odn!
|  |  |  |     |     +--rw route-policy    -> /l2vpn-route-policy/name
|  |  |  |     |     +--rw attach-point
|  |  |  |     |        +--rw (parent-rr-route-policy-choice)?
|  |  |  |     |           +--:(parent-rr-route-policy)
|  |  |  |     |              +--rw parent-rr-route-policy?   string
|  |  |  |     +--:(preferred-path)
|  |  |  |        +--rw preferred-path!
|  |  |  |           +--rw policy      string
|  |  |  |           +--rw fallback?   enumeration
|  |  |  +--rw pw-class                string
|  |  |  +--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 y-1731-profile* [name]
|  |  |  |        +--rw name         -> /y-1731-profile/name
|  |  |  |        +--ro statistic* [type]
|  |  |  |           +--ro type            string
|  |  |  |           +--ro statistic-id?   uint32
|  |  |  +--rw action
|  |  |  |  +---x error-recovery
|  |  |  |     +---w input
|  |  |  |     |  +---w sync-direction    enumeration
|  |  |  |     +--ro output
|  |  |  |        +--ro success    boolean
|  |  |  |        +--ro detail?    string
|  |  |  +--rw evi-source              uint32
|  |  |  +--rw evi-target              uint32
|  |  +--rw remote-site!
|  |     +--rw pe                      -> /core-fp-common:dispatch-map/device
|  |     +--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 if-type                 enumeration
|  |     +--rw if-id                   string
|  |     +--rw if-description?         string
|  |     +--rw if-encap                l2-serv-encap-type
|  |     +--rw multi-home!
|  |     |  +--rw esi-value    string
|  |     +--rw vlan-id                 int32
|  |     +--rw sub-if-id?              int32
|  |     +--rw rewrite!
|  |     |  +--rw ingress!
|  |     |     +--rw (tag-choice)?
|  |     |     |  +--:(pop)
|  |     |     |  |  +--rw pop          enumeration
|  |     |     |  +--:(push)
|  |     |     |  |  +--rw push         empty
```

```
|  |     |     |  +--:(translate)
|  |     |     |     +--rw translate    enumeration
|  |     |     +--rw dot1q              uint16
|  |     |     +--rw mode?              enumeration
|  |     +--rw mtu?                    uint16
|  |     +--rw xconnect-group-name     string
|  |     +--rw p2p-name                string
|  |     +--rw sr-te!
|  |     |  +--rw (type)?
|  |     |     +--:(odn)
|  |     |     |  +--rw odn!
|  |     |     |     +--rw route-policy    -> /l2vpn-route-policy/name
|  |     |     |     +--rw attach-point
|  |     |     |        +--rw (parent-rr-route-policy-choice)?
|  |     |     |           +--:(parent-rr-route-policy)
|  |     |     |              +--rw parent-rr-route-policy?   string
|  |     |     +--:(preferred-path)
|  |     |        +--rw preferred-path!
|  |     |           +--rw policy      string
|  |     |           +--rw fallback?   enumeration
|  |     +--rw pw-class                string
|  |     +--rw ethernet-service-oam!
|  |     |  +--rw md-name     string
|  |     |  +--rw md-level    uint8
|  |     |  +--rw y-1731* [maid]
|  |     |     +--rw maid              string
|  |     |     +--rw mep-id            uint16
|  |     |     +--rw id-type?                  enumeration
|  |     |     +--ro sman-id-allocation-data
|  |     |     |  +--ro icc-based-id?   string
|  |     |     |  +--ro number-id?      string
|  |     |     +--rw message-period?   string
|  |     |     +--rw y-1731-profile* [name]
|  |     |        +--rw schedule
|  |     |        |  +--rw interval?   uint8
|  |     |        |  +--rw duration?   union
|  |     |        +--rw name         -> /y-1731-profile/name
|  |     |        +--ro statistic* [type]
|  |     |           +--ro type            string
|  |     |           +--ro statistic-id?   uint32
|  |     +--rw action
|  |        +---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 flat-L2vpn-actions
     +---x update-internal-cfp-configurations
     +---x cleanup
     |  +---w input
     |  |  +---w service             string
     |  |  +---w site?            -> /core-fp-common:dispatch-map/device
     |  |  +---w no-networking       boolean
     |  +--ro output
     |     +--ro success    boolean
     |     +--ro detail?    string
     +---x error-recovery
        +---w input
        |  +---w service                  string
        |  +---w sync-direction           enumeration
        |  +---w (site-type)?
        |     +--:(remote-site-only)
        |     |  +---w remote-site-only?   -> /core-fp-common:dispatch-
map/device
        |     +--:(local-site-only)
        |        +---w local-site-only?    -> /core-fp-common:dispatch-
map/device
        +--ro output
           +--ro success    boolean
           +--ro detail?    string
```

Flat L2VPN Service Plan Model

```
module: cisco-flat-L2vpn-fp
  +--ro flat-L2vpn-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 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
```

Flat-L3VPN Yang Models

This section includes:

- **Flat-L3VPN Service Model**
- **Flat-L3VPN Service Plan Model**

Flat-L3VPN Service Model

```
module: cisco-flat-L3vpn-fp
  +--rw l3vpn-route-policy* [name]
  |  +--rw name     service-name
  |  +--rw color* [id]
  |     +--rw id      uint32
  |     +--rw ipv4!
  |     |  +--rw prefix*   tailf:ipv4-address-and-prefix-length
  |     +--rw ipv6!
  |     |  +--rw ipv6-prefix*   tailf:ipv6-address-and-prefix-length
  |     +--rw exclusive?     boolean
  |     +--rw description?   string
  |  +--rw extra-policy* [name]
  |     +--rw name         string
  |     +--rw operation?   enumeration
  |     +--rw address?     enumeration
  +--rw flat-L3vpn* [name]
  |  +--rw name                       service-name
  |  +--rw service-assurance!
  |  |  +--rw monitoring-state?   aa-monitoring-state
```

```
|  |  +--rw profile-name?       string
|  |  +--rw rule-name?          string
|  +--rw service-status
|  |  +--ro ops
|  |     +--ro status?      operational-type
|  |     +--ro timestamp?   yang:date-and-time
|  +--rw endpoint* [endpoint-name]
|  |  +--rw endpoint-name              string
|  |  +--rw access-pe                  -> /core-fp-common:dispatch-map/device
|  |  +---x error-recovery
|  |  |  +---w input
|  |  |  |  +---w sync-direction    enumeration
|  |  |  +--ro output
|  |  |     +--ro success    boolean
|  |  |     +--ro detail?    string
|  |  +--rw if-type                    enumeration
|  |  +--rw if-id                      string
|  |  +--rw mtu?                       uint16
|  |  +--rw pe-ip-addr?                tailf:ipv4-address-and-prefix-length
|  |  +--rw pe-ipv6-addr?              tailf:ipv6-address-and-prefix-length
|  |  +--rw (as-choice)?
|  |  |  +--:(as-no)
|  |  |  |  +--rw as-no?               as-no-type
|  |  |  +--:(as-no-from-device)
|  |  |     +--rw as-no-from-device?   empty
|  |  +--rw vlan-id?                   int32
|  |  +--rw BDI?                       uint16
|  |  +--rw ce-pe-prot
|  |  |  +--rw (routing)?
|  |  |     +--:(e-bgp)
|  |  |        +--rw e-bgp
|  |  |           +--rw neighbor-ipv4?    inet:ipv4-address
|  |  |           +--rw neighbor-ipv6?    inet:ipv6-address
|  |  |           +--rw remote-as-ipv4    as-no-type
|  |  |           +--rw remote-as-ipv6    as-no-type
|  |  |           +--rw ebgp-multihop!
|  |  |           |  +--rw ttl-value            uint8
|  |  |           |  +--rw mpls-deactivation?   boolean
|  |  |           +--rw update-source!
|  |  |              +--rw if-type      enumeration
|  |  |              +--rw if-id        string
|  |  |              +--rw sub-if-id?   int32
|  |  +--rw vrf
|  |  |  +--rw vrf-definition         string
```

```
|  |  |  +--rw route-distinguisher?   asn-ip-type
|  |  |  +--rw vpn-id?                vpn-id-type
|  |  |  +--rw address-family* [address-family]
|  |  |     +--rw address-family            address-family-type
|  |  |     +--rw redistribute-connected?   empty
|  |  |     +--rw metric?                   Bgp-default-metric-range
|  |  |     +--rw srv6!
|  |  |     |  +--rw locator-name?   string
|  |  |     +--rw vpn-target* [rt-value]
|  |  |        +--rw rt-type     bgp-rt-type
|  |  |        +--rw rt-value    asn-ip-type
|  |  +--rw sr-te!
|  |  |  +--rw export-route-policy?   -> /l3vpn-route-policy/name
|  |  |  +--rw import-route-policy?   -> /l3vpn-route-policy/name
|  |  +--rw l2-attachment-circuit* [name]
|  |     +--rw name       string
|  |     +--rw if-type    enumeration
|  |     +--rw if-id      string
|  |     +--rw vlan-id?   int32
|  |     +--rw rewrite!
|  |        +--rw ingress!
|  |           +--rw (tag-choice)?
|  |           |  +--:(pop)
|  |           |  |  +--rw pop          enumeration
|  |           |  +--:(push)
|  |           |  |  +--rw push         empty
|  |           |  +--:(translate)
|  |           |     +--rw translate    enumeration
|  |           +--rw dot1q              uint16
|  |           +--rw mode?              enumeration
|  +---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 flat-L3vpn-actions
   +---x cleanup
   |  +---w input
```

```
     |  |  +---w service          string
     |  |  +---w endpoint?        string
     |  |  +---w no-networking    boolean
     |  +--ro output
     |     +--ro success    boolean
     |     +--ro detail?    string
     +---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?         string
     |  |  +---w sync-direction    enumeration
     |  +--ro output
     |     +--ro success    boolean
     |     +--ro detail?    string
     +---x update-internal-cfp-configurations
```

Flat-L3VPN Service Plan Model

```
module: cisco-flat-L3vpn-fp
  +--ro flat-L3vpn-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 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-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 head-end and tail-end leaf elements for the devices are used as source and destination.

This section includes:

- **IETF-TE Service Model**
- **IETF-TE Service Plan Model**
- **IETF-TE Deviation Model**

IETF-TE Service Model

```
module: ietf-te
  +--rw te!
  |  +--rw tunnels
  |  |  +--rw tunnel* [name]
  |  |  |  +--ro operational-state?          identityref
  |  |  |  +--rw head-end                    -> /core-fp-common:dispatch-
map/device
  |  |  |  +--rw tail-end?                   -> /core-fp-common:dispatch-
map/device
  |  |  |  +--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 setup-priority?             uint8
  |  |  |  +--rw hold-priority?              uint8
  |  |  |  +--rw signaling-type?             identityref
  |  |  |  +--rw te-bandwidth
  |  |  |  |  +--rw (technology)?
  |  |  |  |     +--:(generic)
  |  |  |  |        +--rw generic?   te-bandwidth
  |  |  |  +--rw p2p-primary-paths
  |  |  |  |  +--rw p2p-primary-path* [name]
```

```
  |  |  |  |     +--rw name                            string
  |  |  |  |     +--rw path-computation-method?        identityref
  |  |  |  |     +--rw optimizations
  |  |  |  |     |  +--rw (algorithm)?
  |  |  |  |     |     +--:(metric) {path-optimization-metric}?
  |  |  |  |     |     |  +--rw optimization-metric* [metric-type]
  |  |  |  |     |     |  |  +--rw metric-type
identityref
  |  |  |  |     +--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
  |  |  |  +---x error-recovery
  |  |  |  |  +---w input
  |  |  |  |  |  +---w sync-direction       enumeration
  |  |  |  |  |  +---w (node-type)?
  |  |  |  |  |     +--:(source)
  |  |  |  |  |     |  +---w source?        te-types:te-node-id
  |  |  |  |  |     +--:(destination)
  |  |  |  |  |        +---w destination?   te-types:te-node-id
  |  |  |  |  +--ro output
  |  |  |  |     +--ro success    boolean
  |  |  |  |     +--ro detail?    string
  |  |  |  +--rw action
  |  |  |  |  +---x self-test
  |  |  |  |     +--ro output
  |  |  |  |        +--ro status?    string
  |  |  |  |        +--ro message?   string
  |  |  |  +--rw traffic-steering
  |  |  |     +--rw (steering-choice)?
  |  |  |        +--:(autoroute)
  |  |  |        |  +--rw autoroute
  |  |  |        |     +--rw announce
```

```
  |  |  |        |     |  +--rw enable?   boolean
  |  |  |        |     |  +--rw metric!
  |  |  |        |     |     +--rw (metric-choice)?
  |  |  |        |     |        +--:(constant)
  |  |  |        |     |        |  +--rw constant?   uint32
  |  |  |        |     |        +--:(absolute)
  |  |  |        |     |        |  +--rw absolute?   uint32
  |  |  |        |     |        +--:(relative)
  |  |  |        |     |           +--rw relative?   int8
  |  |  |        |     +--rw destination* [address]
  |  |  |        |        +--rw address    inet:ipv4-address
  |  |  |        +--:(forwarding-adjacency)
  |  |  |           +--rw forwarding-adjacency!
  |  |  |              +--rw holdtime?       uint16
  |  |  |              +--rw include-ipv6?   empty
```

IETF-TE Service Plan Model

```
module: ietf-te
  +--rw te!
  |  +--rw tunnels
  |     +--ro tunnel-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 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-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 ietf-te { prefix te; }

  description "IETF TE Cisco NSO Deviations";
  revision 2020-09-04 {
      description "Initial revision.";
  }

  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:p2p-primary-paths/te:p2p-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";
      }
    }
  }
}
```

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 Service Model

```
module: ietf-l2vpn-ntw
  +--rw l2vpn-ntw
  |  +--rw id-pools!
  |  |  +--rw evi-id-pool-name?              -> /ralloc:resource-
pools/idalloc:id-pool/name
  |  |  +--rw evi-source-target-pool-name?   -> /ralloc:resource-
pools/idalloc:id-pool/name
  |  +--rw vpn-services
  |  |  +--rw vpn-service* [vpn-id]
  |  |  |  +--rw vpn-id                                      string
  |  |  |  +--ro evi-allocation-data
  |  |  |  |  +--ro evi-id?       string
  |  |  |  |  +--ro evi-source?   string
  |  |  |  |  +--ro evi-target?   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
  |  |  |  +--rw (evi-id-choice)?
  |  |  |  |  +--:(auto-evi-id)
  |  |  |  |  |  +--rw auto-evi-id?                            empty
  |  |  |  |  +--:(evi-id)
  |  |  |  |     +--rw evi-id                                  uint16
  |  |  |  +--rw (evi-source-choice)?
  |  |  |  |  +--:(auto-evi-source)
  |  |  |  |  |  +--rw auto-evi-source?                        empty
  |  |  |  |  +--:(evi-source)
  |  |  |  |     +--rw evi-source                              uint32
  |  |  |  +--rw (evi-target-choice)?
  |  |  |  |  +--:(auto-evi-target)
  |  |  |  |  |  +--rw auto-evi-target?                        empty
  |  |  |  |  +--:(evi-target)
  |  |  |  |     +--rw evi-target                              uint32
  |  |  |  +---x self-test
  |  |  |  |  +--ro output
  |  |  |  |     +--ro status?    string
  |  |  |  |     +--ro message?   string
```

```
  |  |  |  +--rw control-word?                                enumeration
  |  |  |  +--rw vpn-svc-type?                                enumeration
  |  |  |  +--rw status
  |  |  |  |  +--ro oper-status
  |  |  |  |     +--ro status?      identityref
  |  |  |  |     +--ro timestamp?   yang:date-and-time
  |  |  |  +--rw vpn-nodes
  |  |  |  |  +--rw vpn-node* [vpn-node-id ne-id]
  |  |  |  |     +--rw vpn-node-id             -> /core-fp-common:dispatch-
map/device
  |  |  |  |     +--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
  |  |  |  |     +---x error-recovery
  |  |  |  |     |  +---w input
  |  |  |  |     |  |  +---w sync-direction    enumeration
  |  |  |  |     |  +--ro output
  |  |  |  |     |     +--ro success    boolean
  |  |  |  |     |     +--ro detail?    string
  |  |  |  |     +--rw multi-home!
  |  |  |  |     |  +--rw esi-value    string
  |  |  |  |     +--rw te-service-mapping
  |  |  |  |     |  +--rw te-mapping
  |  |  |  |     |     +--rw (te)?
  |  |  |  |     |        +--:(sr-policy)
  |  |  |  |     |        |  +--rw sr-policy!
  |  |  |  |     |        |     +--rw policy      string
  |  |  |  |     |        |     +--rw fallback?   enumeration
  |  |  |  |     |        +--:(te-tunnel-list)
  |  |  |  |     |        |  +--rw te-tunnel-list!
  |  |  |  |     |        |     +--rw (tunnel-te-id-source)
  |  |  |  |     |        |     |  +--:(te-tunnel-id)
  |  |  |  |     |        |     |  |  +--rw te-tunnel-id?      uint16
  |  |  |  |     |        |     |  +--:(ietf-te-service)
  |  |  |  |     |        |     |     +--rw ietf-te-service?   string
  |  |  |  |     |        |     +--rw fallback?                enumeration
```

```
  |  |  |  |     |        +--:(odn)
  |  |  |  |     |           +--rw odn!
  |  |  |  |     |              +--rw route-policy    -> /cisco-flat-L2vpn-
fp:l2vpn-route-policy/name
  |  |  |  |     |              +--rw attach-point
  |  |  |  |     |                 +--rw (parent-rr-route-policy-choice)?
  |  |  |  |     |                    +--:(parent-rr-route-policy)
  |  |  |  |     |                       +--rw parent-rr-route-policy?   string
  |  |  |  |     +--rw ne-id                  -> ../vpn-node-id
  |  |  |  |     +--rw signaling-options* [type]
  |  |  |  |     |  +--rw type         enumeration
  |  |  |  |     |  +--rw evpn-bgp
  |  |  |  |     |  |  +--rw type?   enumeration
  |  |  |  |     |  +--rw t-ldp-pwe
  |  |  |  |     |     +--rw ac-pw-list* [peer-addr vc-id]
  |  |  |  |     |        +--rw peer-addr     inet:ip-address
  |  |  |  |     |        +--rw vc-id         vpn-common:svc-id
  |  |  |  |     |        +--rw mpls-label?   uint32
  |  |  |  |     +--rw vpn-network-accesses
  |  |  |  |        +--rw vpn-network-access* [id]
  |  |  |  |           +--rw id                      vpn-common:svc-id
  |  |  |  |           +--rw Interface-mtu?          uint16
  |  |  |  |           +--rw connection
  |  |  |  |           |  +--rw encapsulation-type?   enumeration
  |  |  |  |           |  +--rw dot1q-interface
  |  |  |  |           |  |  +--rw l2-access-type?   enumeration
  |  |  |  |           |  |  +--rw dot1q {dot1q}?
  |  |  |  |           |  |     +--rw physical-inf?   string
  |  |  |  |           |  |     +--rw c-vlan-id?      uint32
  |  |  |  |           |  |     +--rw rewrite!
  |  |  |  |           |  |        +--rw ingress!
  |  |  |  |           |  |           +--rw (tag-choice)?
  |  |  |  |           |  |           |  +--:(pop)
  |  |  |  |           |  |           |  |  +--rw pop          enumeration
  |  |  |  |           |  |           |  +--:(push)
  |  |  |  |           |  |           |  |  +--rw push         empty
  |  |  |  |           |  |           |  +--:(translate)
  |  |  |  |           |  |           |     +--rw translate    enumeration
  |  |  |  |           |  |           +--rw dot1q              uint16
  |  |  |  |           |  |           +--rw mode?              enumeration
  |  |  |  |           |  +--rw untagged-interface
  |  |  |  |           |     +--rw l2-access-type?   enumeration
  |  |  |  |           |     +--rw untagged
  |  |  |  |           |        +--rw physical-inf?   string
```

```
  |  |  |  |           |        +--rw sub-if-id?      uint32
  |  |  |  |           |        +--rw rewrite!
  |  |  |  |           |           +--rw ingress!
  |  |  |  |           |              +--rw (tag-choice)?
  |  |  |  |           |              |  +--:(pop)
  |  |  |  |           |              |  |  +--rw pop          enumeration
  |  |  |  |           |              |  +--:(push)
  |  |  |  |           |              |  |  +--rw push         empty
  |  |  |  |           |              |  +--:(translate)
  |  |  |  |           |              |     +--rw translate    enumeration
  |  |  |  |           |              +--rw dot1q              uint16
  |  |  |  |           |              +--rw mode?              enumeration
  |  |  |  |           +--rw ethernet-service-oam
  |  |  |  |              +--rw md-name?    string
  |  |  |  |              +--rw md-level?   uint8
  |  |  |  |              +--rw y-1731* [maid]
  |  |  |  |                 +--rw maid
string
  |  |  |  |                 +--rw mep-id
uint16
  |  |  |  |                 +--rw l2vpn-ntw-augmentations:id-type?
enumeration
  |  |  |  |                 +--ro l2vpn-ntw-augmentations:sman-id-allocation-
data
  |  |  |  |                 |  +--ro l2vpn-ntw-augmentations:icc-based-id?
string
  |  |  |  |                 |  +--ro l2vpn-ntw-augmentations:number-id?
string
  |  |  |  |                 +--rw message-period?
string
  |  |  |  |                 +--rw l2vpn-ntw-augmentations:y-1731-profile*
[name]
  |  |  |  |                    +--rw l2vpn-ntw-augmentations:name         ->
/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw-augmentations:y-1731-profile/name
  |  |  |  |                    +--rw l2vpn-ntw-augmentations:schedule
  |  |  |  |                    |  +--rw l2vpn-ntw-augmentations:interval?
uint8
  |  |  |  |                    |  +--rw l2vpn-ntw-augmentations:duration?
union
  |  |  |  +--rw l2vpn-ntw-augmentations:service-assurance!
  |  |  |     +--rw l2vpn-ntw-augmentations:monitoring-state?   aa-monitoring-
state
  |  |  |     +--rw l2vpn-ntw-augmentations:profile-name?       string
  |  |  |     +--rw l2vpn-ntw-augmentations:rule-name?          string
  |  +--rw l2vpn-ntw-augmentations:y-1731-profile* [name]
  |     +--rw l2vpn-ntw-augmentations:schedule
  |     |  +--rw l2vpn-ntw-augmentations:interval?   uint8
```

```
|     |  +--rw l2vpn-ntw-augmentations:duration?   union
|     +--rw l2vpn-ntw-augmentations:name                        string
|     +--rw l2vpn-ntw-augmentations:type                        enumeration
|     +--rw l2vpn-ntw-augmentations:probe
|     |  +--rw l2vpn-ntw-augmentations:type?                   enumeration
|     |  +--rw l2vpn-ntw-augmentations:burst
|     |  |  +--rw l2vpn-ntw-augmentations:message-count?    uint16
|     |  |  +--rw l2vpn-ntw-augmentations:message-period?   uint32
|     |  +--rw l2vpn-ntw-augmentations:measurement-interval?   uint32
|     |  +--rw l2vpn-ntw-augmentations:frame-size?             uint16
|     |  +--rw l2vpn-ntw-augmentations:priority?               uint8
|     +--rw l2vpn-ntw-augmentations:delay-params
|     |  +--rw l2vpn-ntw-augmentations:statistic* [type]
|     |  |  +--rw l2vpn-ntw-augmentations:type    enumeration
|     |  +--rw l2vpn-ntw-augmentations:version?      enumeration
|     +--rw l2vpn-ntw-augmentations:loss-params
|     |  +--rw l2vpn-ntw-augmentations:statistic* [type]
|     |     +--rw l2vpn-ntw-augmentations:type    enumeration
|     +--rw l2vpn-ntw-augmentations:bucket-details
|        +--rw l2vpn-ntw-augmentations:bucket-size?      uint8
|        +--rw l2vpn-ntw-augmentations:bucket-archive?   uint8
+--rw l2nm-actions
   +---x cleanup
   |  +---w input
   |  |  +---w service          string
   |  |  +---w no-networking    boolean
   |  |  +---w vpn-node?        -> /core-fp-common:dispatch-map/device
   |  +--ro output
   |     +--ro success    boolean
   |     +--ro detail?    string
   +---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 vpn-node?         string
      |  +---w sync-direction    enumeration
      +--ro output
         +--ro success    boolean
         +--ro detail?    string
```

