

Cisco NSO T-SDN 機能パックバンドル インストールガイド

バージョン 4.1.0

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報と推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザー側の責任となります。

対象製品のソフトウェア ライセンスと限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

シスコが導入する TCP ヘッダー圧縮は、カリフォルニア大学バークレー校 (UCB) により、UNIX オペレーティング システムの UCB パブリック ドメイン バージョンの一部として開発されたプログラムを適応したものです。All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルとソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスと電話番号は、実際のアドレスと電話番号を示すものではありません。マニュアル内の例、コマンド表示出力、ネットワーク ポジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

シスコは世界各国に 200 カ所を超えるオフィスを開設しています。各オフィスの住所、電話番号、FAX 番号については当社の Web サイト (www.cisco.com/go/offices/) をご覧ください。

Cisco および Cisco ロゴは、シスコまたはその関連会社の米国およびその他の国における商標または登録商標です。シスコの商標の一覧は、www.cisco.com/go/trademarks でご確認ください。記載されている第三者機関の商標は、それぞれの所有者に帰属します。「パートナー」という用語の使用はシスコと他社との間のパートナーシップ関係を意味するものではありません。(1721R)

Copyright

© 2022 Cisco Systems, Inc. All rights reserved.

目次

はじめに.....	6
バイアスのないドキュメントポリシー	7
インストール方法	8
インストールの概要	8
インストール要件	9
システム要件.....	9
Cisco NSO と Cisco NED の要件	9
Cisco T-SDN FP バンドルをインストールするための NSO 環境の準備	11
単一の NSO インスタンスでの Cisco NSO T-SDN 機能パックバンドルのインストール とアンインストール	14
単一の NSO インスタンス上のパッケージカテゴリとパッケージ	14
単一の NSO インスタンスでの NCS 構成ファイルの編集	16
単一の NSO インスタンスへのコア機能パックのインストール	18
SR-TE CFP-IOSXR CLI のインストール.....	18
SR-TE CFP-IOSXR CLI のインストール後のタスクの実行.....	20
SR-TE CFP-IOSXR CLI のインストール後のタスクの確認.....	22
SR-TE CFP-IOSXR NC のインストール.....	25
SR-TE CFP-IOSXR NC インストールのインストール後のタスクの実行.....	26
SR-TE CFP-IOSXR NC のインストール後のタスクの確認.....	26
SR-TE CFP-IOSXE CLI のインストール	26
SR-TE CFP-IOSXE CLI のインストール後のタスクの実行.....	27
SR-TE CFP-IOSXE CLI のインストール後のタスクの確認.....	28
単一の NSO インスタンスへのサンプル機能パックのインストール	29
自動アシュアランスサービスのインストール	30
L2VPN/L2NM サービスのインストール	30
L2VPN-IOSXR CLI のインストール	30
L2NM-IOSXR CLI のインストール.....	35

L2VPN-IOSXR NC のインストール	38
L2NM-IOSXR NC のインストール.....	40
L2VPN-IOSXE CLI のインストール	43
L2NM-IOSXE CLI のインストール.....	44
L3VPN/L3NM サービスのインストール	46
L3VPN-IOSXR CLI のインストール	46
L3NM-IOSXR CLI のインストール.....	50
L3VPN-IOSXR NC のインストール	53
L3NM-IOSXR NC のインストール.....	55
L3VPN-IOSXE CLI のインストール	57
L3NM-IOSXE CLI のインストール.....	58
IETF-TE サービスのインストール	59
IETF TE-IOSXR CLI のインストール	59
IETF TE-IOSXR NC のインストール	64
IETF-TE-IOSXE CLI のインストール.....	65
単一の NSO インスタンスでの Cisco T-SDN FP バンドルのアンインストール.....	66
単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す	67
サンプル機能パックのアンインストール	68
L2VPN/L2NM サービスのアンインストール.....	68
L3VPN/L3NM サービスのアンインストール.....	76
IETF-TE サービスのアンインストール	82
コア機能パックのアンインストール.....	86
SR-TE CFP-IOSXE CLI のアンインストール.....	86
SR-TE CFP-IOSXR NC のアンインストール.....	86
SR-TE CFP-IOSXR CLI のアンインストール.....	87
LSA モデルでの T-SDN FP バンドルのインストールとアンインストール.....	90
下位ノードへの T-SDN FP バンドルのインストール.....	91
パッケージカテゴリとパッケージ - 下位ノード.....	91
下位ノードでの NCS 構成ファイルの変更.....	92
下位ノードへの T-SDN FP バンドルのインストール	95
下位ノードでのインストール後のタスクの実行.....	98
下位ノードでのインストールの確認.....	99

上位ノードへの T-SDN FP バンドルのインストール.....	102
パッケージカテゴリとパッケージ - 上位ノード.....	102
上位ノードでの NCS 構成ファイルの変更.....	102
上位ノードへの T-SDN FP バンドルのインストール.....	105
上位ノードでのインストール後のタスクの実行.....	108
上位ノードでのインストールの確認.....	111
LSA モデルでの T-SDN FP バンドルのアンインストール.....	112
上位ノードからの T-SDN FP バンドルのアンインストール.....	112
下位ノードからの T-SDN FP バンドルのアンインストール.....	115
アップグレード.....	117
前提条件.....	117
Cisco T-SDN FP バンドルの CFP を v4.1.0 にアップグレードする.....	117
アップグレードの準備.....	117
T-SDN FP バンドルのアップグレード.....	118
T-SDN FP バンドルサービスのアップグレード.....	122
付録 A : Python 起動コマンド構成の変更.....	123
付録 B : commit-queue async フラグを渡す.....	124

はじめに

概要

この **Cisco Network Service Orchestrator Transport-SDN 機能パック バンドル インストール ガイド**には、Cisco NSO Transport SDN 機能パック (T-SDN FP) バンドル v4.1.0 のインストールに役立つ情報が含まれています。

対象読者

このドキュメントは、T-SDN 自動化機能をシスコの顧客にインストールするシスコ アドバンスドサービスのデベロッパー、ネットワークエンジニア、およびシステムエンジニアを対象としています。

その他の資料

このドキュメントでは、読者が NSO と、NSO のドキュメントで説明されているその使用方法について十分に理解している必要があります。

シリアル番号	資料
1.	Cisco Transport SDN Function Pack Bundle User Guide
2.	NSO Installation Guide
3.	NSO User Guide

バイアスのないドキュメントポリシー

シスコは、バイアスのないドキュメントポリシーに従います。このポリシーに従って、シスコは、人種、肌の色、祖先、出身国、年齢、性別、市民権、退役軍人の地位、婚姻状況、性的指向、身体的または精神的な能力、宗教的信条、または病状に関係なく、すべての人を尊重して扱います。他人に不快感を与える言語またはグラフィック要素は、当社のビジネス哲学および当社のポリシーに違反します。

インストール方法

NSO での T-SDN FP バンドルのインストールは、次の 2 つの方法で実行できます。

- システムインストール
- ローカルインストール

システムインストールは、リアルタイムの実稼働環境用であり、推奨されるインストール方法です。

ローカルインストールは、インストールのデモバージョンです。

インストールを実行し、インストールコマンドを実行するには、**sudo** ユーザー権限が必要です。T-SDN FP バンドルのシステムインストールは、1 台のマシンまたは複数のマシンに同時に実行できます。システムインストールは、単一コントローラホストから複数のホスト/VM に NSO をインストールするために使用されます。

このドキュメントでは、T-SDN FP バンドルのシステムインストールを実行する方法について説明します。ローカルインストールについては、シスコの担当者にお問い合わせください。

インストールの概要

NSO T-SDN FP バンドルのシステムインストールでは、SR-TE CFP サービスとサンプル機能パックをインストールできます。

Cisco SR-TE CFP-IOSXR CLI は 製品化され、SR-TE 自動化の実装に対応しており、Cisco SR-ODN および Cisco SR-Policy サービスで構成されています。

レイヤ 2 VPN (L2VPN) 、L2NM、レイヤ 3 VPN (L3VPN) 、L3NM、および IETF - トラフィック エンジニアリング (IETF-TE) のサンプル機能パックは、特定の要件に合わせてカスタマイズすることを目的としています。

この製品の詳細な概要については、『**Cisco NSO T-SDN FP Bundle User Guide**』を参照してください。

Layered Service Architecture (LSA) 導入モデルを使用して、または単一の NSO インスタンスで、NSO T-SDN FP バンドルシステムのインストールを実行します。このドキュメントの続くトピックでは、それぞれのインストール手順について詳しく説明します。

単一の NSO インスタンスに T-SDN FP バンドルをインストールする方法については、「**単一の NSO インスタンスでの Cisco NSO T-SDN 機能パックバンドルのインストールとアンインストール**」を参照してください。

LSA モデルで T-SDN FP バンドルをインストールする方法については、「**LSA モデルでの T-SDN FP バンドルのインストールとアンインストール**」を参照してください。

サンプル機能パック（フレーバー）は、基本フレーバー SR-TE CFP-IOSXR CLI の上にインストールできます。

フレーバーをインストールするには、SR-TE CFP-IOSXR CLI のインストール中またはインストール後に、フレーバーに必要なパッケージをコピーします。さまざまなフレーバーをインストールする方法の詳細については、「[単一の NSO インスタンスへのサンプル機能パック](#)」を参照してください。

この章では、各フレーバーに必要なパッケージカテゴリと、基本フレーバー（SR-TE CFP-IOSXR CLI）をインストールする手順について説明します。

インストール要件

このトピックでは、T-SDN FP バンドルをインストールするために必要なシステム要件、NSO バージョン、および NED バージョンについて説明します。

システム要件

このセクションでは、Cisco NSO T-SDN FP バンドル v4.1.0 を NSO v5.7.6 に正常にインストールするためのハードウェア要件、ソフトウェア要件、およびプラットフォームの依存関係について概説します。

項目	要件
オペレーティングシステム	NSO および T-SDN FP バンドルは、すべての Linux ディストリビューションで利用でき、次のオペレーティングシステムでサポートされています。 Ubuntu v18.04 LTS (バイオニック) CentOS v7.x RedHat Linux v7.x
ソフトウェア	Open JDK v11 以降 Python v3.8 以降
NSO の ulimit 値	64000 (最小)

Cisco NSO と Cisco NED の要件

ソフトウェア/ドライバ	バージョン
Cisco NSO	5.7.6
IOSXR CLI NED	7.40.1

IOSXR Netconf NED (マルチベンダー拡張の例)	7.3.2、7.4.2、7.5.2、 7.6.1、7.6.2、7.7.1
IOSXE CLI NED (マルチベンダー拡張の例)	6.77.9

Cisco T-SDN FP バンドルをインストールするための NSO 環境の準備

NSO 5.7.6 に T-SDN FP バンドルをインストールするには、最初に NSO 環境を準備する必要があります。

NSO 環境を準備するには、次の手順を実行します。

1. NSO 5.7.6 インストール bin ファイルを取得し、システムインストールガイド「**NSO Installation Guide**」に記載されている手順に従って NSO をインストールします。

2. NSO 5.7.6 をインストールして確認します。

```
$ ncs --version  
5.7.6
```

3. Python および Python 3 のバージョンが 3.8 以上であることを確認してください。以下の手順に従って Python のバージョンを確認します。

```
$ python --version  
3.8
```

デフォルトの Python は Python 3 を指す必要があります。デフォルトの Python を Python 3 に変更できない場合は、Python スタートアップ コマンド コンフィギュレーションを変更します。詳細については、「[付録 A : Python 起動コマンド構成の変更](#)」を参照してください。

CentOS を使用している場合は、次の手順を実行して Python 3.8 をインストールします。

注： CentOS が正しく機能するには、Python 2 が必要です。したがって、Python の代替を使用して Python 2 および Python 3.8 を登録してから、使用する Python バージョンを選択する必要があります。

- a. 列挙モジュール Enum 34 がシステムに存在することを確認します。

```
$ pip install enum34
```

モジュールがシステムに存在する場合、要件が満たされていることを示すメッセージが表示されます。モジュールがシステムに存在しない場合、コマンドはモジュールに必要なパッケージをダウンロードしてインストールします。

- b. 代替として Python 2 を登録します。

```
$ alternatives --install /usr/bin/python python /usr/bin/python2 50
```

- c. 代替として Python 3.8 を登録します。

```
$ alternatives --install /usr/bin/python python /usr/bin/python3.8  
60
```

- d. 次のコマンドを実行し、使用する Python バージョンを選択します。NSO T-SDN FP バンドルには Python 3.8 が必要です。

```
$ alternatives --config python
```

4. OpenJDK 11 以降がインストールされていることを確認します。以下の手順に従って OpenJDK のバージョンを確認します。

```
$ Java --version  
openjdk 11.0.7
```

5. インストールを実行するための **sudo** ユーザー権限があることを確認します。このユーザーは、**ncsadmin** グループにも属している必要があります。

6. (LSA インストールの場合のみ) Network Time Protocol (NTP) を設定します。

7. 次のように、NSO とオペレーティングシステムの ulimit レベル値を **/etc/init.d/ncs** に追加します。

```
...  
ncsdir=/opt/ncs/current  
confdir=/etc/ncs  
rundir=/var/opt/ncs  
logdir=/var/log/ncs  
  
ncs=${ncsdir}/bin/ncs  
ulimit -n 65535  
prog=ncs  
conf="-c ${confdir}/ncs.conf"  
heart="--heart"  
...
```

8. オペレーティングシステムの ulimit 値を追加します。次に、例を示します。

- a. **/etc/security/limits.conf** ファイルを編集し、次の行を追加します。

```
* soft nproc 65535  
* hard nproc 65535  
* soft nofile 65535  
* hard nofile 65535  
* hard memlock 65536  
* soft memlock 65536
```

- b. **sysctl -p** スクリプトを実行してパラメータを設定します。

- c. システムからログアウトし、再度ログインして新しい値を適用します。

9. ulimit 値が適用されていることを確認します。

```
[admin@cns0-60-52 ~]$ ulimit -a
```

```
core file size          (blocks, -c) 0
data seg size          (kbytes, -d) unlimited
scheduling priority    (-e) 0
file size              (blocks, -f) unlimited
pending signals        (-i) 95697
max locked memory      (kbytes, -l) 65536
max memory size        (kbytes, -m) unlimited
open files             (-n) 65535
pipe size              (512 bytes, -p) 8
POSIX message queues   (bytes, -q) 819200
real-time priority     (-r) 0
stack size             (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
max user processes     (-u) 4096
virtual memory         (kbytes, -v) unlimited
file locks             (-x) unlimited
```

10. 必要に応じて、ncs サービスのタイムアウト値を設定します。ncs サービスのデフォルトのタイムアウト値は 300 秒です。

```
sudo mkdir /etc/systemd/system/ncs.service.d
echo -e "[Service]\nTimeoutStartSec=<timeout_in_seconds>" | sudo tee
/etc/systemd/system/ncs.service.d/startup-timeout.conf
sudo systemctl daemon-reload
```

11. タイムアウト値が設定されていることを確認します。

```
sudo systemctl show ncs | grep ^Timeout
```

単一の NSO インスタンスでの Cisco NSO T-SDN 機能 パックバンドルのインストールとアンインストール

このトピックでは、単一の NSO インスタンスに NSO T-SDN FP バンドルをインストールする
ために必要なパッケージと、そのようなインストールに向けて NSO 環境を整えるための情報に
ついて説明します。

単一の NSO インスタンス上のパッケージカテゴリとパッ ッケージ

このセクションでは、必要なパッケージカテゴリと関連するパッケージについて説明します。こ
れらのパッケージは、SR-TE CFP-IOSXR CLI のインストール中またはインストール後に抽出さ
れます。

SR-TE CFP-IOSXR CLI のインストールの詳細については、「[SR-TE CFP-IOSXR CLI のインス
トール](#)」を参照してください。

次の表は、コア機能パックカテゴリのパッケージとパッケージカテゴリの例を示しています。

SR-TE CFP をインストールするには、コア機能パックパッケージが必要です。パッケージ例に
は、インストールすることを選択したサンプル機能パック (L2VPN など) のパッケージが含ま
れています。

コア機能パック	
パッケージカテゴリ	パッケージ
T-SDN FP バンドルパッケージ	ncs-5.7.6-cisco-sr-te-cfp-4.1.0.tar.gz ncs-5.7.6-cisco-sr-te-cfp-internal- 4.1.0.tar.gz
T-SDN FP バンドル共通パッケージ	ncs-5.7-core-fp-plan-notif-generator- 1.0.6.tar.gz ncs-5.7.2-custom-template-utils-2.0.9.tar.gz ncs-5.7.2.1-lsa-utils-1.0.1.tar.gz ncs-5.7-core-fp-common-1.30.0.tar.gz ncs-5.7.6-cisco-tdsn-core-fp-common- 4.1.0.tar.gz
NED	IOSXR CLI NED : ncs-5.7.5-cisco-iosxr- 7.40.1.tar.gz IOSXE CLI NED : ncs-5.7-cisco-ios- 6.77.9.tar.gz IOSXR NC NED : ncs-5.7-cisco-iosxr_netconf-7.3.2.tar.gz ncs-5.7.2-cisco-iosxr_netconf-7.4.2.tar.gz

	ncs-5.7.3-cisco-iosxr_netconf-7.5.2.tar.gz ncs-5.7.4-cisco-iosxr-nc-7.6.2.19l.tar.gz ncs-5.7.5-cisco-iosxr_netconf-7.7.1.tar.gz ncs-5.7-cisco-iosxr_netconf-7.315.tar.gz
IOSXR NETCONF NED および IOSXE CLI NED を使 用する T-SDN FP バンドルマルチベンダー	ncs-5.7.6-sr-te-multi-vendors-4.1.0.tar.gz

サンプルパッケージ	
パッケージカテゴリ	パッケージ名
L2NM、L3NM、flat-L2、flat-L3 の自動ア シュアランスパッケージ	ncs-5.7.6-cisco-aa-service-assurance-EXAMPLE- 4.1.0.tar.gz
IOSXR CLI NED を使用する L2VPN サンプル パッケージ	ncs-5.7.6-cisco-flat-L2vpn-fp-EXAMPLE- 4.1.0.tar.gz ncs-5.7.6-cisco-flat-L2vpn-fp-internal-EXAMPLE- 4.1.0.tar.gz
IOSXR NETCONF NED および IOSXE CLI NED を使用する L2VPN マルチベンダーサンプル	ncs-5.7.6-flat-l2vpn-multi-vendors-EXAMPLE- 4.1.0.tar.gz
IOSXR CLI NED を使用する L3VPN サンプル パッケージ	ncs-5.7.6-cisco-flat-L3vpn-fp-EXAMPLE- 4.1.0.tar.gz ncs-5.7.6-cisco-flat-L3vpn-fp-internal-EXAMPLE- 4.1.0.tar.gz
IOSXR NETCONF NED および IOSXE CLI NED を使用する L3VPN マルチベンダーサンプル	ncs-5.7.6-flat-l3vpn-multi-vendors-EXAMPLE- 4.1.0.tar.gz
L2NM の Resource Manager	ncs-5.7.1-resource-manager-3.5.7.tar.gz
L2/L3 インターフェイスの共有削除ハンドラ パッケージ	ncs-5.7-core-fp-delete-tag-service-1.0.5.tar.gz
IETF L2NM インターフェイスを介した T-SDN Flat L2VPN サービスの作成を処理するための ノースバウンド/外部サービスパッケージ	ncs-5.7.6-ietf-l2vpn-nm-EXAMPLE-4.1.0.tar.gz
IETF L3NM インターフェイスを介した T-SDN Flat L3VPN サービスの作成を処理するための ノースバウンド/外部サービスパッケージ	ncs-5.7.6-ietf-l3vpn-nm-EXAMPLE-4.1.0.tar.gz
IOSXR CLI NED を使用する IETF-TE サンプル パッケージ	ncs-5.7.6-ietf-te-fp-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-rsvp-te-fp-EXAMPLE-4.1.0.tar.gz
IOSXR NETCONF NED および IOSXE CLI を使 用する RSVP-TE マルチベンダーサンプル	ncs-5.7.6-rsvp-te-multi-vendors-EXAMPLE- 4.1.0.tar.gz

単一の NSO インスタンスでの NCS 構成ファイルの編集

NCS 構成ファイルを編集して、Cisco NSO T-SDN 機能パックバンドルの設定を追加または更新します。ncs.conf ファイルの詳細については、**volume5** の **nso_man-<version>.pdf** ドキュメントを参照してください。

/etc/ncs/ncs.conf ファイルを次のように編集します。

1. **<stream>** の下に **service-state-changes** を追加して、サービス状態の変更に関する通知を生成します。

```
<notifications>
  <event-streams>
    <stream>
      <name>service-state-changes</name>
      <description>Plan state transitions according to
tailf-ncs-plan.yang</description>
      <replay-support>true</replay-support>
      <builtin-replay-store>
        <enabled>true</enabled>
        <dir>./state</dir>
        <max-size>S10M</max-size>
        <max-files>50</max-files>
      </builtin-replay-store>
    </stream>
```

2. 自動アシュアランス (AA) のインストールを選択した場合は、**<notifications>** -> **<event streams>** の下に AA 通知を追加して、AA 構成の変更を通知します。

```
<stream>
  <name>service-aa-changes</name>
  <description>Notifications relating to the service aa configuration
change</description>
  <replay-support>true</replay-support>
  <builtin-replay-store>
    <enabled>true</enabled>
    <dir>./state</dir>
    <max-size>S10M</max-size>
    <max-files>50</max-files>
  </builtin-replay-store>
</stream>
```

3. 次のとおりに **<hide-group>** 情報を追加します。

```
<hide-group>
  <name>debug</name>
```



```
</hide-group>
<hide-group>
  <name>tsdn</name>
</hide-group>
<hide-group>
  <name>fastmap-private</name>
</hide-group>
</ncs-config>
```

