Cisco APIC トラブルシューティング ガイド、リリース 4.0(1)

初版：2018 年 10 月 24 日

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
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

目 次

はじめに

この前書きは、次の項で構成されています。

対象読者

このガイドは、データ システム、ネットワーク、ストレージ システムのトラブルシューティングに関して経歴があるシステムおよびネットワーク エンジニアを対象としています。

表記法

コマンドの説明には、次のような表記法が使用されます。

表記法	説明
bold	太字の文字は、表示どおりにユーザが入力するコマンドおよびキーワードです。
italic	イタリック体の文字は、ユーザが値を入力する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角かっこで囲んで示しています。
[x | y]	いずれか1つを選択できる省略可能なキーワードや引数は、角カッコで囲み、縦棒で区切って示しています。
{x | y}	必ずいずれか1つを選択しなければならない必須キーワードや引数は、波かっこで囲み、縦棒で区切って示しています。

表記法	説明
[x {y | z}]	角かっこまたは波かっこが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。角かっこ内の波かっこと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
`variable`	ユーザが値を入力する変数であることを表します。イタリック体が使用できない場合に使用されます。
string	引用符を付けない一組の文字。string の前後には引用符を使用しません。引用符を使用すると、その引用符も含めて string とみなされます。

例では、次の表記法を使用しています。

表記法	説明
`screen フォント`	スイッチが表示する端末セッションおよび情報は、スクリーンフォントで示しています。
太字の `screen` フォント	ユーザが入力しなければならない情報は、太字のスクリーンフォントで示しています。
イタリック体の *screen* フォント	ユーザが値を指定する引数は、イタリック体の screen フォントで示しています。
< >	パスワードのように出力されない文字は、山カッコ（< >）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

このマニュアルでは、次の表記法を使用しています。

（注） 「注釈」です。役立つ情報やこのマニュアルに記載されていない参照資料を紹介しています。

注意 「要注意」の意味です。機器の損傷またはデータ損失を予防するための注意事項が記述されています。

警告　安全上の重要事項

「危険」の意味です。人身事故を予防するための注意事項が記述されています。装置の取り扱い作業を行うときは、電気回路の危険性に注意し、一般的な事故防止策に留意してください。各警告の最後に記載されているステートメント番号を基に、装置に付属の安全についての警告を参照してください。

これらの注意事項を保管しておいてください。

関連資料

Cisco Application Centric Infrastructure（ACI）Documentation

ACI のマニュアルは、次の URL で入手できます。http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html。

シスコ アプリケーション セントリック インフラストラクチャ（ACI）シミュレータのマニュアル

Cisco ACI Simulator のマニュアルは、次の URL から入手できます：http://www.cisco.com/c/en/us/support/cloud-systems-management/application-centric-infrastructure-simulator/tsd-products-support-series-home.html

Cisco Nexus 9000 シリーズ スイッチのマニュアル

Cisco Nexus 9000 シリーズ スイッチのマニュアルは、次の URL で入手できます。http://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/tsd-products-support-series-home.html

Cisco Application Virtual Switch のマニュアル

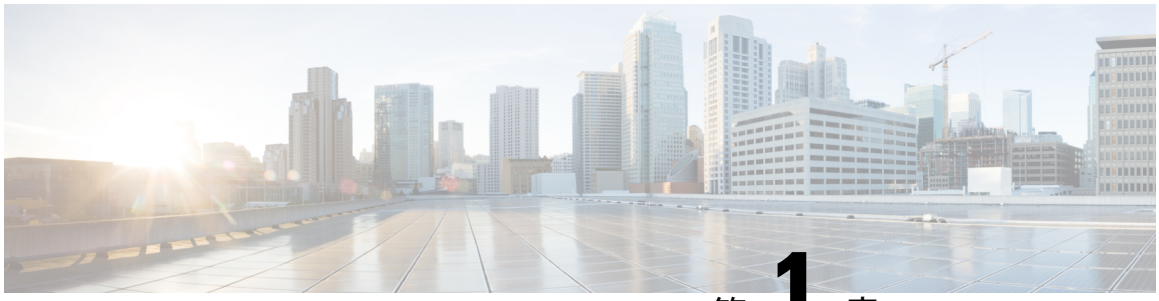
Cisco Application Virtual Switch（AVS）のマニュアルは、次の URL で入手できます。http://www.cisco.com/c/en/us/support/switches/application-virtual-switch/tsd-products-support-series-home.html

シスコ アプリケーション セントリック インフラストラクチャ（ACI）と OpenStack の統合に関するマニュアル

Cisco ACI と OpenStack の統合に関するマニュアルは、次の URL から入手できます。http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html

マニュアルに関するフィードバック

このマニュアルに関する技術的なフィードバック、または誤りや記載もれなどお気づきの点がございましたら、apic-docfeedback@cisco.com までご連絡ください。ご協力をよろしくお願いいたします。



第 1 章

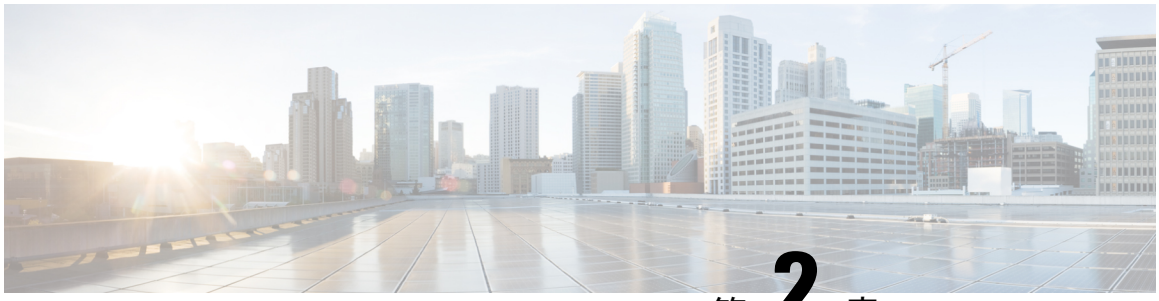
新機能および変更された機能

新機能および変更された機能に関する情報

次の表は、この最新リリースまでのガイドでの主な変更点の概要を示したものです。ただし、このリリースまでのこのガイドの変更点や新機能の中には一部、この表に記載されていないものもあります。

Cisco APIC のリリース バージョン	特長	説明
4.0 (1x)	**ユーザー アクティビティ**機能は、管理者がユーザー アクションの 2 週間の履歴を表示できるトラブルシューティング ツールです。	ユーザー アクティビティの表示 （144 ページ）
	構成同期の問題機能を使用すると、GUI でユーザー設定可能なオブジェクトを含むトランザクションの遅延を確認できます。	設定の同期の問題の確認 （144 ページ）

新機能および変更された機能に関する情報



第 2 章

トラブルシューティングの概要

このガイドの各章では、Cisco APIC の特定の機能に関する一般的なトラブルシューティングのヒントを示し、問題のトラブルシューティングに使用できるモニタリングツールについて説明します。

このガイドで取り上げる機能、問題、およびタスクを以下に示します。

- **_ui_ Objects**：**拡張モード**を使用する前に、**基本モード**または NX-OS CLI で変更を加えたために発生した不要な _ui_ objects を削除する方法について説明します。
- **Acidiag**：Cisco APIC での操作をトラブルシューティングするための acidiag コマンドの使用方法について説明します。
- **Cisco APIC クラスタ**：クラスタの障害を診断し、一般的なクラスタの問題をトラブルシューティングする方法について説明します。基本的なクラスタ管理情報については、このガイドの付録を参照してください。
- **Cisco APIC パスワード回復と緊急/非表示ログイン アクセス**：パスワードを回復する方法、レスキュー ユーザー ログインにアクセスしてトラブルシューティング コマンドを実行する方法（構成の消去を含む）、およびロックアウトの場合に非表示のログイン ドメインにアクセスする方法について説明します。
- **Cisco APIC トラブルシューティング操作**：スイッチに関する情報を収集する方法と、システムのシャットダウン、Cisco APIC コントローラのシャットダウン、APIC コントローラのリロード、LED ロケータのオンなどのトラブルシューティング操作を実行する方法について説明します。
- **Cisco APIC トラブルシューティング ツール**：Cisco APIC トラブルシューティング ツールを使用して、デバッグ、トラフィックの監視、ユーザー アクティビティ履歴の表示、ポリシー マネージャとポリシー ディストリビュータの同期の遅延のチェック、およびトラフィック ドロップ、誤ルーティング、ブロックされたパス、およびアップリンクの障害などの問題の検出を行う方法について説明します。
- **CRC エラー**：CRC エラーを表示する方法を説明します。
- **エンドポイント接続**：traceroute、アトミック カウンタ、SPAN などの Cisco APIC トラブルシューティング ツールを使用してエンドポイント接続をトラブルシューティングする方法と、SFP モジュールを新しいカードに接続する方法について説明します。

（注） Cisco APIC トラブルシューティング ツールに関する情報は、この Cisco APIC トラブルシューティングツールの使用 （27 ページ） 章にあります。

- **EVPN タイプ 2 ホスト ルート**：この機能の検証手順を提供します。
- **エクスポート ポリシー**：エクスポートの統計情報、テクニカル サポート収集、障害、イベントをエクスポートし、ファブリックから外部ホストにコア ファイルとデバッグ データを処理できます。
- **ファブリックの再構築**：ファブリックを再構築する方法について説明します。
- **障害が発生したライン カードの識別** ：ループバック障害の原因となった可能性のあるライン カードを識別する手順について説明します。
- **IP ベースの EPG** ：Cisco APIC GUI およびスイッチ コマンドを使用して、IP ベースの EPG が正しく設定されていることを確認する方法について説明します。
- **リーフ接続**：REST API を使用して切断されたリーフを復元する方法を説明します。
- PIM インターフェイス：L3Out、マルチキャスト トンネル インターフェイス、またはマルチキャスト対応ブリッジ ドメインに対して PIM インターフェイスが作成されていない場合にチェックする内容について説明します。
- **ポート セキュリティ**：ポート セキュリティ ハードウェアおよびソフトウェアのインストールを確認する方法について説明します。
- **無効なインターフェイスと廃止されたスイッチの削除**：GUI で無効なポート エントリを削除する方法を説明します。
- **スイッチの廃止と再開**：ポッド内のノードの廃止および再開方法について説明します。 このタスクのユース ケースは、より論理的でスケーラブルな番号付け規則でポッド内のノードの番号を付け直すことです。
- **Cisco APIC SSD の交換**：GUI で SSD を削除する方法を説明します。
- **QoS**：この機能の特定のトラブルシューティング シナリオを提供します。
- **SSL 暗号**：SSL 暗号がサポートされているかどうかを判断する方法について説明します。
- **スイッチ インベントリ**：スイッチのシリアル番号とモデル番号を見つける方法を説明します。 これは、TAC が発生する可能性のある問題のトラブルシューティングに役立ちます。
- **予想される出力エラー** ：ACI モードの Cisco Nexus 93180YC-EX および ACI 93180YC-FX リーフ スイッチのアップリンクの内部カウンタ インターフェイスから観察される予想される出力エラーの例を示します。

- トラブルシューティングの基本 （5 ページ）

トラブルシューティングの基本

トラブルシューティングの基本的な手順は次のとおりです。

始める前に

- Cisco APIC トラブルシューティングツールの使用 （27 ページ） にリストされたツールに習熟してください。
- Cisco APIC トラブルシューティング オペレーション （23 ページ） の内容についてよく理解しておきます。
- 特定のフィーチャに関する問題については、そのフィーチャに関するこのガイドの主な内容を確認してください。トラブルシューティングのヒントは、フィーチャごとにリストされています。

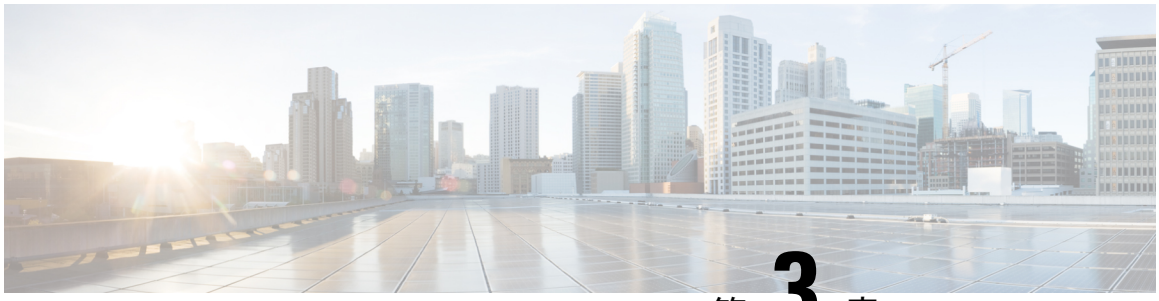
ステップ 1 特定の現象に関する情報を収集します。

（注） 多くの場合、Cisco APIC トラブルシューティングツールの使用 （27 ページ）の章にリストおよび説明されているツールを使用して、有用なトラブルシューティング情報を収集できます。