IETF-L2VPN-NM Service Plan Model

```
module: ietf-l2vpn-ntw
  +--rw l2vpn-ntw
  |  +--rw vpn-services
  |     +--ro vpn-service-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 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 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 tailf-common { prefix tailf; }
  import ietf-l2vpn-ntw { prefix l2vpn-ntw; }
  import core-fp-common { prefix core-fp-common; }

  description "IETF L2VPN NTW Cisco NSO Deviations";
```

```
  revision 2021-09-28 {
    description
       "Added: deviation must conditions for vpn-network-accesses -> vpn-
network-access -> ethernet-service-oam";
  }

  revision 2021-09-17 {
    description "Added deviation for vpn-service -> vpn-service -> vpn-nodes ->
vpn-node -> te-service-mapping -> te-mapping -> sr-policy -> policy length
1..max";
  }

  revision 2021-08-31 {
    description
      "Added: deviation must conditions for vpn-services -> vpn-service -> vpn-
svc-type and IOSXE device check for EVPN-VPWS
       Added: deviation for connection -> untagged-interface -> untagged ->
physical-inf
              and connection -> dot1q-interface -> dot1q -> physical-inf string
pattern";
  }

  revision 2021-07-08 {
    description "Added deviation for vpn-service -> vpn-id length 1..32";
  }

  revision 2021-06-22 {
    description
      "Added: deviation must conditions for vpn-network-accesses -> vpn-
network-access -> ethernet-service-oam
              have same maid, md-name and md-level on all vpn-nodes and unique
mep-id on each vpn-node
       Added: deviation for vpn-network-accesses -> vpn-network-access ->
ethernet-service-oam -> md-level
              to be mandatory";
  }

  revision 2021-06-02 {
    description
      "Added: deviation must condition for vpn-service";
  }

  revision 2021-06-01 {
    description
      "Added: deviation for vpn-network-accesses -> vpn-network-access ->
ethernet-service-oam";
```

```
  }

  revision 2021-04-27 {
      description "Added deviation for vpn-service->vpn-id to filter invalid
id";
  }

  revision 2021-03-25 {
      description "Added deviation to ethernet-service-oam/y-1731 related
elements";
  }

  revision 2021-03-18 {
      description "Updated deviation for vpn-network-access and added min-
elements as 1";
  }

  revision 2020-11-16 {
      description "Added deviation for l2-access-type for untagged-interface";
  }

  revision 2020-10-19 {
      description "Added restriction of min-elements 2 on vpn-node,
      Added type deviation for Interface-mtu,
      Added must condition on c-vlan-id - can be set when encapsulation-type is
'vpn-common:dot1q' and c-vlan-id is required when encapsulation-type is 'vpn-
common:dot1q'";
  }

  revision 2020-10-13 {
      description "Added replace deviation for type of l2vpn-ntw:signaling-
options/l2vpn-ntw:evpn-bgp/l2vpn-ntw:type node";
  }

  revision 2020-09-11 {
      description "Initial revision.";
  }

  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"
{
    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/l2vpn-ntw:vpn-node/l2vpn-ntw:vpn-node-id" {
    deviate replace {
      type leafref {
        path "/core-fp-common:dispatch-map/core-fp-common:device";
      }
    }
    deviate add {
      must "../../../l2vpn-ntw:vpn-svc-type='vpn-common:t-ldp' or
(../../../l2vpn-ntw:vpn-svc-type='vpn-common:evpn-bgp' and not(contains(/core-
fp-common:dispatch-map
                 [core-fp-common:device=current()]/core-fp-common:ned-id,
                 'cisco-ios-cli-')))"{
        error-message "Service Type EVPN-VPWS not supported on IOSXE devices'";
      }
    }

  }
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:ne-id" {
    deviate replace {
      type leafref {
        path "../l2vpn-ntw:vpn-node-id";
      }
    }
  }
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node" {
    deviate add {
```

```
      min-elements 2;
      max-elements 2;
    }
  }
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-options" {
    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:signaling-
options/l2vpn-ntw:t-ldp-pwe/l2vpn-ntw:ac-pw-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:signaling-
options/l2vpn-ntw:type" {
    deviate replace {
      type enumeration {
        enum vpn-common:t-ldp;
        enum vpn-common:evpn-bgp;
      }
    }
  }
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:signaling-
options/l2vpn-ntw:evpn-bgp/l2vpn-ntw:type" {
    deviate replace {
      type enumeration {
        enum evpn-vpws;
      }
    }
  }
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-nodes/l2vpn-ntw:vpn-node/l2vpn-ntw:te-service-
mapping/l2vpn-ntw:te-mapping/l2vpn-ntw:te/l2vpn-ntw:sr-policy/l2vpn-ntw:sr-
policy/l2vpn-ntw:policy" {
    deviate replace {
      type string {
        length "1..max";
      }
    }
  }
```

```
  deviation "/l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-
service/l2vpn-ntw:vpn-svc-type" {
    deviate replace {
      type enumeration {
        enum vpn-common:t-ldp;
        enum vpn-common:evpn-bgp;
      }
    }
  }
  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;
      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:connection/l2vpn-
ntw:encapsulation-type" {
    deviate replace {
      type enumeration {
        enum vpn-common:dot1q;
        enum vpn-common:untagged-int;
      }
    }
  }
  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:dot1q-
interface/l2vpn-ntw:l2-access-type" {
    deviate replace {
      type enumeration {
        enum 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:Interface-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:connection/l2vpn-ntw:dot1q-
interface/l2vpn-ntw:dot1q/l2vpn-ntw:c-vlan-id" {
    deviate add {
      must "../../../l2vpn-ntw:encapsulation-type='vpn-common:dot1q'"{
        error-message "c-vlan-id can only be set 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:dot1q-
interface/l2vpn-ntw:dot1q" {
    deviate add {
      must "../../l2vpn-ntw:encapsulation-type='vpn-common:dot1q' and l2vpn-
ntw:c-vlan-id"{
        error-message "c-vlan-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:untagged-
interface/l2vpn-ntw:l2-access-type" {
    deviate replace {
      type enumeration {
        enum vpn-common:untagged-int;
      }
    }
  }


  // ====================== y-1731-profile-identifier ======================
  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";
      }
    }
    deviate add {
      must "not(./md-name) or (./md-name and not(contains(/core-fp-
common:dispatch-map[core-fp-common:device=current()/../../../ne-id]/core-fp-
common:ned-id,'cisco-ios-cli-')))" {
```

```
        error-message "Y.1731 not supported for XE devices.";
      }
    }
  }
  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 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/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)';
      }
    }
    deviate add {
      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:connection/l2vpn-ntw:dot1q-
interface/l2vpn-ntw:dot1q/l2vpn-ntw:physical-inf" {
    deviate replace {
      type string {
        pattern "(Bundle-
Ether|FiftyGigE|FortyGigE|FourHundredGigE|HundredGigE|TenGigE|TwentyFiveGigE|Tw
oHundredGigE|GigabitEthernet|Ethernet)[0-9]+(/[0-9]+)*";
      }
    }
  }
```

```
  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:untagged-
interface/l2vpn-ntw:untagged/l2vpn-ntw:physical-inf" {
    deviate replace {
      type string {
        pattern "(Bundle-
Ether|FiftyGigE|FortyGigE|FourHundredGigE|HundredGigE|TenGigE|TwentyFiveGigE|Tw
oHundredGigE|GigabitEthernet|Ethernet)[0-9]+(/[0-9]+)*";
      }
    }
  }
}
```

IETF-L2VPN-NM Augmentations

```
module ietf-l2vpn-ntw-cisco-augmentations {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/ietf-l2vpn-ntw-cisco-
augmentations";
  prefix l2vpn-ntw-augmentations;

  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 core-fp-common { prefix core-fp-common; }

  description "IETF L2VPN NTW Cisco NSO Augmentations";

  revision 2021-09-28 {
    description
      "Removed: leaf delay-type under list y-1731-profile -> delay-params
       Removed: leaf cos under list y-1731-profile -> delay-params -> statistic
       Removed: leaf cos list y-1731-profile -> loss-params -> statistic
       Removed: list statistic under vpn-services -> vpn-service -> vpn-nodes -
> vpn-node -> vpn-network-accesses ->
               vpn-network-access -> ethernet-service-oam -> y-1731 -> y-1731-
profile";
  }

  revision 2021-09-24 {
    description
      "Added: must condition to y-1731-profile -> measurment-internal to y-
1731-profile -> probe -> type";
   }
```

```
  revision 2021-09-14 {
    description
      "Modified: description/tailf:info for y-1731-profile -> message-period
       Modified: range for leaf interval in vpn-services -> vpn-service -> vpn-
nodes -> vpn-node -> vpn-network-accesses ->
                 vpn-network-access -> ethernet-service-oam -> y-1731 -> y-
1731-profile -> schedule";
  }

  revision 2021-08-31 {
    description
      "Added: Augmentation for vpn-services -> vpn-service -> vpn-nodes -> vpn-
node -> vpn-network-accesses ->
        vpn-network-access -> ethernet-service-oam -> y-1731 -> id-type -> id-
type enum (icc-based|number)";
  }

  revision 2021-08-27 {
    description
      "Added: container probe in list y-1731-profile
       Added: leaf message-count in y-1731-profile -> probe
       Moved: y-1731-profile -> measurment-internal to y-1731-profile -> probe
-> measurement-interval
       Moved: y-1731-profile -> message-period to y-1731-profile -> probe ->
burst -> message-period
       Moved: y-1731-profile -> frame-size to y-1731-profile -> probe -> frame-
size
       Moved: y-1731-profile -> priority to y-1731-profile -> probe -> priority
       Added: grouping y-1731-profile-schedule
       Added: uses y-1731-profile-schedule to l2vpn-ntw -> y-1731-profile
       Added: uses y-1731-profile-schedule to l2vpn-ntw -> vpn-services -> vpn-
service
                -> vpn-nodes -> vpn-node -> vpn-network-accesses -> vpn-
network-access
                -> ethernet-service-oam -> y-1731-profile
       Added: container bucket-details under list y-1731-profile";
  }

  revision 2021-06-25 {
    description
      "Added: list y-1731-profile augmenting l2vpn-ntw
       Added: leaf y-1731-profile augmenting l2vpn-ntw -> vpn-services -> vpn-
service
                -> vpn-nodes -> vpn-node -> vpn-network-accesses -> vpn-
network-access
                -> ethernet-service-oam -> y-1731";
  }
```

```
  revision 2021-05-11 {
      description "Initial revision.";
  }

  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 {
        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" {
  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 ncs:service-data;
    ncs:servicepoint "l2vpn-ntw-augmentations-y1731-servicepoint";

    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 uint32 {
            range "1..3600";
          }
          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";
        }
        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 version {
          description
            "Delay Version";
          tailf:info
            "Delay Version";
          type enumeration {
            enum 0;
            enum 1;
          }
          default 1;
        }
      }
      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';
            }
          }
        }
      }
      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/l2vpn-ntw:vpn-service" {
    uses tsdn-core-fp-common:service-assurance-grouping {
      when "/tsdn-core-fp-common:enable-service-assurance = '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;
      }
      when "not(contains(/core-fp-common:dispatch-map[core-fp-
common:device=current()/../../../../../l2vpn-ntw:ne-id]/core-fp-common:ned-
id,'cisco-ios-cli-'))";
      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[id-type=current()])=2) or " +
           "(contains(/core-fp-common:dispatch-map[core-fp-
common:device=current()/../../../../../../l2vpn-ntw:vpn-node[1]/l2vpn-ntw:ne-
id]/core-fp-common:ned-id,'cisco-ios-cli-')) or " +
           "(contains(/core-fp-common:dispatch-map[core-fp-
common:device=current()/../../../../../../l2vpn-ntw:vpn-node[2]/l2vpn-ntw:ne-
id]/core-fp-common:ned-id,'cisco-ios-cli-'))" {
            error-message "y-1731 (icc-based|number) type must be same for each
vpn-node";
          }
    }
    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/l2vpn-ntw-augmentations:y-1731-
profile/l2vpn-ntw-augmentations:name";
        }
        description
          "References L2NM y-1731-profile definition";
        tailf:info
          "References L2NM y-1731-profile definition";
      }
    }
  }
}
```