4. CLI、webui、および netconf ノースバウンドパラメータの場合、SSH ポートの設定はオプションです。これらのパラメータについて、必要に応じて SSH ポートの設定を有効または無効にすることができます。

デフォルトでは、これらのパラメータの SSH ポートの設定は無効になっています。これらのパラメータの詳細については、NSO のドキュメントを参照してください。

次に、SSH ポートの設定を有効化する方法を示します。要件に応じてポート番号を指定します。

CLI の SSH ポート

```
<cli>
  <enabled>true</enabled>
  <!-- Use the builtin SSH server -->
  <ssh>
    <enabled>true</enabled>
    <ip>0.0.0.0</ip>
    <port>${North_Bound_CLI_SSH_Port}</port>
  </ssh>
```

webui

TCP または SSL のいずれかで webui を有効にすることができます。

```
<webui>
  <enabled>true</enabled>
  <transport>
    <tcp>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>${North_Bound_Web_UI_Port}</port>
    </tcp>

    <ssl>
  <enabled>true</enabled>
    <ip>0.0.0.0</ip>
    <port>${SSL_port}</port>
    <key-file>${NCS_CONFIG_DIR}/ssl/cert/host.key</key-file>
```

```
<cert-file>${NCS_CONFIG_DIR}/ssl/cert/host.cert</cert-file>  
</ssl>  
</transport>
```

netconf northbound

```
<netconf-north-bound>  
  <enabled>true</enabled>  
  <transport>  
    <ssh>  
      <enabled>true</enabled>  
      <ip>0.0.0.0</ip>  
      <port>${Netconf_North_Bound_port}</port>  
    </ssh>
```

5. <python-vm> の下の **ncs.conf** に start-timeout を追加/更新します。

```
<python-vm>  
  <start-timeout>PT300S</start-timeout>  
</python-vm>
```

単一の NSO インスタンスへのコア機能パックのインストール

Cisco NSO T-SDN FP バンドルをインストールするには、SR-TE CFP-IOSXR CLI をインストールする必要があります。SR-TE CFP-IOSXR CLI は、プライマリ CFP または T-SDN FP バンドルの主要コンポーネントです。これは、他の CFP (SR-TE CFP-IOSXR NC および SR-TE CFP IETF-TE) とサンプル機能パックがインストールされる基本フレーバーです。

このトピックでは、コア機能パックをインストールする手順について説明します。サンプル機能パックのインストール方法については、「[単一の NSO インスタンスへのサンプル機能パック](#)」を参照してください。

SR-TE CFP-IOSXR CLI のインストール

このセクションの情報を使用して、SR-TE CFP-IOSXR CLI をインストールします。このインストールは、他のすべてのフレーバーのベースです。

SR-TE CFP-IOSXR CLI をインストールするには：

- 次のセクションで説明されているタスクをすでに実行したことを確認してください。
 - Cisco T-SDN FP バンドルをインストールするための NSO 環境の準備**
 - 単一の NSO インスタンスでの NCS 構成ファイルの編集**
- ncsadmin** ユーザーグループのメンバーでもある **sudo** ユーザーとしてホストマシンにログインします。

- 署名付き bin ファイル **nso-<version>-tsdn-<version>.signed.bin** パッケージをシスコの Web サイトから入手してダウンロードし、ホストサーバーにコピーします。

たとえば、**nso-5.7.6-tsdn-4.1.0.signed.bin** パッケージをシスコの Web サイトから入手してダウンロードし、ホストサーバーにコピーします。

- bin ファイルの内容を現在のディレクトリに抽出します。

```
$ sh nso-<version>-tsdn-<version>.signed.bin
```

これにより、製品の真正性が確認されます。ネットワーク接続の問題が発生した場合は、次のコマンドを実行してこの検証をスキップします。

```
$ sh nso-<version>-tsdn-<version>.signed.bin --skip-verification
```

- T-SDN FP バンドルインストーラの **tar.gz** ファイルを現在のディレクトリに展開します。フォルダがすでに存在する場合は、既存のフォルダのバックアップを作成してください。

```
$ tar -xvf nso-<version>-tsdn-<version>.tar.gz
```

- パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/core-fp-packages/
```

- 次のパッケージをコピーしてリンクし、SR-TE CFP-IOSXR CLI をインストールします。他のパッケージは、さまざまなインストールフレーバー用です。

```
sudo cp ncs-xxx-cisco-sr-te-cfp-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-xxx.tar.gz
```

```
sudo cp ncs-xxx-cisco-sr-te-cfp-internal-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-internal-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-internal-xxx.tar.gz
```

```
sudo cp ncs-xxx-custom-template-utils-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-custom-template-utils-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-custom-template-utils-xxx.tar.gz
```

```
sudo cp ncs-xxx-cisco-tsdn-core-fp-common-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-tsdn-core-fp-common-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-tsdn-core-fp-common-xxx.tar.gz
```

```
sudo cp ncs-xxx-core-fp-common-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz
```

```
sudo cp ncs-xxx-cisco-iosxr-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz
```

```
sudo cp ncs-xxx-core-fp-plan-notif-generator-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-xxx.tar.gz
```

```
sudo cp ncs-xxx-lsa-utils-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz
```

注： 1つ以上のインストールフレーバーをインストールするには、そのフレーバーに必要なパッケージを **EXAMPLE** フォルダからコピーしてリンクします。フレーバーと必要なパッケージの詳細については、「[単一の NSO インスタンス上のパッケージカテゴリとパッケージ](#)」を参照してください。

8. (オプション) 300 秒のデフォルトのタイムアウト後に NCS サービスが失敗した場合、NCS サービスのデフォルトのタイムアウトを変更します。

```
sudo mkdir /etc/systemd/system/ncs.service.d  
echo -e "[Service]\nTimeoutStartSec=<timeout_in_seconds>" | sudo tee  
/etc/systemd/system/ncs.service.d/startup-timeout.conf  
sudo systemctl daemon-reload
```

新しいタイムアウトが適用されていることを確認します。

```
sudo systemctl show ncs | grep ^Timeout
```

9. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

T-SDN FP バンドルのインストールが完了し、SR-TE CFP-IOSXR CLI がシステムにインストールされました。

10. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs> show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

11. SR-TE CFP-IOSXR CLI のインストールはこれで完了です。インストール後のタスクを実行します。

SR-TE CFP-IOSXR CLI のインストール後のタスクの実行

SR-TE CFP-IOSXR CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. **SR-plan-notification-settings.xml** ファイルを load-merge して、通知をアクティブに
します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge SR-plan-notification-settings.xml
admin@ncs% commit
```

3. **SR-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブに
します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge SR-status-codes.xml
admin@ncs% commit
```

4. **SR-cfp-configuration-kicker.xml** ファイルを load-merge して、キッカーをアクティブに
します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge SR-internal-plan-kicker.xml
admin@ncs% load merge SR-cfp-configuration-kicker.xml
admin@ncs% commit
```

5. 次のブートストラップデータを設定します。

- a. **commit-queue 設定**

```
admin@ncs% load merge commit-queue-settings.xml
admin@ncs% commit
```

- b. **AA 通知設定**

AA パッケージをインストールした場合は、この設定を構成します。

```
unhide tsdn
load merge config/IETF-L2NM-AA-notification-settings.xml
load merge config/IETF-L3NM-AA-notification-settings.xml
load merge config/L2VPN-AA-notification-settings.xml
load merge config/L3VPN-AA-notification-settings.xml
commit
hide tsdn
```

- c. **ユーザーの NACM ルールを設定します。**

ユーザーは、**ncsadmin** グループのメンバーでもある **sudo** ユーザーである必要
があります。

```
admin@ncs% set nacm groups group ncsadmin user-name [ <user name>  
private ]  
admin@ncs% commit
```

- d. SSH-RSA アルゴリズムの公開キーを設定します。このアルゴリズムは、グローバル
レベルで、またはオンボーディングされる特定のデバイスに対して設定できます。

アルゴリズムをグローバルに設定する

```
% show devices global-settings ssh-algorithms public-key  
public-key [ ssh-ed25519 ecdsa-sha2-nistp256 ecdsa-sha2-nistp384  
ecdsa-sha2-nistp521 rsa-sha2-512 rsa-sha2-256 ];
```

```
% set devices global-settings ssh-algorithms public-key [ ssh-  
ed25519 ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521  
rsa-sha2-512 rsa-sha2-256 ssh-rsa ]  
% commit
```

```
% show device global-settings ssh-algorithms public-key  
public-key [ ssh-ed25519 ecdsa-sha2-nistp256 ecdsa-sha2-nistp384  
ecdsa-sha2-nistp521 rsa-sha2-512 rsa-sha2-256 ssh-rsa ];
```

アルゴリズムを特定のデバイスに対して設定する

```
% set devices device <DEVICE_NAME> ssh-algorithms public-key [ ssh-  
rsa ]  
% commit
```

```
% show device device <DEVICE_NAME> ssh-algorithms public-key  
public-key [ ssh-rsa ];
```

SR-TE CFP-IOSXR CLI のインストール後のタスクの確認

実行したインストール後のタスクが正しいことを確認します。

次の手順を実行します。

1. キッカーの構成を確認します。

```
admin@ncs% show kickers  
data-kicker sr-te-cfp-configuration-kicker {  
    monitor      /cisco-sr-te-cfp:cfp-configurations;  
    kick-node    /cisco-sr-te-cfp:sr-te;  
    action-name  update-internal-cfp-configurations;
```

```
}
data-kicker sr-te-odn-internal-plan-kicker {
    monitor /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-odn-
internal:odn/cisco-sr-te-cfp-sr-odn-internal:odn-template-plan;
    kick-node /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/cisco-sr-
te-cfp-sr-odn:actions;
    action-name internal-plan-change-handler;
}
data-kicker sr-te-policy-internal-plan-kicker {
    monitor /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-policies-
internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-plan;
    kick-node /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/cisco-sr-te-cfp-sr-policies:actions;
    action-name internal-plan-change-handler;
}
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes
core-function-pack SR {
    status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/sr_te_status_codes;
    status-code 301 {
        reason "Device unreachable";
        category device;
        severity ERROR;
        recommended-actions "Check device connectivity from NSO and perform
recovery steps.";
    }
    status-code 302 {
        reason "Device out of sync";
        category device;
        severity ERROR;
        recommended-actions "Check sync between device and NSO, and perform
recovery steps.";
    }
    ...
    ...
}
[ok]
```

3. プラン通知を確認します。

```
admin@ncs% run show configuration services plan-notifications
```

```
subscription sr-odn-notif {
    service-type /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-odn:odn/cisco-sr-
te-cfp-sr-odn:odn-template;
}
subscription sr-policy-notif {
    service-type /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/cisco-sr-te-cfp-sr-policies:policy;
}
[ok]
admin@ncs% run show configuration plan-path-for-notification
plan-path-for-notification /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
odn:odn/odn-template-plan {
    service-path /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
odn:odn/odn-template;
    service-key-elements [ name ];
}
plan-path-for-notification /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy-plan {
    service-path /cisco-sr-te-cfp:sr-te/cisco-sr-te-cfp-sr-
policies:policies/policy;
    service-key-elements [ name ];
}
```

4. デバイスリストを確認してください。リストに設定されているデバイスが表示されます。このリス
トのデバイスは、ディスパッチマップにも入力する必要があります。デバイスが設定されてい
ない場合、リストは空です。

```
admin@ncs> show devices list
```

5. ブートストラップデータが正常にロードされたことを確認します。

- a. AA 通知設定

```
admin@ncs% unhide tsdn
admin@ncs% run show service-path-for-subscription
```

SERVICE PATH	LSA			LSA			DEVICE	CUSTOMER	PLAN	CONFIG									
	DEVICES	SERVICES	SERVICES	DEVICES	SERVICES	SERVICES	LIST	SERVICE	LOCATION	ID	STATUS	NAME	TIME	DATA	ERROR	WHEN	TYPE	LEVEL	MESSAGE
/cisco-flat-12vpn-fp:flat-12vpn	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
/cisco-flat-13vpn-fp:flat-13vpn	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
/12vpn-ntw:12vpn-ntw/vpn-services/vpn-service	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
/13vpn-ntw:13vpn-ntw/vpn-services/vpn-service	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

```
[ok]
```

```
admin@ncs% hide tsdn
```

- b. ディスパッチマップを確認します。ディスパッチマップには、デバイスリストからデ
バイスを読み込まれます。デバイスリストにデバイスがない場合、ディスパッチマッ
プは空です。

- c. 以下は、2 つの PIOSXR デバイス用に正しく構成されたディスパッチマップのサ
ンプル出力です。

```
admin@ncs% show devices lsa dispatch-map PIOSXR-0 {
    ned-id cisco-iosxr-cli-7.33:cisco-iosxr-cli-7.33;
}
dispatch-map PIOSXR-1 {
    ned-id cisco-iosxr-cli-7.33:cisco-iosxr-cli-7.33;
}
```

6. **commit-queue** 設定の構成を確認します。

```
admin@ncs% show devices global-settings commit-queue
enabled-by-default false;
async;
atomic                false;
retry-attempts        0;
retry-timeout         30;
error-option          stop-on-error;
[ok]
```

SR-TE CFP-IOSXR NC のインストール

このセクションでは、SR-TE CFP-IOSXR NC をインストールして検証するためにコピーする必
要があるパッケージについて説明します。

SR-TE CFP-IOSXR NC では、SR-TE CFP-IOSXR CLI をインストールする必要があります。
SR-TE CFP-IOSXR CLI のインストール方法の詳細については、「[SR-TE CFP-IOSXR CLI のイ
ンストール](#)」を参照してください。

SR-TE CFP-IOSXR NC をインストールするには：

1. NETCONF 7.3 NED を取得して NCS にロードします。
2. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/core-fp-packages/
```

3. 次のパッケージをコピーしてリンクし、SR-TE CFP-IOSXR NC をインストールします。

```
sudo cp ncs-xxx-sr-te-multi-vendors-xxx.tar.gz /opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz
/var/opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz

sudo cp ncs-xxx-cisco-iosxr_netconf-xxx.tar.gz /opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr_netconf-xxx.tar.gz
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr_netconf-xxx.tar.gz
```

4. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

SR-TE CFP-IOXR NC はこれで完了です。

5. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs
version | select build-info file | select build-info package sha1 | select
oper-status error-info | select oper-status up | tab
```

6. SR-TE CFP-IOXR NC のインストールはこれで完了です。インストール後のタスクを実行し ます。

SR-TE CFP-IOXR NC インストールのインストール後のタスクの実行 SR-TE CFP-IOXR NC をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. **SR-multi-vendor-iosxr-netconf.xml** ファイルを load-merge して、動的マッピングを構成し ます。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge SR-multi-vendor-iosxr-netconf.xml
admin@ncs% commit
```

SR-TE CFP-IOXR NC のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
admin@ncs% show cisco-sr-te-cfp:cfp-configurations
dynamic-device-mapping cisco-iosxr-nc-7.3:cisco-iosxr-nc-7.3 {
  python-impl-class-name sr_te_multi_vendors.NativeXR;
}
dynamic-device-mapping cisco-iosxr-nc-7.4:cisco-iosxr-nc-7.4 {
  python-impl-class-name sr_te_multi_vendors.NativeXR;
}
...
[ok]
```

SR-TE CFP-IOXE CLI のインストール

このセクションでは、必要なパッケージをコピーし、SR-TE CFP-IOXE CLI をインストールし
て検証する手順について説明します。

SR-TE CFP-IOXE CLI をインストールする前に、SR-TE CFP-IOXR CLI がインストールされ
ていることを確認してから、このトピックで説明されているタスクを続行します。SR-TE CFP-

IOSXR CLI のインストール方法の詳細については、「**SR-TE CFP-IOSXR CLI のインストール**」を参照してください。

SR-TE CFP-IOSXE CLI をインストールするには：

1. IOSXE CLI 6.77 NED を取得して NCS にロードします。

2. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/core-fp-packages/
```

3. 次のパッケージをコピーしてリンクし、SR-TE CFP-IOSXE CLI をインストールします。

```
sudo cp core-fp-packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz  
/opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz
```

```
sudo cp core-fp-packages/ncs-xxx-cisco-ios-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

4. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

SR-TE CFP-IOSXE CLI はこれで完了です。

5. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

6. SR-TE CFP-IOSXE CLI のインストールはこれで完了です。インストール後のタスクを実行します。

SR-TE CFP-IOSXE CLI のインストール後のタスクの実行

SR-TE CFP-IOSXE CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. **SR-multi-vendor-iosxe-cli.xml** ファイルを load-merge して、動的マッピングを構成します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
```

```
admin@ncs> configure
```

```
admin@ncs% load merge SR-multi-vendor-iosxe-cli.xml
```

```
admin@ncs% commit
```

SR-TE CFP-IOSXE CLI のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
admin@ncs% show cisco-sr-te-cfp:cfp-configurations
dynamic-device-mapping cisco-ios-cli-6.77:cisco-ios-cli-6.77 {
  python-impl-class-name sr_te_multi_vendors.IosXE;
}
[ok]
```

単一の NSO インスタンスへのサンプル機能パックのイン ストール

SR-TE CFP-IOSXR-CLI は基本フレーバーであり、スタンドアロンサービスを除き、サンプル機能パック（フレーバー）をインストールするための前提条件です。

基本フレーバー SR-TE CFP-IOSXR CLI の上に 1 つ以上のフレーバーをインストールすることを選択できます。フレーバーをインストールするには、SR-TE CFP-IOSXR CLI のインストール中またはインストール後に、フレーバーのパッケージをコピーします。

さらに、サービス (L2VPN/L3VPN/IETF-TE) の IOSXR-NC フレーバーまたは IOSXE-CLI フレーバーには、そのサービスの IOSXR-CLI フレーバーが必要です。たとえば、L2VPN-IOSXR NC フレーバーをインストールするには、次のことを行う必要があります。

1. 基本フレーバー (SR-TE CFP-IOSXR CLI) をインストールする
2. 基本フレーバーに L2VPN-IOSXR CLI フレーバーをインストールする
3. L2VPN-IOSXR NC フレーバーをインストールする

このトピックでは、次のフレーバーをインストールする手順について説明します。

- IOSXR-CLI/IOSXR-NC/IOSXE-CLI を使用する L2VPN サービス
 - IOSXR-CLI/IOSXR-NC/IOSXE-CLI を使用する L2NM サービス
- IOSXR-CLI/IOSXR-NC/IOSXE-CLI を使用する L3VPN サービス
 - IOSXR-CLI/IOSXR-NC/IOSXE-CLI を使用する L3NM サービス
- IOSXR-CLI/IOSXR-NC/IOSXE-CLI を使用する IETF-TE サービス

注： L2VPN サービスおよび L3VPN サービスについては、これらのサービスの管理に使用するモデルを確認してください。L2VPN サービスまたは L3VPN サービスを作成する前に、ネイティブモデルの実装または IETF の実装を選択します。サービス間でこれらの実装モデルを切り替えないでください。

スタンドアロンのフレーバー

スタンドアロンインストールを実行する場合は、SR-TE CFP-IOSXR CLI のインストール中に **cisco-sr-te-cfp** パッケージをコピーしないでください。IOSXR-CLI、IOSXR-NC、または IOSXE-CLI をスタンドアロンサービスとして使用して、L2VPN サービス、L3VPN サービス、および IETF-TE サービスをインストールできます。

IOSXR-CLI を使用したサービスのスタンドアロンインストールは基本であり、IOSXR-NC または IOSXE-CLI フレーバーを使用したサービスのスタンドアロンインストールを実行するための前提条件です。

たとえば、L2VPN-IOSXR NC をスタンドアロンサービスとしてインストールするには、最初に L2VPN-IOSXR CLI をスタンドアロンサービスとしてインストールしてから、L2VPN-IOSXR NC サービスをインストールする必要があります。

自動アシュアランスサービスのインストール

自動アシュアランス機能はオプション機能であり、Flat L2VPN、L2NM、Flat L3VPN、および L3NM にのみ適用されます。この機能は、**ncs-xxx-cisco-aa-service-assurance-EXAMPLE-<version>.tar.gz** パッケージが抽出され、サンプルサービスをインストールするパッケージと共にコピーされた場合にインストールされます。

L2VPN/L2NM サービスのインストール

このセクションでは、SR-TE CFP-IOSXR CLI の上に、またはスタンドアロンプレーヤーとして L2VPN/L2NM サービスをインストールするために必要なパッケージと、それを確認する手順について説明します。

L2VPN-IOSXR CLI のインストール

基本プレーヤーに L2VPN-IOSXR CLI をインストールするには、SR-TE CFP-IOSXR CLI がインストールされていることを確認してから、このトピックで説明されているタスクを続行します。SR-TE CFP-IOSXR CLI のインストール方法の詳細については、「**SR-TE CFP-IOSXR CLI のインストール**」を参照してください。

L2VPN-IOSXR CLI をスタンドアロンプレーヤーとしてインストールするには、「**SR-TE CFP-IOSXR CLI のインストール**」セクションの**ステップ 1**から**ステップ 5**のタスクを実行してから、このトピックで説明されているタスクを続行します。

L2VPN-IOSXR CLI をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. 必要に応じて次のいずれかの作業を行います。

- a. SR-TE CFP-IOSXR CLI の上に L2VPN-IOSXR CLI をインストールするには、以下のパッケージをコピーしてリンクします。

注： AA 機能をインストールするには、AA パッケージをコピーしてリンクします。ただし、AA パッケージのロードはオプションです。

```
sudo cp example-packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-xxx.tar.gz  
sudo cp example-packages/ncs-xxx-cisco-flat-L2vpn-fp-internal-EXAMPLE-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-internal-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-  
fp-internal-EXAMPLE-xxx.tar.gz  
sudo cp example-packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-  
service-xxx.tar.gz  
sudo cp example-packages/ncs-xxx-resource-manager-xxx.tar.gz  
/opt/ncs/packages  
sudo ln -s /opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz  
sudo cp example-packages/ncs-xxx-cisco-aa-service-assurance-  
EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-aa-service-assurance-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-aa-service-  
assurance-EXAMPLE-xxx.tar.gz
```