ステップ 2 現象の原因となり得る潜在的な問題をすべて識別します。

ステップ 3 現象が見られなくなるまで、潜在的な問題を系統的に 1 つずつ（最も可能性の高いものから低いものの順に）排除していきます。

（注） このガイドでは、ポートセキュリティ、エンドポイント接続、PIM、IP ベースの EPG などの特定の機能のインストールと設定を確認するための手順を順を追って説明します。指示に従うと、発生している問題を絞り込んで解決するのに役立ちます。



第 3 章

APICクラッシュ シナリオのトラブルシューティング

この章には、さまざまな障害またはクラッシュのシナリオと考えられる復元ソリューションに関する情報が含まれています。

この章は、次の項で構成されています。

Cisco APIC クラスタの障害シナリオ

クラスタのトラブルシューティング シナリオ

次の表は、Cisco APIC に共通するクラスタのトラブルシューティングのシナリオを示します。

問題	ソリューション
APIC ノードはクラスタ内でエラーが発生します。たとえば、5 つの APIC のクラスタのノード 2 がエラーを起こすとします。	2 つの解決策があります。 • 目標サイズはそのままにし、APIC を交換します。 • クラスタ サイズを 4 に減らし、コントローラ 5 をデコミッションし、APIC 2 として再コミッションします。ターゲット サイズは 4 のままで、再構成された APIC がアクティブになったときの運用サイズは 4 です。 （注） クラスタに交換する APIC を追加し、目標サイズと動作サイズを増大することができます。新しい APIC を追加する方法については、『*Cisco APIC Management, Installation, Upgrade, and Downgrade Guide*』を参照してください。

問題	ソリューション
新しい APIC はファブリックに接続し、リーフ スイッチへの接続は失われます。	インフラ（インフラストラクチャ）VLAN の不一致があるかを確認するには、次のコマンドを使用します。 • cat /mit/sys/lldp/inst/if-\[eth1--1\]/ctrlradj/summary：リーフ スイッチ上で構成された VLAN を表示します。 • cat /mit/sys/lldp/inst/if-\[eth1--1\]/ctrlradj/summary：接続された APIC によってアドバタイズされるインフラ（インフラストラクチャ）VLAN を表示します。 これらのコマンドの出力が異なる VLAN を表示する場合、新しい APIC は正しいインフラ（インフラストラクチャ）VLAN で設定されていません。この問題を解決するには、次の手順に従います。 • レスキューユーザーを使用して APIC にログインします。 （注） APIC はファブリックの一部ではないため、管理者のログイン情報は機能しません。 • 構成を消去し、 **acidiag touch setup** コマンドを使用して APIC を再起動します。 • APIC を再構成します。ファブリック名、TEP アドレス、およびクラスタの APIC にマッチするインフラ（インフラストラクチャ）VLAN を確認します。 • リーフ ノードをリロードします。25-03-2015 22:13
2 つの APIC は、再起動後に通信できません。	この問題は次の一連のイベントの後に発生することがあります。 • APIC1 と APIC2 が相互に検出します。 • APIC1 がリブートし、新しいシャーシ ID （APIC1a）でアクティブになる。 • 2 つの APIC が通信しなくなる。 このシナリオでは、APIC1a が APIC2 を検出しますが、APIC2 はオフラインと見なされる APIC1 があるクラスタ内に存在するので使用できません。その結果、APIC1a は APIC2 からのメッセージを受け入れません。 この問題を解決するには、APIC2 上の APIC1 をデコミッションし、再度 APIC1 を稼働させます。

問題	ソリューション
デコミッションされた APIC がクラスタに参加します。	この問題は次の一連のイベントの後に発生することがあります。 • クラスタのメンバーが使用できなくなるか、クラスタが分割されます。 • APIC はデコミッションされます。 • クラスターが回復すると、デコミッションされた APIC が自動的に試運転されます。 この問題を解決するには、クラスタの回復後に APIC をデコミッションします。
再起動後の ChassisID が一致しません。	この問題は、APICがクラスタで登録されたシャーシIDと異なるシャーシIDで起動したときに起こります。その結果、このAPICからのメッセージが廃棄されます。 この問題を解決するには、リブートの前に APIC が解放されていることを確認してください。
APIC はクラスタ サイズの変更時のエラーを表示します。	さまざまな条件が、AdminstrativeClusterSize に合わせたクラスタによる OperationalClusterSize の拡張の妨げになる可能性があります。詳細については、障害を調べて、*Cisco APIC* ベーシック コンフィギュレーション ガイド の「クラスタ障害」セクションを確認してください。
APIC がクラスタに参加できない	この問題は、クラスタを拡大するときに 2 つの APIC が同じクラスタ ID で設定されると起こります。その結果、2 つのうち 1 つの APIC がクラスタに参加できず、拡張競合シャーシ ID 不一致のエラーが表示されます。 この問題を解決するには、新しいクラスタ ID でクラスタの外側に APIC を設定します。

問題	ソリューション
APIC がクラスタで到達不能です。	この問題を診断するには、次の設定を確認してください。 • ファブリック検出が完了していることを確認します。 • ファブリックから欠落しているスイッチを特定します。 • スイッチが APIC からの IP アドレスを要求し、受信したかどうかを確認します。 • スイッチがソフトウェア イメージをロードしたことを確認します。 • スイッチがアクティブになっている時間を確認します。 • すべてのプロセスがスイッチ上で動作していることを確認します。詳細については、*Cisco APIC* ベーシック コンフィギュレーション ガイド の「acidiag コマンド」セクションを参照してください。 • 欠落しているスイッチに正しい日付と時刻が設定されていることを確認します。 • スイッチが他の APIC と通信できることを確認します。
クラスタは拡張しません。	この問題は、次の状況で発生します。 • OperationalClusterSize が APIC の数より少ない。 • 拡張候補はありません (たとえば、管理サイズが 5 であり、clusterID が 4 の APIC がありません。 • クラスタと新しい APIC の間に接続がない • 新しい APIC によってハートビート メッセージが拒否される • システムが正常ではありません。 • 使用できないアプライアンスは、再配置に関連するデータ サブセットを保持しています。 • 再配置に関連するデータ サブセットを持つアプライアンスでサービスがダウンしています。 • 再配置に関する不健全なデータ サブネット

問題	ソリューション
APIC がダウンしています。	次の点を確認します。 • 接続の問題：ping を使用して接続を確認します。 • インターフェイス タイプの不一致：すべての APIC がインバンド通信になっていることを確認します。 • ファブリック接続：ファブリック接続が正常であること、およびファブリック検出が完了していることを確認します。 • 拒否されたハートビート：fltInfraIICIMsgSrcOutsider エラーを確認します。一般的なエラーには、動作クラスタ サイズ、シャーシ ID の不一致、動作クラスタ サイズの外の送信元 ID 、承認されていない送信元、およびファブリック ドメインの不一致が含まれます。

クラスタの障害

APIC は、クラスタの問題の診断に役立つさまざまなエラーをサポートします。ここでは、2 つの主要なクラスタのエラーの種類について説明します。

エラーの破棄

APIC は現在のクラスタのピアまたはクラスタ拡大候補以外からのクラスタ メッセージを破棄します。APIC によりメッセージを破棄した場合、発信元の APIC のシリアル番号、クラスタ ID、タイムスタンプを含むエラーが発生します。次の表で、破棄されるメッセージのエラーを要約します。

Fault	意味
expansion-contender-chassis-id-mismatch	送信側 APIC のシャーシ ID が拡大のためにクラスタが認識するシャーシ ID と一致しません。
expansion-contender-fabric-domain-mismatch	送信側 APIC のファブリック ID が拡大のためにクラスタが認識するファブリック ID と一致しません。
expansion-contender-id-is-not-next-to-oper-cluster-size	送信側 APIC に拡大に不適切なクラスタ ID があります。値は、現在の OperationalClusterSize よりも 1 大きい必要があります。
expansion-contender-message-is-not-heartbeat	送信側 APIC が継続的ハートビート メッセージを送信しません。
fabric-domain-mismatch	送信側 APIC のファブリック ID がクラスタのファブリック ID と一致しません。
operational-cluster-size-distance-cannot-be-bridged	送信側 APIC に、受信側 APIC のものとは 1 以上違う OperationalClusterSize があります。受信側 APIC は要求を拒否します。

Fault	意味
source-chassis-id-mismatch	送信側 APIC のシャーシ ID がクラスタに登録されたシャーシ ID と一致しません。
source-cluster-id-illegal	送信側 APIC に許可されていないクラスタ ID 値があります。
source-has-mismatched-target-chassis-id	送信側 APIC の目標シャーシ ID が受信側 APIC のシャーシ ID に一致しません。
source-id-is-outside-operational-cluster-size	送信側 APIC に、クラスタの OperationalClusterSize 外のクラスタ ID があります。
source-is-not-commissioned	送信側 APIC にクラスタで現在解放されている ID があります。

クラスタ変更時エラー

次のエラーは、APIC のクラスタ サイズの変更時のエラーがある場合に適用されます。

Fault	意味
cluster-is-stuck-at-size-2	このエラーは、OperationalClusterSize が拡張期間にわたり 2 のままになると発行されます。問題を解決するには、クラスタの目標サイズをリストアします。
most-right-appliance-remains-commissioned	クラスタ内の最後の APIC が稼働中で、クラスタの縮小を妨げています。
no-expansion-contender	クラスタがより大きいクラスタ ID を持つ APIC を検出できず、クラスタの拡張を行えません。
service-down-on-appliance-carrying-replica-related-to-relocation	移動するデータのサブセットは、障害が起きているサービス上にコピーがあります。APIC に複数のこのような障害があることを示します。
unavailable-appliance-carrying-replica-related-to-relocation	移動するデータのサブセットは、使用できない APIC 上にコピーがあります。このエラーを解決するには、使用できない APIC を復元します。
unhealthy-replica-related-to-relocation	移動するデータのサブセットは、正常でない APIC 上にコピーがあります。このエラーを解決するには、障害の根本原因を特定します。

APIC 使用不可

次のクラスタのエラーは、APIC が使用できない場合に適用できます。

Fault	意味
fltInfraReplicaReplicaState	クラスタがデータのサブセットを起動できません。
fltInfraReplicaDatabaseState	データ ストア サービスの破損を示します。

Fault	意味
fltInfraServiceHealth	データのサブセットが完全には機能していないことを示します。
fltInfraWiNodeHealth	APIC が完全には機能していないことを示します。

アプリケーション セントリック インフラストラクチャ クラッシュ シナリオのトラブルシューティング

ファブリック ノードとプロセス クラッシュのトラブルシューティング

ACI スイッチ ノードには、システムのさまざまな機能面を制御する多数のプロセスがあります。システムの特定のプロセスでソフトウェア障害が発生した場合、コア ファイルが生成され、プロセスがリロードされます。

プロセスが Data Management Engine（DME）プロセスの場合、DME プロセスは自動的に再起動します。プロセスが非 DME プロセスの場合、プロセスは自動的に再起動せず、スイッチが再起動して回復します。

このセクションでは、さまざまなプロセスの概要、プロセスがコア化したことを検出する方法、およびこれが発生したときに取るべきアクションについて説明します。

DME プロセス

APIC で実行されている重要なプロセスは、CLI で見つけることができます。APIC とは異なり、 **FABRIC** > **INVENTORY** > **Pod 1** > ***node*** の GUI を介して表示できるプロセスには、リーフで実行されているすべてのプロセスが表示されます。

ps-ef | grep svc_ifc を経由：

```
rtp_leaf1# ps -ef |grep svc_ifc
root 3990 3087 1 Oct13 ? 00:43:36 /isan/bin/svc_ifc_policyelem --x
root 4039 3087 1 Oct13 ? 00:42:00 /isan/bin/svc_ifc_eventmgr --x
root 4261 3087 1 Oct13 ? 00:40:05 /isan/bin/svc_ifc_opflexelem --x -v
dptcp:8000
root 4271 3087 1 Oct13 ? 00:44:21 /isan/bin/svc_ifc_observerelem --x
root 4277 3087 1 Oct13 ? 00:40:42 /isan/bin/svc_ifc_dbgrelem --x
root 4279 3087 1 Oct13 ? 00:41:02 /isan/bin/svc_ifc_confelem --x
rtp_leaf1#
```