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 Service Model

```
module: ietf-l3vpn-ntw
  +--rw l3vpn-ntw
  |  +--rw vpn-profiles
```

```
  |  |  +--rw valid-provider-identifiers
  |  |     +--rw routing-profile-identifier* [id]
  |  |        +--rw id    string
  |  +--rw vpn-services
  |     +--rw vpn-service* [vpn-id]
  |     |  +--rw service-status
  |     |  |  +--ro ops
  |     |  |     +--ro status?      operational-type
  |     |  |     +--ro timestamp?   yang:date-and-time
  |     |  +--rw vpn-id                      l3vpn-svc:svc-id
  |     |  +--rw ie-profiles
  |     |  |  +--rw ie-profile* [ie-profile-id]
  |     |  |     +--rw ie-profile-id    string
  |     |  |     +--rw rd?              rt-types:route-distinguisher
  |     |  |     +--rw vpn-targets
  |     |  |        +--rw vpn-target* [id]
  |     |  |        |  +--rw id                   int8
  |     |  |        |  +--rw route-targets* [route-target]
  |     |  |        |  |  +--rw route-target    rt-types:route-target
  |     |  |        |  +--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 vpn-nodes
  |     |  |  +--rw vpn-node* [ne-id]
  |     |  |     +---x error-recovery
  |     |  |     |  +---w input
  |     |  |     |  |  +---w vpn-network-access-id    string
  |     |  |     |  |  +---w sync-direction           enumeration
  |     |  |     |  +--ro output
  |     |  |     |     +--ro success    boolean
  |     |  |     |     +--ro detail?    string
  |     |  |     +--rw local-autonomous-system?   inet:as-number
  |     |  |     +--rw ne-id                      string
  |     |  |     +--rw rd?                        rt-types:route-distinguisher
  |     |  |     +--rw vpn-targets
  |     |  |     |  +--rw vpn-target* [id]
  |     |  |     |  |  +--rw id                   int8
  |     |  |     |  |  +--rw route-targets* [route-target]
  |     |  |     |  |  |  +--rw route-target    rt-types:route-target
  |     |  |     |  |  +--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 vpn-network-accesses
  |     |  |     |  +--rw vpn-network-access* [id]
  |     |  |     |     +--rw id                   l3vpn-svc:svc-id
  |     |  |     |     +--rw port-id?             l3vpn-svc:svc-id
  |     |  |     |     +--rw connection
  |     |  |     |     |  +--rw encapsulation-type?   identityref
  |     |  |     |     |  +--rw tagged-interface
  |     |  |     |     |     +--rw type?                identityref
  |     |  |     |     |     +--rw dot1q-vlan-tagged {dot1q}?
  |     |  |     |     |        +--rw cvlan-id?   uint16
  |     |  |     |     |        +--rw BDI?        uint16
  |     |  |     |     +--rw ip-connection
  |     |  |     |     |  +--rw ipv4 {l3vpn-svc:ipv4}?
  |     |  |     |     |  |  +--rw address-allocation-type?   identityref
  |     |  |     |     |  |  +--rw static-addresses
  |     |  |     |     |  |     +--rw primary-address?   -> ../address/address-
id
  |     |  |     |     |  |     +--rw address* [address-id]
  |     |  |     |     |  |        +--rw address-id          string
  |     |  |     |     |  |        +--rw provider-address?   inet:ipv4-address
  |     |  |     |     |  |        +--rw prefix-length?      uint8
  |     |  |     |     |  +--rw ipv6 {l3vpn-svc:ipv6}?
  |     |  |     |     |     +--rw address-allocation-type?   identityref
  |     |  |     |     |     +--rw static-addresses
  |     |  |     |     |        +--rw primary-address?   -> ../address/address-
id
  |     |  |     |     |        +--rw address* [address-id]
  |     |  |     |     |           +--rw address-id          string
  |     |  |     |     |           +--rw provider-address?   inet:ipv6-address
  |     |  |     |     |           +--rw prefix-length?      uint8
  |     |  |     |     +--rw routing-protocols
  |     |  |     |        +--rw routing-protocol* [id]
  |     |  |     |           +--rw id      string
  |     |  |     |           +--rw type?   identityref
  |     |  |     |           +--rw bgp {l3vpn-svc:rtg-bgp}?
  |     |  |     |              +--rw peer-autonomous-system
inet:as-number
  |     |  |     |              +--rw address-family*                  l3vpn-
svc:address-family
  |     |  |     |              +--rw redistribute-connected-ipv4-af!
  |     |  |     |              |  +--rw metric?   uint32
  |     |  |     |              +--rw redistribute-connected-ipv6-af!
```

```
  |     |  |     |              |  +--rw metric?   uint32
  |     |  |     |              +--rw update-source!
  |     |  |     |              |  +--rw if-type      enumeration
  |     |  |     |              |  +--rw if-id        string
  |     |  |     |              |  +--rw sub-if-id?   int32
  |     |  |     |              +--rw mpls-deactivation?                boolean
  |     |  |     |              +--rw neighbor*
inet:ip-address
  |     |  |     |              +--rw multihop?                         uint8
  |     |  |     |              +--rw l3vpn-ntw-augmentations:srv6!
  |     |  |     |                +--rw l3vpn-ntw-augmentations:address-family*
[name]
  |     |  |     |                  +--rw l3vpn-ntw-augmentations:name
  |     |  |     |                         -> ../../../l3vpn-ntw:address-family
  |     |  |     |                  +--rw l3vpn-ntw-augmentations:locator-name?
string
  |     |  |     +--rw node-ie-profile?           -> ../../../ie-profiles/ie-
profile/ie-profile-id
  |     |  +--rw service-assurance!
  |     |  |  +--rw monitoring-state?   aa-monitoring-state
  |     |  |  +--rw profile-name?       string
  |     |  |  +--rw rule-name?          string
  |     |  +---x self-test
  |     |  |  +--ro output
  |     |  |     +--ro status?    string
  |     |  |     +--ro message?   string
  |     |  +---x error-recovery
  |     |     +---w input
  |     |     |  +---w sync-direction    enumeration
  |     |     +--ro output
  |     |        +--ro success    boolean
  |     |        +--ro detail?    string
  +--rw l3nm-actions
     +---x cleanup
     |  +---w input
     |  |  +---w service                 string
     |  |  +---w vpn-node?               string
     |  |  +---w vpn-network-access-id   string
     |  |  +---w no-networking           boolean
     |  +--ro output
     |     +--ro success    boolean
     |     +--ro detail?    string
     +---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 vpn-node?                string
   |  +---w vpn-network-access-id    string
   |  +---w sync-direction           enumeration
   +--ro output
      +--ro success    boolean
      +--ro 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>.

```
module: ietf-l3vpn-ntw
    +--rw l3vpn-ntw
    |     +--ro vpn-service-plan* [vpn-id]
    |        +--ro vpn-id          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 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-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 ietf-l3vpn-ntw; }
    import core-fp-common { prefix core-fp-common; }
    import tailf-common { prefix tailf; }
    import cisco-flat-L3vpn-fp { prefix cisco-flat-L3vpn-fp; }

    description "IETF L3VPN NTW Cisco NSO Deviations";

    revision 2021-08-31 {
      description "Added deviation for l3vpn-ntw -> vpn-services -> vpn-service
-> vpn-nodes
                        -> vpn-node -> vpn-network-accesses -> vpn-network-
access -> port-id
                        string pattern";
    }

    revision 2021-06-11 {
      description "Modified: added no-leafref-check to l3vpn-ntw -> vpn-
profiles ->
                   valid-provider-identifiers -> routing-profile-identifier -
>id";
    }

    revision 2021-03-24 {
      description "Added: vpn-node -> local-autonomous-system deviation
mandatory true
                   Added: dot1q-vlan-tagged -> cvlan-id deviation mandatory
true";
    }

    revision 2021-03-19 {
      description "Removed deviation for bgp -> local-autonomous-system as the
field.";
    }

    revision 2021-02-22 {
      description "Corrected vpn-id regex pattern (removed extra backslash)";
    }

    revision 2021-02-01 {
```

```
        description "Removed: 'vpn-network-access -> service' deviations";
    }
    revision 2020-11-16 {
        description "Added: vpn-node -> rd and route-target deviation to match
string pattern from Flat L3 model
                     Added: ie-profile -> rd and route-target deviation to
match string pattern from Flat L3 model
                     Added: vpn-node -> local-autonomous-system deviation to
match union type of Flat L3 Model
                     Added: bgp -> local-autonomous-system deviation to match
union type of Flat L3 Model
                     Added: provider-address and prefix-length mandatory true
                     Added: must condition where at least one static-address ->
primary-address must be defined
                     Added: dot1q-vlan-tagged -> cvlan-id 1-4000 range
constraint";
    }

    revision 2020-10-21 {
        description "Modified: removed vpn-network-access deviation max-
elements 1";
    }

    revision 2020-10-08 {
        description "Added: vpn-network-access deviation min-elements 1, max-
elements 1
                     Added: routing-protocol deviation min-elements 1
                     Added: routing-profile-identifier id deviation leafref to
l3vpn-route-policy
                     Added: vpn-id deviation to match flat l3vpn name string
pattern
                     Added: ne-id deviation mandatory true
                     Added: port-id deviation mandatory true";
    }

    revision 2020-09-04 {
        description "Initial revision.";
    }

    // ====================== routing-protocol ======================
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:routing-protocols/ietf-l3vpn-ntw:routing-protocol" {
      deviate add {
        max-elements 1;
      }
    }
```

```
    // ====================== routing-profile-identifier ======================
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-profiles/ietf-
l3vpn-ntw:valid-provider-identifiers/ietf-l3vpn-ntw:routing-profile-
identifier/ietf-l3vpn-ntw:id" {
      deviate replace {
        type leafref {
          tailf:no-leafref-check;
          path "/cisco-flat-L3vpn-fp:l3vpn-route-policy/cisco-flat-L3vpn-
fp:name";
        }
      }
    }

    // ====================== vpn-service ======================
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw: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\-_]+';
        }
      }
    }

    // ====================== ip-connection ======================
    // ipv4 prefix-length mandatory when provider-address is defined
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:ip-connection/ietf-l3vpn-ntw:ipv4/ietf-l3vpn-ntw:static-addresses/ietf-
l3vpn-ntw:address/ietf-l3vpn-ntw:provider-address" {
      deviate add {
        mandatory true;
      }
    }
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:ip-connection/ietf-l3vpn-ntw:ipv4/ietf-l3vpn-ntw:static-addresses/ietf-
l3vpn-ntw:address/ietf-l3vpn-ntw:prefix-length" {
      deviate add {
        mandatory true;
      }
    }
    // ipv6 prefix-length mandatory when provider-address is defined
```

```
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:ip-connection/ietf-l3vpn-ntw:ipv6/ietf-l3vpn-ntw:static-addresses/ietf-
l3vpn-ntw:address/ietf-l3vpn-ntw:provider-address" {
      deviate add {
        mandatory true;
      }
    }
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:ip-connection/ietf-l3vpn-ntw:ipv6/ietf-l3vpn-ntw:static-addresses/ietf-
l3vpn-ntw:address/ietf-l3vpn-ntw:prefix-length" {
      deviate add {
        mandatory true;
      }
    }
    // At least one primary-address must be defined
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:ip-connection" {
      deviate add {
        must "ipv4/static-addresses/primary-address or ipv6/static-
addresses/primary-address" {
          error-message "At least one ip-connection primary-address must be
defined";
        }
      }
    }

    // ====================== connection ======================
    // L3VPN vlan-id has range 1-4000, L3NM dot1q cvlan-id is unrestricted
uint16
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:connection/ietf-l3vpn-ntw:tagged-interface/ietf-l3vpn-ntw:dot1q-vlan-
tagged/ietf-l3vpn-ntw:cvlan-id" {
      deviate replace {
        type uint16 {
          range "1..4000";
        }
      }
    }
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
```

```
ntw:connection/ietf-l3vpn-ntw:tagged-interface/ietf-l3vpn-ntw:dot1q-vlan-
tagged/ietf-l3vpn-ntw:cvlan-id" {
      deviate add {
        mandatory true;
      }
    }


    // ====================== ie-profile ======================
    // L3VPN route-distinguisher supports RD and RT type 0,1,2. So we have to
remove other L3NM supported types
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:ie-profiles/ietf-l3vpn-ntw:ie-
profile/ietf-l3vpn-ntw: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 "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:ie-profiles/ietf-l3vpn-ntw:ie-
profile/ietf-l3vpn-ntw:vpn-targets/ietf-l3vpn-ntw:vpn-target/ietf-l3vpn-
ntw:route-targets/ietf-l3vpn-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]|[1-9][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 "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:ne-id" {
      deviate replace {
```

```
        type leafref {
          path "/core-fp-common:dispatch-map/core-fp-common:device";
        }
      }
    }
    // L3VPN route-distinguisher supports RD and RT type 0,1,2. So we have to
remove other L3NM supported types
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw: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 "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
```

```
l3vpn-ntw:vpn-targets/ietf-l3vpn-ntw:vpn-target/ietf-l3vpn-ntw:route-
targets/ietf-l3vpn-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]|[1-9][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-autonomous-
system type uint32
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:local-autonomous-system" {
      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 "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:local-autonomous-system" {
      deviate add {
        //  L3NM local-autonomous-system is mandatory
        mandatory true;
      }
    }
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:ne-id" {
      deviate add {
        // ne-id maps to Flat L3VPN flat-L3vpn -> endpoint -> access-pe which
        //    is mandatory
        mandatory true;
      }
    }

    // ====================== vpn-network-access ======================
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access" {
      deviate add {
        // port-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 "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:port-id" {
      deviate add {
        // port-id maps to Flat L3VPN flat-L3vpn -> endpoint -> if-type which
        //    is mandatory
        mandatory true;
      }
    }
```

```
    deviation "/ietf-l3vpn-ntw:l3vpn-ntw/ietf-l3vpn-ntw:vpn-services/ietf-
l3vpn-ntw:vpn-service/ietf-l3vpn-ntw:vpn-nodes/ietf-l3vpn-ntw:vpn-node/ietf-
l3vpn-ntw:vpn-network-accesses/ietf-l3vpn-ntw:vpn-network-access/ietf-l3vpn-
ntw:port-id" {
      deviate replace {
        type string {
          pattern "(Bundle-
Ether|BVI|FiftyGigE|FortyGigE|FourHundredGigE|HundredGigE|Loopback|TenGigE|Twen
tyFiveGigE|TwoHundredGigE|GigabitEthernet|Ethernet)[0-9]+(/[0-9]+)*";
        }
      }
    }
}
```

IETF-L3VPN-NM Augmentations Model

```
module ietf-l3vpn-ntw-cisco-augmentations {

  yang-version 1.1;
  namespace "http://cisco.com/ns/nso/fp/examples/ietf-l3vpn-ntw-cisco-
augmentations";
  prefix l3vpn-ntw-augmentations;

  import ietf-l3vpn-ntw { prefix l3vpn-ntw; }
  import core-fp-common { prefix core-fp-common; }
  import cisco-tsdn-core-fp-common { prefix tsdn-core-fp-common; }
  import tailf-common { prefix tailf; }

  description "IETF L2VPN NTW Cisco NSO Augmentations";

  revision 2021-09-30 {
    description
      "Added: container srv6 under l3vpn-ntw -> vpn-services -> vpn-service ->
vpn-nodes
              -> vpn-node -> vpn-network-accesses -> vpn-network-access ->
routing-protocols
              -> routing-protocol -> bgp";
  }

  revision 2021-05-11 {
      description "Initial revision.";
  }

  augment "/l3vpn-ntw:l3vpn-ntw/l3vpn-ntw:vpn-services/l3vpn-ntw:vpn-service" {
    uses tsdn-core-fp-common:service-assurance-grouping {
      when "/tsdn-core-fp-common:enable-service-assurance = 'true'";
    }
  }
```

```
  augment "/l3vpn-ntw:l3vpn-ntw/l3vpn-ntw:vpn-services/l3vpn-ntw:vpn-
service/l3vpn-ntw:vpn-nodes"
        + "/l3vpn-ntw:vpn-node/l3vpn-ntw:vpn-network-accesses/l3vpn-ntw:vpn-
network-access"
        + "/l3vpn-ntw:routing-protocols/l3vpn-ntw:routing-protocol/l3vpn-
ntw:bgp" {
    container srv6 {
      presence true;

      tailf:info "Associate SRv6 Policy";
      description "Associate SRv6 Policy";

      must "not(contains(/core-fp-common:dispatch-map[core-fp-
common:device=current()/../../../../../../l3vpn-ntw:ne-id]/core-fp-common:ned-
id, 'cisco-ios-cli-'))" {
        error-message "SRv6 not supported on XE devices";
      }

      list address-family {
        min-elements 1;

        key name;
        leaf name {
          tailf:info
            "BGP activated address-family.";
          description
            "BGP activated address-family.";
          type leafref {
            path "../../../l3vpn-ntw: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;
          }
        }
      }
    }
  }
}
```

Appendix D – Sample Custom-Template Payloads

SR-TE CFP

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**

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-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>
        <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>
        <head-end>
          <name>PIOSXR-0</name>
        </head-end>
        ...
      </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-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>
        <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-6.74="http://tail-f.com/ns/ned-id/cisco-ios-cli-
6.74">cisco-ios-cli-6.74:cisco-ios-cli-6.74</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-6.74="http://tail-f.com/ns/ned-id/cisco-ios-cli-
6.74">cisco-ios-cli-6.74:cisco-ios-cli-6.74</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-tsdn-sr-te">
    <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-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-tsdn-sr-te">
        <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-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>
```

Example Function Packs

The custom template payloads for following services are discussed in this topic.

- **L2VPN Services**
- **L3VPN Services**
- **IETF-TE Services**
- **IETF-L2VPN-NM Services**
- **IETF-L3VPN-NM Services**

L2VPN Services

Sample service payload using custom-template

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
    <name>L2vpn-dynamic-02</name>
    <service-type>evpn-vpws</service-type>
    <custom-template>
========> APPLYING CUSTOM TEMPLATE AT GLOBAL LEVEL FOR ALL SITES
      <name>CT-CLI-logging</name>
    </custom-template>
    <flat-L2vpn-evpn-vpws>
      <evi-id>1000</evi-id>
```

```
      <local-site>
        <pe>PIOSXR-0</pe>
        <custom-template>
=========> APPLYING CUSTOM TEMPLATE AT LOCAL LEVEL FOR LOCAL SITE
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome</value>
          </variable>
        </custom-template>
        <if-type>GigabitEthernet</if-type>
        <if-id>1000</if-id>
        <if-description>L2VPN-Dynamic-02</if-description>
        <if-encap>dot1q</if-encap>
        <vlan-id>2</vlan-id>
        <sub-if-id>3</sub-if-id>
        <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
        <p2p-name>c</p2p-name>
        <evi-source>2</evi-source>
        <evi-target>6</evi-target>
      </local-site>
      <remote-site>
        <pe>PIOSXR-1</pe>
        <custom-template>
========> APPLYING CUSTOM TEMPLATE AT LOCAL LEVEL FOR REMOTE SITE
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome</value>
          </variable>
        </custom-template>
        <if-type>FortyGigE</if-type>
        <if-id>12</if-id>
        <if-description>L2VPN-Dynamic-02</if-description>
        <if-encap>untagged</if-encap>
        <xconnect-group-name>evpn_vpws_nso</xconnect-group-name>
        <p2p-name>EVPN-PIOSXR-1</p2p-name>
      </remote-site>
    </flat-L2vpn-evpn-vpws>
  </flat-L2vpn>
</config>
```

L3VPN Services

Sample custom-template for all nodes

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L3vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L3vpn">
    <name>L3</name>
    <custom-template>
======> APPLYING CUSTOM TEMPLATE AT GLOBAL LEVEL FOR ALL NODES
      <name>CT-CLI-logging</name>
    </custom-template>
…
…
          <metric>6</metric>
        </address-family>
      </vrf>
    </endpoint>
  </flat-L3vpn>
</config>
```

Sample custom-template per node

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L3vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L3vpn">
    <name>L3</name>
    <endpoint>
      <endpoint-name>cli-0</endpoint-name>
      <access-pe>PIOSXR-0</access-pe>
      <custom-template>
========> APPLYING CUSTOM TEMPLATE AT LOCAL LEVEL FOR GIVEN NODE
        <name>CT-CLI-logging</name>
      </custom-template>
      <if-type>Loopback</if-type>
      <if-id>7</if-id>
      <pe-ip-addr>169.1.1.1/32</pe-ip-addr>
      <as-no>65001</as-no>
…
…
          <vpn-target>
            <rt-value>100:102</rt-value>
            <rt-type>import</rt-type>
          </vpn-target>
        </address-family>
        <address-family>
          <address-family>ipv6</address-family>
```

```
          <redistribute-connected/>
          <metric>6</metric>
        </address-family>
      </vrf>
    </endpoint>
  </flat-L3vpn>
</config>
```

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 vpn-nodes
        <name>CT-CLI-banner</name>
        <variable>
          <name>BANNER_TEXT</name>
          <value>Welcome</value>
        </variable>
      </custom-template>
    </tunnel>
  </tunnels>
</te>
```