- b. L2VPN-IOSXR CLI をスタンドアロンフレーバーとしてインストールするには、次の
パッケージをコピーしてリンクします。

```
sudo cp example-packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-  
EXAMPLE-xxx.tar.gz  
sudo cp example-packages/ncs-xxx-cisco-flat-L2vpn-fp-internal-  
EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-internal-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-  
fp-internal-EXAMPLE-xxx.tar.gz  
sudo cp example-packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-  
service-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-core-fp-plan-notif-generator-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-plan-notif-  
generator-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-custom-template-utils-xxx.tar.gz  
/opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-custom-template-utils-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-custom-template-utils-  
xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L2VPN-IOSXR CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

注： スタンドアロン L2VPN インストールでは、cisco-sr-te-cfp パッケージは表示され
ません。

5. インストール後のタスクを実行します。

L2VPN-IOSXR CLI のインストール後のタスクの実行

L2VPN-IOSXR CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tdsn-<release>/tdsn/bootstrap-data
```

2. (スタンドアロンインストールの場合のみ) 「SR-TE CFP-IOSXR CLI のインストール後のタ
スクの実行」セクションの説明に従って、一般的なブートストラップデータを設定します。

3. L2VPN-plan-notification-settings.xml ファイルを load-merge して、通知をアクティブにし
ます。

```
$ /opt/ncs/current/bin/ncs_cli -u admin  
admin@ncs> configure
```



```
admin@ncs% load merge L2VPN-plan-notification-settings.xml
admin@ncs% commit
```

4. **L2VPN-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge L2VPN-status-codes.xml
admin@ncs% commit
```

5. **L2VPN-internal-plan-kicker.xml** ファイルと **L2VPN-cfp-configuration-kicker.xml** ファイルを load-merge して、キッカー設定をアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge L2VPN-internal-plan-kicker.xml
admin@ncs% load merge L2VPN-cfp-configuration-kicker.xml
admin@ncs% commit
```

6. AA がインストールされている場合は、**L2VPN-AA-notification-settings.xml** ファイルを load-merge して AA 通知を構成します。

```
/opt/ncs/current/bin/ncs_cli -u admin
unhide tsdn
configure
load merge config/L2VPN-AA-notification-settings.xml
commit
```

7. Y1731 の SMAN ID リソースプールを構成します。

```
ncs_cli -u admin
configure
load merge config/xr-sman-id-resource-pool.xml
commit
```

L2VPN-IOSXR CLI のインストール後のタスクの確認

次のように、インストール後のタスクを確認します。

1. キッカーの構成を確認します。

```
unhide debug
admin@ncs% show kickers
data-kicker flat-L2vpn-internal-local-site-plan-kicker {
```

```
    monitor      /cisco-flat-L2vpn-fp-internal-local-site:flat-L2vpn-internal-  
local-site/cisco-flat-L2vpn-fp-internal-local-site:flat-L2vpn-plan;  
    kick-node    /cisco-flat-L2vpn-fp:actions;  
    action-name  l2vpn-internal-plan-change-handler;  
  }  
data-kicker flat-L2vpn-internal-remote-site-plan-kicker {  
    monitor      /cisco-flat-L2vpn-fp-internal-remote-site:flat-L2vpn-  
internal-remote-site/cisco-flat-L2vpn-fp-internal-remote-site:flat-L2vpn-  
plan;  
    kick-node    /cisco-flat-L2vpn-fp:actions;  
    action-name  l2vpn-internal-plan-change-handler;  
  }  
data-kicker flat-l2vpn-cfp-configuration-kicker {  
    monitor      /cisco-flat-L2vpn-fp:cfp-configurations;  
    kick-node    /cisco-flat-L2vpn-fp:flat-L2vpn-actions;  
    action-name  update-internal-cfp-configurations;  
  }  
[ok]
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes  
core-function-pack L2VPN {  
    status-code-enum-path cisco-tdsn-core-fp-common/python/cisco_tsdn-  
__core_fp_common/status_codes/flat_L2vpn_status_codes;  
    status-code 301 {  
        reason          "Device unreachable";  
        category        device;  
        severity        ERROR;  
        recommended-actions "Check device connectivity from NSO and perform  
recovery steps.";  
    }  
    status-code 302 {  
        reason          "Device out of sync";  
        category        device;  
        severity        ERROR;  
        recommended-actions "Check sync between device and NSO, and perform  
recovery steps.";  
    }  
    ...  
    ...  
}
```

```
}  
[ok]
```

3. プラン通知を確認します。プラン通知には、AA パッケージがインストールされている場合にのみ AA モデルが表示されます。

```
admin@ncs% run show configuration services plan-notifications  
subscription l2vpn-notif {  
    service-type /cisco-flat-L2vpn-fp:flat-L2vpn;  
}  
[ok]
```

```
admin@ncs% show plan-path-for-notification  
plan-path-for-notification /cisco-flat-L2vpn-fp:flat-L2vpn-plan {  
    service-path          /cisco-flat-L2vpn-fp:flat-L2vpn;  
    service-key-elements [ name ];  
}  
[ok]
```

4. Y1731 の SMAN ID リソースプールを確認します。

```
admin@ncs% show resource-pools id-pool sman-id-pool  
range {  
    start 1;  
    end 65535;  
}  
[ok]
```

L2NM-IOSXR CLI のインストール

L2NM は、L2VPN 実装の標準化された IETF バージョンを採用しています。このセクションでは、L2NM-IOSXR CLI サービスをインストールして検証するためにコピーする必要があるパッケージについて説明します。

L2NM をインストールするには、L2VPN-IOSXR CLI をインストールする必要があります。L2VPN-IOSXR CLI のインストール方法の詳細については、「[L2VPN-IOSXR CLI のインストール](#)」を参照してください。

L2NM-IOSXR CLI をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages/
```

2. 次のパッケージをコピーしてリンクし、IOSXR CLI を使用する L2NM をインストールします。

```
sudo cp example-packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

L2NM-IOSXR CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs
version | select build-info file | select build-info package sha1 | select
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

L2NM-IOSXR CLI インストール後のタスクの実行

L2NM-IOSXR CLI サービスをインストールした後、次の手順を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/bootstrap-data
```

2. **IETF-L2NM-plan-notification-settings.xml** ファイルを load-merge して、通知をアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge IETF-L2NM-plan-notification-settings.xml
admin@ncs% commit
```

3. **IETF-L2NM-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-L2NM-status-codes.xml
admin@ncs% commit
```

4. **IETF-L2NM-internal-plan-kicker.xml** ファイルを load-merge して、キッカー設定をアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-L2NM-internal-plan-kicker.xml
```

L2NM-IOSXR CLI のインストール後のタスクの確認

次のように、インストール後のタスクを確認します。

1. キッカーの構成を確認します。

```
unhide debug
admin@ncs% show kickers
data-kicker l2nm-internal-plan-kicker {
    monitor      /cisco-flat-L2vpn-fp:flat-L2vpn-plan;
    kick-node    /l2vpn-ntw:l2nm-actions;
    action-name  internal-plan-change-handler;
}
[ok]
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes
core-function-pack IETF-L2NM {
    status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/ietf_l2vpn_nm_status_cod
es;
    status-code 400 {
        reason          "Status code mapping has not been loaded for
function pack during install";
        category        user;
        severity        ERROR;
        recommended-actions "Bootstrap status code mapping";
    }
    status-code 404 {
        reason          "Input element's value is not supported";
        category        validation;
        severity        ERROR;
        recommended-actions "Verify that input element's value is supported
in the payload";
    }
    ...
    ...
}
[ok]
```

3. プラン通知を確認します。プラン通知には、AA パッケージがインストールされている場合にのみ AA モデルが表示されます。

```
admin@ncs% run show configuration services plan-notifications
subscription l2nm-notif {
```

```
    service-type /l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-  
service;  
}  
[ok]  
admin@ncs% run show plan-path-for-notification  
plan-path-for-notification /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service-plan  
{  
    service-path          /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service;  
    service-key-elements [ vpn-id ];  
}  
[ok]
```

L2VPN-IOSXR NC のインストール

L2VPN-IOSXR NC をインストールするには、L2VPN-IOSXR CLI をインストールする必要があります。L2VPN-IOSXR CLI のインストール方法の詳細については、「[L2VPN-IOSXR CLI のインストール](#)」を参照してください。

L2VPN-IOSXR NC をインストールするには：

1. Netconf NED **cisco-iosxr_nc-7.3** を入手して NCS にダウンロードします。

2. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/example-packages
```

3. 以下のパッケージをコピーしてリンクし、L2VPN-IOSXR NC をインストールします。

```
sudo cp ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz
```

```
sudo cp core-fp-packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz
```

4. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L2VPN-IOSXR NC のインストールはこれで完了です。

5. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

注： 出力には、スタンドアロン L2VPN インストール用の cisco-sr-te-cfp パッケージは表示されません。

6. インストール後のタスクを実行します。

L2VPN-IOSXR NC のインストール後のタスクの実行

L2VPN-IOSXR NC をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/bootstrap-data
```

2. **L2VPN-multi-vendor-iosxr-netconf.xml** ファイルを load-merge して、動的マッピングを構成します。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% load merge L2VPN-multi-vendor-iosxr-netconf.xml  
admin@ncs% commit
```

3. Y1731 の SMAN ID リソースプールを構成します。

```
$ ncs_cli -u admin  
configure  
load merge config/xr-sman-id-resource-pool.xml  
commit
```

L2VPN-IOSXR NC のインストール後のタスクの確認

次の手順を実行します。

1. 次のように動的マッピングを確認します。通知には、AA パッケージがインストールされている場合にのみ AA モデルが表示されます。

```
unhide tsdn  
admin@ncs% show cisco-flat-L2vpn-fp:cfp-configurations  
dynamic-device-mapping cisco-iosxr_nc-7.3:cisco-iosxr_nc-7.3 {  
    python-impl-class-name flat_l2vpn_multi_vendors.NativeXR;  
}  
dynamic-device-mapping cisco-iosxr_nc-7.4:cisco-iosxr_nc-7.4 {  
    python-impl-class-name flat_l2vpn_multi_vendors.NativeXR;
```

```
}  
[ok]
```

2. Y1731 の SMAN ID リソースプールを確認します。

```
admin@ncs% show resource-pools id-pool sman-id-pool  
range {  
    start 1;  
    end 65535;  
}  
[ok]
```

L2NM-IOSXR NC のインストール

L2NM は、L2VPN 実装の標準化された IETF バージョンを採用しています。このセクションでは、L2NM-IOSXR NC サービスをインストールして検証するためにコピーする必要があるパッケージについて説明します。

L2NM NC をインストールするには、L2VPN-IOSXR NC をインストールする必要があります。L2VPN-IOSXR NC のインストール方法の詳細については、「[L2VPN-IOSXR NC のインストール](#)」を参照してください。

L2NM-IOSXR NC をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/example-packages
```

2. 以下のパッケージをコピーしてリンクし、L2NM-IOSXR NC をインストールします。

```
sudo cp ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz  
sudo cp ncs-xxx-resource-manager-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L2NM-IOSXR NC のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```


注： スタンドアロン L2VPN インストールでは、cisco-sr-te-cfp パッケージは表示されません。

5. インストール後のタスクを実行します。

L2NM-IOSXR NC のインストール後のタスクの実行

L2NM-IOSXR NC をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/bootstrap-data
```

2. **IETF-L2NM-plan-notification-settings.xml** ファイルを load-merge して、通知をアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge IETF-L2NM-plan-notification-settings.xml
admin@ncs% commit
```

3. **IETF-L2NM-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-L2NM-status-codes.xml
admin@ncs% commit
```

4. **IETF-L2NM-internal-plan-kicker.xml** ファイルを load-merge して、キッカー設定をアクティブにします。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-L2NM-internal-plan-kicker.xml
admin@ncs% commit
```

L2NM-IOSXR NC のインストール後のタスクの確認

次の手順を実行します。

1. キッカーの構成を確認します。

```
unhide debug
admin@ncs% show kickers
data-kicker l2nm-internal-plan-kicker {
    monitor /cisco-flat-L2vpn-fp:flat-L2vpn-plan;
```

```
kick-node /l2vpn-ntw:l2nm-actions;  
action-name internal-plan-change-handler;  
}  
[ok]
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes  
core-function-pack IETF-L2NM {  
    status-code-enum-path cisco-tsdn-core-fp-  
common/python/cisco_tsdn_core_fp_common/status_codes/ietf_l2vpn_nm_status_cod  
es;  
    status-code 400 {  
        reason "Status code mapping has not been loaded for  
function pack during install";  
        category user;  
        severity ERROR;  
        recommended-actions "Bootstrap status code mapping";  
    }  
    status-code 404 {  
        reason "Input element's value is not supported";  
        category validation;  
        severity ERROR;  
        recommended-actions "Verify that input element's value is supported  
in the payload";  
    }  
    :::  
    }  
}  
[ok]
```

3. プラン通知を確認します。

```
admin@ncs% run show configuration services plan-notifications  
subscription l2nm-notif {  
    service-type /l2vpn-ntw:l2vpn-ntw/l2vpn-ntw:vpn-services/l2vpn-ntw:vpn-  
service;  
}  
[ok]  
  
admin@ncs% show plan-path-for-notification  
plan-path-for-notification /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service-plan  
{  
    service-path /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-service;  
    service-key-elements [ vpn-id ];  
}  
[ok]
```

L2VPN-IOSXE CLI のインストール

このセクションでは、パッケージをコピーし、SR-TE CFP-IOSXR CLI の上に、またはスタンドアロンフレーバーとして L2VPN-IOSXE CLI サービスをインストールして検証する手順について説明します。

L2VPN-IOSXE CLI をインストールするには、L2VPN-IOSXR CLI をインストールする必要があります。L2VPN-IOSXR CLI のインストール方法の詳細については、「[L2VPN-IOSXR CLI のインストール](#)」を参照してください。

L2VPN-IOSXE CLI をインストールするには：

1. IOSXE-CLI 6.77 NED を取得して NCS にロードします。

2. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

3. 次のパッケージをコピーしてリンクし、L2VPN-IOSXE CLI をインストールします。

```
sudo cp example-packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz
```

```
sudo cp core-fp-packages/ncs-xxx-cisco-ios-xxx.tar.gz /opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

4. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
```

```
Restarting ncs (via systemctl):
```

```
[ OK ]
```

L2VPN-IOSXE CLI のインストールはこれで完了です。

5. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

注： スタンドアロン L2VPN インストールでは、cisco-sr-te-cfp パッケージは表示されません。

6. インストール後のタスクを実行します。

L2VPN-IOSXE CLI のインストール後のタスクの実行

L2VPN-IOSXE CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. L2VPN の動的マッピングを設定します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge L2VPN-multi-vendor-iosxe-cli.xml
admin@ncs% commit
```

L2VPN-IOSXE CLI のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn
admin@ncs% show cisco-flat-L2vpn-fp:cfp-configurations
dynamic-device-mapping cisco-ios-cli-6.77:cisco-ios-cli-6.77 {
    python-impl-class-name flat_l2vpn_multi_vendors.IosXE;
}
[ok]
```

L2NM-IOSXE CLI のインストール

L2NM は、L2VPN 実装の標準化された IETF バージョンを採用しています。このセクションでは、L2NM-IOSXE CLI サービスをインストールして検証するためにコピーする必要があるパッケージについて説明します。

L2NM-IOSXE CLI をインストールするには、L2NM-IOSXR CLI と L2VPN IOSXE CLI をインストールする必要があります。詳細については、「[L2NM-IOSXR CLI のインストール](#)」および「[L2VPN-IOSXE CLI のインストール](#)」をそれぞれ参照してください。

L2NM-IOSXE CLI をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/example-packages
```

2. 次のパッケージをコピーしてリンクし、L2NM-IOSXE CLI をインストールします。

```
sudo cp example-packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
```

```
sudo cp core-fp-packages/ncs-xxx-cisco-ios-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L2NM-IOSXE CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

L2NM-IOSXE CLI のインストール後のタスクの実行

L2NM-IOSXE CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/bootstrap-data
```

2. **L2VPN-multi-vendor-iosxe-cli.xml** ファイルを load-merge して、動的マッピングを構成します。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% load merge L2VPN-multi-vendor-iosxe-cli.xml  
admin@ncs% commit
```

L2NM-IOSXE CLI のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn  
admin@ncs% show cisco-flat-L2vpn-fp:cfp-configurations  
dynamic-device-mapping cisco-ios-cli-6.77:cisco-ios-cli-6.77 {  
    python-impl-class-name flat_l2vpn_multi_vendors.IosXE;  
}  
[ok]
```

L3VPN/L3NM サービスのインストール

このセクションでは、SR-TE CFP-IOSXR CLI の上に、またはスタンドアロンフレーバーとして L3VPN/L3NM サービスをインストールするためにコピーする必要があるパッケージと、それを確認する手順について説明します。

L3VPN-IOSXR CLI のインストール

基本フレーバーに L3VPN-IOSXR CLI をインストールするには、SR-TE CFP-IOSXR CLI がインストールされていることを確認してから、このセクションで説明されているタスクを続行します。SR-TE CFP-IOSXR CLI のインストール方法の詳細については、「**SR-TE CFP-IOSXR CLI のインストール**」を参照してください。

L3VPN-IOSXR CLI をスタンドアロンフレーバーとしてインストールするには、「**SR-TE CFP-IOSXR CLI のインストール**」セクションの**ステップ 1**から**ステップ 5**のタスクを実行してから、このトピックで説明されているタスクを続行します。

L3VPN-IOSXR CLI をインストールするには：

1. IOSXE-CLI 6.77 NED を取得して NCS にロードします。

2. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

3. 必要に応じて次のいずれかの作業を行います。

- a. 次のパッケージをコピーしてリンクし、SR-TE CFP-IOSXR CLI に L3VPN-IOSXR CLI をインストールします。

注： AA 機能をインストールするには、AA パッケージをコピーしてリンクします。ただし、AA パッケージのロードはオプションです。

```
sudo cp ncs-xxx-cisco-flat-L3vpn-fp-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-  
EXAMPLE-xxx.tar.gz  
sudo cp ncs-xxx-cisco-flat-L3vpn-fp-internal-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-internal-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-  
fp-internal-EXAMPLE-xxx.tar.gz  
sudo cp ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-  
service-xxx.tar.gz  
sudo cp ncs-xxx-cisco-aa-service-assurance-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-aa-service-assurance-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-aa-service-  
assurance-EXAMPLE-xxx.tar.gz
```

- b. 次のパッケージをコピーしてリンクし、L3VPN-IOSXR CLI をスタンドアロンフレー
バーとしてインストールします。

```
sudo cp ncs-xxx-cisco-flat-L3vpn-fp-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-  
EXAMPLE-xxx.tar.gz  
sudo cp ncs-xxx-cisco-flat-L3vpn-fp-internal-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-internal-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-  
fp-internal-EXAMPLE-xxx.tar.gz  
sudo cp ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-  
service-ncs-xxx-cisco-iosxr-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-core-fp-plan-notif-generator-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-plan-notif-  
generator-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-custom-template-utils-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-custom-template-utils-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-custom-template-utils-  
xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz  
sudo cp example-packages/ncs-xxx-cisco-aa-service-assurance-  
EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-aa-service-assurance-  
EXAMPLE-xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-aa-service-  
assurance-EXAMPLE-xxx.tar.gz
```

```
sudo cp core-fp-packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz
```

4. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L3VPN-IOSXR CLI のインストールはこれで完了です。

5. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

スタンドアロン L3VPN インストールでは、**cisco-sr-te-cfp** パッケージは表示されません。

6. インストール後のタスクを実行します。

L3VPN-IOSXR CLI のインストール後のタスクの実行

L3VPN-IOSXR CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. (L3VPN スタンドアロンインストールの場合のみ) 「SR-TE CFP-IOSXR CLI のインストール後のタスクの実行」セクションの説明に従って、一般的なブートストラップデータを設定します。

3. **L3VPN-plan-notification-settings.xml** ファイルを load-merge して、通知をアクティブにします。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% load merge L3VPN-plan-notification-settings.xml  
admin@ncs% commit
```

4. **L3VPN-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブにします。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% unhide debug  
admin@ncs% load merge L3VPN-status-codes.xml  
admin@ncs% commit
```

5. キッカーの xml ファイルを load-merge して、キッカー設定をアクティブにします。

```
$ ncs_cli -u admin
```



```
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge L3VPN-internal-plan-kicker.xml
admin@ncs% load merge L3VPN-cfp-configuration-kicker.xml
admin@ncs% commit
```

6. AA がインストールされている場合は、**L3VPN-AA-notification-settings.xml** ファイルを **load-merge** して AA 通知を構成します。

```
/opt/ncs/current/bin/ncs_cli -u admin
unhide tsdn
configure
load merge config/L3VPN-AA-notification-settings.xml
commit
```

L3VPN-IOSXR CLI のインストール後のタスクの確認

次のように、インストール後のタスクを確認します。

1. キッカーの構成を確認します。

```
unhide debug
admin@ncs% show kickers
data-kicker flat-l3vpn-cfp-configuration-kicker {
    monitor      /cisco-flat-L3vpn-fp:cfp-configurations;
    kick-node    /cisco-flat-L3vpn-fp:flat-L3vpn-actions;
    action-name  update-internal-cfp-configurations;
}
data-kicker flat-l3vpn-internal-plan-kicker {
    monitor      /cisco-flat-L3vpn-fp-internal:flat-L3vpn-internal/cisco-flat-
L3vpn-fp-internal:flat-L3vpn-plan;
    kick-node    /cisco-flat-L3vpn-fp:flat-L3vpn-actions;
    action-name  internal-plan-change-handler;
}
[ok]
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes
core-function-pack L3VPN {
    status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/flat_L3vpn_status_codes;
    status-code 301 {
        reason          "Device unreachable";
        category        device;
    }
}
```

```
        severity          ERROR;
        recommended-actions "Check device connectivity from NSO and perform
recovery steps.";
    }
    status-code 302 {
        reason              "Device out of sync";
        category            device;
        severity            ERROR;
        recommended-actions "Check sync between device and NSO, and perform
recovery steps.";
    }
    ...
}
[ok]
```