スイッチで実行されている各プロセスは、システムのログ ファイルにアクティビティを書き込みます。これらのログ ファイルは、techsupport ファイルの一部として処理されていますが、CLI アクセスを介して /tmp/logs/ ディレクトリにあります。たとえば、ポリシー エレメントのプロセス ログ出力は、/tmp/logs/svc_ifc_policyelem.log に書き込まれます。

以下は、システムで実行されている DME プロセスの簡単な説明です。これは、特定のプロセスのトラブルシューティング時にどのログ ファイルを参照するかを理解したり、プロセスがクラッシュした場合のシステムへの影響を理解したりするのに役立ちます。

プロセス	機能
ポリシー要素	ポリシー要素: APIC からの論理 MO を処理し、具体的なモデルをスイッチにプッシュします
eventmgr	イベント マネージャ：ローカルの障害、イベント、ヘルス スコアを処理します
opflexelem	Opflex 要素: スイッチ上の Opflex サーバ
observerelem	オブザーバ要素: APIC に送信されたローカル統計を処理します
dbgrelem	デバッガー要素：コア ハンドラ
nginx	スイッチと APIC 間のトラフィックを処理する Web サーバ

プロセスがいつクラッシュしたかを特定する

プロセスがクラッシュしてコア ファイルが生成されると、イベントだけでなく障害も生成されます。APIC からの次の syslog 出力に示されているように、特定のプロセスの障害は「プロセス クラッシュ」として表示されます。

```
Oct 16 03:54:35 apic3 %LOG_LOCAL7-3-SYSTEM_MSG [E4208395][process-crash][major]
[subj-[dbgs/cores/node-102-card-1-svc-policyelem-ts-2014-10-16T03:54:55.000+00:00]/
rec-12884905092]Process policyelem cored
```

スイッチのプロセスがクラッシュすると、コア ファイルが圧縮され、APIC にコピーされます。syslog メッセージ通知は APIC から送信されます。

プロセスがクラッシュしたときに生成される障害は、プロセスが再起動された Cisco Application Centric Infrastructure 275 のトラブルシューティングでクリアされます。障害は、**[ファブリック（FABRIC）]** > **[インベントリ（INVENTORY）]** > **[ポッド 1 （Pod 1）]** でファブリック履歴タブの GUI を介して表示できます。

コア ファイルの収集

APIC GUI は、ファブリック ノードのコア ファイルを収集するための中心的な場所を提供します。

エクスポート ポリシーは、**ADMIN** > **IMPORT/EXPORT** > **Export Policies** > **Core** から作成されます。ただし、ファイルを直接ダウンロードできるデフォルトのコア ポリシーがあります。

コア ファイルには、コア ファイルが配置されている APIC の /data/techsupport にある APIC を介して SSH/SCP 経由でアクセスできます。コア ファイルは、クラスタ内の 1 つの APIC の /data/techsupport で入手できることに注意してください。コア ファイルが存在する正確な APIC

は、GUIに表示されるエクスポートロケーションパスで見つけることができます。たとえば、エクスポート先が「files/3/」で始まる場合、ファイルはノード3（APIC3）にあります。

APIC プロセスのクラッシュの検証と再起動

症状 1

スイッチファブリックのプロセスがクラッシュします。プロセスが自動的に再起動するか、スイッチがリロードして復元します。

- **検証**：

  概要セクションに示されているように、DME プロセスがクラッシュした場合、スイッチを再起動せずに自動的に再起動する必要があります。非 DME プロセスがクラッシュした場合、プロセスは自動的に再起動せず、スイッチが再起動して回復します。

  どのプロセスがクラッシュするかによって、プロセス コアの影響は異なります。

  非DMEプロセスがクラッシュすると、通常コンソールに表示されるようにHAPリセットが発生します。

```
[ 1130.593388] nvram_klm wrote rr=16 rr_str=ntp hap reset to nvram
[ 1130.599990] obfl_klm writing reset reason 16, ntp hap reset
[ 1130.612558] Collected 8 ext4 filesystems
```

- **プロセス ログの確認**：

  クラッシュするプロセスには、クラッシュ前に何らかのレベルのログ出力が必要です。スイッチのログの出力は、/tmp/logsディレクトリに書き込まれます。プロセス名はファイル名の一部になります。たとえば、ポリシー エレメント プロセスの場合、ファイルは svc_ifc_policyelem.log です。

```
rtp_leaf2# ls -l |grep policyelem
-rw-r--r-- 2 root root 13767569 Oct 16 00:37 svc_ifc_policyelem.log
-rw-r--r-- 1 root root 1413246 Oct 14 22:10 svc_ifc_policyelem.log.1.gz
-rw-r--r-- 1 root root 1276434 Oct 14 22:15 svc_ifc_policyelem.log.2.gz
-rw-r--r-- 1 root root 1588816 Oct 14 23:12 svc_ifc_policyelem.log.3.gz
-rw-r--r-- 1 root root 2124876 Oct 15 14:34 svc_ifc_policyelem.log.4.gz
-rw-r--r-- 1 root root 1354160 Oct 15 22:30 svc_ifc_policyelem.log.5.gz
-rw-r--r-- 2 root root 13767569 Oct 16 00:37 svc_ifc_policyelem.log.6
-rw-rw-rw- 1 root root 2 Oct 14 22:06 svc_ifc_policyelem.log.PRESERVED
-rw-rw-rw- 1 root root 209 Oct 14 22:06 svc_ifc_policyelem.log.stderr
rtp_leaf2#
```

  /tmp/logsにあるプロセスごとにいくつかのファイルがあります。ログ ファイルのサイズが大きくなるにつれて、ログ ファイルは圧縮され、古いログ ファイルはローテーションされなくなります。コア ファイルの作成時刻（GUI とコア ファイル名に表示される）を確認して、ファイルのどこを確認すればよいかを理解します。また、プロセスが最初に起動しようとすると、ログファイルに「クラッシュ後にプロセスが再起動しています」というエントリが記録されます。このエントリを使用して、クラッシュの前に何が起こったかを遡って検索できます。

- **アクティビティをチェック**：

実行中のプロセスに変更が加えられたため、クラッシュが発生しました。多くの場合、変更はシステムの構成アクティビティによるものである可能性があります。システムで発生したアクティビティは、システムの監査ログ履歴で確認できます。

- **TAC に連絡する**：

  通常、プロセスのクラッシュは発生しません。上記の手順を超える理由をよりよく理解するには、コア ファイルをデコードする必要があります。この時点で、ファイルを収集して、さらに処理するために TAC に提供する必要があります。

  上記の方法でコア ファイルを収集し、TAC でケースをオープンします。

症状 2

ファブリック スイッチが継続的にリロードするか、BIOS ローダー プロンプトでスタックします。

- **検証**：

  DME プロセスがクラッシュした場合、スイッチの再起動をせずに自動的に再起動する必要があります。非DMEプロセスがクラッシュした場合、プロセスは自動的に再起動せず、スイッチが再起動して回復します。ただし、いずれの場合でもプロセスが継続的にクラッシュすると、スイッチは継続的なリロードループに入るか、BIOS ローダー プロンプトで終了する可能性があります。

```
[ 1130.593388] nvram_klm wrote rr=16 rr_str=policyelem hap reset to nvram
[ 1130.599990] obfl_klm writing reset reason 16, policyelem hap reset
[ 1130.612558] Collected 8 ext4 filesystems
```

- **HAP リセット ループを破る**：

  最初のステップは、スイッチをさらに情報を収集できる状態に戻すことです。

  スイッチが継続的に再起動している場合、スイッチの起動時に、スイッチが起動サイクルの最初の部分である場合 CTRL C を入力して、コンソールから BIOS ローダー プロンプトに侵入します。

  スイッチがローダー プロンプトに表示されたら、次のコマンドを入力します。

  - cmdline no_hap_reset
  - ブート

  cmdline コマンドは、hap リセットが呼び出されたときにスイッチがリロードするのを防ぎます。2番目のコマンドでは、システムを起動します。リロードによって入力されたcmdlineオプションが削除されるため、ローダーでのリロードの代わりに boot コマンドが必要であることに注意してください。

  これで、システムはデータを収集するためのより適切なアクセスを許可するようになったはずですが、プロセスがクラッシュするとスイッチの機能に影響を与えます。

前の表のように、プロセス ログ、アクティビティを確認し、TACの手順に連絡してください。

APIC プロセス クラッシュのトラブルシューティング

APIC には、システムのさまざまな機能的側面を制御する一連のデータ管理エンジン（DME）プロセスがあります。システムの特定のプロセスでソフトウェア障害が発生すると、コア ファイルが生成され、プロセスが再ロードされます。

次のセクションでは、システム プロセスのクラッシュやソフトウェアの障害に関連する潜在的な問題について説明します。まず、さまざまなシステム プロセスの概要、プロセスがコア化されたことを検出する方法、およびこれが発生したときに取るべきアクションについて説明します。正常に動作しているシステムの表示は、突然終了した可能性のあるプロセスを特定するために使用できます。

DME プロセス

APIC で実行されている重要なプロセスは、GUI または CLI のいずれかで見つけることができます。GUI を使用すると、実行中のプロセスとプロセス ID が **[システム（System）] > [コントローラ（Controllers）] > [プロセス（Processes）]** に表示されます。

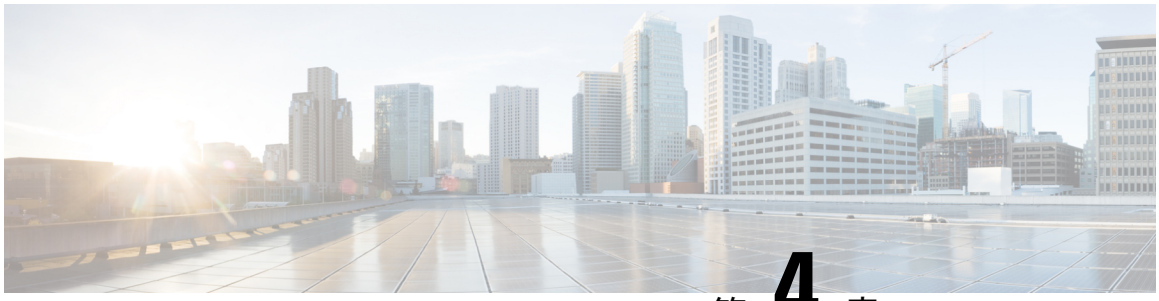
CLI を使用すると、プロセスとプロセス ID は、/aci/system/controllers/1/processes（APIC1 の場合）のサマリ ファイルにあります。

```
admin@RTP_Apic1:processes> cat summary
processes:
process-id process-name max-memory-allocated state
---------- ---------------- -------------------- -------------------
0 KERNEL 0 interruptible-sleep
331 dhcpd 108920832 interruptible-sleep
336 vmmmgr 334442496 interruptible-sleep
554 neo 398274560 interruptible-sleep
1034 ae 153690112 interruptible-sleep
1214 eventmgr 514793472 interruptible-sleep
2541 bootmgr 292020224 interruptible-sleep
4390 snoopy 28499968 interruptible-sleep
5832 scripthandler 254308352 interruptible-sleep
19204 dbgr 648941568 interruptible-sleep
21863 nginx 4312199168 interruptible-sleep
32192 appliancedirector 136732672 interruptible-sleep
32197 sshd 1228800 interruptible-sleep
32202 perfwatch 19345408 interruptible-sleep
32203 observer 724484096 interruptible-sleep
32205 lldpad 1200128 interruptible-sleep
32209 topomgr 280576000 interruptible-sleep
32210 xinetd 99258368 interruptible-sleep
32213 policymgr 673251328 interruptible-sleep
32215 reader 258940928 interruptible-sleep
32216 logwatch 266596352 interruptible-sleep
32218 idmgr 246824960 interruptible-sleep
32416 keyhole 15233024 interruptible-sleep
admin@apic1:processes>
```