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>
==========> 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>
===============> APPLYING CUSTOM TEMPLATE AT LOCAL LEVEL FOR Vpn-node
            <name>CT-CLI-banner</name>
            <variable>
              <name>BANNER_TEXT</name>
              <value>Welcome_B</value>
            </variable>
          </custom-template>
          <signaling-options>
            <type>vpn-common:t-ldp</type>
            …
            </vpn-network-access>
          </vpn-network-accesses>
        </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>
          <name>CT-CLI-banner</name>
          <variable>
            <name>BANNER_TEXT</name>
            <value>Welcome_A</value>
          </variable>
        </custom-template>
        <vpn-nodes>
          <vpn-node>
            <ne-id>PIOSXR-1</ne-id>
            <custom-template>
              <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>
```

Appendix E: Use Cases

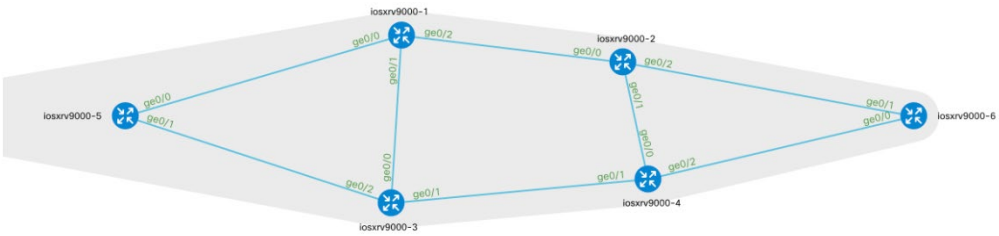
The use cases in this section provide an insight to the T-SDN FPs workflow. These use cases provide complete functional solution and thus help to understand and implement the various service models in the CFP.

To explain these use cases, the network orchestration and virtualization platform Virtual Internet Routing Lab (VIRL) for IOSXR devices and Cisco Modeling Labs (CML) for IOSXE devices are used to design the network topology, configure the virtual machines, and instantiate the network.

The following criteria was considered to design the topology and configure the virtual machines.

1. Install T-SDN FPs with the required NED packages. For more information on the installation, see the **Cisco T-SDN FP Bundle Installation Guide**.
2. Use the image versions supported for the T-SDN FP Bundle 3.0.0 from the Cisco website.
3. Copy the IOSXRv9000 image.
4. Create flavor before starting the simulation on VIRL/CML (Flavor - 16384 MB, 4 CPUs, and 80 GB Disk)
5. Start the VIRL/CML simulation with the topology configuration. The topology configuration is available in TSDN-0_0_11.virl.
6. Configure the network topology with Intermediate System to Intermediate System (isis), BGP protocol, and Segment Routing infrastructure setup.

   The reference topology appears as follows:



7. Ensure NSO VM is reachable to IOSXRV9000 spun in VIRL setup.
   Verify the configuration of the devices in the topology. Make sure the segment routing label table on isis is listing the devices connected in the topology.

Device	Verification
IOSXRv9000-1 (Core-R1)	RP/0/RP0/CPU0:Core-R1#show isis segment-routing label table Tue May 5 19:47:57.076 UTC IS-IS 1 IS Label Table

	Label Prefix/Interface ---------- ---------- 17001 Loopback0 18001 100.100.100.2/32 19001 100.100.100.3/32 20001 100.100.100.4/32 21001 100.100.100.5/32 22001 100.100.100.6/32
IOSXRv9000-2 **(Core-R2)**	RP/0/RP0/CPU0:Core-R2#show isis segment-routing label table Tue May 5 19:48:17.641 UTC IS-IS 1 IS Label Table Label Prefix/Interface ---------- ---------- 17001 100.100.100.1/32 18001 Loopback0 19001 100.100.100.3/32 20001 100.100.100.4/32 21001 100.100.100.5/32 22001 100.100.100.6/32
IOSXRv9000-3 **(Core-R3)**	RP/0/RP0/CPU0:Core-R3#show isis segment-routing label table Tue May 5 19:48:27.695 UTC IS-IS 1 IS Label Table Label Prefix/Interface ---------- ---------- 17001 100.100.100.1/32 18001 100.100.100.2/32 19001 Loopback0 20001 100.100.100.4/32 21001 100.100.100.5/32 22001 100.100.100.6/32
IOSXRv9000-4 **(Core-R4)**	RP/0/RP0/CPU0:Core-R4#show isis segment-routing label table Tue May 5 19:48:46.989 UTC IS-IS 1 IS Label Table

	Label Prefix/Interface ---------- ---------- 17001 100.100.100.1/32 18001 100.100.100.2/32 19001 100.100.100.3/32 20001 Loopback0 21001 100.100.100.5/32 22001 100.100.100.6/32
IOSXRv9000-5 **(PE-R5)**	RP/0/RP0/CPU0:PE-R5#show isis segment-routing label table Tue May 5 19:50:06.116 UTC IS-IS 1 IS Label Table Label Prefix/Interface ---------- ---------- 17001 100.100.100.1/32 18001 100.100.100.2/32 19001 100.100.100.3/32 20001 100.100.100.4/32 21001 Loopback0 22001 100.100.100.6/32
IOSXRv9000-6 **(PE-R6)**	RP/0/RP0/CPU0:PE-R6#show isis segment-routing label table Tue May 5 19:50:09.317 UTC IS-IS 1 IS Label Table Label Prefix/Interface ---------- ---------- 17001 100.100.100.1/32 18001 100.100.100.2/32 19001 100.100.100.3/32 20001 100.100.100.4/32 21001 100.100.100.5/32 22001 Loopback0

8. Add the devices configured in the VIRL topology (TSDN-0_0_11.virl) to the NSO device tree. The IOSXR9000v devices are added as CLI type devices.

```
configure
set devices authgroups group iosxr_authgroup default-map remote-name
admin remote-password cisco

set devices device Core-R1 authgroup iosxr_authgroup address 172.16.1.31
port 22 state admin-state unlocked
set devices device Core-R1 device-type cli ned-id cisco-iosxr-cli-7.33
set devices device Core-R1 trace raw

set devices device Core-R2 authgroup iosxr_authgroup address 172.16.1.32
port 830 state admin-state unlocked
set devices device Core-R2 device-type cli ned-id cisco-iosxr-cli-7.33
set devices device Core-R2 trace raw

set devices device Core-R3 authgroup iosxr_authgroup address 172.16.1.33
port 830 state admin-state unlocked
set devices device Core-R3 device-type cli ned-id cisco-iosxr-cli-7.33
set devices device Core-R3 trace raw

set devices device Core-R4 authgroup iosxr_authgroup address 172.16.1.34
port 830 state admin-state unlocked
set devices device Core-R4 device-type cli ned-id cisco-iosxr-cli-7.33
set devices device Core-R4 trace raw

set devices device PE-5 authgroup iosxr_authgroup address 172.16.1.35
port 830 state admin-state unlocked
set devices device PE-5 device-type cli ned-id cisco-iosxr-cli-7.33
set devices device PE-5 trace raw

set devices device PE-6 authgroup iosxr_authgroup address 172.16.1.36
port 22 state admin-state unlocked
set devices device PE-6 device-type cli ned-id cisco-iosxr-cli-7.33
set devices device PE-6 trace raw
commit
Run the following commands:
request devices fetch-ssh-host-keys
request devices connect
request devices sync-from
```

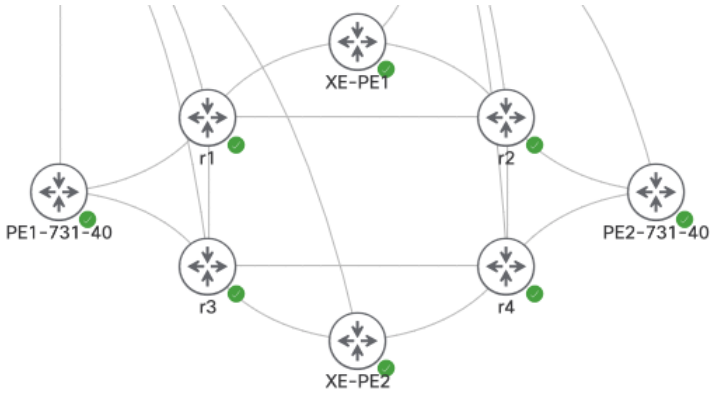
9. Add the interfaces in the IGP device's part to **rsvp** and **mpls traffic-eng** as follows for the end-to-end traffic on RSVP-TE tunnel to come up.

```
On PE/CE:
router isis 1
address-family ipv4 unicast
  mpls traffic-eng level-1-2
  mpls traffic-eng router-id Loopback0
!
!
ipv4 unnumbered mpls traffic-eng Loopback0
rsvp      -----------------> add all interfaces in device part of IGP.
interface GigabitEthernet0/0/0/0
  bandwidth percentage 90
!
interface GigabitEthernet0/0/0/1
  bandwidth percentage 90
!
interface GigabitEthernet0/0/0/2
  bandwidth percentage 90
!
!
mpls traffic-eng   -----------------> add all interfaces in device part
of IGP.
 interface GigabitEthernet0/0/0/0
!
interface GigabitEthernet0/0/0/1
!
interface GigabitEthernet0/0/0/2
!
!
End
```

10. Configure the network topology with Intermediate System to Intermediate System (isis), BGP protocol, and Segment Routing infrastructure setup.

11. IOSXR and IOSXE devices are set up according to the setup in CML.

    The reference topology for the IOSXR and IOSXE devices appear as follows. The PE1-731-40 and PE2-731-40 are the IOSXR devices.



The following table shows the configuration on IOSXE.

XE-PE1	**XE-PE-2**
XE-PE1#show segment-routing mpls connected-prefix-sid-map ipv4 PREFIX_SID_CONN_MAP ALGO_0 PREFIX_SID_CONN_MAP ALGO_0 Prefix/masklen SID Type Range Flags SRGB 100.100.100.7/32 3501 Indx 1 Y PREFIX_SID_PROTOCOL_ADV_MAP ALGO_0 Prefix/masklen SID Type Range Flags SRGB Source 100.100.100.1/32 501 Indx 1 Y IS-IS Level 2 0000.0000.0001 100.100.100.2/32 1001 Indx 1 Y IS-IS Level 2 0000.0000.0002 100.100.100.3/32 1501 Indx 1 Y IS-IS Level 2 0000.0000.0003 100.100.100.4/32 2001 Indx 1 Y IS-IS Level 2 0000.0000.0004 100.100.100.5/32 2501 Indx 1 Y IS-IS Level 2 0000.0000.0005 100.100.100.6/32 3001 Indx 1 Y IS-IS Level 2 0000.0000.0006	XE-PE2#show segment-routing mpls connected-prefix-sid-map ipv4 PREFIX_SID_CONN_MAP ALGO_0 Prefix/masklen SID Type Range Flags SRGB 100.100.100.8/32 4001 Indx 1 Y PREFIX_SID_PROTOCOL_ADV_MAP ALGO_0 Prefix/masklen SID Type Range Flags SRGB Source 100.100.100.1/32 501 Indx 1 Y IS-IS Level 2 0000.0000.0001 100.100.100.2/32 1001 Indx 1 Y IS-IS Level 2 0000.0000.0002 100.100.100.3/32 1501 Indx 1 Y IS-IS Level 2 0000.0000.0003 100.100.100.4/32 2001 Indx 1 Y IS-IS Level 2 0000.0000.0004 100.100.100.5/32 2501 Indx 1 Y IS-IS Level 2 0000.0000.0005 100.100.100.6/32 3001 Indx 1 Y IS-IS Level 2 0000.0000.0006

100.100.100.7/32 3501 Indx 1 Y BGP 0.0.0.0 100.100.100.8/32 4001 Indx 1 Y BGP 100.100.100.8 PREFIX_SID_CONN_MAP ALGO_128 Prefix/masklen SID Type Range Flags SRGB 100.100.100.7/32 19701 Abs 1 Y PREFIX_SID_PROTOCOL_ADV_MAP ALGO_128 Prefix/masklen SID Type Range Flags SRGB Source 100.100.100.7/32 3701 Indx 1 Y IS-IS Level 1 0000.0000.0007 100.100.100.8/32 4201 Indx 1 Y IS-IS Level 2 0000.0000.0008	100.100.100.7/32 3501 Indx 1 Y BGP 100.100.100.7 100.100.100.8/32 4001 Indx 1 Y BGP 0.0.0.0 PREFIX_SID_CONN_MAP ALGO_128 Prefix/masklen SID Type Range Flags SRGB 100.100.100.8/32 20201 Abs 1 Y PREFIX_SID_PROTOCOL_ADV_MAP ALGO_128 Prefix/masklen SID Type Range Flags SRGB Source 100.100.100.7/32 3701 Indx 1 Y IS-IS Level 2 0000.0000.0007 100.100.100.8/32 4201 Indx 1 Y IS-IS Level 1 0000.0000.0008

12. Onboard the devices on the NSO device tree to use CLI based NED for IOSXE. In this topology XE-PE1 and XE-PE2 are onboarded on the NSO device tree.

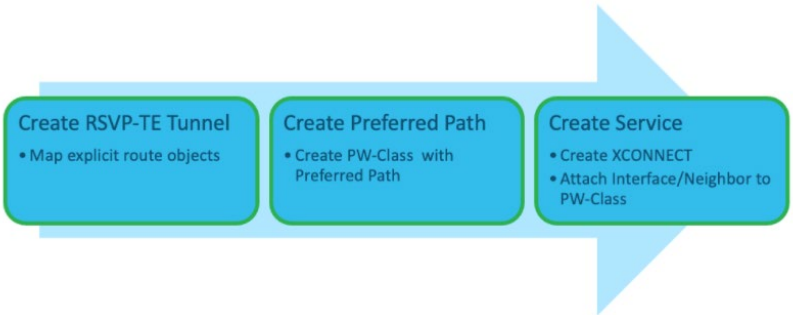
Use Cases

This section covers use cases for the various models and their implementations in T-SDN FPs.

Use Case 1: Instantiating L2VPN over RSVP-TE Tunnel – IOSXR

L2VPN allows dissimilar systems to be connected in a way that it appears as if they are connected to a common LAN segment. For more information about how to create L2VPN service, see section **Creating Flat L2VPN Services** in this documentation.

This use case discusses how to instantiate L2VPN service over the RSVP-TE tunnel on an IOSXR device. The following diagram shows the required tasks for this use case.



The detailed workflow includes:

1. Create and verify a bidirectional RSVP-TE service.
2. Perform a self-test for the end-end functionality of the tunnel.
3. Create Flat-L2VPN service.

4. Associate RSVP-TE to L2VPN service and verify the same.
5. Perform an end-to-end self-test on the L2VPN service.

To instantiate L2VPN over RSVP-TE tunnel on an IOSXR device:

1. Configure the TE tunnel for the explicit tunnel hop between the two PE devices – PE-5 and PE-6. This configuration establishes a bi-directional RSVP-TE service.

```
<te xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
  <tunnels>
    <tunnel>
      <name>EXPLICIT-TUNNEL-LOOSE-HOPS</name>
      <identifier>4332</identifier>
      <description>Explicit with tunnel-loose-hops</description>
      <source>200.200.200.5</source>
      <destination>200.200.200.6</destination>
      <bidirectional>true</bidirectional>
      <te-bandwidth>
        <generic>10</generic>
      </te-bandwidth>
      <signaling-type>te-types:path-setup-rsvp</signaling-type>
      <p2p-primary-paths>
        <p2p-primary-path>
          <name>10</name>
          <path-computation-method xmlns:te-
types="urn:ietf:params:xml:ns:yang:ietf-te-types">te-types:path-locally-
computed</path-computation-method>
          <preference>10</preference>
          <explicit-route-objects-always>
            <route-object-include-exclude>
              <index>10</index>
              <numbered-node-hop>
                <node-id>200.200.200.10</node-id>
                <hop-type>strict</hop-type>
              </numbered-node-hop>
            </route-object-include-exclude>
            <route-object-include-exclude>
              <index>20</index>
              <numbered-node-hop>
                <node-id>200.200.200.11</node-id>
                <hop-type>strict</hop-type>
              </numbered-node-hop>
            </route-object-include-exclude>
          </explicit-route-objects-always>
        </p2p-primary-path>
```

```
        <p2p-primary-path>
          <name>20</name>
          <path-computation-method xmlns:te-
types="urn:ietf:params:xml:ns:yang:ietf-te-types">te-types:path-locally-
computed</path-computation-method>
        </p2p-primary-path>
      </p2p-primary-paths>
      <head-end>PE-5</head-end>
      <tail-end>PE-6</tail-end>
    </tunnel>
  </tunnels>
</te>
```

2. Perform a dry-run and verify the tunnel configuration on the PE-5 and PE-6 devices as follows.

```
admin@ncs% commit dry-run
cli {
    local-node {
        data  devices {
                  device PE-5 {
                      config {
                          interface {
             +                tunnel-te 4332 {
             +                    description "Explicit with tunnel-
loose-hops";
             +                    ipv4 {
             +                        unnumbered {
             +                            Loopback 0;
             +                        }
             +                    }
             +                    signalled-name EXPLICIT-TUNNEL-LOOSE-
HOPS;
             +                    signalled-bandwidth {
             +                        bandwidth 10;
             +                    }
             +                    priority {
             +                        setup 7;
             +                        hold-value 7;
             +                    }
             +                    autoroute {
             +                        announce {
             +                            exclude-traffic {
             +                                segment-routing;
             +                            }
             +                        }
             +                    }
```

```
+                    destination 200.200.200.6;
+                    path-option 1 {
+                        dynamic {
+                        }
+                    }
+                    path-option 10 {
+                        dynamic {
+                        }
+                    }
+                }
            }
        }
    }
    device PE-6 {
        config {
            interface {
+               tunnel-te 4332 {
+                   description "Explicit with tunnel-
loose-hops";
+                   ipv4 {
+                       unnumbered {
+                           Loopback 0;
+                       }
+                   }
+                   signalled-name EXPLICIT-TUNNEL-LOOSE-
HOPS;
+                   signalled-bandwidth {
+                       bandwidth 10;
+                   }
+                   priority {
+                       setup 7;
+                       hold-value 7;
+                   }
+                   autoroute {
+                       announce {
+                           exclude-traffic {
+                               segment-routing;
+                           }
+                       }
+                   }
+                   destination 200.200.200.5;
+                   path-option 1 {
+                       dynamic {
+                       }
+                   }
```

```
+                    path-option 10 {
+                        dynamic {
+                        }
+                    }
+                }
                }
            }
        }
    }
    te {
        tunnels {
+           tunnel EXPLICIT-TUNNEL-LOOSE-HOPS {
+               identifier 4332;
+               description "Explicit with tunnel-loose-hops";
+               source 200.200.200.5;
+               destination 200.200.200.6;
+               bidirectional true;
+               te-bandwidth {
+                   generic 10;
+               }
+               signaling-type te-types:path-setup-rsvp;
+               p2p-primary-paths {
+                   p2p-primary-path 10 {
+                       path-computation-method path-locally-
computed;
+                       preference 10;
+                       explicit-route-objects-always {
+                           route-object-include-exclude 10 {
+                               numbered-node-hop {
+                                   node-id 200.200.200.10;
+                                   hop-type strict;
+                               }
+                           }
+                           route-object-include-exclude 20 {
+                               numbered-node-hop {
+                                   node-id 200.200.200.11;
+                                   hop-type strict;
+                               }
+                           }
+                       }
+                   }
+                   p2p-primary-path 20 {
+                       path-computation-method path-locally-
computed;
+                   }
```

```
                +             }
                +             head-end PE-5;
                +             tail-end PE-6;
                +         }
                      }
                  }
        }
    }
```

3. Commit the configuration to deploy the tunnel service between the two IOSXR PE devices.

```
admin@ncs% commit commit-queue async
Commit complete.
```

4. Verify the plan status for the tunnel service. The plan displays a **reached** status for a successful deployment of the service.

```
admin@ncs% run show te tunnels tunnel-plan EXPLICIT-TUNNEL-LOOSE-HOPS plan

                          BACK        STATUS
TYPE        NAME          TRACK  GOAL CODE   STATE                                          STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------
self        self          false  -    -      init                                           reached  2021-02-09T00:31:57
                                             ready                                          reached  2021-02-09T00:32:12
source      200.200.200.5 false  -    -      init                                           reached  2021-02-09T00:31:57
                                             ietf-te-fp-tunnel-nano-plan-services:config-apply reached 2021-02-09T00:31:57
                                             ready                                          reached  2021-02-09T00:32:12
destination 200.200.200.6 false  -    -      init                                           reached  2021-02-09T00:31:57
                                             ietf-te-fp-tunnel-nano-plan-services:config-apply reached 2021-02-09T00:31:57
                                             ready                                          reached  2021-02-09T00:32:12
```

5. Perform a self-test on the tunnel service to ensure the TE tunnel is set up between both the PE devices.