3. プラン通知を確認します。プラン通知には、AA パッケージがインストールされている場合にのみ AA モデルが表示されます。

```
admin@ncs% run show configuration services plan-notifications
subscription l3vpn-notif {
    service-type /cisco-flat-L3vpn-fp:flat-L3vpn;
}
[ok]
admin@ncs% show plan-path-for-notification
plan-path-for-notification /cisco-flat-L3vpn-fp:flat-L3vpn-plan {
    service-path      /cisco-flat-L3vpn-fp:flat-L3vpn;
    service-key-elements [ name ];
}
[ok]
```

L3NM-IOSXR CLI のインストール

L3NM は、L3VPN 実装の標準化された IETF バージョンを採用しています。このセクションでは、L3NM-IOSXR CLI サービスをインストールして検証するためにコピーする必要があるパッケージについて説明します。

L3NM をインストールするには、L3VPN-IOSXR CLI をインストールする必要があります。L3VPN-IOSXR CLI のインストール方法の詳細については、「[L3VPN-IOSXR CLI のインストール](#)」を参照してください。

L3NM-IOSXR CLI をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages/
```

2. 次のパッケージをコピーしてリンクし、L3NM-IOSXR CLI をインストールします。

```
sudo cp ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L3NM-IOSXR CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

L3NM-IOSXR CLI のインストール後のタスクの実行

IOSXR CLI を使用して L3NM をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. **IETF-L3NM-plan-notification-settings.xml** ファイルを load-merge して、通知をアクティブにします。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% load merge IETF-L3NM-plan-notification-settings.xml  
admin@ncs% commit
```

3. **IETF-L3NM-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブにします。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% unhide debug  
admin@ncs% load merge IETF-L3NM-status-codes.xml  
admin@ncs% commit
```

4. **IETF-L3NM-internal-plan-kicker.xml** ファイルを load-merge して、キッカー設定をアクティブにします。

```
$ ncs_cli -u admin  
admin@ncs> configure
```

```
admin@ncs% unhide debug
admin@ncs% load merge IETF-L3NM-internal-plan-kicker.xml
admin@ncs% commit
```

5. AA がインストールされている場合は、**IETF-L3NM-AA-notification-settings.xml** ファイルを load-merge して AA 通知を構成します。

```
/opt/ncs/current/bin/ncs_cli -u admin
unhide tsdn
configure
load merge IETF-L3NM-AA-notification-settings.xml
commit
```

L3NM-IOSXR CLI のインストール後のタスクの確認

次の手順を実行します。

1. キッカーの構成を確認します。

```
unhide debug
admin@ncs% show kickers
data-kicker l3nm-internal-plan-kicker {
    monitor      /cisco-flat-L3vpn-fp:flat-L3vpn-plan;
    kick-node    /l3vpn-ntw:l3nm-actions;
    action-name  internal-plan-change-handler;
}
[ok]
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes
core-function-pack IETF-L3NM {
    status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/ietf_l3vpn_nm_status_cod
es;
    status-code 400 {
        reason          "Status code mapping has not been loaded for
function pack during install";
        category        user;
        severity        ERROR;
        recommended-actions "Bootstrap status code mapping";
    }
    status-code 404 {
        reason          "Input element's value is not supported";
        category        validation;
    }
}
```

```
        severity          ERROR;
        recommended-actions "Verify that input element's value is supported
in the payload";
    }
...
...
    }
}
[ok]
```

3. プラン通知を確認します。プラン通知には、AA パッケージがインストールされている場合にのみ AA モデルが表示されます。

```
admin@ncs% run show configuration services plan-notifications
subscription l3nm-notif {
    service-type /l3vpn-ntw:l3vpn-ntw/l3vpn-ntw:vpn-services/l3vpn-ntw:vpn-
service;
}
[ok]

admin@ncs% show plan-path-for-notification
plan-path-for-notification /l3vpn-ntw:l3vpn-ntw/vpn-services/vpn-service-plan
{
    service-path          /l3vpn-ntw:l3vpn-ntw/vpn-services/vpn-service;
    service-key-elements [ vpn-id ];
}
[ok]
```

L3VPN-IOSXR NC のインストール

このセクションでは、SR-TE CFP-IOSXR CLI で、またはスタンドアロンフレーバーとして、L3VPN-IOSXR NC サービスをインストールおよび検証するためにコピーするパッケージについて説明します。

L3VPN-IOSXR NC をインストールするには、L3VPN-IOSXR CLI をインストールする必要があります。L3VPN-IOSXR CLI のインストール方法の詳細については、「[L3VPN-IOSXR CLI のインストール](#)」を参照してください。

L3VPN-IOSXR NC をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. 以下のパッケージをコピーしてリンクし、L3VPN-IOSXR NC をインストールします。

```
sudo cp ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz  
  
sudo cp core-fp-packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz  
/opt/ncs/packages/  
  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L3VPN-IOSXR NC のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

注： スタンドアロン L3VPN インストールでは、cisco-sr-te-cfp パッケージは表示され
ません。

5. インストール後のタスクを実行します。

L3VPN-IOSXR NC のインストール後のタスクの実行

L3VPN-IOSXR NC をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. **L3VPN-multi-vendor-iosxr-netconf.xml** ファイルを load-merge して、動的マッピングを構成
します。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% load merge L3VPN-multi-vendor-iosxr-netconf.xml  
admin@ncs% commit
```

L3VPN-IOSXR NC のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn  
  
show cisco-flat-L3vpn-fp:cfp-configurations  
dynamic-device-mapping cisco-iosxr-nc-7.3:cisco-iosxr-nc-7.3 {  
    python-impl-class-name flat_l3vpn_multi_vendors.NativeXR;
```

```
}  
dynamic-device-mapping cisco-iosxr-nc-7.4:cisco-iosxr-nc-7.4 {  
    python-impl-class-name flat_l3vpn_multi_vendors.NativeXR;  
}  
}
```

L3NM-IOSXR NC のインストール

L3NM は、L3VPN 実装の標準化された IETF バージョンを採用しています。このセクションでは、L3NM-IOSXR NC サービスをインストールして検証するためにコピーする必要があるパッケージについて説明します。

L3NM-IOSXR NC をインストールするには、L3VPN-IOSXR NC をインストールする必要があります。L3VPN-IOSXR NC のインストール方法の詳細については、「[L3VPN-IOSXR NC のインストール](#)」を参照してください。

L3NM-IOSXR NC をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. 以下のパッケージをコピーしてリンクし、L3NM-IOSXR NC をインストールします。

```
sudo cp ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L3NM-IOSXR NC のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

L3NM-IOSXR NC のインストール後のタスクの実行

L3NM-IOSXR NC をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<release>/tsdn/bootstrap-data
```

2. **IETF-L3NM-plan-notification-settings.xml** ファイルを load-merge して、通知をアクティブにします。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge IETF-L3NM-plan-notification-settings.xml
admin@ncs% commit
```

3. **IETF-L3NM-status-codes.xml** ファイルを load-merge して、ステータスコードをアクティブにします。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-L3NM-status-codes.xml
admin@ncs% commit
```

4. **IETF-L3NM-internal-plan-kicker.xml** ファイルを load-merge して、キッカーをアクティブにします。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-L3NM-internal-plan-kicker.xml
admin@ncs% commit
```

L3NM-IOSXR NC のインストール後のタスクの確認

次のように、インストール後のタスクを確認します。

プラン通知を確認します。

```
admin@ncs% run show configuration services plan-notifications
subscription l3nm-notif {
    service-type /l3vpn-ntw:l3vpn-ntw/l3vpn-ntw:vpn-services/l3vpn-ntw:vpn-
service;
}
[ok]

admin@ncs% run show configuration plan-path-for-notification
plan-path-for-notification /l3vpn-ntw:l3vpn-ntw/vpn-services/vpn-service-plan
{
    service-path /l3vpn-ntw:l3vpn-ntw/vpn-services/vpn-service;
    service-key-elements [ vpn-id ];
}
[ok]
```


L3VPN-IOSEXE CLI のインストール

L3VPN-IOSEXE CLI をインストールするには、L3VPN-IOSEXR CLI をインストールする必要があります。L3VPN-IOSEXR CLI のインストール方法の詳細については、「[L3VPN-IOSEXR CLI のインストール](#)」を参照してください。

L3VPN-IOSEXE CLI をインストールするには：

1. IOSEXE-CLI 6.77 NED を取得して NCS にロードします。

2. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

3. 次のパッケージをコピーしてリンクし、L3VPN-IOSEXE CLI をインストールします。

```
sudo cp ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
  
sudo ln -s /opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-  
xxx.tar.gz  
  
sudo cp core-fp-packages/ncs-xxx-cisco-ios-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

4. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L3VPN-IOSEXE CLI のインストールはこれで完了です。

5. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

注： スタンドアロン L3VPN インストールでは、cisco-sr-te-cfp パッケージは表示されません。

6. インストール後のタスクを実行します。

L3VPN-IOSEXE CLI のインストール後のタスクの実行

L3VPN-IOSEXE CLI をインストールした後、次のタスクを実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. **L3VPN-multi-vendor-iosxe-cli.xml** ファイルを load-merge して、動的マッピングを構成し
ます。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge L3VPN-multi-vendor-iosxe-cli.xml
admin@ncs% commit
```

L3VPN-IOSXE CLI のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn
show cisco-flat-L3vpn-fp:cfp-configurations
dynamic-device-mapping cisco-ios-cli-6.77:cisco-ios-cli-6.77 {
    python-impl-class-name flat_l3vpn_multi_vendors.IosXE;
}
```

L3NM-IOSXE CLI のインストール

L3NM は、L3VPN 実装の標準化された IETF バージョンを採用しています。このセクションで
は、L3NM-IOSXE CLI サービスをインストールして検証するためにコピーする必要があるパッ
ケージについて説明します。

L3NM-IOSXE CLI をインストールするには、L3NM-IOSXR CLI と L3VPN-IOSXE CLI をインス
トールする必要があります。詳細については、「[L3NM-IOSXR CLI のインストール](#)」および
「[L3VPN-IOSXE CLI のインストール](#)」をそれぞれ参照してください。

L3NM-IOSXE CLI をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages/
```

2. 次のパッケージをコピーしてリンクし、L3NM-IOSXE CLI をインストールします。

```
sudo cp ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

L3NM-IOSXE CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

L3NM-IOSXE CLI のインストール後のタスクの実行

L3NM-IOSXE CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. **L3VPN-multi-vendor-iosxe-cli.xml** ファイルを load-merge して、動的マッピングを構成します。

```
$ ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% load merge L3VPN-multi-vendor-iosxe-cli.xml  
admin@ncs% commit
```

L3NM-IOSXE CLI のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn  
show cisco-flat-L3vpn-fp:cfp-configurations  
dynamic-device-mapping cisco-ios-cli-6.77:cisco-ios-cli-6.77 {  
    python-impl-class-name flat_l3vpn_multi_vendors.IosXE;  
}
```

IETF-TE サービスのインストール

このセクションでは、パッケージをコピーし、SR-TE CFP-IOSXR CLI の上に、またはスタンドアロンフレーバーとして IETF-TE-IOSXR CLI サービスをインストールして検証する手順について説明します。

IETF TE-IOSXR CLI のインストール

基本フレーバーに IETF-TE-IOSXR CLI をインストールするには、SR-TE CFP-IOSXR CLI がインストールされていることを確認してから、このトピックで説明されているタスクを続行します。SR-TE CFP-IOSXR CLI のインストール方法の詳細については、「**SR-TE CFP-IOSXR CLI のインストール**」を参照してください。

IETF-TE-IOSXR CLI をスタンドアロンフレーバーとしてインストールするには、「**SR-TE CFP-
IOSXR CLI のインストール**」セクションの**ステップ 1**から**ステップ 5**のタスクを実行してか
ら、このトピックで説明されているタスクを続行します。

IETF TE-IOSXR CLI をインストールして確認するには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. 必要に応じて次のいずれかの作業を行います。

- a. 次のパッケージをコピーしてリンクし、SR-TE CFP-IOSXR CLI の上に IETF TE-
IOSXR CLI をインストールします。

```
sudo cp ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-  
xxx.tar.gz
```

```
sudo cp ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz
```

- b. 次のパッケージをコピーしてリンクし、IETF-TE-IOSXR CLI をスタンドアロンフレー
バーとしてインストールします。

```
sudo cp ncs-xxx-cisco-rsvp-te-fp-EXAMPLE- xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-  
xxx.tar.gz  
sudo cp ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz  
sudo cp ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-  
service-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz  
sudo cp core-fp-packages/ncs-xxx-core-fp-plan-notif-generator-  
xxx.tar.gz /opt/ncs/packages/  
sudo ln -s /opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-core-fp-plan-notif-  
generator-xxx.tar.gz
```

```
sudo cp core-fp-packages/ncs-xxx-custom-template-utils-xxx.tar.gz  
/opt/ncs/packages/  
  
sudo ln -s /opt/ncs/packages/ncs-xxx-custom-template-utils-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-custom-template-utils-  
xxx.tar.gz  
  
sudo cp core-fp-packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/opt/ncs/packages/  
  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz  
  
sudo cp core-fp-packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz /opt/ncs/packages/  
  
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz /var/opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-  
xxx.tar.gz  
  
sudo cp core-fp-packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/opt/ncs/packages/  
  
sudo ln -s /opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz  
/var/opt/ncs/packages/ncs-xxx-lsa-utils-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

IETF TE-IOSXR CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs  
version | select build-info file | select build-info package sha1 | select  
oper-status error-info | select oper-status up | tab
```

注： スタンドアロンインストールの場合、cisco-sr-te-cfp パッケージは表示されません。

5. インストール後のタスクを実行します。

IETF TE-IOSXR CLI のインストール後のタスクの実行

IETF TE-IOSXR CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tdsn-<version>/tdsn/example-packages
```

2. (スタンドアロンインストールの場合のみ) 「**SR-TE CFP-IOSXR CLI のインストール後のタスクの実行**」セクションの説明に従って、一般的なブートストラップデータを設定します。

3. **IETF-TE-plan-notification-settings.xml** ファイルを load-merge して、プラン通知を設定します。

```
$ ncs_cli -u admin
```

```
admin@ncs> configure
admin@ncs% load merge IETF-TE-plan-notification-settings.xml
admin@ncs% commit
```

4. 次の xml ファイルを load-merge して、ステータスコードを構成します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge RSVP-TE-status-codes.xml
load merge IETF-TE-status-codes.xml
admin@ncs% commit
```

5. 次の xml ファイルを load-merge して、キッカーを構成します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% load merge IETF-TE-internal-plan-kicker.xml
admin@ncs% load merge IETF-TE-cfp-configuration-kicker.xml
admin@ncs% commit
```

IETF-TE IOSXR CLI のインストール後のタスクの確認

次の手順を実行します。

1. キッカーの構成を確認します。

```
admin@ncs% show kickers
data-kicker ietf-te-fp-configuration-kicker {
    monitor      /te:cfp-configurations;
    kick-node    /te:ietf-te-actions;
    action-name  update-internal-fp-configurations;
}
data-kicker ietf-te-internal-plan-kicker {
    monitor      /cisco-rsvp-te-fp:rsvp-te/cisco-rsvp-te-fp:tunnel-te-plan;
    kick-node    /te:ietf-te-actions;
    action-name  internal-plan-change-handler;
}
[ok]
```

2. ステータスコードを確認します。

```
admin@ncs% show status-codes
core-function-pack IETF-TE {
    status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/ietf_te_status_codes;
    status-code 301 {
```

```
        reason          "Device unreachable";
        category         device;
        severity         ERROR;
        recommended-actions "Check device connectivity from NSO and perform
recovery steps.";
    }
    status-code 302 {
        reason          "Device out of sync";
        category         device;
        severity         ERROR;
        recommended-actions "Check sync between device and NSO, and perform
recovery steps.";
    }
    ...
}
}
core-function-pack RSVP-TE {
    status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/rsvp_te_status_codes;
    status-code 301 {
        reason          "Device unreachable";
        category         device;
        severity         ERROR;
        recommended-actions "Check device connectivity from NSO and perform
recovery steps.";
    }
    status-code 302 {
        reason          "Device out of sync";
        category         device;
        severity         ERROR;
        recommended-actions "Check sync between device and NSO, and perform
recovery steps.";
    }
    ...
}
}
[ok]
```

3. プラン通知を確認します。

```
admin@ncs% run show configuration services plan-notifications
subscription ietf-te-notif {
    service-type /te:te/te:tunnels/te:tunnel;
}
[ok]
```

```
admin@ncs% run show configuration plan-path-for-notification

plan-path-for-notification /te:te/tunnels/tunnel-plan {
  service-path          /te:te/tunnels/tunnel;
  service-key-elements [ name ];
}
[ok]
```

IETF TE-IOSXR NC のインストール

IETF-TE-IOSXR NC をインストールするには、IETF-TE-IOSXR CLI をインストールする必要があります。IETF-TE-IOSXR CLI のインストール方法の詳細については、「[IETF TE-IOSXR CLI のインストール](#)」を参照してください。

IETF TE-IOSXR NC をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. 以下のパッケージをコピーしてリンクし、IETF TE-IOSXR NC をインストールします。

```
sudo cp core-fp-packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz
/opt/ncs/packages/
```

```
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz
/var/opt/ncs/packages/ncs-xxx-cisco-iosxr-netconf-xxx.tar.gz
```

```
sudo cp ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz /opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz
/var/opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
          [ OK ]
```

IETF TE-IOSXR NC のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs
version | select build-info file | select build-info package sha1 | select
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

IETF TE-IOSXR NC のインストール後のタスクの実行

IETF TE-IOSXR NC をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. IETF-TE-multi-vendor-iosxr-netconf.xml ファイルを load-merge して、動的マッピングを構成します。

```
$ ncs_cli -u admin
configure
unhide debug
admin@ncs% load merge IETF-TE-multi-vendor-iosxr-netconf.xml
admin@ncs% commit
```

IETF TE-IOSXR NC のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn
admin@ncs% show cisco-rsvp-te-fp:cfp-configurations
dynamic-device-mapping cisco-iosxr-nc-7.3:cisco-iosxr-nc-7.3 {
    python-impl-class-name rsvp_te_multi_vendors.NativeXR;
}
dynamic-device-mapping cisco-iosxr-nc-7.4:cisco-iosxr-nc-7.4 {
    python-impl-class-name rsvp_te_multi_vendors.NativeXR;
}
```

IETF-TE-IOSEX CLI のインストール

IETF-TE-IOSEX CLI をインストールするには、IETF-TE-IOSXR CLI をインストールする必要があります。IETF-TE-IOSXR CLI のインストール方法の詳細については、「[IETF TE-IOSXR CLI のインストール](#)」を参照してください。

IETF-TE-IOSEX CLI をインストールするには：

1. パッケージディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/example-packages
```

2. 次のパッケージをコピーしてリンクし、IETF-TE-IOSEX CLI をインストールします。

```
sudo cp ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz /opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz
/var/opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz
sudo cp core-fp-packages/ncs-xxx-cisco-ios-xxx.tar.gz /opt/ncs/packages/
sudo ln -s /opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
/var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

3. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

IETF-TE-IOSXE CLI のインストールはこれで完了です。

4. インストールを確認し、パッケージが稼働していることを確認します。

```
admin@ncs% run show packages package package-version | select build-info ncs
version | select build-info file | select build-info package sha1 | select
oper-status error-info | select oper-status up | tab
```

5. インストール後のタスクを実行します。

IETF-TE-IOSXE CLI のインストール後のタスクの実行

IETF-TE-IOSXE CLI をインストールした後、次を実行します。

1. 現在のディレクトリを次のディレクトリに変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn/bootstrap-data
```

2. **IETF-TE-multi-vendor-iosxe-cli.xml** ファイルを load-merge して、動的マッピングを構成します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% load merge IETF-TE-multi-vendor-iosxe-cli.xml
admin@ncs% commit
```

IETF-TE-IOSXE CLI のインストール後のタスクの確認

次のように動的マッピングを確認します。

```
unhide tsdn
admin@ncs% show cisco-rsvp-te-fp:cfp-configurations
dynamic-device-mapping cisco-ios-cli-6.77:cisco-ios-cli-6.77 {
    python-impl-class-name rsvp_te_multi_vendors.IosXE;
}
```

単一の NSO インスタンスでの Cisco T-SDN FP バンドルのアンインストール

Cisco NSO T-SDN FP バンドルをアンインストールするには、最初に関連するサービスと関連するデバイスをシステムから削除する必要があります。サービスに対してゾンビサービスが実行されていないこと、およびすべてのデバイスがデバイスツリーから削除されていることを確認します。

SR-TE CFP-IOSXR CLI の上にインストールされているサービス（フレーバー）をアンインストールしても、システムは SR-TE CFP-IOSXR CLI サービスのレンダリングを継続します。ただし、フレーバーをアンインストールせずに SR-TE CFP-IOSXR CLI をアンインストールすると、フレーバーによってレンダリングされるサービスのみが使用可能になります。このような場合、共通パッケージとサービスに必要なパッケージが利用可能な場合にのみ、フレーバーはスタンドアロンサービスとして機能します。