APIC で実行されている各プロセスは、システムのログ ファイルに書き込みます。これらのログ ファイルは、APIC techsupport ファイルの一部としてバンドルできますが、/var/log/dme/log の SSH シェル アクセスを介して確認することもできます。たとえば、Policy Manager プロセス ログ出力は /var/log/dme/log/svc_ifc_policymgr.bin.log に書き込まれます。

以下は、システムで実行されているプロセスの簡単な説明です。これは、特定のプロセスのトラブルシューティング時にどのログ ファイルを参照するかを理解したり、プロセスがクラッシュした場合のシステムへの影響を理解したりするのに役立ちます。

プロセス	機能
カーネル	Linux カーネル
dhcpd	APIC がインフラ アドレスを割り当てるために実行されている DHCP プロセス
vmmmgr	APIC とハイパーバイザ間のプロセスを処理します
neo	Shell CLI インタープリタ
ae	ローカル APIC アプライアンスの状態とインベントリを処理します
eventmgr	システム上のすべてのイベントと障害を処理します
bootmgr	ファブリック ノードでの起動とファームウェアの更新を制御します
snoopy	Shell CLI ヘルプ、タブ コマンド補完
scripthandler	L4-L7 デバイスのスクリプトと通信を処理します
dbgr	プロセスがクラッシュしたときにコア ファイルを生成します
nginx	Web サービス処理 GUI および REST API アクセス
appliancedirector	APIC クラスタの形成と制御を処理します
sshd	APIC への SSH アクセスを有効化
perfwatch	Linux cgroup 技術情報の使用法を監視します
observer	ファブリック システムと状態、統計、正常性のデータ処理を監視します
lldpad	LLDP エージェント
topomgr	ファブリックのトポロジとインベントリを維持します



第 4 章

Cisco APIC パスワードの復元と特別ログインのアクセス

この章では、Cisco APIC パスワードを復元する方法、レスキューユーザー ログインにアクセスしてトラブルシューティング コマンドを実行する方法（構成を消去するコマンドを含む）、およびロックアウトの場合のローカル ユーザー データベースを使用してログインできる非表示のログイン ドメインにアクセスする方法について説明します。

この章は、次の項で構成されています。

APICパスワードの回復

これらの手順に従い、APIC パスワードを復元します。

ステップ 1 「aci-admin-passwd-reset.txt」という名前の空のファイルを作成して保存します。

ステップ 2 ファイルを USB ドライブに追加します。USB ドライブを FAT または FAT32 にフォーマットできます。

ステップ 3 USB ドライブを Cisco APIC の背面 USB ポートの 1 つに接続します。

ステップ 4 Cisco Integrated Management Controller（CIMC）を使用するか、デバイスの電源を入れ直して Cisco APIC を再起動します。

ステップ 5 左上に表示される 10 秒のカウントダウン タイマーの間に **[Esc]** キーを押して、ブート ターゲットのリストを表示します。

ステップ 6 **[e]** キーを押して、デフォルトの grub 行を編集します。

ステップ 7 「linux」で始まる行に移動します。**[End]** キーまたは**右矢印**キーを使用して、カーソルをその行の最後に移動し、「aci-admin-passwd-reset」を追加します。

ステップ 8 **[Ctrl+X]** を押してエントリを起動します。

新しいパスワードを有効にするには数分かかる場合があります。

Rescue-user アカウントを使用し NX-OS スタイルの CLI を使用した Cisco APIC 構成を消去する

rescue-user は、クラスタにない場合でも Cisco APIC へのアクセスを提供する緊急ログインです。このログインを使用して、構成の消去を含むトラブルシューティング コマンドを実行できます。

Note スタンバイ Cisco APIC の場合、SSH を使用して、ユーザー名「rescue-user」でパスワードなしでログインできます。スタンバイ Cisco APIC が以前にファブリックの一部であった場合、キーボード、ビデオ、マウス（KVM）コンソールを使用してオペレーティング システムを再インストールしない限り、「rescue-user」アカウントは古い管理者パスワードを保持します。

ステップ 1 Cisco Integrated Management Controller（CIMC）コンソールを使用して APIC にアクセスします。

ステップ 2 rescue-user としてログインします。

Note 管理者パスワードが構成されていて、Cisco APIC がファブリックにログオンしている場合、rescue-user パスワードは管理者パスワードと同じです。それ以外の場合、rescue-user のパスワードはありません。

ステップ 3 **acidiag touch** コマンドを使用して、構成をクリアします。

Example:

```
apic1# acidiag touch setup
```

フォールバック ログイン ドメインを使用してローカル データベースにログインする

「フォールバック」という名前の隠しログイン ドメインがあり、ロックアウトの場合にローカル ユーザー データベースを使用してログインできます。認証方法に使用されるユーザー名の形式は `apic#fallback\\<username>` です。

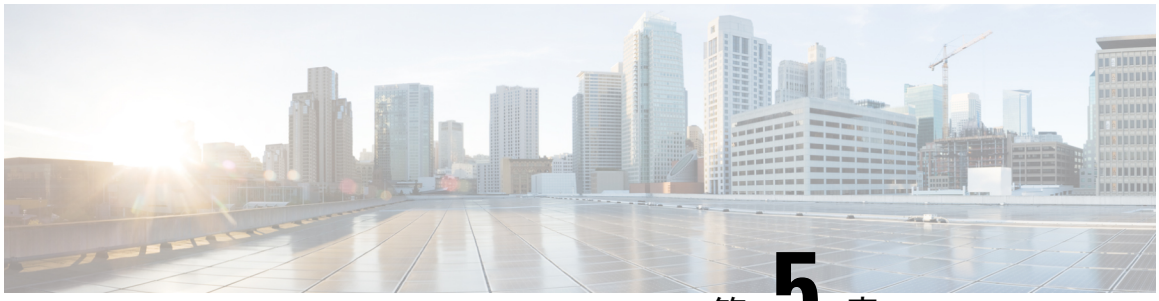
ステップ 1 次に示すように、フォールバック ログイン ドメインを使用して GUI のローカル データベースにログインするか、NX-OS スタイルの CLI を使用してフォールバック ログイン ドメインにログインします。

```
apic1(config)# aaa authentication login domain fallback
apic1(config-domain)# ?
group Set provider group for login domain
realm Specify server realm
```

ステップ 2 必要に応じて、代わりに REST API を使用して、次のようにフォールバック ログイン ドメインにログインできます。

- URL: `https://ip_address/api/aaaLogin.xml`
- データ：

```
<aaaUser name="apic#fallback\\admin"
pwd="passwordhere"/>
```



第 5 章

Cisco APIC トラブルシューティング オペレーション

この章では、基本的なトラブルシューティング操作を実行する方法について説明し、次のセクションで構成されています。

Cisco APIC システムのシャットダウン

この手順では、Cisco Application Policy Infrastructure Controller（APIC）システムをシャットダウンします。システムをシャットダウンした後、ファブリック全体を再配置してから電源を入れ、それに応じてタイム ゾーンおよび/または NTP サーバーを更新します。

始める前に

クラスタの健全性が完全に適合していることを確認します。

ステップ 1 メニュー バーで、**[システム（System）] > [コントローラ（Controllers）]** を選択します。
ステップ 2 ナビゲーション ウィンドウで、**[コントローラ（Controllers）] > *apic_name*** を選択します。
ステップ 3 Cisco APIC を右クリックし、**[シャットダウン（Shutdown）]** を選択します。
ステップ 4 Cisco APIC を再配置してから、電源を入れます。
ステップ 5 クラスタが完全に収束したことを確認します。
ステップ 6 次の Cisco APIC についてこの手順を繰り返します。

GUI を使用した Cisco APIC のシャットダウン

この手順では、Cisco Application Policy Infrastructure Controller（APIC）をシャットダウンします。この手順では、Cisco APIC システム全体ではなく、1 つの Cisco APIC システムのみがシャットダウンされます。この手順に従うと、コントローラはすぐにシャットダウンします。コントローラを元に戻すには、実際のマシンから実行するしかないため、シャットダウンの実行には注意が必要です。マシンにアクセスする必要がある場合は、「GUI を使用した LED ロケータの制御 （24 ページ）」を参照してください。

（注） 可能であれば、Cisco APIC を 1 つずつ移動します。クラスタ内にオンラインの Cisco APIC が少なくとも 2 つある限り、読み取り/書き込みアクセスが可能です。一度に複数の Cisco APIC を再配置する必要がある場合、これにより、1 つまたはすべてのコントローラがオンラインになり、ファブリックはシャットダウン時に読み取り専用モードになります。この間、エンドポイントの移動（仮想マシンの移動を含む）を含むポリシーの変更はできません。

ステップ 1 メニュー バーで、**[システム（System）] > [コントローラ（Controllers）]** を選択します。
ステップ 2 ナビゲーション ウィンドウで、**[コントローラ（Controllers）] >** ***apic_name*** を選択します。
ステップ 3 Cisco APIC を右クリックし、**[シャットダウン（Shutdown）]** を選択します。
ステップ 4 Cisco APIC を再配置してから、電源を入れます。
ステップ 5 クラスタが完全に収束したことを確認します。

GUI を使用した APIC リロード オプションの使用

この手順では、GUI を使用して、Cisco APIC システム全体ではなく Cisco Application Policy Infrastructure Controller（APIC）をリロードします。

ステップ 1 メニュー バーで、**[システム（System）] > [コントローラ（Controllers）]** を選択します。
ステップ 2 ナビゲーション ウィンドウで、**[コントローラ（Controllers）] >** ***apic_name*** を選択します。
ステップ 3 Cisco APIC を右クリックし、**[リロード（Reload）]** を選択します。

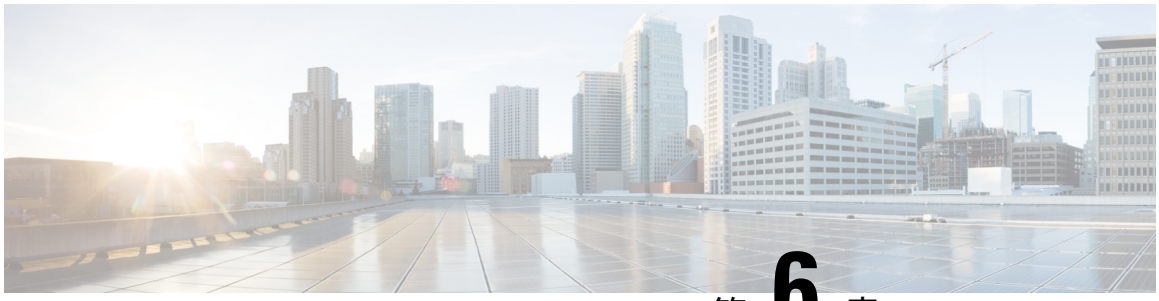
GUI を使用した LED ロケータの制御

この手順では、GUI を使用して Cisco Application Policy Infrastructure Controller（APIC）の LED ロケータをオンまたはオフにします。

ステップ 1 メニュー バーで、**[システム（System）]** > **[コントローラ（Controllers）]** を選択します。

ステップ 2 ナビゲーション ウィンドウで、**[コントローラ（Controllers）]** > ***apic_name*** を選択します。

ステップ 3 Cisco APIC を右クリックし、必要に応じて **[ロケータ LED をオンにする（Turn On Locator LED）]** または **[ロケータ LED をオンにする（Turn On Locator LED）]** を選択します。



第 6 章

Cisco APIC トラブルシューティングツールの使用

この章では、発生する可能性のある問題のトラブルシューティングに一般的に使用されるツールと方法を紹介します。これらのツールは、トラフィックの監視、デバッグ、およびトラフィック ドロップ、誤ルーティング、ブロックされたパス、アップリンク障害などの問題の検出に役立ちます。この章で説明するツールの概要については、以下のツールを参照してください。