```
admin@ncs% run request te tunnels tunnel EXPLICIT-TUNNEL-LOOSE-HOPS
action self-test
status success
```

6. Create a L2VPN service.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
  <name>L2VPN-P2P-Service</name>
  <service-type>p2p</service-type>
  <flat-L2vpn-p2p>
    <pw-id>200</pw-id>
    <local-site>
      <pe>PE-5</pe>
      <if-type>GigabitEthernet</if-type>
      <if-id>0/0/0/3</if-id>
      <if-description>L2vpn-p2p-flat-remote</if-description>
      <if-encap>dot1q</if-encap>
      <vlan-id>300</vlan-id>
      <xconnect-group-name>TSDN-L2vpn</xconnect-group-name>
      <xconnect-encapsulation>mpls</xconnect-encapsulation>
      <p2p-name>p2p-flat</p2p-name>
```

```
        <control-word>no</control-word>
        <pw-class>TSDN-L2vpn-p2p</pw-class>
        <xconnect-local-ip>200.200.200.5</xconnect-local-ip>
        <xconnect-remote-ip>200.200.200.6</xconnect-remote-ip>
        <mpls-local-label>101</mpls-local-label>
        <mpls-remote-label>202</mpls-remote-label>
      </local-site>
      <remote-site>
        <pe>PE-6</pe>
        <if-type>GigabitEthernet</if-type>
        <if-id>0/0/0/3</if-id>
        <if-description>L2vpn-p2p-flat-remote</if-description>
        <if-encap>dot1q</if-encap>
        <vlan-id>300</vlan-id>
        <xconnect-group-name>TSDN-L2vpn</xconnect-group-name>
        <p2p-name>p2p-flat</p2p-name>
        <pw-class>TSDN-L2vpn-p2p</pw-class>
      </remote-site>
    </flat-L2vpn-p2p>
  </flat-L2vpn>
  </config>
```

7. Associate RSVP-TE service to the L2VPN service. This allows the L2VPN service to use the tunnel service.

```
<flat-L2vpn xmlns="http://cisco.com/ns/nso/fp/examples/cisco-tsdn-flat-
L2vpn">
  <name>L2VPN-P2P-Service</name>
  <service-type>p2p</service-type>
  <flat-L2vpn-p2p>
    <local-site>
      <rsvp-te>
        <preferred-path>
          <ietf-te-service>EXPLICIT-TUNNEL-LOOSE-HOPS</ietf-te-service>
        </preferred-path>
      </rsvp-te>
    </local-site>
  </flat-L2vpn-p2p>
</flat-L2vpn>
```

8. Perform a dry-run and verify the configuration for the rsvp-te association.

```
admin@ncs% commit dry-run
cli {
    local-node {
        data +flat-L2vpn L2VPN-P2P-Service {
             +    service-type p2p;
             +    flat-L2vpn-p2p {
```

```
+        pw-id 200;
+        local-site {
+            pe PE-5;
+            if-type GigabitEthernet;
+            if-id 0/0/0/3;
+            if-description L2vpn-p2p-flat-remote;
+            if-encap dot1q;
+            vlan-id 300;
+            xconnect-group-name TSDN-L2vpn;
+            xconnect-encapsulation mpls;
+            p2p-name p2p-flat;
+            rsvp-te {
+                preferred-path {
+                    ietf-te-service EXPLICIT-TUNNEL-LOOSE-
HOPS;
+                }
+            }
+            control-word no;
+            pw-class TSDN-L2vpn-p2p;
+            xconnect-local-ip 200.200.200.5;
+            xconnect-remote-ip 200.200.200.6;
+            mpls-local-label 101;
+            mpls-remote-label 202;
+        }
+        remote-site {
+            pe PE-6;
+            if-type GigabitEthernet;
+            if-id 0/0/0/3;
+            if-description L2vpn-p2p-flat-remote;
+            if-encap dot1q;
+            vlan-id 300;
+            xconnect-group-name TSDN-L2vpn;
+            p2p-name p2p-flat;
+            pw-class TSDN-L2vpn-p2p;
+        }
+    }
+}
 devices {
     device PE-5 {
         config {
             interface {
                 GigabitEthernet-subinterface {
+                    GigabitEthernet 0/0/0/3.300 {
+                        mode l2transport;
```

```
+                        description L2vpn-p2p-flat-remote;
+                        encapsulation {
+                            dot1q {
+                                vlan-id 300;
+                            }
+                        }
+                    }
                    }
                }
                l2vpn {
+                   pw-class TSDN-L2vpn-p2p {
+                       encapsulation {
+                           mpls {
+                               preferred-path {
+                                   interface tunnel-te; -----
------------------> Associating TE Tunnel id based on TE Tunnel Service.
+                                   id 4332;
+                               }
+                           }
+                       }
+                   }
                    xconnect {
+                       group TSDN-L2vpn {
+                           p2p p2p-flat {
+                               interface
GigabitEthernet0/0/0/3.300;
+                               neighbor 200.200.200.6 200 {
+                                   ip-version ipv4;
+                                   mpls {
+                                       static {
+                                           label {
+                                               local 101;
+                                               remote 202;
+                                           }
+                                       }
+                                   }
+                                   pw-class TSDN-L2vpn-p2p;
+                               }
+                           }
+                       }
                    }
                }
            }
        }
        device PE-6 {
```

```
                        config {
                            interface {
                                GigabitEthernet-subinterface {
                +                   GigabitEthernet 0/0/0/3.300 {
                +                       mode l2transport;
                +                       description L2vpn-p2p-flat-remote;
                +                       encapsulation {
                +                           dot1q {
                +                               vlan-id 300;
                +                           }
                +                       }
                +                   }
                                }
                            }
                            l2vpn {
                +               pw-class TSDN-L2vpn-p2p {
                +                   encapsulation {
                +                       mpls {
                +                       }
                +                   }
                +               }
                                xconnect {
                +                   group TSDN-L2vpn {
                +                       p2p p2p-flat {
                +                           interface
GigabitEthernet0/0/0/3.300;
                +                           neighbor 200.200.200.5 200 {
                +                               ip-version ipv4;
                +                               mpls {
                +                                   static {
                +                                       label {
                +                                           local 202;
                +                                           remote 101;
                +                                       }
                +                                   }
                +                               }
                +                               pw-class TSDN-L2vpn-p2p;
                +                           }
                +                       }
                +                   }
                                }
                            }
                        }
                    }
                }
            }
        }
```

9. Verify the plan status for the L2VPN service. The plan displays a **reached** status for a successful deployment of the L2VPN service.

```
admin@ncs% run show flat-L2vpn-plan L2VPN-P2P-Service plan

                    BACK         STATUS
TYPE         NAME   TRACK  GOAL  CODE    STATE                                                  STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------
self         self   false  -     -       init                                                   reached  2021-02-09T00:50:06
                                         ready                                                  reached  2021-02-09T00:50:17
local-site   PE-5   false  -     -       init                                                   reached  2021-02-09T00:50:06
                                         cisco-flat-L2vpn-fp-nano-plan-services:config-apply    reached  2021-02-09T00:50:06
                                         ready                                                  reached  2021-02-09T00:50:17
remote-site  PE-6   false  -     -       init                                                   reached  2021-02-09T00:50:06
                                         cisco-flat-L2vpn-fp-nano-plan-services:config-apply    reached  2021-02-09T00:50:06
                                         ready                                                  reached  2021-02-09T00:50:17
```

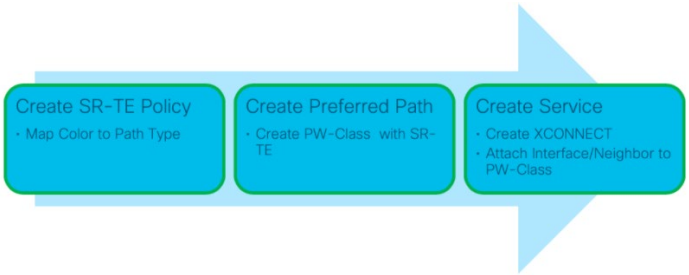
10. Perform an end-to-end self-test to ensure the tunnel service for L2VPN is set up on both the PE devices.

```
admin@ncs# flat-L2vpn L2VPN-P2P-Service action self-test
status success
message local-site: success. remote-site: success
```

Use Case 2: Instantiating L2NM Service Over SR-TE – IOSXE

The IETF-L2VPN-NM service provides an IETF model overlay of the flat L2VPN configuration.

This use case discusses how to instantiate L2NM over SR-TE on an IOSXE device. The following diagram shows the required tasks for this use case.



The detailed workflow includes:

1. Create a SR-TE policy with metric latency for color 40.
2. Verify the SR-TE policy admin and operational status.
3. Create Flat-L2NM service.
4. Attach the SR-TE policy to the L2NM service.
5. Perform an end-to-end self-test on the L2VPN service.

To instantiate L2NM service over SR-TE:

1. Create SR-Policy with metric-latency for color 40 on the IOSXE devices: XE-PE1 and XE-PE2 devices.

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

```
  <sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
  <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te-sr-
policies">
    <policy>
      <name>DYN-LATENCY-XE-1</name>
      <head-end>
        <name>XE-PE1</name>
      </head-end>
      <tail-end>100.100.100.8</tail-end>
      <color>40</color>
      <binding-sid>15000</binding-sid>
      <path>
        <preference>100</preference>
        <dynamic>
          <metric-type>latency</metric-type>
        </dynamic>
      </path>
    </policy>
    <policy>
      <name>DYN-LATENCY-XE-2</name>
      <head-end>
        <name>XE-PE2</name>
      </head-end>
      <tail-end>100.100.100.7</tail-end>
      <color>40</color>
      <binding-sid>15000</binding-sid>
      <path>
        <preference>100</preference>
        <dynamic>
          <metric-type>latency</metric-type>
        </dynamic>
      </path>
    </policy>
  </policies>
</sr-te>
</config>
```

2. Perform a dry-run and verify the configuration on the devices.

```
admin@ncs% commit dry-run
cli {
    local-node {
        data  devices {
                  device XE-PE1 {
                      config {
                          segment-routing {
```

```
                    traffic-eng {
+                       policy DYN-LATENCY-XE-1 {
+                           color {
+                               id 40;
+                               end-point 100.100.100.8;
+                           }
+                           binding-sid {
+                               mpls 15000;
+                           }
+                           candidate-paths {
+                               preference 100 {
+                                   constraints {
+                                       segments {
+                                           dataplane {
+                                               mpls;
+                                           }
+                                       }
+                                   }
+                                   dynamic {
+                                       metric {
+                                           type delay;
+                                       }
+                                   }
+                               }
+                           }
+                       }
                    }
                }
            }
        }
        device XE-PE2 {
            config {
                segment-routing {
                    traffic-eng {
+                       policy DYN-LATENCY-XE-2 {
+                           color {
+                               id 40;
+                               end-point 100.100.100.7;
+                           }
+                           binding-sid {
+                               mpls 15000;
+                           }
+                           candidate-paths {
+                               preference 100 {
```

```
+                                   constraints {
+                                       segments {
+                                           dataplane {
+                                               mpls;
+                                           }
+                                       }
+                                   }
+                                   dynamic {
+                                       metric {
+                                           type delay;
+                                       }
+                                   }
+                               }
+                           }
+                       }
                    }
                }
            }
        }
    }
    cisco-sr-te-cfp:sr-te {
        policies {
+           policy DYN-LATENCY-XE-1 {
+               head-end XE-PE1;
+               tail-end 100.100.100.8;
+               color 40;
+               binding-sid 15000;
+               path 100 {
+                   dynamic {
+                       metric-type latency;
+                   }
+               }
+           }
+           policy DYN-LATENCY-XE-2 {
+               head-end XE-PE2;
+               tail-end 100.100.100.7;
+               color 40;
+               binding-sid 15000;
+               path 100 {
+                   dynamic {
+                       metric-type latency;
+                   }
+               }
+           }
```

```
                        }
                    }
        }
    }
    }
```

3. Commit the configuration to deploy the service.

```
admin@ncs% commit commit-queue async
Commit complete.
```

4. Verify the plan status for the SR-TE policy. Here, the policy is attached to both the PE devices. The plan status displays **reached** status upon successful implementation.

```
admin@ncs% run show cisco-sr-te-cfp:sr-te policies policy-plan DYN-LATENCY-XE-1 plan

                                                          BACK        STATUS
TYPE                                                NAME  TRACK  GOAL CODE   STATE                                                    STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------------------------------------
self                                                self  false  -    -      init                                                     reached  2021-02-09T18:44:23
                                                                             ready                                                    reached  2021-02-09T18:44:29
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end  XE-PE1  false  -    -      init                                                     reached  2021-02-09T18:44:23
                                                                             cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply  reached  2021-02-09T18:44:23
                                                                             ready                                                    reached  2021-02-09T18:44:29
```

```
admin@ncs% run show cisco-sr-te-cfp:sr-te policies policy-plan DYN-LATENCY-XE-2 plan

                                                          BACK        STATUS
TYPE                                                NAME  TRACK  GOAL CODE   STATE                                                    STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------------------------------------
self                                                self  false  -    -      init                                                     reached  2021-02-09T18:44:24
                                                                             ready                                                    reached  2021-02-09T18:44:29
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end  XE-PE2  false  -    -      init                                                     reached  2021-02-09T18:44:24
                                                                             cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply  reached  2021-02-09T18:44:24
                                                                             ready                                                    reached  2021-02-09T18:44:29
```

5. Verify the SR-TE policy admin and operational status on both devices.

```
XE-PE1#show segment-routing traffic-eng policy name DYN-LATENCY-XE-1
Name: DYN-LATENCY-XE-1 (Color: 40 End-point: 100.100.100.8)
  Owners : CLI
  Status:
    Admin: up, Operational: up for 00:00:27 (since 02-09 18:43:55.692)
  Candidate-paths:
    Preference 100 (CLI):
      Dynamic (active)
        Metric Type: DELAY, Path Accumulated Metric: 30
          20001 [Prefix-SID, 100.100.100.8]
  Attributes:
    Binding SID: 15000
      Allocation mode: explicit
      State: Programmed

XE-PE2#show segment-routing traffic-eng policy name DYN-LATENCY-XE-2
Name: DYN-LATENCY-XE-2 (Color: 40 End-point: 100.100.100.7)
  Owners : CLI
  Status:
    Admin: up, Operational: up for 00:01:09 (since 02-09 18:43:56.013)
  Candidate-paths:
    Preference 100 (CLI):
      Dynamic (active)
```

```
          Metric Type: DELAY, Path Accumulated Metric: 30
            19501 [Prefix-SID, 100.100.100.7]
      Attributes:
        Binding SID: 15000
          Allocation mode: explicit
          State: Programmed
```

6. Create the flat L2NM 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-p2p-XE</vpn-id>
        <vpn-svc-type>vpn-common:t-ldp</vpn-svc-type>
        <vpn-nodes>
          <vpn-node>
            <vpn-node-id>XE-PE1</vpn-node-id>
            <ne-id>XE-PE1</ne-id>
            <signaling-options>
              <type>vpn-common:t-ldp</type>
              <t-ldp-pwe>
                <ac-pw-list>
                  <peer-addr>100.100.100.8</peer-addr>
                  <vc-id>1001</vc-id>
                  <mpls-label>111</mpls-label>
                </ac-pw-list>
              </t-ldp-pwe>
            </signaling-options>
            <vpn-network-accesses>
              <vpn-network-access>
                <id>l2vpn-p2p-ac1</id>
                <connection>
                  <encapsulation-type>vpn-common:dot1q</encapsulation-
type>
                  <dot1q-interface>
                    <l2-access-type>vpn-common:dot1q</l2-access-type>
                    <dot1q>
                      <physical-inf>GigabitEthernet4</physical-inf>
                      <c-vlan-id>601</c-vlan-id>
                    </dot1q>
                  </dot1q-interface>
                </connection>
              </vpn-network-access>
            </vpn-network-accesses>
          </vpn-node>
```

```
            <vpn-node>
              <vpn-node-id>XE-PE2</vpn-node-id>
              <ne-id>XE-PE2</ne-id>
              <signaling-options>
                <type>vpn-common:t-ldp</type>
                <t-ldp-pwe>
                  <ac-pw-list>
                    <peer-addr>100.100.100.7</peer-addr>
                    <vc-id>1001</vc-id>
                    <mpls-label>112</mpls-label>
                  </ac-pw-list>
                </t-ldp-pwe>
              </signaling-options>
              <vpn-network-accesses>
                <vpn-network-access>
                  <id>l2vpn-p2p-ac1</id>
                  <connection>
                    <encapsulation-type>vpn-common:dot1q</encapsulation-
type>
                    <dot1q-interface>
                      <l2-access-type>vpn-common:dot1q</l2-access-type>
                      <dot1q>
                        <physical-inf>GigabitEthernet4</physical-inf>
                        <c-vlan-id>601</c-vlan-id>
                      </dot1q>
                    </dot1q-interface>
                  </connection>
                </vpn-network-access>
              </vpn-network-accesses>
            </vpn-node>
          </vpn-nodes>
        </vpn-service>
      </vpn-services>
    </l2vpn-ntw>
  </config>
```

7. Perform a dry-run to verify the L2NM service configuration.

```
admin@ncs% commit dry-run
  cli {
      local-node {
          data +flat-L2vpn L2NM-l2nm-p2p-XE-internal {
                +    service-type p2p;
                +    flat-L2vpn-p2p {
                +        pw-id 1001;
                +        local-site {
```

```
+            pe XE-PE1;
+            if-type GigabitEthernet;
+            if-id 4;
+            if-encap dot1q;
+            vlan-id 601;
+            sub-if-id 601;
+            xconnect-group-name l2vpn-p2p-ac1;
+            p2p-name l2vpn-p2p-ac1;
+            control-word no;
+            pw-class pw-class-l2vpn-p2p-ac1;
+            xconnect-local-ip 100.100.100.7;
+            xconnect-remote-ip 100.100.100.8;
+            mpls-local-label 111;
+            mpls-remote-label 112;
+        }
+        remote-site {
+            pe XE-PE2;
+            if-type GigabitEthernet;
+            if-id 4;
+            if-encap dot1q;
+            vlan-id 601;
+            sub-if-id 601;
+            xconnect-group-name l2vpn-p2p-ac1;
+            p2p-name l2vpn-p2p-ac1;
+            control-word no;
+            pw-class pw-class-l2vpn-p2p-ac1;
+        }
+    }
+}
 devices {
     device XE-PE1 {
         config {
             interface {
                 GigabitEthernet 4 {
                     service {
+                        instance 601 {
+                            ethernet;
+                            encapsulation {
+                                dot1q {
+                                    id 601;
+                                }
+                            }
+                        }
                     }
```

```
                                    }
                +                   pseudowire 1001 {
                +                       source {
                +                           template {
                +                               type pseudowire;
                +                               name pw-class-l2vpn-p2p-ac1;
                +                           }
                +                       }
                +                       encapsulation mpls;
                +                       neighbor {
                +                           address 100.100.100.8;
                +                           vcid 1001;
                +                       }
                +                       label {
                +                           local-pseudowire-label 111;
                +                           remote-pseudowire-label 112;
                +                       }
                +                   }
                                }
                +               template pw-class-l2vpn-p2p-ac1 {
                +                   type pseudowire;
                +                   encapsulation mpls;
                +                   signaling {
                +                       protocol none;
                +                   }
                +               }
                                l2vpn-xconnect {
                                    l2vpn {
                                        xconnect {
                +                           context l2vpn-p2p-ac1 {
                +                               member {
                +                                   interface-list
pseudowire1001;
                +                                   member-list
GigabitEthernet4 601;
                +                               }
                +                           }
                                        }
                                    }
                                }
                            }
                        }
                        device XE-PE2 {
                            config {
                                interface {
```

```
                                GigabitEthernet 4 {
                                    service {
                +                       instance 601 {
                +                           ethernet;
                +                           encapsulation {
                +                               dot1q {
                +                                   id 601;
                +                               }
                +                           }
                +                       }
                                    }
                                }
                +               pseudowire 1001 {
                +                   source {
                +                       template {
                +                           type pseudowire;
                +                           name pw-class-l2vpn-p2p-ac1;
                +                       }
                +                   }
                +                   encapsulation mpls;
                +                   neighbor {
                +                       address 100.100.100.7;
                +                       vcid 1001;
                +                   }
                +                   label {
                +                       local-pseudowire-label 112;
                +                       remote-pseudowire-label 111;
                +                   }
                +               }
                            }
                +           template pw-class-l2vpn-p2p-ac1 {
                +               type pseudowire;
                +               encapsulation mpls;
                +               signaling {
                +                   protocol none;
                +               }
                +           }
                            l2vpn-xconnect {
                                l2vpn {
                                    xconnect {
                +                       context l2vpn-p2p-ac1 {
                +                           member {
                +                               interface-list
pseudowire1001;
```

```
                +                                   member-list
GigabitEthernet4 601;
                +                              }
                +                          }
                                       }
                                   }
                               }
                           }
                       }
                   }
               }
           }
       }
   }
```

8. Commit to deploy the L2NM service.