このセクションでは、サンプル機能パック（フレーバー）とコア機能パックをアンインストールする手順について説明します。

単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す
フレーバーと Cisco NSO T-SDN FP バンドルをアンインストールする前に、`ncs.config` ファイルへの変更を元に戻します。

NCS 構成ファイルへの変更を元に戻すには、次の手順を実行します。

1. NCS を停止します。
2. `/etc/ncs/ncs.conf` を次のように編集します。
 - a. `python <start-command>` 構成を更新します。

```
<python-vm>
<start-command>DEFAULT</start-command>
<run-in-terminal>
  <terminal-command>DEFAULT</terminal-command>
</run-in-terminal>
<logging>
  <log-file-prefix>${NCS_LOG_DIR}/ncs-python-vm</log-file-prefix>
</logging>
</python-vm>
```

- b. 以下のように、`<notifications > <event-streams>` グループの下で、`<replay-support>` および `<builtin-replay-store>` パラメータを `false` に設定します。

```
<notifications>
  <event-streams>
    <stream>
      <name>service-state-changes</name>
      <description>Plan state transitions according to
tailf-ncs-plan.yang</description>
      <replay-support>>false</replay-support>
      <builtin-replay-store>
        <enabled>>false</enabled>
        <dir>./state</dir>
```

```
<max-size>S10M</max-size>  
<max-files>50</max-files>  
</builtin-replay-store>  
</stream>
```

- c. ファイルの最後にある <hide-group> セクションを削除します。

```
<hide-group>  
  <name>debug</name>  
</hide-group>  
<hide-group>  
  <name>tsdn</name>  
</hide-group>  
<hide-group>  
  <name>fastmap-private</name>  
</hide-group>
```

- d. CLI、webui、およびノースパウンド通知用の SSH ポート構成を有効にした場合、構成を元に戻すことはオプションです。

サンプル機能パックのアンインストール

SR-TE CFP-IOSXR CLI にインストールされているフレーバーをアンインストールしても、システムは SR-TE CFP-IOSXR CLI サービスのレンダリングを継続します。フレーバーをアンインストールする前に、フレーバーのすべてのサービスを削除します。

L2VPN/L2NM サービスのアンインストール

このセクションの情報を使用して、IOSXR CLI/IOSXR NC/IOSXE CLI で L2VPN/L2NM サービスをアンインストールします。

L2NM-IOSXR CLI のアンインストール

L2NM-IOSXR CLI フレーバーをアンインストールする前に、関連するすべてのサービスとデバイスをデバイスツリーから削除してください。

L2NM-IOSXR CLI をアンインストールすると、システムは L2VPN-IOSXR CLI フレーバーに戻ります。L2VPN-IOSXR CLI のアンインストール方法の詳細については、「[L2VPN-IOSXR CLI のアンインストール](#)」を参照してください。

L2NM-IOSXR CLI をアンインストールするには :

1. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。
2. プラン通知を削除します。

```
$ ncs_cli -u admin
```

```
admin@ncs> configure
admin@ncs% delete services plan-notifications subscription l2nm-notif
admin@ncs% delete plan-path-for-notification /l2vpn-ntw:l2vpn-ntw/vpn-
services/vpn-service-plan
admin@ncs% commit
```

3. ステータスコードを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete status-codes core-function-pack IETF-L2NM
admin@ncs% delete status-code-cfp IETF-L2NM
admin@ncs% commit
```

4. キッカーを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
delete kickers data-kicker l2nm-internal-plan-kicker
admin@ncs% commit
```

5. AA がインストールされている場合は、AA モジュールの通知を削除します。

```
unhide debug
/opt/ncs/current/bin/ncs_cli -u admin
configure
delete service-path-for-subscription /cisco-flat-L2vpn-fp:flat-L2vpn
delete service-path-for-subscription /l2vpn-ntw:l2vpn-ntw/vpn-services/vpn-
service
commit
```

6. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz
```

7. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

8. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

L2VPN-IOSXR CLI のアンインストール

このフレーバーをアンインストールすると、システムは SR-TE CFP-IOSXR CLI に戻ります。SR-TE CFP-IOSXR CLI のアンインストール方法の詳細については、「[SR-TE CFP-IOSXR CLI のアンインストール](#)」を参照してください。

L2VPN-IOSXR CLI をアンインストールするには：

1. L2NM-IOSXR CLI サービスがインストールされている場合は、アンインストールします。L2NM-IOSXR CLI のアンインストール方法の詳細については、「[L2NM-IOSXR CLI のアンインストール](#)」を参照してください。
2. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。
3. プラン通知を削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete services plan-notifications subscription l2vpn-notif
admin@ncs% delete plan-path-for-notification /cisco-flat-L2vpn-fp:flat-L2vpn-
plan
admin@ncs% commit
```

4. ステータスコードを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete status-codes core-function-pack L2VPN
admin@ncs% delete status-code-cfp L2VPN
admin@ncs% commit
```

5. キッカーを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete kickers data-kicker flat-L2vpn-internal-local-site-plan-
kicker
admin@ncs% delete kickers data-kicker flat-L2vpn-internal-remote-site-plan-
kicker
```

```
admin@ncs% delete kickers data-kicker flat-l2vpn-cfp-configuration-kicker
admin@ncs% commit
```

6. AA がインストールされている場合は、AA モジュールの通知を削除します。

```
unhide debug
/opt/ncs/current/bin/ncs_cli -u admin
configure
delete service-path-for-subscription /cisco-flat-L2vpn-fp:flat-L2vpn
commit
```

7. `/var/opt/ncs/packages` にある以下のパッケージのリンクを解除し、`/opt/ncs/packages/` ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-xxx.tar.gz /opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-EXAMPLE-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-internal-EXAMPLE-xxx.tar.gz /opt/ncs/packages/ncs-xxx-cisco-flat-L2vpn-fp-internal-EXAMPLE-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz
```

AA がインストールされている場合は、以下のパッケージのリンクを解除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-aa-service-assurance-EXAMPLE-xxx.tar.gz /opt/ncs/packages/ncs-xxx-cisco-aa-service-assurance-EXAMPLE-xxx.tar.gz
```

8. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show zombies
sudo /etc/init.d/ncs stop
```

9. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

L2NM-IOSXR NC のアンインストール

L2NM-IOSXR NC フレーバーをアンインストールするには、最初に NC NED で L2NM サービスを削除する必要があります。

このフレーバーをアンインストールすると、システムは L2VPN-IOSXR NC フレーバーに戻ります。L2VPN-IOSXR NC をアンインストールする方法の詳細については、「[L2VPN-IOSXR NC のアンインストール](#)」を参照してください。

L2NM-IOSXR NC フレーバーをアンインストールする前に、関連するすべてのサービスとデバイスをデバイスツリーから削除してください。

L2NM-IOSXR NC をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO** インスタンスでの NCS 構成ファイルへの変更を元に戻す」を参照してください。

2. プラン通知を削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete services plan-notifications subscription l2nm-notif
admin@ncs% delete plan-path-for-notification /l2vpn-ntw:l2vpn-ntw/vpn-
services/vpn-service-plan
admin@ncs% commit
```

3. L2NM のステータスコードを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete status-codes core-function-pack IETF-L2NM
admin@ncs% delete status-code-cfp IETF-L2NM
admin@ncs% commit
```

4. キッカーを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete kickers data-kicker l2nm-internal-plan-kicker
admin@ncs% commit
```

5. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-ietf-l2vpn-nm-EXAMPLE-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-resource-manager-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```


7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

L2VPN-IOSXR NC のアンインストール

このフレーバーをアンインストールすると、システムは L2VPN-IOSXR CLI フレーバーに戻ります。L2VPN-IOSXR CLI フレーバーをアンインストールするには、「[L2VPN-IOSXR CLI のアンインストール](#)」を参照してください。

L2NM がインストールされている場合は、最初に L2NM をアンインストールする必要があります。

L2VPN-IOSXR NC をアンインストールするには：

1. L2NM-IOSXR NC がインストールされている場合は、アンインストールします。L2NM-IOSXR NC のアンインストール方法の詳細については、「[L2NM-IOSXR NC のアンインストール](#)」を参照してください。
2. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。
3. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-flat-L2vpn-fp:cfp-configurations dynamic-device-
mapping cisco-iosxr_netconf-xxx:cisco-iosxr_netconf-xxx
admin@ncs% commit
```

4. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
```

5. Netconf NED を削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show zombies
sudo /etc/init.d/ncs stop
```

7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

L2NM-IOSXE CLI のアンインストール

L2NM-IOSXE CLI フレーバーをアンインストールすると、システムは L2VPN-IOSXE CLI フレーバーに戻ります。L2VPN-IOSXE CLI のアンインストール方法の詳細については、「[L2VPN-IOSXE CLI のアンインストール](#)」を参照してください。

L2NM-IOSXE CLI フレーバーをアンインストールする前に、関連するすべてのサービスとデバイスをデバイスツリーから削除してください。

L2NM-IOSXE CLI をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。

2. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-flat-L2vpn-fp:cfp-configurations dynamic-device-
mapping cisco-ios-cli-xxx:cisco-ios-cli-xxx
admin@ncs% commit
```

3. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
```

4. CLI NED を削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

L2VPN-IOSXE CLI のアンインストール

このフレーバーをアンインストールすると、システムは L2VPN-IOSXR CLI に戻ります。L2VPN-
IOSXR CLI フレーバーのアンインストール方法の詳細については、「[L2VPN-IOSXR CLI のアン
インストール](#)」を参照してください。

L2NM がインストールされている場合は、最初に L2NM をアンインストールする必要があります。

L2VPN-IOSXE CLI をアンインストールするには：

1. L2NM-IOSXE CLI がインストールされている場合は、アンインストールします。L2NM-IOSXE
CLI のアンインストール方法の詳細については、「[L2NM-IOSXE CLI のアンインストール](#)」を参
照してください。
2. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構
成ファイルへの変更を元に戻す](#)」を参照してください。

3. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-flat-l2vpn-fp:cfp-configurations dynamic-device-
mapping cisco-ios-cli-xxx:cisco-ios-cli-xxx
admin@ncs% commit
```

4. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディ
レクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-flat-l2vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
```

5. CLI NED を削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

L3VPN/L3NM サービスのアンインストール

このセクションの情報を使用して、IOSXR CLI/IOSXR NC/IOSXE CLI で L3VPN/L3NM サービスをアンインストールします。

L3NM-IOSXR CLI のアンインストール

このフレーバーをアンインストールすると、システムは L3VPN-IOSXR CLI に戻ります。L3VPN-IOSXR CLI フレーバーのアンインストール方法の詳細については、「**L3VPN-IOSXR CLI のアンインストール**」を参照してください。

L3NM-IOSXR CLI フレーバーをアンインストールする前に、関連するすべてのサービスとデバイスをデバイスツリーから削除してください。

L3NM-IOSXR CLI をアンインストールするには :

1. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す**」を参照してください。

2. プラン通知を削除します。

```
$ ncs_cli -u admin
admin@ncs> configure

admin@ncs% delete services plan-notifications subscription l3nm-notif
admin@ncs% delete plan-path-for-notification /l3vpn-ntw:l3vpn-ntw/vpn-
services/vpn-service-plan
admin@ncs% commit
```

3. ステータスコードを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure

admin@ncs% unhide debug

admin@ncs% delete status-codes core-function-pack IETF-L3NM
admin@ncs% delete status-code-cfp IETF-L3NM
admin@ncs% commit
```

4. キッカーを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure

admin@ncs% unhide debug

admin@ncs% delete kickers data-kicker l3nm-internal-plan-kicker
admin@ncs% commit
```

5. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz  
/opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN  
### Make sure there are no zombie services by running the command: show  
zombies  
sudo /etc/init.d/ncs stop
```

7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart  
Restarting ncs (via systemctl):  
[ OK ]
```

L3VPN-IOSXR CLI のアンインストール

このフレーバーをアンインストールすると、システムは SR-TE CFP-IOSXR CLI に戻ります。SR-TE CFP-IOSXR CLI のアンインストール方法の詳細については、「**SR-TE CFP-IOSXR CLI のアンインストール**」を参照してください。

L3NM がインストールされている場合は、最初に L3NM-IOSXR CLI をアンインストールする必要があります。

L3VPN-IOSXR CLI をアンインストールするには：

1. L3NM-IOSXR CLI サービスがインストールされている場合は、アンインストールします。L3NM-IOSXR CLI のアンインストール方法の詳細については、「**L3NM-IOSXR CLI のアンインストール**」を参照してください。
2. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す**」を参照してください。
3. プラン通知を削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% delete services plan-notifications subscription l3vpn-notif  
admin@ncs% delete plan-path-for-notification /cisco-flat-L3vpn-fp:flat-L3vpn-  
plan  
admin@ncs% commit
```

4. ステータスコードを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin  
admin@ncs> configure  
admin@ncs% unhide debug
```

```
admin@ncs% delete status-codes core-function-pack L3VPN
admin@ncs% delete status-code-cfp L3VPN
admin@ncs% commit
```

5. キッカーを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete kickers data-kicker flat-l3vpn-cfp-configuration-kicker
admin@ncs% delete kickers data-kicker flat-l3vpn-internal-plan-kicker
admin@ncs% commit
```

6. `/var/opt/ncs/packages` にある以下のパッケージのリンクを解除し、`/opt/ncs/packages/` ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-EXAMPLE-xxx.tar.gz /opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-EXAMPLE-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz /opt/ncs/packages/ncs-xxx-core-fp-delete-tag-service-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-internal-EXAMPLE-xxx.tar.gz /opt/ncs/packages/ncs-xxx-cisco-flat-L3vpn-fp-internal-EXAMPLE-xxx.tar.gz
```

7. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show zombies
sudo /etc/init.d/ncs stop
```

8. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

L3NM-IOSXR NC のアンインストール

このフレーバーをアンインストールすると、システムは L3VPN-IOSXR NC フレーバーに戻ります。L3VPN-IOSXR NC をアンインストールする方法の詳細については、「[L3VPN-IOSXR NC のアンインストール](#)」を参照してください。

L3NM-IOSXR NC フレーバーをアンインストールする前に、関連するすべてのサービスとデバイスをデバイスツリーから削除してください。

L3NM-IOSXR NC をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す**」を参照してください。

2. プラン通知を削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete services plan-notifications subscription l3nm-notif
admin@ncs% delete plan-path-for-notification /l3vpn-ntw:l3vpn-ntw/vpn-
services/vpn-service-plan
admin@ncs% commit
```

3. L3NM のステータスコードを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete status-codes core-function-pack IETF-L3NM
admin@ncs% delete status-code-cfp IETF-L3NM
admin@ncs% commit
```

4. キッカーを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete kickers data-kicker l3nm-internal-plan-kicker
admin@ncs% commit
```

5. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-ietf-l3vpn-nm-EXAMPLE-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

L3VPN-IOSXR NC のアンインストール

このフレーバーをアンインストールすると、システムは L3VPN-IOSXR CLI に戻ります。
L3VPN-IOSXR CLI のアンインストール方法の詳細については、「[L3VPN-IOSXR CLI のアンインストール](#)」を参照してください。

L3NM がインストールされている場合は、最初に L3NM-IOSXR NC をアンインストールする
必要があります。

L3VPN-IOSXR NC をアンインストールするには：

1. L3NM-IOSXR NC がインストールされている場合は、アンインストールします。L3NM-IOSXR
NC のアンインストール方法の詳細については、「[L3NM-IOSXR NC のアンインストール](#)」を参
照してください。
2. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構
成ファイルへの変更を元に戻す](#)」を参照してください。
3. L3VPN の動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-flat-L3vpn-fp:cfp-configurations dynamic-device-
mapping cisco-iosxr-nc-xxx:cisco-iosxr-nc-xxx
admin@ncs% commit
```

4. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディ
レクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
```

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```


L3NM-IOSXE CLI のアンインストール

L3NM-IOSXE CLI フレーバーをアンインストールすると、システムは L3VPN-IOSXE CLI フレーバーに戻ります。L3VPN-IOSXE CLI をアンインストールする方法については、「**L3VPN-IOSXE CLI のアンインストール**」を参照してください。

L3NM-IOSXE CLI フレーバーをアンインストールする前に、関連するすべてのサービスとデバイスをデバイスツリーから削除してください。

L3NM-IOSXE CLI をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す**」を参照してください。

2. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-flat-L3vpn-fp:cfp-configurations dynamic-device-
mapping cisco-ios-cli-xxx:cisco-ios-cli-xxx
admin@ncs% commit
```

3. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

4. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

5. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

L3VPN-IOSXR CLI のアンインストール

このフレーバーをアンインストールすると、システムは L3VPN-IOSXR CLI に戻ります。

L3VPN-IOSXR CLI フレーバーをアンインストールする方法については、「**L3VPN-IOSXR CLI のアンインストール**」を参照してください。

L3NM がインストールされている場合は、最初に L3NM をアンインストールする必要があります。

L3VPN-IOSXE CLI をアンインストールするには：

1. L3NM-IOSXE CLI がインストールされている場合は、アンインストールします。L3NM-IOSXE CLI をアンインストールする方法の詳細については、「**L3NM-IOSXE CLI のアンインストール**」を参照してください。
2. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す**」を参照してください。

3. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-flat-L3vpn-fp:cfp-configurations dynamic-device-
mapping cisco-ios-cli-xxx:cisco-ios-cli-xxx
admin@ncs% commit
```

4. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-flat-l3vpn-multi-vendors-EXAMPLE-
xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

IETF-TE サービスのアンインストール

このセクションの情報を使用して、IOSXR CLI/IOSXR NC/IOSXE CLI で IETF-TE サービスをアンインストールします。

IETF-TE-IOSXR-CLI のアンインストール

このフレーバーをアンインストールすると、システムは SR-TE CFP-IOSXR CLI に戻ります。SR-TE CFP-IOSXR CLI のアンインストール方法の詳細については、「[SR-TE CFP-IOSXR CLI のアンインストール](#)」を参照してください。

IOSXR-CLI で IETF-TE をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。

2. プラン通知を削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete services plan-notifications subscription ietf-te-notif
admin@ncs% delete plan-path-for-notification /te:te/tunnels/tunnel-plan
admin@ncs% commit
```

3. IETF-TE のステータスコードを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete status-codes core-function-pack IETF-TE
admin@ncs% delete status-code-cfp IETF-TE
admin@ncs% delete status-codes core-function-pack RSVP-TE
admin@ncs% delete status-code-cfp RSVP-TE
admin@ncs% commit
```

4. キッカーを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete kickers data-kicker ietf-te-fp-configuration-kicker
admin@ncs% delete kickers data-kicker ietf-te-internal-plan-kicker
admin@ncs% commit
```

5. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-ietf-te-fp-EXAMPLE-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-rsvp-te-fp-EXAMPLE-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

IETF-TE-IOXR NC のアンインストール

このフレーバーをアンインストールすると、システムは IETF-TE IOSXR CLI に戻ります。IETF-TE IOSXR CLI をアンインストールする方法については、「[IETF-TE-IOXR-CLI のアンインストール](#)」を参照してください。

IETF-TE-IOXR NC をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。
2. IETF-TE の動的マッピングを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-rsvp-te-fp:cfp-configurations dynamic-device-mapping
cisco-iosxr-nc-xxx:cisco-iosxr-nc-xxx
admin@ncs% delete te:cfp-configurations dynamic-device-mapping cisco-iosxr-
nc-xxx:cisco-iosxr-nc-xxx
admin@ncs% commit
```

3. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz
```

4. 他のサービスで使用されていない Netconf NED を削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
```

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

IETF-TE-IOSXE-CLI のアンインストール

このフレーバーをアンインストールすると、システムは IETF-TE IOSXR CLI に戻ります。IETF-TE IOSXR CLI をアンインストールする方法については、「[IETF-TE-IOSXR-CLI のアンインストール](#)」を参照してください。

IETF-TE-IOSXE-CLI をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す](#)」を参照してください。

2. 動的マッピングを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-rsvp-te-fp:cfp-configurations dynamic-device-mapping
cisco-ios-cli-xxx:cisco-ios-cli-xxx
admin@ncs% commit
```

3. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz /opt/ncs/packages/ncs-xxx-rsvp-te-multi-vendors-EXAMPLE-xxx.tar.gz
```

4. 他のサービスで使用されていない場合は、XE CLI NED を削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz /opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
      [ OK ]
```

コア機能パックのアンインストール

このセクションの情報を使用して、IOSXE CLI/IOSXR NC/IOSXR-CLI で SR-TE CFP をアンインストールします。

SR-TE CFP IOSXR CLI をアンインストールすると、Cisco T-SDN FP バンドルもアンインストールされます。

SR-TE CFP-IOSXE CLI のアンインストール

このフレーバーをアンインストールすると、システムは SR-TE CFP-IOSXR CLI に戻ります。

SR-TE CFP-IOSXE CLI をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO** インスタンスでの NCS 構成ファイルへの変更を元に戻す」を参照してください。

2. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-sr-te-cfp:cfp-configurations dynamic-device-mapping
cisco-ios-cli-xxx:cisco-ios-cli-xxx
admin@ncs% commit
```

3. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz
```

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-ios-xxx.tar.gz
```

4. すべてのサービスとデバイスを削除します。

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

SR-TE CFP-IOSXR NC のアンインストール

SR-TE CFP-IOSXR NC をアンインストールすると、システムは SR-TE CFP-IOSXR CLI に戻ります。

SR-TE CFP-IOSXR NC をアンインストールするには：

1. **ncs.config** ファイルを元に戻します。詳細については、「**単一の NSO インスタンスでの NCS 構成ファイルへの変更を元に戻す**」を参照してください。

2. フレーバーの動的マッピングを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete cisco-sr-te-cfp:cfp-configurations dynamic-device-mapping
cisco-iosxr-nc-xxx:cisco-iosxr-nc-xxx
admin@ncs% commit
```