- **[ACL コントラクト許可と拒否ログ（ACL Contract Permit and Deny Logs）]**：コントラクト許可ルールのために送信が許可されているパケットまたはフローのロギング、またはタブー コントラクト拒否ルールのためにドロップされているパケットまたはフローのロギングの有効化します。
- **[アトミック カウンタ（Atomic Counters）]**：ドロップ検出のフローの間のトラフィックの統計を収集することを有効化。ファブリックのミスルーティングの統計を収集。クイック デバッギングとアプリケーション接続問題の隔離の有効化。
- **[デジタル オプティカル モニタリング（Digital Optical Monitoring）]**：物理インターフェイスに関するデジタル オプティカル モニタリング（DOM）統計を表示できます。
- **[正常性スコア（Health Score）]**：ネットワーク階層をドリルダウンして障害を特定の管理対象オブジェクト（MO）に分離することにより、パフォーマンスの問題を分離できます。
- **[ポート トラッキング（Port Tracking）]**：アップリンクの障害を検出するために、リーフ スイッチとスパイン スイッチ間のリンクのステータスをモニタできます。
- **[SNMP]**：Simple Network Management Protocol (SNMP) は、個々のホスト（APIC またはその他のホスト）をリモートでモニタし、特定のノードの状態を確認できます。
- **[SPAN]**：Switchport Analizer（SPAN）は、詳細なトラブルシューティングの実行または特定のアプリケーション ホストからトラフィックのサンプルを取得し、プロアクティブなモニタリングと分析を行うことができます。
- **[統計（Statistics）]**：監視対象オブジェクトのリアルタイム測定が提供されます。統計の表示により、トレンド分析とトラブルシューティングの実行が可能になります。

- **[Syslog]**： 送信されるメッセージの重大度の最小値、syslog メッセージに含める項目、および syslog の接続先を指定できます。NX-OS CLI フォーマットで表示することもできます。
- **[トレースルート（Traceroute）]**：パケットが接続先に移動するときに実際にたどるルートを探すことできます。
- **[トラブルシューティング ウィザード（Troubleshooting Wizard）]**：管理者は、2 つのエンドポイントを選択することで指定できる特定の時間枠内に発生する問題のトラブルシューティングを行うことができます。
- **[設定の同期の問題（Configuration Sync Issues）]**：Cisco APIC のトランザクションがまだ同期されていないかどうかを確認できます。

この章は、次の項で構成されています。

ACL コントラクトおよび拒否ログの有効化および表示

ACL 契約の許可および拒否ログについて

契約ルールのトラフィック フローをログ記録および監視するには、契約の許可ルールのため送信されることが許可されたパケットまたはフローのログを有効化および表示するか、契約の許可ルールのためドロップされたフローのログを有効化および表示できます。

- 禁止契約拒否ルール
- 契約の件名でアクションを拒否する
- 契約または件名の例外

- ACI ファブリックの ACL コントラクト許可は、EX または FX で終わる名前の Nexus 9000 シリーズ スイッチ、およびそれ以降のすべてのモデルでのみサポートされます。たとえば、N9K-C93180LC-EX や N9K-C9336C-FX のように指定してください。
- ACI ファブリックでのログの拒否は、すべてのプラットフォームでサポートされています。
- 管理契約のフィルタでログ directive を使用することはサポートされていません。ログ directive を設定すると、ゾーン分割ルールの展開エラーが発生します。

標準および禁止契約と件名についての詳細は、『*CiscoApplication Centric Infrastructure Fundamentals*』および『*Cisco APIC Basic Configuration Guide*』を参照してください。

ACL 許可および拒否ログ出力に含まれる EPG データ

Cisco APIC、リリース 3.2(1) まで、ACL 許可および拒否ログでは、記録されている契約に関連付けられた EPG を識別していませんでした。リリース 3.2(1) では、送信元 EPG と 送信先 EPG が ACI 許可および拒否ログの出力に追加されます。ACL 許可および拒否ログには、次の制限を持つ関連 EPG を含めます。

- ネットワーク内の EPG の配置によっては、ログの EPG データを使用できない場合があります。
- 設定の変更が発生するとき、ログデータが期限切れになっている可能性があります。安定した状態では、ログ データは正確です。

ログが次にフォーカスされているとき、許可および拒否ログの EPG データは最も正確になります。

- 入力 TOR で入力ポリシーがインストールされており、出力 TOR で出力ポリシーがインストールされている場合の EPG から EPG へのフロー。
- 境界リーフ TOR で 1 個のポリシーが適用され、非 BL TOR で他のポリシーが適用されている場合の EPG から L3Out へのフロー。

ログ出力の EPG は、共有サービス（共有 L3Outs を含む）で使用される uSeg Epg または Epg ではサポートされていません。

GUI を使用してACL 契約の許可とロギングの拒否を有効にする

次の手順では、GUI を使用してACL 契約の許可とロギングの拒否を有効にする方法を表示します。

（注） 許可ロギングを含むテナントは、EPG が関連する VRF を含むテナントです。これは必ずしも EPG と同じテナントや関連する契約である必要はありません。

ステップ 1 メニュー バーで、**[Tenants]** > **[<tenant name>]** の順に選択します。

ステップ 2 [Navigation] ペインで、[Contracts] を展開し、 [Standard] を右クリックして [Create Contract] を選択します。

ステップ 3 [Create Contract] ダイアログボックスで、次の作業を実行します。

a) [Name] フィールドに、契約の名前を入力します。
b) [Scope] フィールドで、そのスコープ（[VRF]、[Tenant]、または [Global]）を選択します
c) オプション。契約に適用するターゲット DSCP または QoS クラスを設定します。
d) [+] アイコンをクリックして、[Subject] を展開します。

ステップ 4 [Create Contract Subject] ダイアログボックスで、次の操作を実行します。

ステップ 5 件名の名前と詳細な説明を入力します。

ステップ 6 オプション。ターゲット DSCP のドロップダウン リストから、件名に適用する DSCP を選択します。

ステップ 7 契約を両方向でなくコンシューマからプロバイダの方向にのみ適用するのでない限り、[Apply Both Directions] はオンにしたままにしておきます。

ステップ 8 [Apply Both Directions] をチェックしてない場合 [Reverse Filter Ports] をチェックしたままにして、ルールがプロバイダから消費者に適用されるようにレイヤ 4 ソースと宛先ポートを交換します。

ステップ 9 [+] アイコンをクリックして、[Filters] を展開します。

ステップ 10 [Name] ドロップダウン リストで、たとえば、**arp**、**default**、**est**、**icmp** などオプションを選択するか、以前設定したフィルタを選択します。

ステップ 11 [Directives] ドロップダウン リストで、[log] をクリックします。

ステップ 12 （任意） この件名で実行するアクションを [Deny] に変更します（またはアクションをデフォルトの [Permit] のままにします。

Directive：ログ有効化により、この件名のアクションが [Permit] になっている場合、ACL は件名と契約により制御されているフローとパケットを追跡します。この件名のアクションが [Deny] の場合、ACL の拒否ログはフローとパケットを追跡します。

ステップ 13 （任意） 件名の優先順位を設定します。

ステップ 14 [Update] をクリックします。

ステップ 15 [OK] をクリックします。

ステップ 16 **[送信（Submit）]** をクリックします。
ロギングがこの契約に対して有効になります。

NX-OS CLI を使用した ACL 契約許可ロギングの有効化

次の例は、NX-OS CLI を使用して契約許可ロギングを有効にする方法を示しています。

ステップ 1 契約許可ルールにより送信できたパケットまたはフローのロギングを有効にするには、次のコマンドを使用します。

```
configure
tenant <tenantName>
contract <contractName> type <permit>
subject <subject Name>
access-group <access-list> <in/out/both> log
```

例：

次に例を示します。

```
apic1# configure
apic1(config)# tenant BDMode1
apic1(config-tenant)# contract Logicmp type permit
apic1(config-tenant-contract)# subject icmp
apic1(config-tenant-contract-subj)# access-group arp both log
```

ステップ 2 許可ロギングを無効にするには、**no** 形式の access-group コマンドを使用します。たとえば、`no access-group arp both log` コマンドを使用します。

REST API を使用した ACL 契約許可ロギングの有効化

次の例は、REST API を使用して許可および拒否ロギングを有効にする方法を示しています。この例では、ACL の許可を設定し、件名 Permit 設定し、設定されたアクションを拒否するには、契約のロギングを拒否します。

この設定では、次の例のように XML で post を送信します。

例：

```
<vzBrCP dn="uni/tn-Tenant64/brc-C64" name="C64" scope="context">
    <vzSubj consMatchT="AtleastOne" name="HTTPSsbj" provMatchT="AtleastOne" revFltPorts="yes"
rn="subj-HTTPSsbj">
         <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes"
priorityOverride="default"
rn="rssubjFiltAtt-PerHTTPS" tDn="uni/tn-Tenant64/flt-PerHTTPS" tRn="flt-PerHTTPS"
tnVzFilterName="PerHTTPS"/>
    </vzSubj>
    <vzSubj consMatchT="AtleastOne" name="httpSbj" provMatchT="AtleastOne" revFltPorts="yes"
rn="subj-httpSbj">
       <vzRsSubjFiltAtt action="deny" directives="log" forceResolve="yes" priorityOverride="default"

rn="rssubjFiltAtt-httpFilter" tDn="uni/tn-Tenant64/flt-httpFilter" tRn="flt-httpFilter"
tnVzFilterName="httpFilter"/>
    </vzSubj>
    <vzSubj consMatchT="AtleastOne" name="subj64" provMatchT="AtleastOne" revFltPorts="yes"
rn="subj-subj64">
       <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes" priorityOverride="default"

rn="rssubjFiltAtt-icmp" tDn="uni/tn-common/flt-icmp" tRn="flt-icmp" tnVzFilterName="icmp"/>
    </vzSubj>
</vzBrCP>
```

GUI を使用した禁止契約拒否ロギングの有効化

次の手順は、GUI を使用して禁止コントラクトの拒否ロギングを有効にする方法を示しています。

ステップ 1 メニュー バーで、**[Tenants]** > **[<tenant name>]** の順に選択します。

ステップ 2 [Navigation] ペインで、[Contracts] を展開します。

ステップ 3 [Taboos] を右クリックし、[Create Taboo Contract] を選択します。

ステップ 4 [Create Taboo Contract] ダイアログ ボックスで、次の操作を実行して禁止契約を指定します。

a) [Name] フィールドに、契約の名前を入力します。
b) オプション。[Description] フィールドに、禁止契約の説明を入力します。
c) [+] アイコンをクリックして、[Subject] を展開します。

ステップ 5 [Create Taboo Contract Subject] ダイアログ ボックスで、次の操作を実行します。

a) [Specify Identity of Subject] 領域に、名前と説明（オプション）を入力します。
b) [+] アイコンをクリックして、[Filters] を展開します。
c) [Name] ドロップダウン リストから、**<tenant_name>/arp**、**<tenant_name>/default**、**<tenant_name>/est**、**<tenant_name>/icmp** などのデフォルト値のいずれかを選択し、以前作成したフィルタか [Create Filter] を選択します。

（注） [Specify Filter Identity] 領域で [Create Filter] を選択した場合、次の操作を実行して、ACL 拒否ルールの基準を指定します。

1. 名前とオプションの説明を入力します。
2. [Entries] を展開し、ルールの名前を入力して、拒否するトラフィックを定義する条件を選択します。
3. [Directives] ドロップダウンリストで [log] を選択します。
4. [Update] をクリックします。
5. [OK] をクリックします。

ステップ 6 **[送信（Submit）]** をクリックします。
ロギングがこの禁止契約に対して有効になります。

NX-OS CLI を使用した禁止契約拒否ロギングの有効化

次の例は、NX-OS CLI を使用して禁止契約拒否ロギングを有効にする方法を示しています。

ステップ 1 禁止契約拒否ルールのためにドロップされたパケットまたはフローのロギングを有効にするには、次のコマンドを使用します。

```
configure
tenant <tenantName>
contract <contractName> type <deny>
subject <subject Name>
access-group <access-list> <both> log
```

例：

次に例を示します。

```
apic1# configure
apic1(config)# tenant BDMode1
apic1(config-tenant)# contract dropFTP type deny
apic1(config-tenant-contract)# subject dropftp
apic1(config-tenant-contract-subj)# access-group ftp both log
```

ステップ 2 拒否ロギングを無効にするには、**no** 形式の access-group コマンドを使用します。たとえば、`no access-group https both log` コマンドを使用します。

REST API を使用した禁止契約拒否ロギングの有効化

次の例は、REST API を使用して禁止契約拒否ロギングを有効にする方法を示しています。

タブー契約を設定するロギングを拒否する、次の例のように XML で post を送信します。

例：

```
<vzTaboo dn="uni/tn-Tenant64/taboo-TCtrctPrefix" name="TCtrctPrefix" scope="context">
    <vzTSubj name="PrefSubj" rn="tsubj-PrefSubj"">
        <vzRsDenyRule directives="log" forceResolve="yes" rn="rsdenyRule-default" tCl="vzFilter"
tDn="uni/tn-common/flt-default" tRn="flt-default"/>
    </vzTSubj>
</vzTaboo>
```