```
admin@ncs% commit commit-queue async
Commit complete.
```

9. Verify the plan status for the L2NM service.

```
admin@ncs% run show l2vpn-ntw vpn-services vpn-service-plan l2nm-p2p-XE plan

                                          BACK        STATUS
TYPE                                NAME  TRACK  GOAL CODE   STATE                                      STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------------
self                                self  false  -    -      init                                       reached  2021-02-09T19:41:09
                                                             ready                                      reached  2021-02-09T19:41:16
ietf-l2vpn-ntw-nano-services:vpn-node XE-PE1 false  -    -      init                                       reached  2021-02-09T19:41:09
                                                             ietf-l2vpn-ntw-nano-services:config-apply  reached  2021-02-09T19:41:09
                                                             ready                                      reached  2021-02-09T19:41:16
ietf-l2vpn-ntw-nano-services:vpn-node XE-PE2 false  -    -      init                                       reached  2021-02-09T19:41:09
                                                             ietf-l2vpn-ntw-nano-services:config-apply  reached  2021-02-09T19:41:09
                                                             ready                                      reached  2021-02-09T19:41:16
```

10. Attach the SR-TE policy to the L2NM service.

```
<l2vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l2vpn-ntw">
  <vpn-services>
    <vpn-service>
      <vpn-id>l2nm-p2p-XE</vpn-id>
      <vpn-nodes>
        <vpn-node>
          <vpn-node-id>XE-PE1</vpn-node-id>
          <ne-id>XE-PE1</ne-id>
          <te-service-mapping>
            <te-mapping>
              <sr-policy>
                <policy>DYN-LATENCY-XE-1</policy>
                <fallback>disable</fallback>
              </sr-policy>
            </te-mapping>
          </te-service-mapping>
        </vpn-node>
        <vpn-node>
```

```
            <vpn-node-id>XE-PE2</vpn-node-id>
            <ne-id>XE-PE2</ne-id>
            <te-service-mapping>
              <te-mapping>
                <sr-policy>
                  <policy>DYN-LATENCY-XE-2</policy>
                  <fallback>disable</fallback>
                </sr-policy>
              </te-mapping>
            </te-service-mapping>
          </vpn-node>
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l2vpn-ntw>
```

11. Perform a dry-run to verify the association.

```
admin@ncs% commit dry-run
  cli {
      local-node {
          data  flat-L2vpn L2NM-l2nm-p2p-XE-internal {
                    flat-L2vpn-p2p {
                        local-site {
                +           sr-te {
                +               preferred-path {
                +                   policy DYN-LATENCY-XE-1;
                +                   fallback disable;
                +               }
                +           }
                        }
                        remote-site {
                +           sr-te {
                +               preferred-path {
                +                   policy DYN-LATENCY-XE-2;
                +                   fallback disable;
                +               }
                +           }
                        }
                    }
                }
                devices {
                    device XE-PE1 {
                        config {
                            template pw-class-l2vpn-p2p-ac1 {
```

```
                    preferred-path {
                        segment-routing {
                            traffic-eng {
+                               policy DYN-LATENCY-XE-1;
                            }
                        }
                    }
                }
            }
        }
        device XE-PE2 {
            config {
                template pw-class-l2vpn-p2p-ac1 {
                    preferred-path {
                        segment-routing {
                            traffic-eng {
+                               policy DYN-LATENCY-XE-2;
                            }
                        }
                    }
                }
            }
        }
    }
    l2vpn-ntw {
        vpn-services {
            vpn-service l2nm-p2p-XE {
                vpn-nodes {
                    vpn-node XE-PE1 {
                        te-service-mapping {
                            te-mapping {
+                               sr-policy {
+                                   policy DYN-LATENCY-XE-1;
+                                   fallback disable;
+                               }
                            }
                        }
                    }
                    vpn-node XE-PE2 XE-PE2 {
                        te-service-mapping {
                            te-mapping {
+                               sr-policy {
+                                   policy DYN-LATENCY-XE-2;
+                                   fallback disable;
```

```
                    +                                    }
                                                    }
                                                }
                                            }
                                        }
                                    }
                                }
                            }
                        }
                    }
```

12. Commit to deploy the service.

```
admin@ncs% commit commit-queue async
Commit complete.
```

13. Perform an end-to-end self-test on L2NM service.

```
admin@ncs# l2vpn-ntw vpn-services vpn-service l2nm-p2p-XE self-test
status success
message local-site: success. remote-site: success
```

Use Case 3: Instantiating L3NM over SR-TE – IOSXE

The IETF-L3VPN-NM service provides an IETF model overlay of the flat L3VPN configuration. This use case discusses how to instantiate L3NM over SR-TE on an IOSXE device. The following diagram shows the required tasks for this use case.

The detailed workflow includes:

1. Create a SR-TE policy with Metric-te for color 30.
2. Verify the SR-TE policy admin and operational status.
3. Create Flat-L3NM service.
4. Attach the SR-TE policy to the L3NM service.
5. Perform an end-to-end self-test on the L3NM service.

To instantiate L3NM over SR-TE:

1. Create the SR-TE policy with metric margin.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
<sr-te xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te">
  <policies xmlns="http://cisco.com/ns/nso/cfp/cisco-tsdn-sr-te-sr-
policies">
    <policy>
      <name>sr-policy-1</name>
      <head-end>
        <name>XE-PE1</name>
      </head-end>
      <tail-end>100.100.100.8</tail-end>
      <color>30</color>
      <path>
        <preference>10</preference>
        <dynamic>
          <metric-type>te</metric-type>
          <metric-margin>
            <absolute>80</absolute>
          </metric-margin>
        </dynamic>
      </path>
    </policy>
  </policies>
</sr-te>
</config>
```

2. Perform a dry-run to verify the configuration.

```
admin@ncs% commit dry-run
  cli {
      local-node {
          data  devices {
                    device XE-PE1 {
                        config {
                            segment-routing {
                                traffic-eng {
              +                   policy sr-policy-1 {
              +                       color {
              +                           id 30;
              +                           end-point 100.100.100.8;
              +                       }
              +                       candidate-paths {
              +                           preference 10 {
              +                               constraints {
```

```
+                                       segments {
+                                           dataplane {
+                                               mpls;
+                                           }
+                                       }
+                                   }
+                                   dynamic {
+                                       metric {
+                                           type te;
+                                           margin {
+                                               absolute 80;
+                                           }
+                                       }
+                                   }
+                               }
+                           }
+                       }
                    }
                }
            }
        }
        cisco-sr-te-cfp:sr-te {
            policies {
+               policy sr-policy-1 {
+                   head-end XE-PE1;
+                   tail-end 100.100.100.8;
+                   color 30;
+                   path 10 {
+                       dynamic {
+                           metric-type te;
+                           metric-margin {
+                               absolute 80;
+                           }
+                       }
+                   }
+               }
            }
        }
    }
}
```

3. Commit the configuration to deploy the service.

```
admin@ncs% commit commit-queue async
Commit complete.
```

4. Verify the plan status for the SR-TE policy.

```
admin@ncs% run show cisco-sr-te-cfp:sr-te policies policy-plan sr-policy-1 plan

                                                                  BACK         STATUS
TYPE                                                   NAME       TRACK  GOAL  CODE    STATE                                                          STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                   self       false  -     -       init                                                           reached  2021-02-09T23:46:45
                                                                                       ready                                                          reached  2021-02-09T23:46:49
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end XE-PE1    false  -     -       init                                                           reached  2021-02-09T23:46:45
                                                                                       cisco-sr-te-cfp-sr-policies-nano-plan-services:config-apply    reached  2021-02-09T23:46:45
                                                                                       ready                                                          reached  2021-02-09T23:46:49
```

5. Verify the SR-TE policy admin and the operational status.

```
XE-PE1#show segment-routing traffic-eng policy name sr-policy-1
Name: sr-policy-1 (Color: 30 End-point: 100.100.100.8)
  Owners : CLI
  Status:
    Admin: up, Operational: up for 00:04:07 (since 02-09 23:46:16.151)
  Candidate-paths:
    Preference 10 (CLI):
      Dynamic (active)
        Metric Type: TE, Path Accumulated Metric: 30
          20001 [Prefix-SID, 100.100.100.8]
  Attributes:
    Binding SID: 1000
      Allocation mode: dynamic
      State: Programmed
```

6. Create L3VPN route-policy for color30 and associate the route-policy to the L3NM service.

```
<config xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
    <vpn-profiles>
      <valid-provider-identifiers>
        <routing-profile-identifier>
          <id>EXPORT_POLICY</id>
        </routing-profile-identifier>
        <routing-profile-identifier>
          <id>IMPORT_POLICY</id>
        </routing-profile-identifier>
      </valid-provider-identifiers>
    </vpn-profiles>
  </l3vpn-ntw>
  <l3vpn-route-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
tsdn-flat-L3vpn">
    <name>EXPORT_POLICY</name>
    <color>
      <id>30</id>
      <ipv4>
        <prefix>120.120.120.1/24</prefix> *** Prefix list to be
permitted ***
```

```
        <prefix>121.121.121.1/24</prefix>
      </ipv4>
    </color>
  </l3vpn-route-policy>
  <l3vpn-route-policy xmlns="http://cisco.com/ns/nso/fp/examples/cisco-
tsdn-flat-L3vpn">
    <name>IMPORT_POLICY</name>
    <color>
      <id>30</id>
      <ipv4>
        <prefix>120.120.120.1/24</prefix>
        <prefix>121.121.121.1/24</prefix>
      </ipv4>
    </color>
  </l3vpn-route-policy>
</config>
```

7. Create L3VPN-NTW for color 30 and associate it with the L3VPN route policy.

```
<l3vpn-ntw xmlns="urn:ietf:params:xml:ns:yang:ietf-l3vpn-ntw">
  <vpn-services>
    <vpn-service>
      <vpn-id>Pepsi</vpn-id>
      <ie-profiles>
        <ie-profile>
          <ie-profile-id>ie_00</ie-profile-id>
          <rd>0:111:111</rd>
          <vpn-targets>
            <vpn-target>
              <id>1</id>
              <route-targets>
                <route-target>0:111:111</route-target>
              </route-targets>
              <route-target-type>both</route-target-type>
            </vpn-target>
            <vpn-policies>
              <export-policy>EXPORT_POLICY</export-policy>
            </vpn-policies>
          </vpn-targets>
        </ie-profile>
      </ie-profiles>
      <vpn-nodes>
        <vpn-node>
          <ne-id>XE-PE1</ne-id>
          <local-autonomous-system>1</local-autonomous-system>
          <vpn-targets>
```

```
            <vpn-policies>
              <import-policy>IMPORT_POLICY</import-policy>
            </vpn-policies>
          </vpn-targets>
          <vpn-network-accesses>
            <vpn-network-access>
              <id>23</id>
              <port-id>GigabitEthernet4</port-id>
              <ip-connection>
                <ipv4>
                  <address-allocation-type xmlns:l3vpn-
svc="urn:ietf:params:xml:ns:yang:ietf-l3vpn-svc">l3vpn-svc:static-
address</address-allocation-type>
                  <static-addresses>
                    <primary-address>XE-PE1-ipv4-address</primary-
address>
                    <address>
                      <address-id>XE-PE1-ipv4-address</address-id>
                      <provider-address>121.121.121.1</provider-address>
                      <prefix-length>24</prefix-length>
                    </address>
                  </static-addresses>
                </ipv4>
              </ip-connection>
              <routing-protocols>
                <routing-protocol>
                  <id>TEST_PROTO</id>
                  <type xmlns:l3vpn-
svc="urn:ietf:params:xml:ns:yang:ietf-l3vpn-svc">l3vpn-svc:bgp</type>
                  <bgp>
                    <peer-autonomous-system>65003</peer-autonomous-
system>
                    <local-autonomous-system>1</local-autonomous-system>
                    <address-family>ipv4</address-family>
                    <neighbor>121.121.121.2</neighbor>
                    <multihop>11</multihop>
                  </bgp>
                </routing-protocol>
              </routing-protocols>
            </vpn-network-access>
          </vpn-network-accesses>
          <node-ie-profile>ie_00</node-ie-profile>
        </vpn-node>
        <vpn-node>
          <ne-id>XE-PE2</ne-id>
```

```
          <local-autonomous-system>1</local-autonomous-system>
          <vpn-targets>
            <vpn-policies>
              <import-policy>IMPORT_POLICY</import-policy>
            </vpn-policies>
          </vpn-targets>
          <vpn-network-accesses>
            <vpn-network-access>
              <id>23</id>
              <port-id>GigabitEthernet4</port-id>
              <ip-connection>
                <ipv4>
                  <address-allocation-type xmlns:l3vpn-
svc="urn:ietf:params:xml:ns:yang:ietf-l3vpn-svc">l3vpn-svc:static-
address</address-allocation-type>
                  <static-addresses>
                    <primary-address>XE-PE2-ipv4-address</primary-
address>
                    <address>
                      <address-id>XE-PE2-ipv4-address</address-id>
                      <provider-address>121.121.121.2</provider-address>
                      <prefix-length>24</prefix-length>
                    </address>
                  </static-addresses>
                </ipv4>
              </ip-connection>
              <routing-protocols>
                <routing-protocol>
                  <id>TEST_PROTO</id>
                  <type xmlns:l3vpn-
svc="urn:ietf:params:xml:ns:yang:ietf-l3vpn-svc">l3vpn-svc:bgp</type>
                  <bgp>
                    <peer-autonomous-system>65003</peer-autonomous-
system>
                    <local-autonomous-system>1</local-autonomous-system>
                    <address-family>ipv4</address-family>
                    <neighbor>121.121.121.1</neighbor>
                    <multihop>11</multihop>
                  </bgp>
                </routing-protocol>
              </routing-protocols>
            </vpn-network-access>
          </vpn-network-accesses>
          <node-ie-profile>ie_00</node-ie-profile>
        </vpn-node>
```

```
        </vpn-nodes>
      </vpn-service>
    </vpn-services>
  </l3vpn-ntw>
```