3. **/var/opt/ncs/packages** にある以下のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-sr-te-multi-vendors-xxx.tar.gz
```

4. マルチベンダーパッケージと共にインストールされた Netconf NED を削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-iosxr-nc-xxx.tar.gz
```

5. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

6. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs restart
Restarting ncs (via systemctl):
[ OK ]
```

SR-TE CFP-IOSXR CLI のアンインストール

SR-TE CFP-IOSXR CLI は、Cisco NSO T-SDN FP バンドルの主要コンポーネントです。SR-TE CFP-IOSXR CLI および関連パッケージをシステムからアンインストールすると、Cisco NSO T-SDN FP バンドルが削除されます。このアンインストールは、**sudo** 権限を持ち、**ncsadmin** ユーザーグループに属しているユーザーのみが実行できます。

T-SDN FP バンドルをアンインストールする前に、まず T-SDN FP バンドルサービスとデバイスをシステムから削除する必要があります。サービスに対してゾンビサービスが実行されていないことを確認します。サービスを削除する方法の詳細については、『Cisco NSO T-SDN FP Bundle User Guide』の「Deleting Services」の章を参照してください。

注： T-SDN FP バンドルをアンインストールした後、スタンドアロンプレーヤーとしてサンプル機能パック（サービス）を引き続き使用する場合は、T-SDN FP バンドルの共通パッケージまたは CLI NED を削除しないでください。

Cisco NSO T-SDN 機能パックバンドルをアンインストールするには、次の手順を実行します。

1. **ncs.conf** ファイルを元に戻します。詳細については、「**単一の NSO** インスタンスでの NCS 構成ファイルへの変更を元に戻す」を参照してください。

2. プラン通知を削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% delete services plan-notifications subscription sr-policy-notif
admin@ncs% delete services plan-notifications subscription sr-odn-notif
admin@ncs% delete plan-path-for-notification /cisco-sr-te-cfp:sr-te/cisco-sr-
te-cfp-sr-odn:odn/odn-template-plan
admin@ncs% delete plan-path-for-notification /cisco-sr-te-cfp:sr-te/cisco-sr-
te-cfp-sr-policies:policies/policy-plan
admin@ncs% commit
```

3. ステータスコードを削除します。

```
$ /opt/ncs/current/bin/ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete status-code-cfp SR
admin@ncs% delete status-codes core-function-pack SR
admin@ncs% commit
```

4. キッカーを削除します。

```
$ ncs_cli -u admin
admin@ncs> configure
admin@ncs% unhide debug
admin@ncs% delete kickers data-kicker sr-te-cfp-configuration-kicker
admin@ncs% delete kickers data-kicker sr-te-odn-internal-plan-kicker
admin@ncs% delete kickers data-kicker sr-te-policy-internal-plan-kicker
admin@ncs% commit
```

5. **/var/opt/ncs/packages** 内のパッケージのリンクを解除し、**/opt/ncs/packages/** ディレクトリからパッケージを削除します。

```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-core-fp-common-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-
xxx.tar.gz /opt/ncs/packages/ncs-xxx-core-fp-plan-notif-generator-xxx.tar.gz
```



```
sudo rm -f /var/opt/ncs/packages/ncs-xxx-custom-template-utils-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-custom-template-utils-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-internal-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-sr-te-cfp-internal-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-iosxr-xxx.tar.gz
sudo rm -f /var/opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-xxx.tar.gz
/opt/ncs/packages/ncs-xxx-cisco-tdsn-core-fp-common-xxx.tar.gz
```

6. NSO を停止します。

```
### Make sure user delete all services and devices from TSDN
### Make sure there are no zombie services by running the command: show
zombies
sudo /etc/init.d/ncs stop
```

7. パッケージの再読み込みで NSO を再起動します。

```
$ sudo NCS_RELOAD_PACKAGES=force
Restarting ncs (via systemctl):
[OK]
```

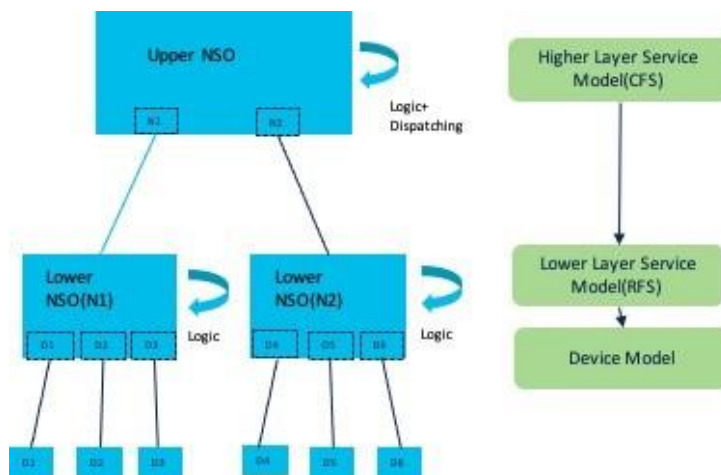
Cisco NSO T-SDN FP バンドルのアンインストールはこれで完了です。

LSA モデルでの T-SDN FP バンドルのインストールとアンインストール

LSA モデルは、T-SDN FP バンドルを 2 つの部分に分割します。1 つの上位レベル顧客向けサービス (CFS) 部分と 2 つの下位レベルリソース向けサービス (RFS) 部分です。

下位ノード層 (RFS ノード) は、デバイスツリーにマウントされたマネージドデバイスで構成されます。下位ノードは、ネットワーク内のデバイスに設定をプッシュします。これらの下位ノードは、デバイスツリーの上位ノードにデバイスとして追加されます。上位ノードには、追加された下位デバイスノード以外のデバイスは含まれません。上位ノードと下位ノードは、Netconf を使用して相互に通信します。

次の図は、LSA 導入モデルを使用した T-SDN FP バンドルのインストールを示しています。



上位ノードは、T-SDN FP バンドルの CFS パッケージ、RFS NED、およびその他の共通パッケージと対応する NED (core-fp-common および core-fp-common-ned など) で構成されます。

下位ノードには、T-SDN FP バンドルの RFS パッケージ、その他の一般的なパッケージ、および必要なデバイス NED があります。

LSA モデルでは、上位ノードと (CFP ごとの) 各下位ノードを個別の NSO インスタンスにインストールします。上位ノードは、両方の下位ノードに共通する 1 つのノードです。先に CFP の下位ノードをインストールし、次に上位ノードをインストールすることを推奨します。これは、下位ノードを上位ノードにデバイスとして追加する必要があるためです。

下位ノードへの T-SDN FP バンドルのインストール

パッケージカテゴリとパッケージ - 下位ノード

次の表は、下位ノードで抽出されたパッケージカテゴリとパッケージを示しています。

RFS ノードパッケージ	
パッケージカテゴリ	パッケージ
T-SDN FP バンドルパッケージ	ncs-5.7.6-cisco-sr-te-cfp-internal-4.1.0.tar.gz ncs-5.7.6-cisco-tdsn-core-fp-common-4.1.0.tar.gz ncs-5.7.6-sr-te-multi-vendors-4.1.0.tar.gz ncs-5.7-core-fp-delete-tag-service-1.0.5.tar.gz
T-SDN FP バンドル共通パッケージ	ncs-5.7-core-fp-plan-notif-generator-1.0.6.tar.gz ncs-5.7-core-fp-common-1.30.0.tar.gz ncs-5.7.2-custom-template-utils-2.0.9.tar.gz ncs-5.7.2.1-lsa-utils-1.0.1.tar.gz
LSA NED パッケージ	ncs-5.7.5-cisco-iosxr-7.40.1.tar.gz ncs-5.7-cisco-iosxr_netconf-7.315.tar.gz ncs-5.7-cisco-iosxr_netconf-7.3.2.tar.gz ncs-5.7.2-cisco-iosxr_netconf-7.4.2.tar.gz ncs-5.7.3-cisco-iosxr_netconf-7.5.2.tar.gz ncs-5.7.4-cisco-iosxr-nc-7.6.2.19l.tar.gz ncs-5.7.5-cisco-iosxr_netconf-7.7.1.tar.gz ncs-5.7-cisco-ios-6.77.9.tar.gz
サンプルパッケージ	ncs-5.7.6-cisco-flat-L2vpn-fp-internal-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-flat-L3vpn-fp-internal-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-rsvp-te-fp-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-flat-l2vpn-multi-vendors-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-flat-l3vpn-multi-vendors-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-rsvp-te-multi-vendors-EXAMPLE-4.1.0.tar.gz

下位ノードでの NCS 構成ファイルの変更

`/etc/ncs/ncs.conf` ファイルをバックアップしてから、CFP の各下位ノードのファイルを変更します。

注： CFP のアンインストール中に、バックアップファイルを使用して構成を復元します。

下位ノードで NCS 構成ファイルを変更するには、次の手順を実行します。

1. (オプション) CLI、webui、および netconf northbound パラメータ用の SSH ポートを設定します。これらのパラメータについて、必要に応じて SSH ポートの設定を有効または無効にすることができます。デフォルトでは、これらのパラメータの SSH ポートの設定は無効になっています。これらのパラメータの詳細については、NSO のドキュメントを参照してください。

次に、SSH ポートの設定を有効化する方法を示します。要件に応じてポート番号を指定します。

CLI の SSH ポート

```
<cli>
  <enabled>true</enabled>
  <!-- Use the builtin SSH server -->
  <ssh>
    <enabled>true</enabled>
    <ip>0.0.0.0</ip>
    <port>${North_Bound_CLI_SSH_Port}</port>
  </ssh>
```

webui

TCP または SSL のいずれかで webui を有効にすることができます。

```
<webui>
  <enabled>true</enabled>
  <transport>
    <tcp>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>${North_Bound_Web_UI_Port}</port>
    </tcp>

    <ssl>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>${SSL_port}</port>
```

```
    <key-file>${NCS_CONFIG_DIR}/ssl/cert/host.key</key-file>
    <cert-file>${NCS_CONFIG_DIR}/ssl/cert/host.cert</cert-file>
  </ssl>
</transport>
```

netconf northbound

```
<netconf-north-bound>
  <enabled>>true</enabled>
  <transport>
    <ssh>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>${Netconf_North_Bound_port}</port>
    </ssh>
```

2. 通知/イベントストリームに以下を追加します。

ディスパッチマップ

```
<stream>
  <name>dispatch-map-events</name>
  <description>Device addition/removal on RFS notified to
  CFS</description>
  <replay-support>true</replay-support>
  <builtin-replay-store>
    <enabled>true</enabled>
    <dir>${NCS_RUN_DIR}/state</dir>
    <max-size>S10M</max-size>
    <max-files>50</max-files>
  </builtin-replay-store>
</stream>
```

カスタム テンプレート イベント

```
<stream>
  <name>custom-template-events</name>
  <description>Custom Template updates on RFS notified to
  CFS</description>
  <replay-support>true</replay-support>
  <builtin-replay-store>
    <enabled>true</enabled>
    <dir>${NCS_RUN_DIR}/state</dir>
    <max-size>S10M</max-size>
    <max-files>50</max-files>
```

```

    </builtin-replay-store>
</stream>

```

3. /log で以下を設定します。

netconf-trace-log

```

<netconf-trace-log>
  <enabled>true</enabled>
  <filename>${NCS_LOG_DIR}/netconf-north.trace</filename>
  <format>pretty</format>
</netconf-trace-log>

```

webui-browser-log

```

<webui-browser-log>
  <enabled>true</enabled>
  <filename>${NCS_LOG_DIR}/webui-browser.log</filename>
</webui-browser-log>

```

4. SSH アルゴリズムを設定します。

```

<ssh>
  <algorithms>
    <kex>diffie-hellman-group14-sha1</kex>
    <mac>hmac-sha2-512,hmac-sha2-256,hmac-sha1</mac>
    <encryption>aes128-ctr,aes192-ctr,aes256-ctr</encryption>
  </algorithms>
</ssh>

```

5. <hide-group> 情報をファイルに追加します。

```

<hide-group>
  <name>tsdn</name>
</hide-group>
<hide-group>
  <name>debug</name>
</hide-group>

```

6. suppress-commit-message-context で、<commit-message> パラメータを追加して false に設定します。

```

<commit-message>>false</commit-message>

```

7. Java-API パラメータを設定します。

```

<japi>
  <new-session-timeout>PT3600S</new-session-timeout>
  <query-timeout>PT3600S</query-timeout>

```

```
<connect-timeout>PT3600S</connect-timeout>
<event-reply-timeout>PT3600S</event-reply-timeout>
</japi>
```

下位ノードへの T-SDN FP バンドルのインストール

下位ノードに T-SDN FP バンドルをインストールするには：

1. 「下位ノードでの NCS 構成ファイルの変更」セクションに記載されているタスクをすでに実行したことを確認してください。
2. ncsadmin ユーザーグループのメンバーでもある sudo ユーザーとしてホストマシンにログインします。
3. 署名付き bin ファイル **nso-<version>-tsdn-<version>** をシスコの Web サイトから入手してダウンロードし、下位ノードにコピーします。
たとえば、**nso-5.7.6-tsdn-4.1.0.signed.bin** ファイルを入手してダウンロードします。
4. bin ファイルの内容を現在のディレクトリに抽出します。フォルダがすでに存在する場合は、既存のフォルダのバックアップを作成します。

```
$ sh nso-<version>-tsdn-<version>.signed.bin
```

これにより、製品の真正性が確認されます。ただし、ネットワーク接続の問題が発生した場合は、次のコマンドを実行してこの検証をスキップします。

```
$ sh nso-<version>-tsdn-<version>.signed.bin --skip-verification
```

5. インストーラ tar.gz ファイルを現在のディレクトリに展開し、T-SDN FP バンドルパッケージを抽出します。フォルダがすでに存在する場合は、既存のフォルダのバックアップを作成してください。

```
$tar -xf nso-<version>-tsdn-<version>.tar.gz
```

6. 下位ノード パッケージ ディレクトリに移動し、現在のディレクトリを次のように変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn-lsa/tsdn-lsa-rfs/core-fp-packages
```

7. T-SDN FP バンドルパッケージを **/opt/ncs/packages/** ディレクトリにコピーし、**/var/opt/ncs/packages** からシンボリックリンクを作成します。

```
sudo cp ncs-<version>-cisco-tsdn-core-fp-common-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-cisco-tsdn-core-fp-common-<version>.tar.gz
```

```
sudo cp ncs-<version>-cisco-sr-te-cfp-internal-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-<version>.tar.gz
```

```
sudo cp ncs-<version>-sr-te-multi-vendors-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-sr-te-multi-vendors-<version>.tar.gz
```

```
sudo cp ncs-<version>-cisco-iosxr-<version>.tar.gz /opt/ncs/packages/ncs-
<version>-cisco-ios-<version>.tar.gz
```

```
sudo cp ncs-<version>-cisco-ios-<version>.tar.gz /opt/ncs/packages/ncs-
<version>-cisco-ios-<version>.tar.gz
```

```
sudo cp ncs-<version>-custom-template-utils-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-custom-template-utils-<version>.tar.gz
```

```
sudo cp ncs-<version>-core-fp-common-<version>.tar.gz /opt/ncs/packages/ncs-  
<version>-core-fp-common-<version>.tar.gz  
sudo cp ncs-<version>-core-fp-plan-notif-generator-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-core-fp-plan-notif-generator-<version>.tar.gz  
sudo cp ncs-<version>-core-fp-delete-tag-service-<version>.tar.gz  
/opt/ncs/packages/ncs-<NSO-Version>-core-fp-delete-tag-service-  
<version>.tar.gz  
sudo cp ncs-<version>-cisco-iosxr_netconf-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-iosxr_netconf-<version>.tar.gz  
sudo cp ncs-<version>-cisco-iosxr-nc-<version>.tar.gz /opt/ncs/packages/ncs-  
<version>-cisco-iosxr-nc-<version>.tar.gz  
sudo cp ncs-<version>-lsa-utils-<version>.tar.gz /opt/ncs/packages/ncs-  
<version>-lsa-utils-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-  
common-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-  
internal-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-sr-te-multi-vendors-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-sr-te-multi-vendors-  
<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<NSO-Version>-cisco-iosxr-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>-cisco-iosxr-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-ios-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>-cisco-ios-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-core-fp-common-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>-core-fp-common-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-core-fp-plan-notif-generator-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-core-fp-plan-notif-  
generator-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-core-fp-delete-tag-service-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-core-fp-delete-tag-  
service-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-iosxr_netconf-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-iosxr_netconf-  
<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-iosxr-nc-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>-cisco-iosxr-nc-<version>.tar.gz  
sudo ln -s /opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz
```


8. example-packages ディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn-lsa/tsdn-lsa-rfs/example-packages
```

9. 必要なサンプル機能パックのパッケージを /opt/ncs/packages/ ディレクトリにコピーし、 /var/opt/ncs/packages からシンボリックリンクを作成します。

```
sudo cp ncs-<version>-cisco-flat-L2vpn-fp-internal-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-internal-EXAMPLE-  
<version>.tar.gz
```

```
sudo cp ncs-<version>-cisco-flat-L3vpn-fp-internal-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-internal-EXAMPLE-  
<version>.tar.gz
```

```
sudo cp ncs-<version>-cisco-rsvp-te-fp-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-rsvp-te-fp-EXAMPLE-<version>.tar.gz
```

```
sudo cp ncs-<version>-flat-l2vpn-multi-vendors-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-flat-l2vpn-multi-vendors-EXAMPLE-  
<version>.tar.gz
```

```
sudo cp ncs-<version>-flat-l3vpn-multi-vendors-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-flat-l3vpn-multi-vendors-EXAMPLE-  
<version>.tar.gz
```

```
sudo cp ncs-<version>-rsvp-te-multi-vendors-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-rsvp-te-multi-vendors-EXAMPLE-  
<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-internal-  
EXAMPLE-<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-flat-  
L2vpn-fp-internal-EXAMPLE-<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-internal-  
EXAMPLE-<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-flat-  
L3vpn-fp-internal-EXAMPLE-<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-rsvp-te-fp-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-rsvp-te-fp-  
EXAMPLE-<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-flat-l2vpn-multi-vendors-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-flat-l2vpn-multi-  
vendors-EXAMPLE-<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-flat-l3vpn-multi-vendors-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-flat-l3vpn-multi-  
vendors-EXAMPLE-<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-rsvp-te-multi-vendors-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-rsvp-te-multi-vendors-  
EXAMPLE-<version>.tar.gz
```

10.package-reload で NSO を再起動します。

```
$ sudo /etc/init.d/ncs restart-with-package-reload
```

T-SDN FP バンドルが下位ノードにインストールされました。

下位ノードでのインストール後のタスクの実行

次の手順を実行します。

1. 次のブートストラップデータを読み込みます。

```
cd nso-<version>-tsdn-<version>/tsdn-lsa/tsdn-lsa-rfs/bootstrap-data
ncs_cli -u <user>
configure
unhide debug
load merge SR-status-codes.xml
load merge SR-internal-plan-monitor.xml
load merge RSVP-TE-status-codes.xml
load merge RSVP-TE-internal-plan-monitor.xml
load merge rfs-custom-template-settings.xml
load merge L3VPN-status-codes.xml
load merge L3VPN-internal-plan-monitor.xml
load merge L2VPN-status-codes.xml
load merge L2VPN-internal-plan-monitor.xml
load merge dispatch-map-settings.xml
load merge commit-queue-settings.xml
load merge bootstrap-autopopulate-dispatch.xml
commit
```

2. NACM ルールを設定します。次の例では、ncsadmin ユーザー ID は admin です。

```
% set nacm groups group ncsadmin user-name admin
% commit
Commit complete.
```

3. ローカル ncsadmin ユーザーを CFP の cfp-local-user として設定し、設定をプッシュするユーザーを識別します。

```
% configure
% set cfp-local-user admin
% commit
```

4. SSH アルゴリズムの公開キーを設定します。

```
# Global settings method
-----
% show devices global-settings ssh-algorithms public-key
public-key [ ssh-ed25519 ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-
nistp521 rsa-sha2-512 rsa-sha2-256 ];
```

```

% set devices global-settings ssh-algorithms public-key [ ssh-ed25519 ecdsa-
sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 rsa-sha2-512 rsa-sha2-
256 ssh-rsa ]
% commit

% show device global-settings ssh-algorithms public-key
public-key [ ssh-ed25519 ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-
nistp521 rsa-sha2-512 rsa-sha2-256 ssh-rsa ];
# Device-specific method
-----
% show devices device PE1 ssh-algorithms public-key
No entries found.

% set devices device <DEVICE_NAME> ssh-algorithms public-key [ ssh-rsa ]
% commit

% show device device <DEVICE_NAME> ssh-algorithms public-key
public-key [ ssh-rsa ];

```

5. タイムアウトのグローバル設定を追加します。

```

configure
set devices global-settings connect-timeout 300
set devices global-settings read-timeout 300
set devices global-settings write-timeout 300

```

下位ノードでのインストールの確認

次の手順を実行します。

1. パッケージがインストールされており、ステータスが UP であることを確認します。

```
admin@ncs> show packages package oper-status | tab
```

2. パッケージ情報を確認します。

```
admin@ncs> show packages package package-version | select build-info ncs
version | select build-info file | select build-info package sha1 | select
oper-status error-info | select oper-status up | tab
```

3. ブートストラップ設定を確認します。

```

% show devices global-settings commit-queue
enabled-by-default false;
async;
atomic                false;
retry-attempts        0;
retry-timeout         30;
error-option          stop-on-error;