GUI を使用した ACL 許可および拒否ログの表示

次の手順は、GUI を使用して、トラフィック フローの ACL 許可および拒否ログを（有効になっていれば）表示する方法を示しています。

ステップ 1 メニュー バーで、**[Tenants]** > **[<tenant name>]** の順に選択します。

ステップ 2 [Navigation] ペインで、[Tenant <tenant name>] をクリックします。

ステップ 3 **Tenants <tenant name> [Work]** ペインで、**[Operational]** タブをクリックします。

ステップ 4 [Operational] タブの下で、[Flows] タブをクリックします。
[Flows] タブの下で、いずれかのタブをクリックして、レイヤ 2 許可ログ（[L2 Permit]）、レイヤ 3 許可ログ（[L3 Permit]）、レイヤ 2 拒否ログ（[L2 Drop]）、またはレイヤ 3 拒否ログ（[L3 Drop]）のログ データを表示します。各タブで、トラフィックがフローしていれば、ACL ロギング データを表示できます。デー

タ ポイントは、ログ タイプと ACL ルールに応じて異なります。たとえば、[L3 Permit] ログおよび [L3 Deny] ログには次のデータ ポイントが含まれます。

- VRF
- Alias
- 送信元 IP アドレス
- 宛先 IP アドレス
- プロトコル
- 送信元ポート
- 宛先ポート
- 送信元 MAC アドレス
- 宛先 MAC アドレス
- Node
- 送信元インターフェイス
- VRF Encap
- 送信元 EPG
- 宛先 EPG
- 送信元 PC タグ
- 宛先 PC タグ

（注） また、[Flows] タブの横の [Packets] タブを使用して、シグニチャ、送信元、および宛先が同じであるパケットのグループ（最大 10 個）の ACL ログにアクセスできます。送信されたりドロップされたりするパケットのタイプを確認できます。

REST API を使用した ACL 許可および拒否ログ

次の例は、REST API を使用して、トラフィック フローのレイヤ 2 拒否ログ データを表示する方法を示しています。次の MO を使用してクエリを送信することができます。

- acllogDropL2Flow
- acllogPermitL2Flow
- acllogDropL3Flow
- acllogPermitL3Flow
- acllogDropL2Pkt

- acllogPermitL2Pkt
- acllogDropL3Pkt
- acllogPermitL3Pkt

始める前に

ACL契約許可および拒否ログのデータを表示する前に、許可または拒否ロギングを有効にする必要があります。

レイヤ 3 ドロップ ログ データを表示するには、REST API を使用して次のクエリを送信します。

```
GET https://apic-ip-address/api/class/acllogDropL3Flow
```

例：

次の例では、サンプル出力をいくつか示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="2">
    <acllogPermitL3Flow childAction="" dn="topology/pod-1/node-101/ndbgs/acllog/tn-common/ctx-inb

/permitl3flow-spctag-333-dpctag-444-sepgname-unknown-depgname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]


-dip-[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-

        [port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
dstMacAddr="00:00:12:00:00:25"
        dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn="" protocol="udp"
srcEpgName="unknown"
       srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0" srcMacAddr="00:00:15:00:00:28"
 srcPcTag="333"
        srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
    <acllogPermitL3Flow childAction="" dn="topology/pod-1/node-102/ndbgs/acllog/tn-common/ctx-inb

/permitl3flow-spctag-333-dpctag-444-sepgname-unknown-depgname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]-dip-


[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-
        [port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
dstMacAddr="00:00:12:00:00:25"
        dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn="" protocol="udp"
srcEpgName="unknown"
       srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0" srcMacAddr="00:00:15:00:00:28"
 srcPcTag="333"
        srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
</imdata>
```

NX-OS CLI を使用した ACL 許可および拒否ログの表示

次の手順は、NX-OS スタイル CLI **show acllog** コマンドを使用して ACL ログの詳細を表示する方法を示しています。

レイヤ 3 コマンドの構文は、**show acllog {permit | deny} l3 {pkt | flow} tenant <tenant_name> vrf <vrf_name> srcip <source_ip> dstip <destination_ip> srcport <source_port> dstport <destination_port> protocol <protocol> srcintf <source_interface> start-time <startTime> end-time <endTime> detail** です。

レイヤ 2 コマンドの構文は、**show acllog {permit | deny} l2 {flow | pkt} tenant <tenant_name> vrf <VRF_name> srcintf <source_interface> vlan <VLAN_number> detail** です。

（注） **show acllog** コマンドの完全な構文は、第二世代 Cisco Nexus 9000 シリーズ スイッチ（N9K-C93180LC-EX など名前の最後に EX または FX がつく。もしくはそれ以降のシリーズ）および Cisco APIC リリース 3.2 以降でのみ使用できます。第一世代のスイッチ（名前の最後に EX または FX が付かない）または 3.2 以前の Cisco APIC リリースでは、使用可能な構文は上記の通りです。

Cisco APIC 3.2 以降では、追加のキーワードが **detail** keyword:**[dstEpgName <destination_EPG_name>| dstmac <destination_MAC_address> | dstpctag <destination_PCTag> | srcEpgName <source_EPG_name> | srcmac <source_MAC_address> | srcpctag <source_PCTag>]** とともにコマンドの両方のバージョンに追加されます。

ステップ 1 次の例では、**show acllog drop l3 flow tenant common vrf default detail** コマンドを使用して、共通テナントのレイヤ 3 拒否ログに関する詳細情報を表示する方法を示します。

例：

```
apic1# show acllog deny l3 flow tenant common vrf default detail
SrcPcTag  : 49153
DstPcTag  : 32773
SrcEPG    : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg6
DstEPG    : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg5
SrcIp     : 16.0.2.10
DstIp     : 19.0.2.10
Protocol  : udp
SrcPort   : 17459
DstPort   : 8721
SrcMAC    : 00:00:15:00:00:28
DstMAC    : 00:00:12:00:00:25
Node      : 101
SrcIntf   : port-channel5
VrfEncap  : VXLAN: 2097153
```

この例では第二世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 2 次の例では、**show acllog deny l2 flow tenant common vrf tsw0connctx0 detail** コマンドを使用して、共通テナントのレイヤ 3 拒否ログに関する詳細情報を表示する方法を示します。

例：

```
apic1# show acllog deny l2 flow tenant common vrf tsw0connctx0 detail
SrcPcTag DstPcTag   SrcEPG           DstEPG          SrcMAC            DstMAC         Node   SrcIntf
   vlan
--------- --------- ----------------  --------------  ------------------  ------------------
-----   ---------  ------
 32773    49153   uni/tn-TSW       uni/tn-TSW  00:00:11:00:00:11  11:00:32:00:00:33   101     port-
     2
```

```
_Tenant0/ap-     _Tenant0/ap-                                                channel8

 tsw0AP0/epg-     tsw0AP0/epg-
 tsw0ctx0BD0epg5  tsw0ctx0BD0epg6
```

この例では第二世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 3 次の例では、**show acllog permit l3 pkt tenant <tenant name> vrf <vrf name> [detail]** コマンドを使用して、送信された一般的な VRF ACL レイヤ 3 許可パケットに関する詳細情報を表示する方法を示しています。

```
apic1# show acllog permit l3 pkt tenant common vrf default detail acllog permit l3 packets detail:
srcIp      : 10.2.0.19
dstIp      : 10.2.0.16
protocol   : udp
srcPort    : 13124
dstPort    : 4386
srcIntf    : port-channel5
vrfEncap   : VXLAN: 2097153
pktLen     : 112
srcMacAddr : 00:00:15:00:00:28
dstMacAddr : 00:00:12:00:00:25
timeStamp  : 2015-03-17T21:31:14.383+00:00
```

この例では第一世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

ステップ 4 次の例では、**show acllog permit l2 pkt tenant <tenant name> vrf <vrf name> srcintf <s interface>** コマンドを使用して、インターフェイス ポートチャネル 15 から送信されたデフォルトの VRF レイヤ 2 パケットに関する情報を表示する方法を示しています。