8. Perform a dry-run to verify the L3NM (L3VPN-NTW) configuration.

```
admin@ncs% commit dry-run
  cli {
      local-node {
          data  devices {
                    device XE-PE1 {
                        config {
                            vrf {
                +               definition Pepsi {
                +                   rd 111:111;
                +                   address-family {
                +                       ipv4 {
                +                           route-target {
                +                               export 111:111;
                +                               import 111:111;
                +                           }
                +                       }
                +                   }
                +               }
                            }
                            ip {
                                prefix-list {
                +                   prefixes SET_COLORv4_EXPORT_POLICY_30
  {
                +                       seq 5 {
                +                           permit {
                +                               ip 120.120.120.1/24;
                +                           }
                +                       }
                +                       seq 10 {
                +                           permit {
                +                               ip 121.121.121.1/24;
                +                           }
                +                       }
                +                   }
                +                   prefixes SET_COLORv4_IMPORT_POLICY_30
  {
                +                       seq 500 {
                +                           permit {
                +                               ip 120.120.120.1/24;
```

```
+                                }
+                            }
+                            seq 505 {
+                                permit {
+                                    ip 121.121.121.1/24;
+                                }
+                            }
+                        }
                    }
                }
                interface {
                    GigabitEthernet 4 {
                        vrf {
+                           forwarding Pepsi;
                        }
                        ip {
                            no-address {
-                               address false;
                            }
                            address {
                                primary {
+                                   address 121.121.121.1;
+                                   mask 255.255.255.0;
                                }
                            }
                        }
                    }
                }
+               route-map SET_COLORv4_EXPORT_POLICY 5 {
+                   operation permit;
+                   match {
+                       ip {
+                           address {
+                               prefix-list
SET_COLORv4_EXPORT_POLICY_30;
+                           }
+                       }
+                   }
+                   set {
+                       extcommunity {
+                           color 30;
+                       }
+                   }
+               }
```

```
+            route-map SET_COLORv4_IMPORT_POLICY 500 {
+                operation permit;
+                match {
+                    ip {
+                        address {
+                            prefix-list
SET_COLORv4_IMPORT_POLICY_30;
+                        }
+                    }
+                }
+                set {
+                    extcommunity {
+                        color 30;
+                    }
+                }
+            }
             mpls {
                 label {
                     mode {
+                        vrf Pepsi {
+                            protocol {
+                                all-afs per-vrf;
+                            }
+                        }
                     }
                 }
             }
             router {
                 bgp 1 {
+                    neighbor 121.121.121.2 {
+                        remote-as 65003;
+                    }
                     address-family {
                         with-vrf {
+                            ipv4 unicast {
+                                vrf Pepsi {
+                                    neighbor 121.121.121.2
{
+                                        remote-as 65003;
+                                        activate;
+                                        ebgp-multihop {
+                                            max-hop 11;
+                                        }
+                                    }
+                                }
```

```
+                                   }
                                }
                                vpnv4 unicast {
+                                   neighbor 121.121.121.2 {
+                                       activate;
+                                       route-map in {
+                                           route-map-name
SET_COLORv4_IMPORT_POLICY;
+                                       }
+                                       route-map out {
+                                           route-map-name
SET_COLORv4_EXPORT_POLICY;
+                                       }
+                                   }
                                }
                            }
                        }
                    }
                }
                device XE-PE2 {
                    config {
                        vrf {
+                           definition Pepsi {
+                               rd 111:111;
+                               address-family {
+                                   ipv4 {
+                                       route-target {
+                                           export 111:111;
+                                           import 111:111;
+                                       }
+                                   }
+                               }
+                           }
                        }
                        ip {
                            prefix-list {
+                               prefixes SET_COLORv4_EXPORT_POLICY_30
{
+                                   seq 5 {
+                                       permit {
+                                           ip 120.120.120.1/24;
+                                       }
+                                   }
+                                   seq 10 {
```

```
+                                   permit {
+                                       ip 121.121.121.1/24;
+                                   }
+                               }
+                           }
+                           prefixes SET_COLORv4_IMPORT_POLICY_30
{
+                               seq 500 {
+                                   permit {
+                                       ip 120.120.120.1/24;
+                                   }
+                               }
+                               seq 505 {
+                                   permit {
+                                       ip 121.121.121.1/24;
+                                   }
+                               }
+                           }
                        }
                    }
                    interface {
                        GigabitEthernet 4 {
                            vrf {
+                               forwarding Pepsi;
                            }
                            ip {
                                no-address {
-                                   address false;
                                }
                                address {
                                    primary {
+                                       address 121.121.121.2;
+                                       mask 255.255.255.0;
                                    }
                                }
                            }
                        }
                    }
+               route-map SET_COLORv4_EXPORT_POLICY 5 {
+                   operation permit;
+                   match {
+                       ip {
+                           address {
+                               prefix-list
SET_COLORv4_EXPORT_POLICY_30;
```

```
+                        }
+                    }
+                }
+                set {
+                    extcommunity {
+                        color 30;
+                    }
+                }
+            }
+            route-map SET_COLORv4_IMPORT_POLICY 500 {
+                operation permit;
+                match {
+                    ip {
+                        address {
+                            prefix-list
SET_COLORv4_IMPORT_POLICY_30;
+                        }
+                    }
+                }
+                set {
+                    extcommunity {
+                        color 30;
+                    }
+                }
+            }
             mpls {
                 label {
                     mode {
+                        vrf Pepsi {
+                            protocol {
+                                all-afs per-vrf;
+                            }
+                        }
                     }
                 }
             }
             router {
                 bgp 1 {
+                    neighbor 121.121.121.1 {
+                        remote-as 65003;
+                    }
                     address-family {
                         with-vrf {
+                            ipv4 unicast {
```

```
                +                                   vrf Pepsi {
                +                                       neighbor 121.121.121.1
{
                +                                           remote-as 65003;
                +                                           activate;
                +                                           ebgp-multihop {
                +                                               max-hop 11;
                +                                           }
                +                                       }
                +                                   }
                                                }
                                                vpnv4 unicast {
                +                                   neighbor 121.121.121.1 {
                +                                       activate;
                +                                       route-map in {
                +                                           route-map-name
SET_COLORv4_IMPORT_POLICY;
                +                                       }
                +                                       route-map out {
                +                                           route-map-name
SET_COLORv4_EXPORT_POLICY;
                +                                       }
                +                                   }
                                                }
                                            }
                                        }
                                    }
                                }
                            }
                         }
                        +cisco-flat-L3vpn-fp:flat-L3vpn L3NM-Pepsi-internal {
                        +    endpoint XE-PE1_23 {
                        +        access-pe XE-PE1;
                        +        if-type GigabitEthernet;
                        +        if-id 4;
                        +        pe-ip-addr 121.121.121.1/24;
                        +        as-no 1;
                        +        ce-pe-prot {
                        +            e-bgp {
                        +                neighbor-ipv4 121.121.121.2;
                        +                remote-as-ipv4 65003;
                        +                ebgp-multihop {
                        +                    ttl-value 11;
                        +                    mpls-deactivation false;
```

```
+                }
+            }
+        }
+        vrf {
+            vrf-definition Pepsi;
+            route-distinguisher 111:111;
+            address-family ipv4 {
+                vpn-target 111:111 {
+                    rt-type both;
+                }
+            }
+        }
+        sr-te {
+            export-route-policy EXPORT_POLICY;
+            import-route-policy IMPORT_POLICY;
+        }
+    }
+    endpoint XE-PE2_23 {
+        access-pe XE-PE2;
+        if-type GigabitEthernet;
+        if-id 4;
+        pe-ip-addr 121.121.121.2/24;
+        as-no 1;
+        ce-pe-prot {
+            e-bgp {
+                neighbor-ipv4 121.121.121.1;
+                remote-as-ipv4 65003;
+                ebgp-multihop {
+                    ttl-value 11;
+                    mpls-deactivation false;
+                }
+            }
+        }
+        vrf {
+            vrf-definition Pepsi;
+            route-distinguisher 111:111;
+            address-family ipv4 {
+                vpn-target 111:111 {
+                    rt-type both;
+                }
+            }
+        }
+        sr-te {
+            export-route-policy EXPORT_POLICY;
```

```
+            import-route-policy IMPORT_POLICY;
+        }
+    }
+}
 l3vpn-ntw {
     vpn-services {
+        vpn-service Pepsi {
+            ie-profiles {
+                ie-profile ie_00 {
+                    rd 0:111:111;
+                    vpn-targets {
+                        vpn-target 1 {
+                            route-targets 0:111:111;
+                            route-target-type both;
+                        }
+                        vpn-policies {
+                            export-policy EXPORT_POLICY;
+                        }
+                    }
+                }
+            }
+            vpn-nodes {
+                vpn-node XE-PE1 {
+                    local-autonomous-system 1;
+                    vpn-targets {
+                        vpn-policies {
+                            import-policy IMPORT_POLICY;
+                        }
+                    }
+                    vpn-network-accesses {
+                        vpn-network-access 23 {
+                            port-id GigabitEthernet4;
+                            ip-connection {
+                                ipv4 {
+                                    address-allocation-
type static-address;
+                                    static-addresses {
+                                        primary-address
XE-PE1-ipv4-address;
+                                        address XE-PE1-
ipv4-address {
+                                            provider-
address 121.121.121.1;
+                                            prefix-length
24;
```

```
            +                                        }
            +                                    }
            +                                }
            +                            }
            +                            routing-protocols {
            +                                routing-protocol
TEST_PROTO {
            +                                    type bgp;
            +                                    bgp {
            +                                        peer-autonomous-
system 65003;
            +                                        local-autonomous-
system 1;
            +                                        address-family [
ipv4 ];
            +                                        neighbor [
121.121.121.2 ];
            +                                        multihop 11;
            +                                    }
            +                                }
            +                            }
            +                        }
            +                    }
            +                    node-ie-profile ie_00;
            +                }
            +                vpn-node XE-PE2 {
            +                    local-autonomous-system 1;
            +                    vpn-targets {
            +                        vpn-policies {
            +                            import-policy IMPORT_POLICY;
            +                        }
            +                    }
            +                    vpn-network-accesses {
            +                        vpn-network-access 23 {
            +                            port-id GigabitEthernet4;
            +                            ip-connection {
            +                                ipv4 {
            +                                    address-allocation-
type static-address;
            +                                    static-addresses {
            +                                        primary-address
XE-PE2-ipv4-address;
            +                                        address XE-PE2-
ipv4-address {
            +                                            provider-
address 121.121.121.2;
```

```
                +                                                   prefix-length
24;
                +                                                   }
                +                                               }
                +                                           }
                +                                       }
                +                                       routing-protocols {
                +                                           routing-protocol
TEST_PROTO {
                +                                               type bgp;
                +                                               bgp {
                +                                                   peer-autonomous-
system 65003;
                +                                                   local-autonomous-
system 1;
                +                                                   address-family [
ipv4 ];
                +                                                   neighbor [
121.121.121.1 ];
                +                                                   multihop 11;
                +                                               }
                +                                           }
                +                                       }
                +                                   }
                +                               }
                +                               node-ie-profile ie_00;
                +                           }
                +                       }
                +                   }
                              }
                            }
                        }
                }
```

9. Commit the configuration to deploy the L3NM service.

```
admin@ncs% commit commit-queue async
Commit complete.
```

10. Verify the plan status for the L3NM service. The plan displays **reached** status upon successful deployment.

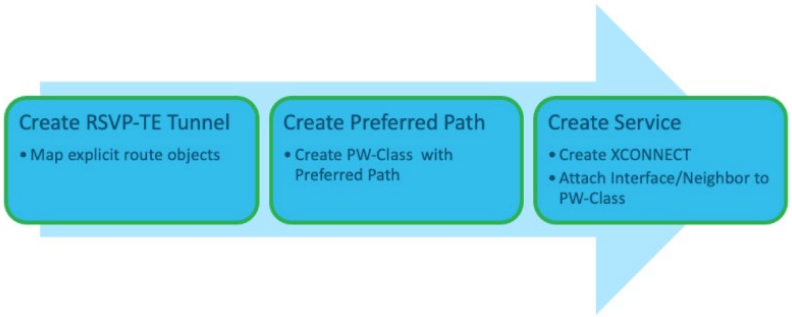
```
admin@ncs% run show l3vpn-ntw vpn-services vpn-service-plan Pepsi plan

                                                BACK        STATUS
TYPE                                NAME        TRACK  GOAL CODE   STATE                                    STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------------------
self                                self        false  -    -      init                                     reached  2021-02-10T04:16:41
                                                                   ietf-l3vpn-ntw-nano-services:config-apply reached  2021-02-10T04:16:41
                                                                   ready                                    reached  2021-02-10T04:16:49
ietf-l3vpn-ntw-nano-services:vpn-node  XE-PE1_23  false  -    -      init                                     reached  2021-02-10T04:16:41
                                                                   ietf-l3vpn-ntw-nano-services:config-apply reached  2021-02-10T04:16:41
                                                                   ready                                    reached  2021-02-10T04:16:49
ietf-l3vpn-ntw-nano-services:vpn-node  XE-PE2_23  false  -    -      init                                     reached  2021-02-10T04:16:41
                                                                   ietf-l3vpn-ntw-nano-services:config-apply reached  2021-02-10T04:16:41
                                                                   ready                                    reached  2021-02-10T04:16:49
```

Use Case 4: Instantiating RSVP-TE Tunnel – IOSXE

The RSVP-TE configuration allows you to set up a bidirectional tunnel by configuring both source and destination devices.

This use case discusses how to instantiate the RSVP-TE tunnel on an IOSXE device. The following diagram shows the required tasks for this use case.



The detailed workflow includes:

1. Add a PCC peer to a PE device.
2. Create and verify the RSVP-TE tunnel.
3. Perform an end-to-end self-test on the RSVP-TE tunnel.

To instantiate RSVP-TE tunnel on an IOSXE device:

1. Configure the Path Computation Client (PCC) peer network.

```
segment-routing mpls
 !
 set-attributes
  address-family ipv4
   sr-label-preferred
  exit-address-family
 !
 global-block 16000 23999
 local-block 15000 15999
 !

segment-routing traffic-eng
 pcc
  pce address 100.100.100.2
  report-all
```

2. Verify the PCC peer is up.

```
XE-PE1#show pce client lsp
PCC's tunnel database:
----------------------
Tunnel Name: cfg_color_30_ep_100.100.100.8_discr_10 client POLICY_MGR
 LSPs:
  LSP[0]:
   source 100.100.100.7, destination 100.100.100.8, tunnel ID 1,LSP ID 0
   State: Admin up, Operation active
   Setup type: SR
   Binding SID: 1000


XE-PE1#show pce client  peer
PCC's peer database:
--------------------

Peer address: 100.100.100.2, Precedence: 255
Client POLICY_MGR
  State up
  Capabilities: Stateful, Update, Segment-Routing, Instantiation
```

3. Configure the RSVP-TE tunnel for explicit tunnel hop.

```
<config xmlns="http://tail-f.com/ns/config/1.0">
  <te xmlns="urn:ietf:params:xml:ns:yang:ietf-te">
    <tunnels>
      <tunnel>
        <name>DYNAMIC-TUNNEL-4</name>
        <identifier>5555</identifier>
        <source>100.100.100.7</source>
        <destination>100.100.100.8</destination>
        <signaling-type>te-types:path-setup-rsvp</signaling-type>
        <p2p-primary-paths>
          <p2p-primary-path>
            <name>Path-1</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>te-types:path-metric-igp</metric-type>
                <weight>1</weight>
              </optimization-metric>
            </optimizations>
          </p2p-primary-path>
```

```
            </p2p-primary-paths>
            <head-end>XE-PE1</head-end>
          </tunnel>
        </tunnels>
      </te>
    </config>
```

4. Perform a dry-run to verify the RSVP-TE tunnel configuration.

```
admin@ncs% commit dry-run
  cli {
      local-node {
          data  devices {
                    device XE-PE1 {
                        config {
                            interface {
                +               Tunnel 5555 {
                +                   ip {
                +                       unnumbered {
                +                           Loopback 0;
                +                       }
                +                   }
                +                   tunnel {
                +                       destination 100.100.100.8;
                +                       mode {
                +                           mpls {
                +                               traffic-eng {
                +                               }
                +                           }
                +                       }
                +                       mpls {
                +                           traffic-eng {
                +                               name DYNAMIC-TUNNEL-4;
                +                               autoroute {
                +                                   announce;
                +                               }
                +                               priority {
                +                                   setup-priority 7;
                +                                   hold-priority 7;
                +                               }
                +                               path-option 1 {
                +                                   dynamic;
                +                                   pce;
                +                               }
                +                               path-selection {
```

```
                +                                   metric igp;
                +                                }
                +                             }
                +                          }
                +                       }
                +                    }
                             }
                          }
                       }
                 }
                 te {
                     tunnels {
                +        tunnel DYNAMIC-TUNNEL-4 {
                +            identifier 5555;
                +            source 100.100.100.7;
                +            destination 100.100.100.8;
                +            signaling-type te-types:path-setup-rsvp;
                +            p2p-primary-paths {
                +                p2p-primary-path Path-1 {
                +                    path-computation-method path-
externally-queried;
                +                    optimizations {
                +                        optimization-metric te-types:path-
metric-igp {
                +                            weight 1;
                +                        }
                +                    }
                +                }
                +            }
                +            head-end XE-PE1;
                +        }
                     }
                 }
     }
}
```

5. Commit the configuration to deploy the RSVP-TE tunnel service.

```
admin@ncs% commit commit-queue async
Commit complete.
```

6. Verify the plan status for the tunnel service. The plan displays **reached** state for successful deployment.

```
admin@ncs% run show te tunnels tunnel-plan DYNAMIC-TUNNEL-4 plan

                      BACK         STATUS
TYPE    NAME          TRACK  GOAL  CODE    STATE                                                  STATUS   WHEN
-----------------------------------------------------------------------------------------------------------------------
self    self          false  -     -       init                                                   reached  2021-02-10T04:56:28
                                           ready                                                  reached  2021-02-10T04:56:33
source  100.100.100.7 false  -     -       init                                                   reached  2021-02-10T04:56:28
                                           ietf-te-fp-tunnel-nano-plan-services:config-apply      reached  2021-02-10T04:56:28
                                           ready                                                  reached  2021-02-10T04:56:33
```

7. Perform an end-to-end self-test on the tunnel service to ensure the TE tunnel is set up on the XE devices.

```
admin@ncs% run request te tunnels tunnel Dynamic-tunnel-4 action self-test

status success
```

Appendix F: Error Recovery

Errors are classified as:

- Transient errors or failures
- Persistent errors or failures

Transient failures are self-correcting failures. In such failures, the device may come back up 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 performed on NSO to bring the device 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 have to 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-tsdn-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. If the device comes back up within the configured time, the recovery flow is executed.
auto-cleanup	If set to true, deleting a service from a device that is offline 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 result to recover 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 that are being currently polled by the CFP.
poll-wait-time	This is the wait time in seconds between each polling request toward the device. The default wait time is set to 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 which is multiplied with 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.

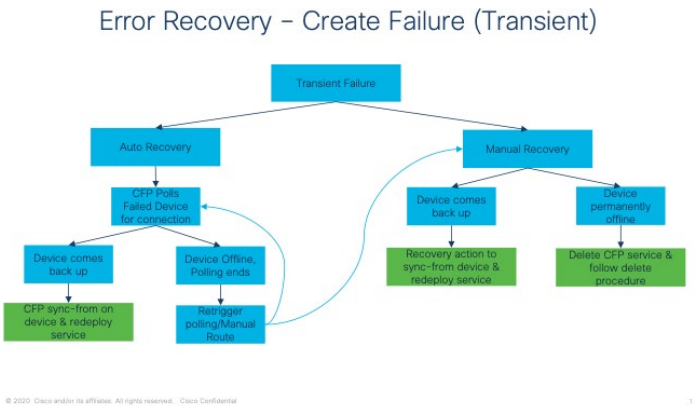
Recovery Flow

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 flow of recovery from a create failure when the error is transient.

The service plan notification shows the device failure. You can choose to perform either the auto-recovery process or the manual recovery process, as required.



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, 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 fails to come back after the polling period, you can either retrigger polling and wait for the device to come back up or manually recover the device once it comes back up.

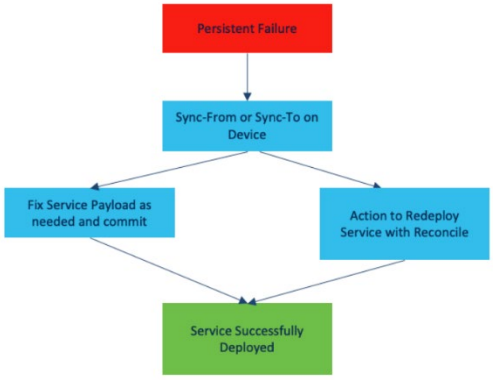
Manual recovery

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

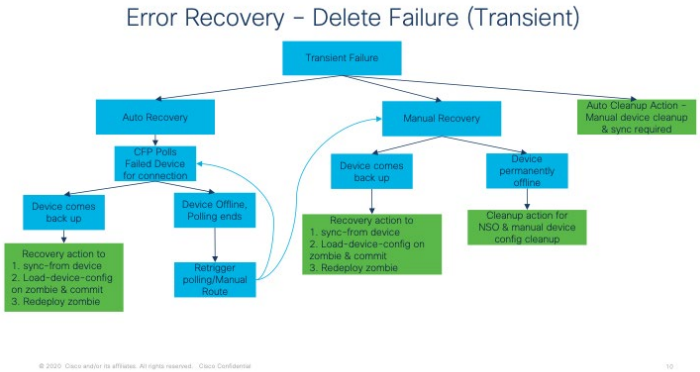
The following diagram shows the flow of recovery of a device from a create failure when the error is persistent.



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

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 fails to come back up after the polling period, you can either retrigger polling and wait for the device to come back up or manually recover the device once it comes back up.

Manual recovery

During the manual recovery process, if the device comes back up, the recovery action performs:

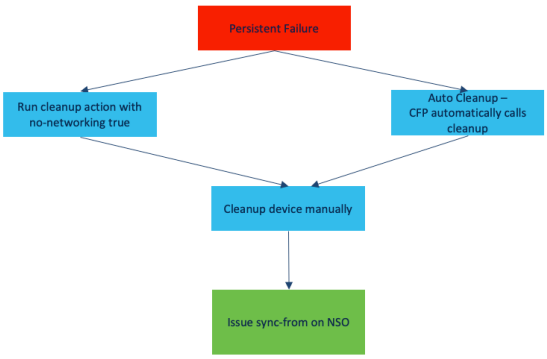
- a sync-from device
- tries to load the zombie (delete) configuration and commits it and
- redeploys the zombie

Redeploying the zombie again pushes a delete on the device configuration.

Use the Cleanup action if the device is permanently offline and you must cleanup 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

In this section, the possible error scenarios are created 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 has failed on the device.