```

```
% show status-code-cfp
status-code-cfp L2VPN;
status-code-cfp L3VPN;
status-code-cfp RSVP-TE;
status-code-cfp SR;

% show rfs-monitor-path
rfs-monitor-path /cisco-flat-L2vpn-fp-internal-local-site:flat-L2vpn-
internal-local-site/cisco-flat-L2vpn-fp-internal-local-site:flat-L2vpn-plan;
rfs-monitor-path /cisco-flat-L2vpn-fp-internal-remote-site:flat-L2vpn-
internal-remote-site/cisco-flat-L2vpn-fp-internal-remote-site:flat-L2vpn-
plan;
rfs-monitor-path /cisco-flat-L2vpn-fp-internal-site:flat-L2vpn-internal-
site/cisco-flat-L2vpn-fp-internal-site:flat-L2vpn-plan;
rfs-monitor-path /cisco-flat-L3vpn-fp-internal:flat-L3vpn-internal/cisco-
flat-L3vpn-fp-internal:flat-L3vpn-plan;
rfs-monitor-path /cisco-rsvp-te-fp:rsvp-te/tunnel-te-plan;
rfs-monitor-path /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-odn-
internal:odn/cisco-sr-te-cfp-sr-odn-internal:odn-template-plan;
rfs-monitor-path /cisco-sr-te-cfp-internal:sr-te/cisco-sr-te-cfp-sr-policies-
internal:policies/cisco-sr-te-cfp-sr-policies-internal:policy-plan;

% show auto-populate-dispatch-map
auto-populate-dispatch-map true;

% show ct-event-stream-enabled
ct-event-stream-enabled true;

% show status-codes | nomore
core-function-pack L2VPN {
  status-code-enum-path cisco-tdsn-core-fp-
common/python/cisco_tsdn_core_fp_common/status_codes/flat_L2vpn_status_codes;
  status-code 301 {
    reason          "Device unreachable";
    category        device;
    severity        ERROR;
    recommended-actions "Check device connectivity from NSO and perform
recovery steps.";
  }
  status-code 302 {
    reason          "Device out of sync";
    category        device;
    severity        ERROR;
  }
}
```

```
        recommended-actions "Check sync between device and NSO, and perform  
recovery steps.";  
    }  
...  
}
```

4. NACM ルールを確認します。

```
% show nacm  
read-default      deny;  
write-default     deny;  
exec-default      deny;  
groups {  
  group ncsadmin {  
    user-name [ admin private ];  
  }  
  group ncsoper {  
    user-name [ public ];  
  }  
}  
...  
}
```

5. cfp-local-user を確認します。

```
% show cfp-local-user  
cfp-local-user admin;
```

6. 下位ノードに LSA ロールが設定されていることを確認します。

```
% show lsa role  
role lower-layer;
```

上位ノードへの T-SDN FP バンドルのインストール

パッケージカテゴリとパッケージ - 上位ノード

次の表は、上位ノードで抽出されたパッケージカテゴリとパッケージを示しています。

CFS ノードパッケージ	
パッケージカテゴリ	パッケージ
T-SDN FP バンドルパッケージ	ncs-5.7.6-cisco-sr-te-cfp-4.1.0.tar.gz ncs-5.7.6-cisco-tdsn-core-fp-common-4.1.0.tar.gz
T-SDN FP バンドル共通パッケージ	ncs-5.7-core-fp-plan-notif-generator-1.0.6.tar.gz ncs-5.7.2-custom-template-utils-2.0.9.tar.gz ncs-5.7-core-fp-common-1.30.0.tar.gz ncs-5.7.6-lsa-utils-ned-1.0.tar.gz
LSA NED パッケージ	ncs-5.7.6-cisco-nso-nc-5.7.tar.gz ncs-5.7.6-cisco-flat-L2vpn-fp-internal-ned-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-flat-L3vpn-fp-internal-ned-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-rsvp-te-fp-internal-ned-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-lsa-utils-ned-1.0.tar.gz ncs-5.7.6-custom-template-utils-ned-1.0.tar.gz ncs-5.7.2.1-lsa-utils-1.0.1.tar.gz ncs-5.7.6-core-fp-common-ned-1.0.tar.gz ncs-5.7.6-cisco-sr-te-cfp-internal-ned-4.1.0.tar.gz
サンプルパッケージ	ncs-5.7.6-cisco-aa-service-assurance-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-flat-L2vpn-fp-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-cisco-flat-L3vpn-fp-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-ietf-l2vpn-nm-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-ietf-l3vpn-nm-EXAMPLE-4.1.0.tar.gz ncs-5.7.6-ietf-te-fp-EXAMPLE-4.1.0.tar.gz ncs-5.7.1-resource-manager-3.5.7.tar.gz

上位ノードでの NCS 構成ファイルの変更

`/etc/ncs/ncs.conf` ファイルをバックアップしてから、上位ノードのファイルを変更します。

注： CFP のアンインストール中に、バックアップファイルを使用して構成を復元します。

上位ノードで NCS 構成ファイルを変更するには、次の手順を実行します。

1. CLI、webui、および netconf northbound パラメータの場合、SSH ポートの設定はオプションです。これらのパラメータについて、必要に応じて SSH ポートの設定を有効または無効にすること

ができます。デフォルトでは、これらのパラメータの SSH ポートの設定は無効になっています。これらのパラメータの詳細については、**NSO のドキュメント**を参照してください。次に、SSH ポートの設定を有効化する方法を示します。要件に応じてポート番号を指定します。

CLI の SSH ポート

```
<cli>
  <enabled>true</enabled>
  <!-- Use the builtin SSH server -->
  <ssh>
    <enabled>true</enabled>
    <ip>0.0.0.0</ip>
    <port>${North_Bound_CLI_SSH_Port}</port>
  </ssh>
```

webui

TCP または SSL のいずれかで webui を有効にすることができます。

```
<webui>
  <enabled>true</enabled>
  <transport>
    <tcp>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>${North_Bound_Web_UI_Port}</port>
    </tcp>

    <ssl>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
      <port>${SSL_port}</port>
      <key-file>${NCS_CONFIG_DIR}/ssl/cert/host.key</key-file>
      <cert-file>${NCS_CONFIG_DIR}/ssl/cert/host.cert</cert-file>
    </ssl>
  </transport>
```

netconf northbound

```
<netconf-north-bound>
  <enabled>true</enabled>
  <transport>
    <ssh>
      <enabled>true</enabled>
      <ip>0.0.0.0</ip>
```

```
<port>${Netconf_North_Bound_port}</port>  
</ssh>
```

2. 次に示すように、ストリーム service-state-changes を追加します。

```
<notifications>  
<event-streams>  
  <stream>  
    <name>service-state-changes< /name >  
    <description>Plan state transitions according to  
tailf-ncs-plan.yang< /description >  
    <replay-support> true < /replay-support >  
    < builtin -replay-store>  
      <enabled> true < /enabled >  
      < dir >${NCS_RUN_DIR}/state < /dir >  
      <max-size>S10M< /max-size >  
      <max-files>50< /max-files >  
    < /builtin-replay-store >  
  < /stream >
```

3. AA がインストールされている場合は、AA 通知ストリームを追加して、AA 設定変更通知を生成します。

```
<stream>  
  <name>service-aa-changes< /name >  
  <description>Notifications relating to the service aa configuration  
change< /description >  
  <replay-support> true < /replay-support >  
  < builtin -replay-store>  
    <enabled> true < /enabled >  
    < dir >${NCS_RUN_DIR} /state < /dir >  
    <max-size>S10M< /max-size >  
    <max-files>50< /max-files >  
  < /builtin-replay-store >  
< /stream >
```

4. <hide-group> 情報をファイルに追加します。

```
<hide-group>  
  <name>tsdn</name>  
</hide-group>  
<hide-group>  
  <name>debug</name>  
</hide-group>
```



```
<hide-group>
  <name>fastmap-private< /name >
< /hide-group >
```

5. <python-vm> 下でパラメータ <start-timeout> を追加または更新します。

```
<python-vm>
  <start-timeout>PT300S< /start-timeout >
</python-vm >
```

上位ノードへの T-SDN FP バンドルのインストール

上位ノードに T-SDN FP バンドルをインストールするには：

1. 下位ノードに T-SDN FP バンドルがインストールされていることを確認してください。これは、下位ノードのデバイスを上位ノードにデバイスとして追加し、下位ノードを設定して、上位ノードに同期する必要があるためです。詳細については、「[下位ノードへの T-SDN FP バンドルのインストール](#)」を参照してください。
2. 「[上位ノードでの NCS 構成ファイルの変更](#)」セクションに記載されているタスクをすでに実行したことを確認してください。
3. ncsadmin ユーザーグループのメンバーでもある sudo ユーザーとしてホストマシンにログインします。
4. 署名付き bin ファイル **nso-<version>-tsdn-<version>** をシスコの Web サイトから入手してダウンロードし、上位ノードにコピーします。
たとえば、**nso-5.7.6-tsdn-4.1.0.signed.bin** ファイルを入手してダウンロードします。

5. bin ファイルの内容を現在のディレクトリに抽出します。フォルダがすでに存在する場合は、既存のフォルダのバックアップを作成します。

```
$ sh nso-<version>-tsdn-<version>.signed.bin
```

これにより、製品の真正性が確認されます。ただし、ネットワーク接続の問題が発生した場合は、次のコマンドを実行してこの検証をスキップします。

```
$ sh nso-<version>-tsdn-<version>.signed.bin --skip-verification
```

6. インストーラ tar.gz ファイルを現在のディレクトリに展開し、T-SDN FP バンドルパッケージを抽出します。フォルダがすでに存在する場合は、既存のフォルダのバックアップを作成してください。

```
$tar -xf nso-<version>-tsdn-<version>.tar.gz
```

7. 上位ノード パッケージ ディレクトリに移動し、現在のディレクトリを次のように変更します。

```
$ cd nso-<version>-tsdn-<version>/tsdn-lsa/tsdn-lsa-cfs/core-fp-packages
```

8. T-SDN FP バンドルパッケージを **/opt/ncs/packages/** ディレクトリにコピーし、**/var/opt/ncs/packages** からシンボリックリンクを作成します。

```
sudo cp ncs-<version>-lsa-utils-ned-<version>.tar.gz /opt/ncs/packages/ncs-<version>-lsa-utils-ned-<version>.tar.gz
```

```
sudo cp ncs-<version>-custom-template-utils-ned-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-custom-template-utils-ned-<version>.tar.gz
sudo cp ncs-<version>-core-fp-common-ned-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-core-fp-common-ned-<version>.tar.gz
sudo cp ncs-<version>-cisco-tdsn-core-fp-common-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-<version>.tar.gz
sudo cp ncs-<version>-cisco-sr-te-cfp-internal-ned-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-ned-<version>.tar.gz
sudo cp ncs-<version>-cisco-sr-te-cfp-<version>.tar.gz /opt/ncs/packages/ncs-
<version>-cisco-sr-te-cfp-<version>.tar.gz
sudo cp ncs-<version>-cisco-nso-nc-<version>.tar.gz /opt/ncs/packages/ncs-
<version>-cisco-nso-nc-<version>.tar.gz
sudo cp ncs-<version>-core-fp-common-<version>.tar.gz /opt/ncs/packages/ncs-
<version>-core-fp-common-<version>.tar.gz
sudo cp ncs-<version>-lsa-utils-<version>.tar.gz /opt/ncs/packages/ncs-
<version>-lsa-utils-<version>.tar.gz
sudo cp ncs-<version>-custom-template-utils-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-custom-template-utils-<version>.tar.gz
sudo cp ncs-<version>-core-fp-plan-notif-generator-<version>.tar.gz
/opt/ncs/packages/ncs-<version>-core-fp-plan-notif-generator-<version>.tar.gz

sudo ln -s /opt/ncs/packages/ncs-<version>-lsa-utils-ned-<version>.tar.gz
/var/opt/ncs/packages/ncs-<version>-lsa-utils-ned-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-custom-template-utils-ned-
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-custom-template-utils-
ned-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-core-fp-common-ned-
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-core-fp-common-ned-
<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-
common-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-ned-
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-
internal-ned-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-<version>.tar.gz
/var/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-cisco-nso-nc-<version>.tar.gz
/var/opt/ncs/packages/ncs-<version>-cisco-nso-nc-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-core-fp-common-<version>.tar.gz
/var/opt/ncs/packages/ncs-<version>-core-fp-common-<version>.tar.gz
sudo ln -s /opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz
/var/opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>-core-fp-plan-notif-generator-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>-core-fp-plan-notif-  
generator-<version>.tar.gz
```

9. example-packages ディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>/tsdn-lsa/tsdn-lsa-rfs/example-packages
```

10. 必要なサンプル機能パックのパッケージを **/opt/ncs/packages/** ディレクトリにコピーし、 **/var/opt/ncs/packages** からシンボリックリンクを作成します。

```
sudo cp ncs-<version>-resource-manager-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-resource-manager-<version>.tar.gz  
  
sudo cp ncs-<version>-ietf-te-fp-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-ietf-te-fp-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>-ietf-l3vpn-nm-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-ietf-l3vpn-nm-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>-ietf-l2vpn-nm-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-ietf-l2vpn-nm-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>cisco-rsvp-te-fp-internal-ned-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>cisco-rsvp-te-fp-internal-ned-EXAMPLE-  
<version>.tar.gz  
  
sudo cp ncs-<version>cisco-flat-L3vpn-fp-internal-ned-EXAMPLE-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>cisco-flat-L3vpn-fp-internal-  
ned-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>cisco-flat-L3vpn-fp-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>cisco-flat-L3vpn-fp-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>cisco-flat-L2vpn-fp-internal-ned-EXAMPLE-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>cisco-flat-L2vpn-fp-internal-  
ned-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>cisco-flat-L2vpn-fp-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>cisco-flat-L2vpn-fp-EXAMPLE-<version>.tar.gz  
  
sudo cp ncs-<version>cisco-aa-service-assurance-EXAMPLE-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>cisco-aa-service-assurance-EXAMPLE-  
<version>.tar.gz  
  
  
sudo ln -s /opt/ncs/packages/ncs-<version>-resource-manager-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>-resource-manager-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>ietf-te-fp-EXAMPLE-<version>.tar.gz  
/var/opt/ncs/packages/ncs-<version>ietf-te-fp-EXAMPLE-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>ietf-l3vpn-nm-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>ietf-l3vpn-nm-EXAMPLE-  
<version>.tar.gz
```

```
sudo ln -s /opt/ncs/packages/ncs-<version>ietf-l2vpn-nm-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>ietf-l2vpn-nm-EXAMPLE-  
<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>cisco-rsvp-te-fp-internal-ned-  
EXAMPLE-<version>.tar.gz /var/opt/ncs/packages/ncs-<version>cisco-rsvp-te-fp-  
internal-ned-EXAMPLE-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>cisco-flat-L3vpn-fp-internal-ned-  
EXAMPLE-<version>.tar.gz /var/opt/ncs/packages/ncs-<version>cisco-flat-L3vpn-  
fp-internal-ned-EXAMPLE-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>cisco-flat-L2vpn-fp-internal-ned-  
EXAMPLE-<version>.tar.gz /var/opt/ncs/packages/ncs-<version>cisco-flat-L2vpn-  
fp-internal-ned-EXAMPLE-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>cisco-flat-L3vpn-fp-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>cisco-flat-L3vpn-fp-  
EXAMPLE-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>cisco-flat-L2vpn-fp-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>cisco-flat-L2vpn-fp-  
EXAMPLE-<version>.tar.gz  
  
sudo ln -s /opt/ncs/packages/ncs-<version>cisco-aa-service-assurance-EXAMPLE-  
<version>.tar.gz /var/opt/ncs/packages/ncs-<version>cisco-aa-service-  
assurance-EXAMPLE-<version>.tar.gz
```

11.package-reload で NSO を再起動します。

```
$ sudo /etc/init.d/ncs restart-with-package-reload
```

T-SDN FP バンドルが上位ノードにインストールされました。

上位ノードでのインストール後のタスクの実行

次の手順を実行します。

1. NACM ルールを設定します。次の例では、ncsadmin ユーザー ID は admin です。

```
% set nacm groups group ncsadmin user-name admin  
% commit  
Commit complete.
```

2. ローカル ncsadmin ユーザーを CFP の cfp-local-user として設定し、設定をプッシュするユーザーを識別します。

```
% configure  
% set cfp-local-user admin  
% commit
```

3. 下位ノードの認証グループを設定します。プロンプトが表示されたら、ユーザーのパスワードを入力します。

```
% configure
```

```
admin@ncs% set devices authgroups group cnc-rfs-auth default-map remote-name  
admin remote-password  
(<AES256 encrypted string>): *****  
[ok]  
admin@ncs% commit  
Commit complete.
```

4. 下位ノードをデバイスとして上位ノードのデバイスツリーにオンボードします。

```
% configure  
set devices device rfs-1 address 172.28.112.228 port 2022 authgroup cnc-rfs-  
auth out-of-sync-commit-behaviour accept  
set devices device rfs-1 device-type netconf ned-id cisco-nso-nc-5.7  
set devices device rfs-1 use-lsa  
set devices device rfs-1 state admin-state unlocked  
set devices device rfs-1 connect-timeout 300 read-timeout 300 write-timeout  
300 connect-retries attempts 2 timeout 300  
admin@ncs% commit  
Commit complete
```

5. 次のコマンドを実行して、受信通知設定を確認します。

```
set devices device rfs-1 netconf-notifications received-notifications max-  
size 500000  
admin@ncs% commit  
Commit complete
```

注： RFS ノードと CFS ノードの両方のタイムスタンプは同じである必要があります。同
じ NTP サーバーを使用して時刻を同期することを推奨します。

6. 下位ノードごとに Netconf 通知サブスクリプションを設定します。すべての通知ストリームの下
位ノード名を追加する必要があります。次の例では、ユーザー admin は、ncsadmin グループに
含まれるローカルユーザーです。

```
set devices device rfs-1 netconf-notifications subscription rfs-cisco-custom-  
template-events stream custom-template-events local-user admin store-in-cdb  
true  
set devices device rfs-1 netconf-notifications subscription rfs-dispatch-map-  
update stream dispatch-map-update local-user admin store-in-cdb true  
set devices device rfs-1 netconf-notifications subscription rfs-kicker-events  
stream kicker-events local-user admin store-in-cdb true
```

注： 受信する可能性のある通知アラームはすべて無視してください。次の手順は、問題の
解決に役立ちます。

7. クラスタを設定します。下位ノードのオンボーディング時に使用したのと同じ認証グループ名
を使用してください。下位ノードの IP アドレスとポート番号を指定します。下位ノードのデバイ

ス名は一意である必要があります。クラスタの fetch ssh host キーは、クラスタを形成し、そのステータスを UP にするためにも不可欠です。

```
set cluster remote-node rfs-1 address 172.28.112.228 port 2022 authgroup cnc-  
rfs-auth username cisco
```

```
set cluster authgroup cnc-rfs-auth default-map remote-name cisco remote-  
password Cisco123\#
```

```
set cluster device-notifications disabled
```

```
set cluster commit-queue enabled
```

```
set cluster global-settings timeouts connect-timeout 300
```

```
commit
```

```
admin@ncs% request cluster remote-node rfs-1 ssh fetch-host-keys
```

```
result updated
```

```
fingerprint {
```

```
    algorithm ssh-ed25519
```

```
    value 83:a0:c2:62:85:dd:ee:bd:12:4f:a1:23:ae:47:d7:ca
```

```
}
```

```
admin@ncs% run show cluster
```

```
RECEIVED
NAME  NAME                STATUS  LAST EVENT                NOTIFICATIONS
-----
rfs-1 device-notifications up      0000-01-01T00:00:00-00:00  0
      ncs-events        up      2022-01-27T10:48:25.148393+00:00  630

REMOTE
NODE  ADDRESS          PORT  CHANNELS  LOCAL  REMOTE  STATUS  TRACE
-----
rfs-1 172.28.112.228  2022  -         cisco  cisco  up      disabled
```

8. 下位デバイスノードから同期し、デバイスツリーを確認します。

```
admin@ncs% request devices fetch-ssh-host-keys
```

```
fetch-result {
```

```
    device rfs-1
```

```
    result updated
```

```
    fingerprint {
```

```
        algorithm ssh-ed25519
```

```
        value ed:7b:1c:e4:77:80:ab:68:3b:17:40:69:68:9e:56:8d
```

```
    }
```

```
}
```

```
[ok]
```

```
admin@ncs% request devices sync-from
```

```

sync-result {
    device rfs-1
    result true
}
[ok]
admin@ncs% run show devices list
NAME      ADDRESS          DESCRIPTION  NED ID          ADMIN STATE
-----
rfs-1     172.28.112.228  -           cisco-nso-nc-5.7  unlocked
[ok]

```

9. ディスパッチマップを同期し、RFS ノードのオンボード済みデバイスが CFS ノードから認識可能かどうかを確認します。

```

% request devices lsa dispatch-map sync
success true
detail Dispatch Map Synced Successfully

% show device lsa dispatch-map
device rfs-1 {
    ned-id cisco-nso-nc-5.7:cisco-nso-nc-5.7;
}

```

10. 追加された下位ノードデバイスに対して Netconf 通知サブスクリプションが実行されていることを確認します。これは、上位ノードと下位ノードが接続され、通信が正常に実行されていることを示しています。

```

admin@ncs% run show devices device rfs-1 netconf-notifications subscription

```

NAME	STATUS	FAILURE REASON	ERROR INFO
rfs-cisco-custom-template-events	running	-	-
rfs-dispatch-map-update	running	-	-
rfs-kicker-events	running	-	-

インストールが完了したら、下位ノードのみからデバイスをオンボードします。

上位ノードでのインストールの確認

次の手順を実行します。

1. パッケージがインストールされており、ステータスが UP であることを確認します。

```
admin@ncs> show packages package oper-status | tab
```

2. パッケージの情報を確認します。

```

admin@ncs> show packages package package-version | select build-info ncs version
| select build-info file | select build-info package sha1 | select oper-status
error-info | select oper-status up | tab

```

3. NACM ルールを確認します。

```
% show nacm
read-default      deny;
write-default     deny;
exec-default      deny;
groups {
  group ncsadmin {
    user-name [ admin private ];
  }
  group ncsoper {
    user-name [ public ];
  }
}
...
```

4. cfp-local-user を確認します。

```
% show cfp-local-user
  cfp-local-user admin;
```

5. 上位ノードに LSA ロールが設定されていることを確認します。

```
% show lsa role
role upper-layer;
```

LSA モデルでの T-SDN FP バンドルのアンインストール

この章では、NSO T-SDN FP バンドルのアンインストール手順について説明します。T-SDN FP バンドルをアンインストールするには、まず上位ノードからバンドルをアンインストールし、その後下位ノードからアンインストールする必要があります。これは、下位ノードが上位ノードのデバイスツリーにデバイスとして追加されるためです。

T-SDN FP バンドルをアンインストールすると、CFP および関連パッケージがシステムから削除されます。このアンインストールプロセスは、sudo 権限を持ち、ncsadmin ユーザーグループに属しているユーザーのみが実行できます。アンインストール手順を実行する前に、関連するすべてのサービス、NSO からのデバイス、およびすべての Day-1 パッケージを削除する必要があります。クラスタに下位ノードが 1 つしかない場合は、そのクラスタを削除します。

注： 他の CFP をインストールしている場合は、共通パッケージを削除しないでください。

上位ノードからの T-SDN FP バンドルのアンインストール

上位ノードから T-SDN FP バンドルをアンインストールするには：

1. kicker 通知を削除します。

```
% unhide debug
% delete kickers notification-kicker remote-dispatch-map-update-notifications
% delete kickers notification-kicker rfs-custom-template-change-notification
```



```
% delete kickers notification-kicker tsdn-kicker-events-notifications
% commit
Commit complete.
```

2. クラスタを削除します。

```
% delete cluster remote-node rfs-1
% commit
Commit complete.