```
apic1# show acllog permit l2 pkt tenant common vrf default srcintf port-channel5
acllog permit L2 Packets
      Node          srcIntf       pktLen     timeStamp
 --------------  --------------  --------  --------------
                 port-channel5      1      2015-03-17T21:
                                           31:14.383+00:00
```

この例では第一世代のスイッチまたは 3.2 以前の Cisco APIC リリースでの出力を示します。

統計情報の収集にアトミック カウンタ ポリシーを使用する

アトミック カウンタ ポリシーを使用すると、エンドポイント、エンドポイント グループ、外部インターフェイス、および IP アドレスの組み合わせ間のトラフィックに関する統計を収集できます。収集された情報で、ファブリック内のドロップや誤ったルーティングを検出できるため、迅速なデバッグを実行し、アプリケーション接続の問題を切り分けることができます。

アトミック カウンタ

アトミック カウンタは、ファブリック内のエンドポイント、EPG、またはアプリケーション間の接続のトラブルシューティングに役立ちます。ユーザー レポート アプリケーションが遅くなったり、2つのエンドポイント間のトラフィック損失を監視するためにアトミック カウンタが必要になる場合があります。アトミック カウンタが提供する機能の1つは、トラブルチケットを予防的な監視モードにする機能です。たとえば、問題が断続的であり、オペレーターがチケットをアクティブに処理しているときに発生するとは限りません。

アトミック カウンタは、ファブリックでのパケット損失の検出に役立ち、接続の問題の原因をすばやく特定できます。アトミック カウンタでは、ファブリックで NTP を有効にする必要があります。

リーフ間（TEP 間）のアトミック カウンタは次を提供できます。

- ドロップ、承認および超過パケットのカウント
- 最後の 30 秒などの短期間のデータ収集、5 分、15 分、またはそれ以上の長期間のデータ収集
- スパイン トラフィックごとの詳細（TEP、リーフ、または VPC の数が 64 未満の場合に使用可能）
- 継続的なモニタリング

リーフ間（TEP間）アトミック カウンタは累積であり、クリアできません。ただし、30 秒のアトミック カウンタは 30 秒間隔でリセットされるため、断続的な問題や再発する問題の分離に使用できます。

テナントのアトミック カウンタは次を提供できます。

- ドロップ、承認および超過パケットを含む、ファブリック全体のトラフィックのアプリケーション固有カウンタ
- モードは次を含みます。
- エンドポイントからエンドポイントへの MAC アドレス、またはエンドポイントからエンドポイントへの IP アドレス。1 つのターゲット エンドポイントに複数の IP アドレスが関連付けられている可能性があることに注意してください。
- オプションのドリル ダウン付きの EPG ツー EPG
- EPG からエンドポイント
- EPG から *（任意）
- エンドポイントから外部 IP アドレス

（注） アトミック カウンタは、2つのエンドポイント間のパケット量を追跡し、これを測定値として使用します。これらは、ハードウェア レベルでのドロップやエラー カウンタを考慮していません。

ドロップされたパケットは、送信元が送信したよりも接続先が受信したパケットが少ない場合に計算されます。

超過パケットは、送信元が送信したよりも接続先が受信したパケットの方が多い場合に計算されます。

アトミック カウンタに関する注意事項および制約事項

- アトミック カウンタの使用は、エンドポイントが異なるテナントまたは同じテナント内の異なるコンテキスト（VRF）にある場合はサポートされません。
- Cisco APIC リリース 3.1(2m) 以降では、ファブリックのライフタイム内のパスで統計情報が生成されなかった場合、そのパスに対するアトミック カウンタは生成されません。また、**[トラフィック マップ（Traffic Map）]**（**[可視化（Visualization）]** タブにあるもので、**[操作（Operations）]** > **[可視化（Visualization）]**を Cisco APIC GUIで選択する）には、すべてのパスではなく、アクティブなパス、つまりファブリックの寿命のいずれかの時点で、トラフィックがあったパスだけが表示されます。
- IP アドレスが学習されない純粋なレイヤ 2 設定（IP アドレスは 0.0.0.0）では、エンドポイント/EPG 間および EPG/エンドポイント間のアトミック カウンタ ポリシーはサポートされません。この場合、エンドポイント間および EPG 間のポリシーはサポートされます。外部ポリシーは学習された IP アドレスが必要な Virtual Routing and Forwarding（VRF）ベースであり、サポートされます。
- アトミック カウンタの送信元または宛先がエンドポイントである場合、そのエンドポイントはスタティックではなくダイナミックである必要があります。ダイナミック エンドポイント（fv:CEp）とは異なり、スタティック エンドポイント（fv:StCEp）にはアトミック カウンタに必要な子オブジェクト（fv:RsCEpToPathEp）がありません。
- 中継トポロジでは、リーフ スイッチはすべてのスパイン スイッチを使用したフル メッシュにはなく、リーフ間（TEP 間）のカウンタは予期どおりに動作しません。
- リーフ間（TEP 間）アトミック カウンタの場合、トンネル数がハードウェア制限を上回ると、システムはモードをトレール モードからパス モードに変更し、ユーザにはスパインごとのトラフィックは表示されなくなります。
- アトミック カウンタはスパイン プロキシ トラフィックはカウントしません。
- ファブリックに入る前、またはリーフ ポートに転送される前にドロップされたパケット、アトミック カウンタによって無視されます。
- ハイパーバイザで切り替えられるパケット（同じポート グループとホスト）はカウントされません。
- アトミック カウンタには、アクティブなファブリック ネットワーク タイム プロトコル（NTP）ポリシーが必要です。
- アトミック カウンタは IPv6 の送信元と接続先で動作しますが、IPv4 アドレスと IPv6 アドレスを混在させて送信元 IP アドレスと接続先 IP アドレスを構成することはできません。
- 送信元または宛先として fvCEp を使用して設定されたアトミック カウンタ ポリシーでは、fvCEp 管理対象オブジェクトに存在する MAC アドレスおよび IP アドレスからのトラフィックと、両者へのトラフィックだけがカウントされます。fvCEp の管理対象オブジェクトで

IP アドレス フィールドが空の場合、その MAC アドレスとの間で送受信されるすべてのトラフィックが IP アドレスに関係なくカウントされます。Cisco APIC が fvCEp について複数の IP アドレスを学習している場合、前述のように、fvCEp 管理対象オブジェクト自体にある 1 つの IP アドレスのみがカウントされます。特定の IP アドレスとの送受信に関連したアトミック カウンタ ポリシーを設定するには、送信元または宛先として fvIp 管理対象オブジェクトを使用します。

- fvCEp の背後に fvIp が存在する場合は、fvCEp ベースのポリシーではなく fvIP ベースのポリシーを追加する必要があります。
- エンドポイントが同じ EPG に属している場合、IPv6 ヘッダーを持つレイヤ 2 ブリッジドトラフィックの、それらのエンドポイント間でのアトミック カウンタ統計は報告されません。
- EPG または ESG から L3Out EPG に流れるトラフィックに対してアトミック カウンタが機能するには、すべてのプレフィックスとマッチさせるため、0/0 ではなく 0/1 および 128/1 を使用して L3Out EPG を設定します。
- Cisco APIC のトラフィック マップ モードが「trial」に設定されていて、Cisco APIC が F1545 障害を生成した場合、この障害をクリアできる唯一の方法は、トラフィック マップ モードを「path」に設定することです。トラフィック マップ モードを変更するには、**[操作（Operations）]** > **[可視化（Visualization）]** に移動し、**[設定（Settings）]** をクリックし、[モード（Mode）] の**パス**を選択して、**[送信（Submit）]** をクリックします。これにより、入力と出力の両方でポートごとのトンネル統計が得られます。

  トライアル モードでは、トンネル論理インターフェイスの最大スケール インデックスに到達する可能性が高くなります。このモードは、より多くのソフトウェアおよびハードウェア リソースを消費します。論理インターフェイスは、ハードウェア内のトンネルに関連付けられている ID です。

  トレイル モードを指定したトンネル エンドポイント（TEP）間に単一のトンネルがある場合は、より多くのハードウェア リソースも消費されます。たとえば、6 つのファブリック ポートと 1 つのトンネルがある場合、ハードウェアは、トンネルの数にファブリック ポートの数を掛けた数に等しいエントリ数を消費します。

  ソフトウェアの場合、割り当てられた論理インターフェイスの数が 2048 を超えると、ハードウェアにエントリを作成できません。その結果、統計情報を取得できません。アトミック カウンタの場合、この問題は減少または超過として表示されることがあります。

  パス モードには、TEP のエントリだけがあります。vPC の場合、2 つのエントリがインストールされます。したがって、上限に達する可能性は低くなります。

アトミック カウンタの構成

ステップ 1 メニュー バーで、[Tenants] をクリックします。

ステップ 2 サブメニュー バーで、必要なテナントをクリックします。

ステップ 3 **Navigation** ウィンドウで、テナントを展開し、**Policies** を展開し、それから **Troubleshoot** を展開します。

ステップ 4 **Troubleshoot** の下で、**Atomic Counter Policy** を展開し、トラフィック トポロジを選択します。

エンドポイントの組み合わせ、エンドポイント グループ、外部インターフェイスおよびIPアドレス間のトラフィックを測定できます。

ステップ 5 必要なトポロジを右クリックして、**Add** *topology* **Policy** を選択し、**Add Policy** ダイアログボックスを開きます。

ステップ 6 [Add Policy] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドにポリシーの名前を入力します。

b) トラフィックの送信元の識別情報を選択するか、入力します。

必要な識別情報のソース（エンドポイント、エンドポイントのグループ、外部インターフェイス、または IP アドレス）によって異なります。

c) トラフィックの宛先の識別情報を選択するか、入力します。

d) （任意）（任意）[Filters]テーブルで+アイコンをクリックし、カウントするトラフィックのフィルタリングを指定します。

表示される [Create Atomic Counter Filter] ダイアログボックスで、IP プロトコル番号（たとえば TCP=6）によるフィルタリング、および送信元と宛先の IP ポート番号によるフィルタリングを指定できます。

e) [Submit] をクリックし、アトミック カウンタ ポリシーを保存します。

ステップ 7 [Navigation] ペインで、選択したトポロジの下の新しいアトミック カウンタ ポリシーを選択します。

ポリシー設定が [Work] ペインに表示されます。

ステップ 8 [Work] ペインで [Operational] タブをクリックし、[Traffic] サブタブをクリックして、アトミック カウンタの統計情報を表示します。

アトミック カウンタの有効化

アトミック カウンタを使用してファブリック内のドロップと誤ルーティングを検出し、アプリケーション接続の問題の迅速なデバッグと分離を可能にするには、次のいずれかのタイプのテナント アトミック カウンタ ポリシーを 1 つ以上作成します。

- EP_to_EP：エンドポイントからエンドポイント（**dbgacEpToEp**）
- EP_to_EPG：エンドポイントからエンドポイント グループ（**dbgacEpToEpg**）
- EP_to_Ext：エンドポイントから外部 IP アドレス（**dbgacEpToExt**）
- EPG_to_EP：エンドポイント グループからエンドポイント（**dbgacEpgToEp**）
- EPG_to_EPG：エンドポイント グループからエンドポイント グループ（**dbgacEpgToEpg**）
- EPG_to_IP：エンドポイント グループから IP アドレス（**dbgacEpgToIp**）
- Ext_to_EP：外部 IP アドレスからエンドポイント（**dbgacExtToEp**）
- IP_to_EPG：IP アドレスからエンドポイント グループ（**dbgacIpToEpg**）
- Any_to_EP：任意の場所からエンドポイント（**dbgacAnyToEp**）

• EP_to_Any ：エンドポイントから任意の場所（**dbgacEpToAny**）

ステップ 1 REST API を使用して EP_to_EP ポリシーを作成するには、次の例のような XML を使用します。

例：

```
<dbgacEpToEp name="EP_to_EP_Policy" ownerTag="" ownerKey=""
dn="uni/tn-Tenant64/acEpToEp-EP_to_EP_Policy" descr="" adminSt="enabled">
<dbgacFilter name="EP_to_EP_Filter" ownerTag="" ownerKey="" descr=""
srcPort="https" prot="tcp" dstPort="https"/>
</dbgacEpToEp>
```

ステップ 2 REST API を使用して EP_to_EPG ポリシーを作成するには、次の例のような XML を使用します。

例：

```
<dbgacEpToEpg name="EP_to_EPG_Pol" ownerTag="" ownerKey=""
dn="uni/tn-Tenant64/epToEpg-EP_to_EPG_Pol" descr="" adminSt="enabled">
<dbgacFilter name="EP_to_EPG_Filter" ownerTag="" ownerKey="" descr=""
srcPort="http" prot="tcp" dstPort="http"/>
<dbgacRsToAbsEpg tDn="uni/tn-Tenant64/ap-VRF64_app_prof/epg-EPG64"/>
</dbgacEpToEpg>
```

REST API でアトミック カウンターを使用したトラブルシューティング

ステップ 1 ファブリック内に展開されたエンドポイント間アトミック カウンタのリストと、ドロップされたパケットの統計情報やパケット数などの関連する詳細を取得するには、次の例のように XML で **dbgEpToEpTsIt** クラスを使用します。

例：

```
https://apic-ip-address/api/node/class/dbgEpToEpRslt.xml
```

ステップ 2 外部 IP からエンドポイントへのアトミック カウンタと関連する詳細のリストを取得するには、次の例のように、XML で **dbgacExtToEp** クラスを使用します。

例：

```
https://apic-ip-address/api/node/class/dbgExtToEpRslt.xml
```

デジタルオプティカルモニタリング統計の有効化と表示

リアルタイムのデジタル オプティカル モニタリング（DOM）データは SFP、SFP+、および XFP から定期的に収集され、警告およびアラームのしきい値テーブル値と比較されます。収集された DOM データは、トランシーバ送信バイアス電流、トランシーバ送信電力、トランシーバ受信電力、およびトランシーバ電源電圧です。

GUI を使用したデジタル オプティカル モニタリングの有効化

物理インターフェイスに関するデジタル オプティカル モニタリング（DOM）統計を表示するには、事前にポリシー グループに関連付けられたスイッチ ポリシーを使用して、リーフ インターフェイスまたはスパイン インターフェイスで DOM を有効にします。

GUI を使用して DOM を有効にするには：

ステップ 1 メニュー バーで、**[Fabric]** > **[Fabric Policies]** の順に選択します。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[モニタリング（Monitoring）]** > **[ファブリック ノード コントロール（Fabric Node Controls）]**を展開します。

ステップ 3 **[ファブリック ノード コントロール（Fabric Node Controls）]**を展開して、既存のポリシーのリストを表示します。

ステップ 4 **[作業（Work）]** ペインで **[アクション（ACTIONS）]** ドロップダウン メニューをクリックして、**[ファブリック ノード コントロールを作成（Create Fabric Node Control）]**を選択します。
[ファブリック ノード コントロールを作成（Create Fabric Node Controls）] ダイアログ ボックスが表示されます。

ステップ 5 **[ファブリック ノード コントロールを作成（Create Fabric Node Control）]** ダイアログ ボックスで、次の操作を実行します：

a) [Name] フィールドにポリシーの名前を入力します。
b) オプション。**[説明]** フィールドに、ポリシーの説明を入力します。
c) **[DOM を有効にする（Enable DOM）]** の横にあるボックスにチェックを入れます。

ステップ 6 **[送信]** をクリックしてポリシーを作成します。
これで、次の手順で説明するように、このポリシーをポリシー グループとプロファイルに関連付けることができます。

ステップ 7 **[ナビゲーション（Navigation）]** ウィンドウで **[スイッチポリシー（Switch Policies）]** > **[ポリシー グループ（Policy Groups）]** を展開します。

ステップ 8 **[作業（Work）]** ペインで、**[アクション（ACTIONS）]** ドロップダウン メニューをクリックし、**[リーフ スイッチ ポリシー グループを作成（Create Leaf Switch Policy Group）]**（スパインの場合は、**[スパイン スイッチ ポリシー グループを作成（Create Spine Switch Policy Group）]**）を選択します。
[リーフ スイッチ ポリシー グループの作成（Create Leaf Switch Policy Group）] または **[スパイン スイッチ ポリシー グループの作成（Create Spine Switch Policy Group）]** ダイアログ ボックスが表示されます。

ステップ 9 ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドにポリシー グループの名前を入力します。
b) **[ノード コントロール ポリシー（Node Control Policy）]** ドロップダウン メニューから、既存のポリシー（先ほど作成したものなど）を選択するか、**[ファブリック ノード コントロールを作成（Create Fabric Node Control）]** を選択して新しいポリシーを選択します。
c) [送信（Submit）] をクリックします。

ステップ 10 作成したポリシー グループを次のようにスイッチにアタッチします。

a) **[ナビゲーション（Navigation）]** ペインで、**[スイッチ ポリシー（Switch Policies）]** > **[プロファイル（Profiles）]** を展開します。

b) **[作業（Work）]**ペインで、**[アクション（ACTIONS）]**ドロップダウン メニューをクリックし、必要に応じて **[リーフ スイッチ プロファイルを作成（Create Leaf Switch Profile）]** または **[スパイン スイッチ プロファイルを作成（Create Spine Switch Profile）]** を選択します。
c) ダイアログボックスの中で、**[名前（Name）]** フィールドにプロファイルのための名前を入力します。field.
d) **[スイッチの関連付け（Switch Associations）]** で、プロファイルに関連付けるスイッチの名前を追加します。
e) **[ブロック（Block）]** プルダウンメニューから、該当するスイッチの横にあるボックスをオンにします。
f) **[ポリシー グループ（Policy Group）]** プルダウンメニューから、前に作成したポリシー グループを選択します。
g) **[アップデート（Update）]** をクリックし、**[送信（Submit）]** をクリックします。

REST API を使用したデジタル オプティカル モニタリングの有効化

物理インターフェイスに関するデジタル オプティカル モニタリング (DOM) 統計を表示するには、インターフェイスで DOM を有効にします。

REST API を使用して DOM を有効にするには：

ステップ 1 次の例のように、ファブリック ノード制御ポリシー（fabricNodeControlPolicy）を作成します。