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 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:

4. 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-autonomous-system
l3vpn-ntw vpn-services vpn-service 0-65008740
 vpn-nodes vpn-node CE-1
  local-autonomous-system 100
 !
 vpn-nodes vpn-node PE-5
  local-autonomous-system 100
 !
!
```

5. 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
     }
 }
```

6. Commit the configuration in commit-queue.

```
admin@ncs% commit commit-queue async
Commit complete.
```

7. Display the plan for service 1. The plan displays as failed with an error message.

```
admin@ncs# show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 plan
Possible completions:
  plan  plan-history

admin@ncs# show l3vpn-ntw vpn-services vpn-service-plan 0-65008740 plan
                                                                                                                                POST
                                                BACK                                                                                                    ACTION
TYPE                                NAME     TRACK  GOAL  STATUS CODE    STATE                                    STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------
self                                self     false  -     -             init                                     reached  2021-03-25T21:56:41  -    -
                                                                        ietf-l3vpn-ntw-nano-services:config-apply reached  2021-03-25T21:56:41  -    -
                                                                        ready                                    failed   2021-03-25T21:56:55  -    -
ietf-l3vpn-ntw-nano-services:vpn-node  CE-1_23  false  -     TSDN-L3VPN-303 init                                     reached  2021-03-25T21:56:41  -    -
                                                                        ietf-l3vpn-ntw-nano-services:config-apply reached  2021-03-25T21:56:41  -    -
                                                                        ready                                    failed   2021-03-25T21:56:55  -    -
ietf-l3vpn-ntw-nano-services:vpn-node  PE-5_23  false  -     TSDN-L3VPN-303 init                                     reached  2021-03-25T21:56:41  -    -
                                                                        ietf-l3vpn-ntw-nano-services:config-apply reached  2021-03-25T21:56:41  -    -
                                                                        ready                                        failed   2021-03-25T21:56:55  -   -

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 status-code-detail endpoint CE-1_23
 code                TSDN-L3VPN-303
 context "Config push failed"
  context-msg "External error in the NED implementation for device CE-1:
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\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."

plan status-code-detail endpoint PE-5_23
 code                TSDN-L3VPN-303
 context "Config push failed"
  context-msg "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"
 severity            ERROR
 recommended-action "Device configuration rejected, fix the service
payload and perform recovery steps."
```

8. 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 65100:87400024;
-                            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;
-                                            }
-                                        }
-                                    }
-                                }
-                            }
-                        }
-                    }
                  }
              }
          }
      }
  }
```

9. Deploy the SR-TE policy service on the PE-5 device.
10. 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
                   !
                  !
                 !
                !
               !
              !
             !
    }
}
```

11. Commit the SR-TE policy service configuration in commit-queue.

```
admin@ncs% commit commit-queue async
Commit complete.
[ok]
```

12. 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
```

```
                                                                                                                                                                                 POST
                                                             BACK                                                                                                               ACTION
TYPE                                                    NAME TRACK GOAL STATUS CODE STATE                                                       STATUS  WHEN                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 PE-5 device or CE-1 device fail.

Solution

This section discusses the steps to recover from the error condition.

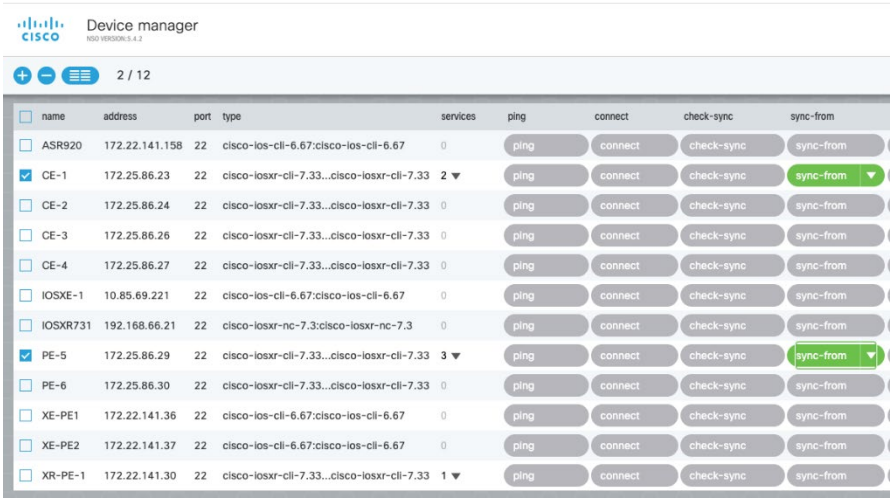
Note: The recovery steps vary based on the errors or scenarios. Be sure to be aware of the actions performed on the service and device.

The following is the workflow to recover from the error condition:

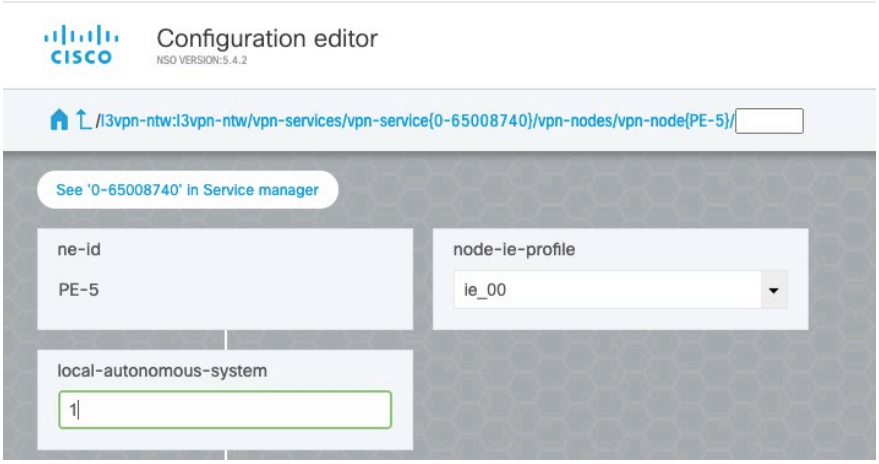
1. Perform a sync-from on devices PE-5 and CE-1.
2. Correct the AS number in the payload for service 1 and redeploy the service.
3. Re-deploy service 2.

The detailed procedure to recover from the error condition is as follows:

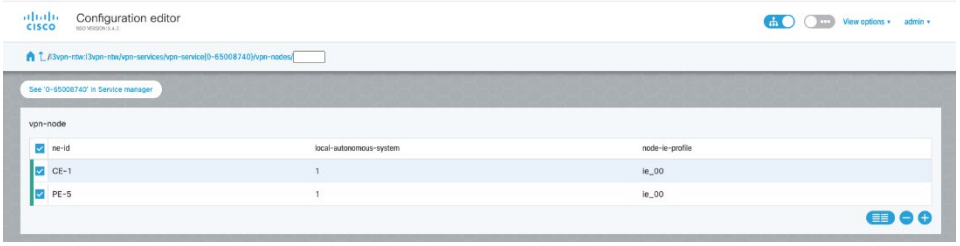
1. Log in to NSO UI with your login credentials.
2. Navigate to the **Device Manager** window and perform a sync-from on the PE-5 and CE-1 devices.



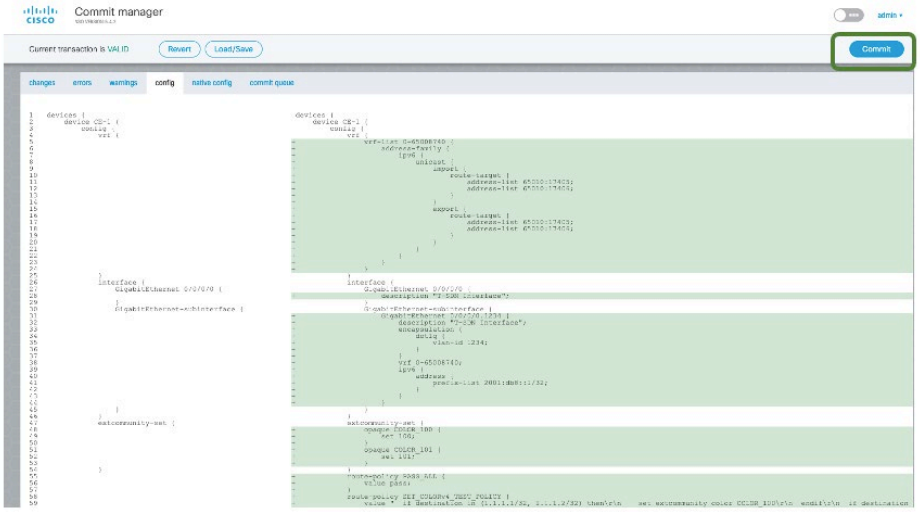
3. Navigate to **Configuration editor** window. Update the value of the **local-autonomous-system** field for both the PE-5 device and CE-1 device with the correct AS number 1.



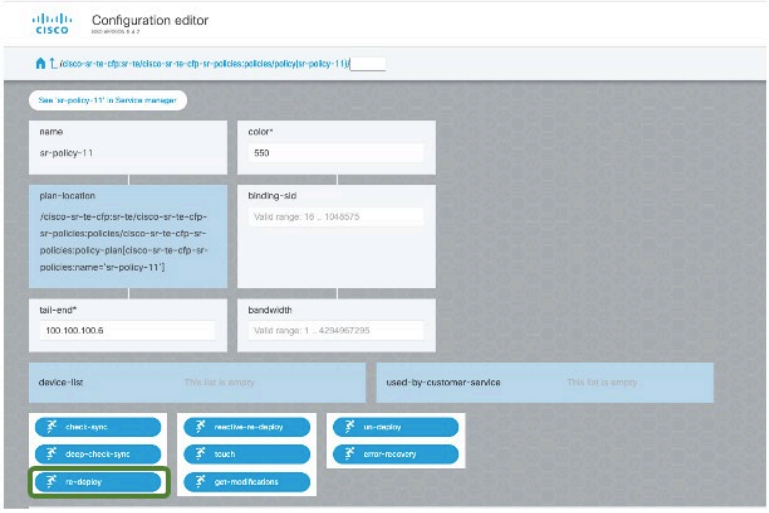
4. Verify the AS number for both the devices is updated.



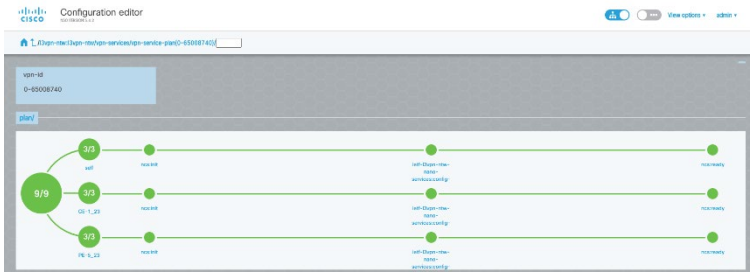
5. In the **Commit Manager** window, commit the configuration to deploy the service.



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 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 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
```

```
                                                        BACK        STATUS                                                                                                  POST  ACTION
TYPE                                                    NAME        TRACK  GOAL  CODE    STATE                                                                  STATUS   WHEN                ref   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
```

```
                                                                                                                                                POST
                                                                                                                                                     ACTION
                                                                BACK
TYPE                                                    NAME    TRACK  GOAL  STATUS CODE  STATE                                          STATUS   WHEN                ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                    self    false  -     -            init                                           reached  2021-03-26T19:52:41  -    -
                                                                                         ready                                          failed   2021-03-26T20:01:13  -    -
```

```
cisco-sr-te-cfp-sr-policies-nano-plan-services:head-end  XR-PE-1  false  -     TSDN-SR-301  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                                                   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
```

```
                                                                                                                                                POST
                                                                BACK        STATUS                                                                  ACTION
TYPE                                                            NAME    TRACK  GOAL  CODE    STATE                                                   STATUS   WHEN                 ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
self                                                            self    false  -     -       init                                                    reached  2021-03-26T19:52:41  -    -
                                                                                             ready                                                   reached  2021-03-26T20:04:14  -    -
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  2021-03-26T20:04:08  -    -
```

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:

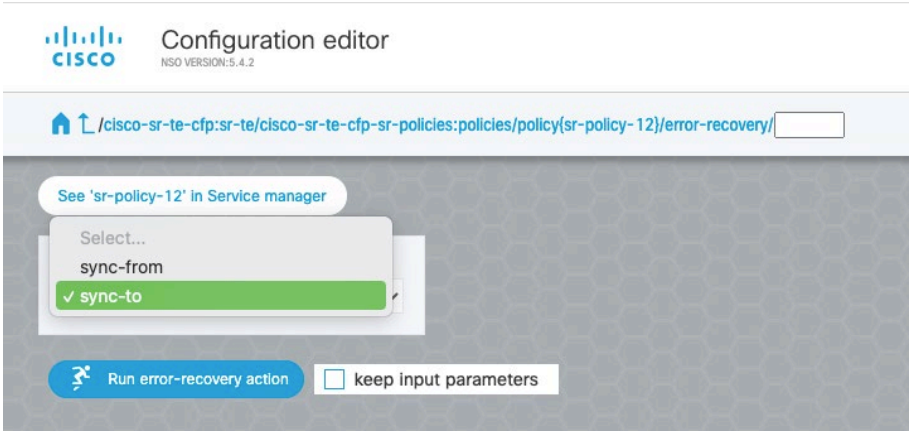
- Use manual recovery
- Change the sync-direction in auto-recovery from sync-from to sync-to.

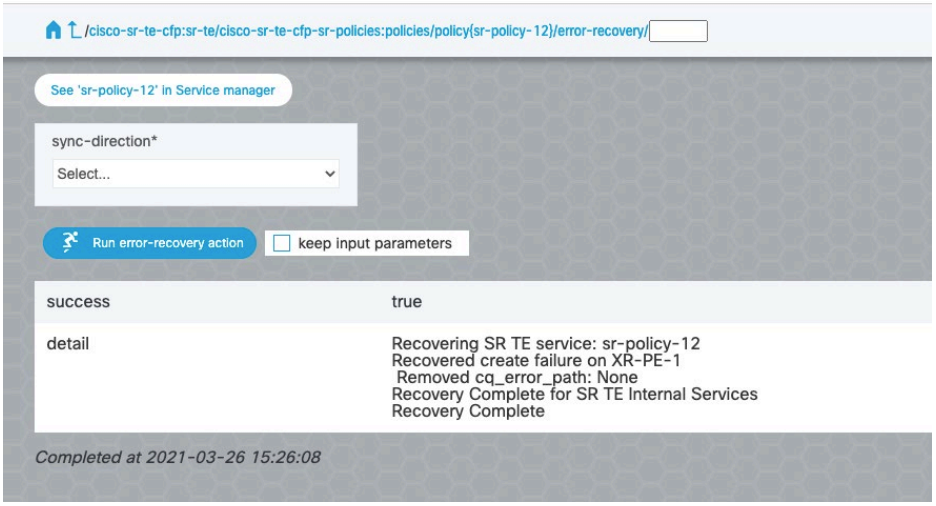
Manual Recovery

1. In the NSO UI, verify the service is marked failed and 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 successful 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.

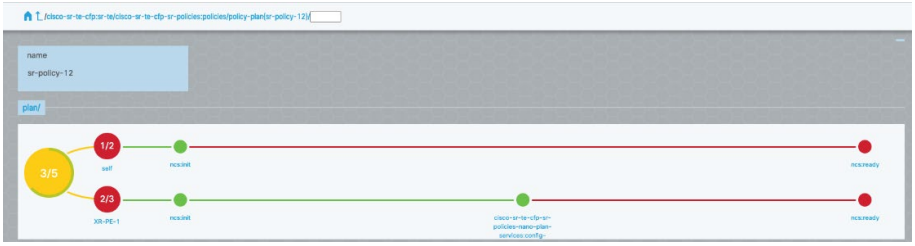
```
** Service Meta Data **
admin@ncs% show devices device XR-PE-1 config segment-routing traffic-eng
policy srte_c_25_ep_100.100.100.44 | display service-meta-data
color {
    /* Refcount: 1 */
    value 25;
    end-point {
        /* Refcount: 1 */
        ipv4 100.100.100.44;
    }
}
/* Refcount: 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;
            }
        }
    }
}
```

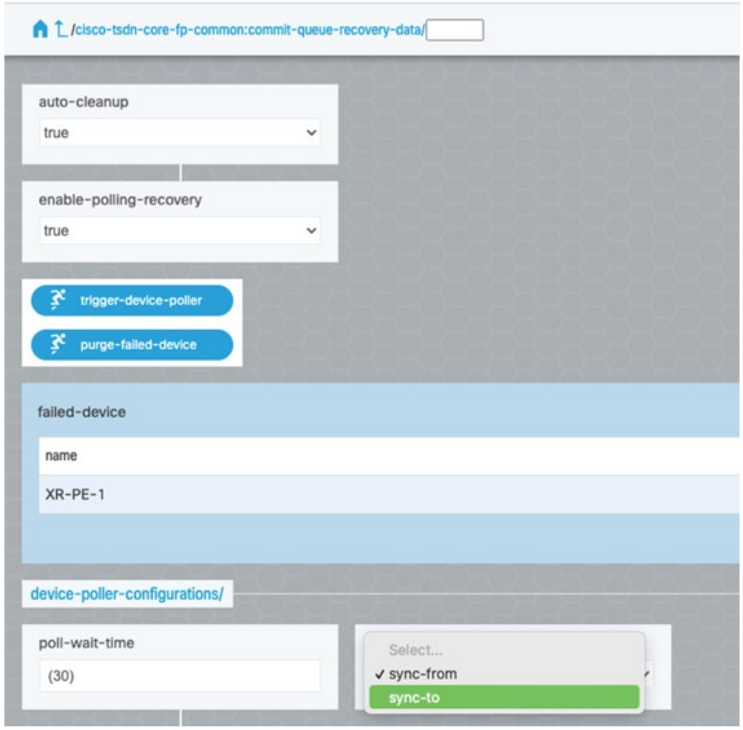
Auto Recovery

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.

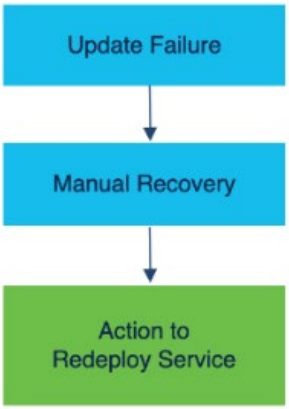
```
** Service Meta Data **
admin@ncs% show devices device XR-PE-1 config segment-routing traffic-eng
policy srte_c_25_ep_100.100.100.44 | display service-meta-data
color {
    /* Refcount: 1 */
    value 25;
    end-point {
        /* Refcount: 1 */
        ipv4 100.100.100.44;
    }
}
/* Refcount: 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;
                }
            }
        }
    }
```

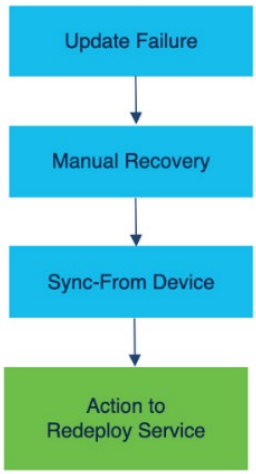
Non Commit-Queue Error Recovery Flow

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

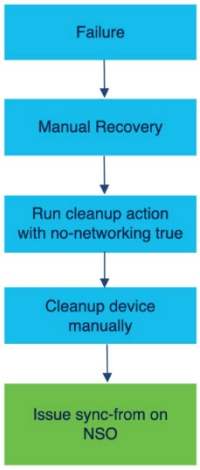


Persistent Update Failure Recovery Flow



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 G - 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 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 ***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 time the slowest device removes the configuration. It is, therefore, recommended to delete the configurations 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.

7. Device is not reachable during deletion.
8. 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** 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 still in 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']
```

```
                                                                                                                POST
                BACK                                                                                            ACTION
TYPE  NAME  TRACK  GOAL  STATE                                                  STATUS       WHEN  ref  STATUS
-----------------------------------------------------------------------------------------------------------------------
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']
```

```
                                                                                                                POST
                                                BACK                                                            ACTION
TYPE                                            NAME      TRACK  GOAL  STATE                                  STATUS       WHEN  ref  STATUS
-----------------------------------------------------------------------------------------------------------------------------------
self                                            self      true   -     init                                    not-reached  -     -    -
                                                                       ready                                   not-reached  -     -    -
cisco-sr-te-cfp-sr-odn-nano-plan-services:head-end  PIOSXR-0  true   -     init                                    reached      -     -    -
                                                                       cisco-sr-te-cfp-sr-odn-nano-plan-services:config-apply  failed       -     -    -
                                                                       ready                                   not-reached  -     -    -
```

```
 plan failed
```

This page is intentionally left blank.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

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