% delete cluster authgroup cluster-rfs-auth
% delete cluster device-notifications enabled
% commit
Commit complete.
```

3. 上位ノードのデバイスツリーから下位ノードのデバイスを削除します。

```
% delete devices device rfs-1
% commit
Commit complete.
```

4. ncs --stop コマンドを使用して NCS を停止します。

5. インストール中に作成した ncs.conf ファイルのバックアップを復元します。詳細については、「上位ノードでの NCS 構成ファイルの変更」を参照してください。

6. /var/opt/ncs/packages ディレクトリ内のパッケージをリンク解除して削除し、/opt/ncs/packages/ ディレクトリからパッケージを削除します。

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-aa-service-assurance-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-aa-service-assurance-EXAMPLE-<version>.tar.gz

sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-EXAMPLE-<version>.tar.gz

sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-EXAMPLE-<version>.tar.gz

sudo rm -rf /var/opt/ncs/packages/ncs-<version>-ietf-l2vpn-nm-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-ietf-l2vpn-nm-EXAMPLE-<version>.tar.gz

sudo rm -rf /var/opt/ncs/packages/ncs-<version>-ietf-l3vpn-nm-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-ietf-l3vpn-nm-EXAMPLE-<version>.tar.gz

sudo rm -rf /var/opt/ncs/packages/ncs-<version>-ietf-te-fp-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-ietf-te-fp-EXAMPLE-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-resource-manager-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-resource-manager-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-cs-sr-te-cfp-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-cs-sr-te-cfp-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-core-fp-common-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-core-fp-common-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-core-fp-plan-notif-generator-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-core-fp-plan-notif-  
generator-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-nso-nc-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-nso-nc-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-internal-  
ned-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-flat-  
L2vpn-fp-internal-ned-EXAMPLE-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-internal-  
ned-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-flat-  
L3vpn-fp-internal-ned-EXAMPLE-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-rsvp-te-fp-internal-  
ned-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-rsvp-te-  
fp-internal-ned-EXAMPLE-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-lsa-utils-ned-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-lsa-utils-ned-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-custom-template-utils-ned-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-custom-template-utils-ned-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-core-fp-common-ned-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-core-fp-common-ned-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-ned-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-  
ned-<version>.tar.gz
```

7. package-reload オプションで NSO を再起動します。

```
$ sudo /etc/init.d/ncs restart-with-package-reload
Restarting ncs (via systemctl):
[ OK ]
```

T-SDN FP バンドルが上位ノードからアンインストールされました。

下位ノードからの T-SDN FP バンドルのアンインストール

クラスタ内の各下位ノードに対して、このセクションで説明されている手順を実行します。

下位ノードから T-SDN FP バンドルをアンインストールするには：

1. 下位ノードのデバイスツリーからデバイスを削除します。

```
% delete devices
% commit
Commit complete.
```

2. ncs --stop コマンドを使用して NSO を停止します。

3. インストール中に作成した ncs.conf ファイルのバックアップを復元します。詳細については、「[下位ノードでの NCS 構成ファイルの変更](#)」を参照してください。

4. /var/opt/ncs/packages ディレクトリ内のパッケージのリンクを解除し、/opt/ncs/packages/ ディレクトリからパッケージを削除します。

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-internal-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-flat-L2vpn-fp-internal-EXAMPLE-<version>
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-internal-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-flat-L3vpn-fp-internal-EXAMPLE-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-rsvp-te-fp-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-rsvp-te-fp-EXAMPLE-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-flat-l2vpn-multi-vendors-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-flat-l2vpn-multi-vendors-EXAMPLE-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-flat-l3vpn-multi-vendors-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-flat-l3vpn-multi-vendors-EXAMPLE-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-rsvp-te-multi-vendors-EXAMPLE-<version>.tar.gz /opt/ncs/packages/ncs-<version>-rsvp-te-multi-vendors-EXAMPLE-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-sr-te-cfp-internal-<version>.tar.gz
```

```
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-tdsn-core-fp-common-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-sr-te-multi-vendors-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-sr-te-multi-vendors-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-custom-template-utils-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-core-fp-common-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-core-fp-common-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-core-fp-delete-tag-service-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-core-fp-delete-tag-service-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-core-fp-plan-notif-generator-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-core-fp-plan-notif-  
generator-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-lsa-utils-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-iosxr-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-iosxr-<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-iosxr_netconf-  
<version>.tar.gz /opt/ncs/packages/ncs-<version>-cisco-iosxr_netconf-  
<version>.tar.gz  
sudo rm -rf /var/opt/ncs/packages/ncs-<version>-cisco-ios-<version>.tar.gz  
/opt/ncs/packages/ncs-<version>-cisco-ios-<version>.tar.gz
```

5. package-reload オプションで NSO を再起動します。

```
$ sudo /etc/init.d/ncs restart-with-package-reload  
Restarting ncs (via systemctl):  
[ OK ]
```

アップグレード

このセクションには、Cisco NSO T-SDN FP バンドルの CFP をバージョン 3.0.0 からバージョン 4.1.0 にアップグレードする方法についての情報が含まれています。アップグレードを実行する前に、環境をバックアップすることをお勧めします。

前提条件

1. NSO 5.7.6 インストーラ bin ファイルと `nso-5.7.6-tdsn-4.1.0-signed.bin` ファイルを取得して `/home/user/` ディレクトリに配置します (セットアップに応じて user を置き換えます)。
2. `/opt/ncs/packages` の下に `upgrade-ned` ディレクトリを作成します。
3. 既存の NED パッケージ (IOSXE CLI および任意のデバイスからの他の NED) を `/opt/ncs/packages` ディレクトリから `/opt/ncs/packages/upgrade-ned` ディレクトリにコピーします。

Cisco T-SDN FP バンドルの CFP を v4.1.0 にアップグレードする

このセクションでは、T-SDN FP バンドルサービスの CFP を v4.1.0 にアップグレードする方法について説明します。

CFP を v4.1.0 にアップグレードするには、次のタスクを記載されている順序で実行します。

1. アップグレードの準備
2. T-SDN FP バンドルのアップグレード
3. T-SDN FP バンドルサービスのアップグレード

アップグレードの準備

アップグレードプロセスを開始する前に、次のことを実行します。

1. 現在の NCS パッケージをバックアップします。

注： データの損失を防ぐために、バックアップ tar ファイルのコピーを `/var/opt/ncs` ディレクトリの外に作成してください。

```
$ mkdir /home/admin/ncsBackup
$ cp /opt/ncs/packages/* /home/admin/ncsBackup/
$ echo "Backup current NCS"
$ sudo /opt/ncs/current/bin/ncs-backup --non-interactive
```

2. 古いステータスコードを削除します。

```
admin@ncs% unhide debug
```

```
admin@ncs% delete status-code-cfp
admin@ncs% delete status-codes
admin@ncs% commit
```

3. 次の手順に従って、lsa-utils ディスパッチマップ内の更新されたパスを参照するように、新しいモデルの lsa-utils ディスパッチマップを設定します。

- a. /var/opt/ncs/packages/ncs-5.7.2.1-lsa-utils-1.0.1.tar.gz から lsa-utils パッケージをコピーします。

- b. NSO をリロードします。

```
admin@ncs> request packages reload
```

- c. lsa-util ディスパッチマップを同期します。

```
admin@ncs> request devices lsa dispatch-map sync
```

T-SDN FP バンドルのアップグレード

T-SDN FP バンドルをアップグレードする前に、NSO を必要なバージョンにアップグレードします。

次の手順を実行します。

1. NSO を停止します。

```
sudo /etc/init.d/ncs stop
```

2. NSO をバージョン 5.7.6 にアップグレードします。NSO のアップグレード方法の詳細については、『NSO Installation Guide』を参照してください。

3. **ncs.conf** ファイルを編集します。詳細については、「[単一の NSO インスタンスでの NCS 構成ファイルの編集](#)」を参照してください。

4. **/opt/ncs/packages** ディレクトリにある古いパッケージを削除します。

```
cd /opt/ncs/packages/
sudo rm *.tar.gz
```

5. **/var/opt/ncs/packages** ディレクトリにあるパッケージの古いシンボリックリンクを削除します。

```
cd /var/opt/ncs/packages/
sudo rm -f *
```

6. ディレクトリを **nso-5.7.6-tdsn-4.1.0** に変更します。

7. T-SDN FP バンドルの bin ファイルのコンテンツを現在のディレクトリに抽出します。

```
$ sh nso-5.7.6-tdsn-4.1.0.signed.bin
```

これにより、製品の真正性が確認されます。ただし、ネットワーク接続の問題が発生した場合は、次のコマンドを実行してこの検証をスキップします。

```
$ sh nso-5.7.6-tdsn-4.1.0.signed.bin --skip-verification
```

8. T-SDN FP バンドルの **tar.gz** ファイルを現在のディレクトリに展開します。フォルダがすでに存在する場合は、既存のフォルダのバックアップを作成してください。

```
$ tar -xvf nso-5.7.6-TSDN-4.1.0.tar.gz
```

9. T-SDN FP バンドル 4.1.0 パッケージを TSDN tar ファイルから **/var/opt/ncs/packages** ディレクトリにコピーします。

```
sudo cp *.tar.gz /opt/ncs/packages/
```

10. **/opt/ncs/packages/upgrade-ned** ディレクトリにある NED (IOSXE CLI NED や任意のデバイスのその他のカスタマイズされた NED など) を NSO v5.7.6 で再コンパイルします。

11. 再コンパイルした NED を **/opt/ncs/packages/upgrade-ned** ディレクトリから **/var/opt/ncs/packages** ディレクトリにコピーします。また、シスコの担当者に連絡して、必要な再コンパイルされた NED パッケージを入手することもできます。

注： 古い NED は 以前にインストールされたバージョンの NSO でコンパイルされているため、NED はこの段階でも古い NED バージョンを表示します。

12. すべてのパッケージのソフトリンクを作成します。

```
cd /var/opt/ncs/packages
sudo ln -s /opt/ncs/packages/*.tar.gz
```

13. **ignore-initial-validation** フラグを設定し、次のように **package-reload** オプションを使用して **ncs** を再起動します。このフラグを設定しないと、アップグレードプロセスはエラーで失敗します。

- a. **start()** 関数に **ignore-initial-validation** フラグを追加します。

```
sudo vi /etc/init.d/ncs
...
start() {
    echo -n "$Starting $prog: "
    . $ncsdir/ncsrc
    NCS_CONFIG_DIR=${confdir}
    NCS_RUN_DIR=${rundir}
    NCS_LOG_DIR=${logdir}
    export NCS_CONFIG_DIR NCS_RUN_DIR NCS_LOG_DIR
    $ncs -cd ${rundir} ${heart} ${conf}
    RETVAL=$?
    Echo
    # [ $RETVAL = 0 ] && touch /var/lock/subsys/ncs
    return $RETVAL
}
...
Change it to
...
start() {
```

```

        echo -n $"Starting $prog: "
        . $ncsdir/ncsrc
        NCS_CONFIG_DIR=${confdir}
        NCS_RUN_DIR=${rundir}
        NCS_LOG_DIR=${logdir}
        export NCS_CONFIG_DIR NCS_RUN_DIR NCS_LOG_DIR
        $ncs -cd ${rundir} ${heart} ${conf} -ignore-initial-
validation
        RETVAL=$?
        Echo
        # [ $RETVAL = 0 ] && touch /var/lock/subsys/ncs
        return $RETVAL
    }
    ...

```

b. systemd をリロードします。

```
sudo systemctl daemon-reload
```

c. package-reload オプションで ncs を再起動します。

```
sudo NCS_RELOAD_PACKAGES=force /etc/init.d/ncs start
```

d. NSO が起動したら、start() 関数スクリプトを元の内容に戻します。

14. systemd をリロードします。

```
sudo systemctl daemon-reload
```

15. パッケージのステータスを確認します。

```
admin@ncs> show packages package oper-status
```

16. IOSXR CLI NED と IOSXE CLI NED を移行します。

```
admin@ncs% request devices device <Device_name> migrate new-ned-id cisco-
iosxr-7.40 no-networking
```

```
admin@ncs% request devices device <Device_name> migrate new-ned-id cisco-ios-
cli-6.77 no-networking
```

17. ssh-rsa を使用しているデバイス用に NSO デバイス SSH アルゴリズムを再定義します。

```
admin@ncs% set devices device PIOSXR-0 ssh-algorithms public-key ssh-rsa
```

```
admin@ncs% set devices device XECLI-0 ssh-algorithms public-key ssh-rsa
```

```
admin@ncs% commit
```

18. XE デバイスを同期して、新しい NED 機能をプルします。たとえば、IOSXE CLI NED 6.77 には、IOSXE CLI NED 6.74 よりも新しい機能が含まれている場合があります。これらの新機能により、Day0 デバイス構成から新しい NSO デバイス構成が導入される可能性があります。したがって、新しい変更をプルして、デバイスを NSO と同期させる必要があります。

注： デバイス構成ノースバウンドを NSO に同期する場合、新しい構成がリハーサルのみを含む Day0 構成であることを確認します。


```
> request devices device XECLI-0 sync-from dry-run
cli config {
    interface {
        GigabitEthernet 1 {
            ip {
                dhcp {
                    client {
                        client-id {
+                 ascii cisco-02bc.9833.b2f9-Gil;
                        }
                    }
                }
            }
        }
    }
    router {
        isis-container {
            isis 1 {
                router-id {
+                 Loopback 0;
                }
            }
        }
    }
}

> request devices sync-from device [ XECLI-0 XECLI-1 ]
sync-result {
    device XECLI-0
    result true
}
sync-result {
    device XECLI-1
    result true
}
```

19. 移行済みのデバイスの古い NED パッケージをクリーンアップし、パッケージをリロードして、古い NED を NSO から削除します。

```
$ rm /var/opt/ncs/packages/ncs-5.7-cisco-ios-cli-6.77.9.tar.gz
$ rm /var/opt/ncs/packages/ncs-5.7.3-cisco-iosxr-cli-7.39.5.tar.gz
admin@ncs> request packages reload force
```

20. ブートストラップデータ、プラン通知、ステータスコード、およびキッカーを構成します。詳細については、「[SR-TE CFP-IOSXR CLI のインストール後のタスクの実行](#)」を参照してください。

21. ディスPATCHマップを同期し、マップに NED ID が入力されていることを確認します。

```
admin@ncs% request devices lsa dispatch-map sync
success true
detail Dispatch Map Synced Successfully
admin@ncs% show devices lsa dispatch-map
dispatch-map <Device_name> {
    ned-id cisco-ios-cli-6.77:cisco-ios-cli-6.77;
    ...
}
```

T-SDN FP バンドルサービスのアップグレード

このセクションでは、以前のバージョンで作成された T-SDN FP バンドルサービスを v4.1.0 にアップグレードする方法について説明します。

次の手順を実行します。

1. デバイス構成とバックポインタが正しいことを確認します。

```
admin@ncs% show devices device <Device_name> config | display service-meta-
data
```

2. ODN サービスの外部プランとポリシーサービスが到達状態にあることを表示して確認します。

```
admin@ncs% run show cisco-sr-te-cfp:sr-te odn odn-template-plan
<ODN_Service_Name> plan
admin@ncs% run show cisco-sr-te-cfp:sr-te policies policy-plan
<Policy_service_Name> plan
```

3. すべての認証グループのパスワードを更新します。

注： 古いパスワードは誤って復号化されるため、パスワードを更新する必要があります。これは、NSO v5.5.2.9 のパスワード暗号化が以前の NSO バージョンと異なるためです。AES 暗号化キーの詳細については、NSO のドキュメントを参照してください。

4. NSO を強制的にリロードして、NSO を実行しているインスタンスから古い NED パッケージを削除します。

```
admin@ncs> request packages reload force
```

Cisco NSO T-SDN FP バンドルのアップグレードはこれで完了です。

付録 A : Python 起動コマンド構成の変更

このセクションの情報は、デフォルトの Python を Python 3 に変更できない場合にのみ使用してください。T-SDN FP バンドルのインストール中に T-SDN FP バンドルパッケージを抽出した後、Python 起動コマンド構成を変更します。T-SDN FP バンドルパッケージを抽出する方法の詳細については、このドキュメントの「**単一の NSO** インスタンスへのコア機能パックのインストール」の章を参照してください。

Python 起動コマンド構成を変更するには :

1. 抽出した TSDN パッケージのディレクトリに移動します。

```
$ cd nso-<version>-tsdn-<version>
```

2. **start-vm** ファイルをコピーします。

```
$ mkdir -p /opt/cisco/nso/tsdn
```

```
$ cp init_data/scripts/ncs-start-python-vm-tsdn /opt/cisco/nso/tsdn/ncs-start-python-vm-tsdn
```

3. 次のように、**/etc/ncs/ncs.conf** ファイルに **<start-command>** を構成します。

```
</java-vm>
```

```
<python-vm>
```

```
<start-command>/opt/cisco/nso/tsdn/ncs-start-python-vm- tsdn</start-command>
```

```
<run-in-terminal>
```

```
<terminal-command>DEFAULT</terminal-command>
```

```
</run-in-terminal>
```

```
<logging>
```

```
<log-file-prefix>${NCS_LOG_DIR}/ncs-python-vm</log-file-prefix>
```

```
</logging>
```

```
</python-vm>
```

注 : ncs.conf ファイルに変更を加えた場合は、ncs を再起動して変更を適用します。

付録 B : commit-queue async フラグを渡す

async フラグは、commit-queue で使用される API 制約です。commit-queue を介して作成、読み取り、更新、削除 (CRUD) 操作をコミットするように、**async** フラグを設定します。

次のコマンドは、さまざまな API で **async** フラグを設定する方法を示しています。

NSO CLI

```
admin@ncs% load merge payload/IETF-TE.xml
[ok]
[edit]
admin@ncs% commit commit-queue async
commit-queue {
    id 1616809621834
    status async
}
Commit complete.
[ok]
```

JSON-RPC

commit-queue async フラグを使用した JSON-RPC コミット呼び出し。詳細については、NSO WebUI のドキュメントを参照してください。

```
{"jsonrpc": "2.0", "id": 497, "method": "validate_commit", "params": {"th": 3, "flags": ["commit-queue=async"]}}
{"jsonrpc": "2.0", "id": 86, "method": "commit", "params": {"th": 3, "flags": ["commit-queue=async"]}}
```

RESTCONF

RESTCONF を使用すると、POST、PUT、および DELETE 呼び出しを、commit-queue async の追加パラメータを使用して送信できます。GET 呼び出しに変更はありません。

```
http://<NSO-IP>:8080/restconf/data/sr-te?async-commit-queue=true
```

Python

commit-queue async フラグを使用した Python API コミット呼び出し。

```
with ncs.maapi.single_write_trans(uinfo.username, "system", db=ncs.RUNNING) as trans:
    root = ncs.maagic.get_root(trans)
    root.ncs__devices.device[input.device].config.asa__banner.login = input.message
    commit_params = ncs.maapi.CommitParams()
    commit_params.commit_queue_async()
    res = trans.apply_params(False, commit_params)
```

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。

リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動 / 変更されている場合がありますことをご了承ください。

あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

このページは意図的に空白になっています。

シスコ コンタクトセンター



自社導入をご検討されているお客様へのお問い合わせ窓口です。

製品に関して | サービスに関して | 各種キャンペーンに関して | お見積依頼 | 一般的なご質問

お問い合わせ先

お電話での問い合わせ

平日 9:00 - 17:00

0120-092-255

お問い合わせウェブフォーム

cisco.com/jp/go/vdc_callback



©2023 Cisco Systems, Inc. All rights reserved.

Cisco, Cisco Systems, およびCisco Systemsロゴは、Cisco Systems, Inc. またはその関連会社の米国およびその他の一定の国における商標登録または商標です。本書類またはウェブサイトに掲載されているその他の商標はそれぞれの権利者の財産です。「パートナー」または「partner」という用語の使用はCiscoと他社との間のパートナーシップ関係を意味するものではありません。(1502R) この資料の記載内容は2023年 03月現在のものです。この資料に記載された仕様は予告なく変更する場合があります。



シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
cisco.com/jp