```
<fabricNodeControl dn="uni/fabric/nodecontrol-testdom" name="testdom" control="1"
rn="nodecontrol-testdom" status="created" />
```

ステップ 2 次のように、ファブリック ノード制御ポリシーをポリシー グループに関連付けます。

```
<?xml version="1.0" encoding="UTF-8" ?>
 <fabricLeNodePGrp dn="uni/fabric/funcprof/lenodepgrp-nodegrp2" name="nodegrp2"
rn="lenodepgrp-nodegrp2" status="created,modified" >

    <fabricRsMonInstFabricPol tnMonFabricPolName="default" status="created,modified" />
    <fabricRsNodeCtrl tnFabricNodeControlName="testdom" status="created,modified" />

</fabricLeNodePGrp>
```

ステップ 3 次のように、ポリシー グループをスイッチに関連付けます（次の例では、スイッチは 103 です）。

```
<?xml version="1.0" encoding="UTF-8" ?>
 <fabricLeafP>
  <attributes>
   <dn>uni/fabric/leprof-leafSwitchProfile</dn>
   <name>leafSwitchProfile</name>
   <rn>leprof-leafSwitchProfile</rn>
   <status>created,modified</status>
  </attributes>
  <children>
   <fabricLeafS>
    <attributes>
     <dn>uni/fabric/leprof-leafSwitchProfile/leaves-test-typ-range</dn>
```

```
    <type>range</type>
    <name>test</name>
    <rn>leaves-test-typ-range</rn>
    <status>created,modified</status>
   </attributes>
   <children>
    <fabricNodeBlk>
     <attributes>
      <dn>uni/fabric/leprof-leafSwitchProfile/leaves-test-typ-range/nodeblk-09533c1d228097da</dn>
      <from_>103</from_>
      <to_>103</to_>
      <name>09533c1d228097da</name>
      <rn>nodeblk-09533c1d228097da</rn>
      <status>created,modified</status>
     </attributes>
    </fabricNodeBlk>
   </children>
   <children>
    <fabricRsLeNodePGrp>
     <attributes>
      <tDn>uni/fabric/funcprof/lenodepgrp-nodegrp2</tDn>
      <status>created</status>
     </attributes>
    </fabricRsLeNodePGrp>
   </children>
  </fabricLeafS>
 </children>
</fabricLeafP>
```

GUI を使用したデジタル オプティカル モニタリング統計の表示

GUI を使用して DOM 統計を表示するには：

始める前に

インターフェイスの DOM 統計を表示するには、事前にインターフェイスのデジタル オプティカル モニタリング（DOM）統計を有効にしておく必要があります。

ステップ 1 メニュー バーから **[ファブリック（Frabric）]** および **[インベントリ（Inventory）]**を選択します。

ステップ 2 [ナビゲーション（Navigation）] ペインで、調査対象の物理インターフェイスがあるポッドおよびリーフ ノードを展開します。

ステップ 3 **[インターフェイス（Interface）]** を展開します。

ステップ 4 **[物理 インターフェイス（Physical Interfaces）]**を拡大します。

ステップ 5 調査対象の物理インターフェイスを展開します。

ステップ 6 **[DOM 統計（DOM Stats）]** を選択します。
インターフェイスの DOM 統計が表示されます。

REST API によるデジタル オプティカル モニタリングを使用したトラブルシューティング

XML REST API クエリを使用して DOM 統計を表示するには：

始める前に

インターフェイスの DOM 統計を表示するには、事前にインターフェイスでデジタル オプティカル モニタリング (DOM) を有効にしておく必要があります。

次の例は、REST API クエリを使用して、ノード 104 の eth1/25 の物理インターフェイスで DOM 統計を表示する方法を示しています。

```
GET https://apic-ip-address/api/node/mo/topology/pod-1/node-104/sys/phys-[eth1/25]/phys/domstats.xml?
query-target=children&target-subtree-class=ethpmDOMRxPwrStats&subscription=yes
```

次の応答が返されます：

```
response : {
   "totalCount":"1",
      "subscriptionId":"72057611234705430",
          "imdata":[
{"ethpmDOMRxPwrStats":{
   "attributes":{
      "alert":"none",
      "childAction":"",
      "dn":"topology/pod-1/node-104/sys/phys[eth1/25]/phys/domstats/rxpower",
      "hiAlarm":"0.158490",
      "hiWarn":"0.079430",
      "loAlarm":"0.001050",
      "loWarn":"0.002630",
      "modTs":"never",
      "status":"",
       "value":"0.139170"}}}]}
```

正常性スコアの概要の表示

APIC は、ポリシー モデルを使用してデータを正常性スコアに組み入れます。正常性スコアはインフラストラクチャ、アプリケーション、またはサービスなどさまざまなエリアで集約できます。正常性スコアを使用すると、ネットワーク階層をドリルダウンして障害を特定の管理対象オブジェクト（MO）に分離することにより、パフォーマンスの問題を分離できます。アプリケーションの状態（テナントごと）またはリーフ スイッチの状態（ポッドごと）を表示することで、ネットワークの状態を表示できます。

正常性スコア、エラー、正常性 スコアの計算については、*Cisco APIC Fundamentals Guide* を参照してください。

正常性スコアのタイプ

APIC は次の正常性スコアのタイプをサポートします。

- システム — ネットワーク全体の正常性を要約します。
- リーフ：ネットワークのリーフ スイッチの正常性を要約します。リーフの正常性には、ファン トレイ、電源、および CPU を含むスイッチのハードウェア正常性が含まれます。
- テナント — テナントとテナントのアプリケーションの正常性を要約します。

正常性スコアによるフィルタ処理

次のツールを使用して、正常性スコアをフィルタ処理できます。

- 正常性スクロール バー：正常性スクロール バーを使って、どのオブジェクトを表示するかを指定できます。スコアを下げれば、正常性スコアの低いオブジェクトだけ見ることができます。
- 劣化した正常性スコアの表示：劣化した正常性スコアを表示するには、ギア アイコンをクリックし、**[劣化した正常性スコアのみを表示（Show only degraded health score）]** を選択します。

テナントの正常性の表示

アプリケーションの正常性を表示するには、メニュー バーで **[テナント（Tenants）]** > *[tenant-name]* をクリックし、次に **[ナビゲーション（Navigation）]** ペインでテナント名をクリックします。GUI がアプリケーションや EPG を含むテナントの正常性の要約を表示します。テナントの構成をドリルダウンするには、正常性スコアをダブルクリックします。

健全性の要約の場合は、**[仕事（Work）]** ペインの **[正常性（Health）]** タブをクリックします。ネットワークのこの表示が正常性スコアとネットワーク上の MO 間の関係を示すので、パフォーマンスの問題を分離し、解決することができます。たとえば、テナントのコンテクストの管理オブジェクトの共通シーケンスは、**[テナント（Tenant）] > [アプリケーション プロファイル（Application profile）] > [アプリケーション EPG（Application EPG）] > [EPP] > [ファブリックの場所（Fabric location）] > [EPG からパス アタッチメント（ EPG to Path Attachment ）] > [ネットワーク パス エンドポイント（Network Path Endpoint）] > [集約インターフェイス（Aggregation Interface）] > [集約されたインターフェイス（Aggregated Interface）] > [集約されたメンバー インターフェイス（Aggregated Member Interface）]** となります。

ファブリックの正常性の表示

ファブリックの正常性を表示するには、メニューバーの **[ファブリック（Fabric）]** をクリックします。**[ナビゲーション（navigation）]** のペインで、ポッドを選択します。GUI は、ノードを含むポッドの正常性の要約を表示します。ファブリック構成の一部をドリル ダウンするには、正常性 スコアをダブルクリックします。

健全性の要約の場合は、**[作業（work）]**ペインの**[正常性（Health）]**タブをクリックします。ネットワークのこの表示が正常性 スコアとネットワーク上の MO 間の関係を示すので、パフォーマンスの問題を分離し、解決することができます。たとえば、ファブリックのコンテキストにおける管理対象オブジェクトの共通シーケンスは、**[ポッド（Pod）]>[リーフ（Leaf）] > [シャーシ（Chassis）] > [ファン トレイ スロット（Fan tray slot）] > [回線モジュールのスロット（Line module slot）]>[回線モジュール（Line module ）]>[ファブリック ポート（Fabric Port）] > [レイヤ 1 物理インターフェイス構成（Layer 1 Physical Interface Configuration）] > [物理インターフェイス実行時間状態（Physical Interface Runtime State）]** です。

（注） 物理ネットワークの問題など、ファブリックの問題は、MO が直接関連するとテナントのパフォーマンスに影響を及ぼすことがあります。

Visore での MO 正常性の表示

Visore で MO の正常性を表示するには、**H** アイコンをクリックします。

次の MO を使って、正常性情報を表示します。

- 正常性：Inst
- 正常性：NodeInst
- オブザーバ：Node
- オブザーバ：Pod

Visore に関する詳細情報については、*Cisco* アプリケーション セントリック インフラストラクチャの基本ガイドを参照してください。

ログを使用する正常性スコアのデバッグ

次のログ ファイルを使用して、APIC の正常性 スコアをデバッグできます。

- svc_ifc_eventmgr.log
- svc_ifc_observer.log

ログを使用して正常性 スコアをデバッグする場合、次の項目を確認してください：

- syslog （エラーまたはイベント）の送信元を確認します。
- APIC で syslog ポリシーが構成されているかどうかを確認します。
- syslogポリシータイプとシビラティ（重大度）が正しく設定されているかどうかを確認します。
- コンソール、ファイル、リモート接続先、プロファイルを指定できます。リモート接続先の場合、syslog サーバーが実行中であり、到達可能であることを確認します。

エラーの表示

次の手順では、障害情報が表示される場所について説明します。

ステップ 1 障害ウィンドウに移動します。

- システム障害（System Faults）：メニュー バーから、**[システム（System）]** > **[障害（Faults）]** をクリックします。
- テナント障害：メニュー バーから：
  1. **[テナント（Tenants）]** > *[tenant-name]*] をクリックします。
  2. **[ナビゲーション（Navigation）]** ペインで、**[テナント（Tenant）]** *[*テナント名（*tenant name*）*]* をクリックします。
  3. **[作業（Work）]** ペインで、**[障害（Faults）]** タブをクリックします。
- ファブリック障害：メニューバーから：
  1. **[ファブリック（Fabric）]** > **[インベントリ（Inventory）]** をクリックします。
  2. **[ナビゲーション（Navigation）]** ペインで、**ポッド**をクリックします。
  3. **[作業（Work）]** ペインで、**[障害（Faults）]** タブをクリックします。

障害のリストが要約表に表示されます。

ステップ 2 障害をダブルクリックします。

ファブリック テーブルとシステム テーブルが変更され、クリックした障害の障害コードに一致する障害が表示されます。

a) ファブリックまたはシステムの障害から、サマリー テーブルの障害をダブルクリックして詳細を表示します。

[障害のプロパティ（Fault Properties）] ダイアログが表示され、次のタブが表示されます。

- **一般（General）**：以下を表示します。
  - **プロパティ（Properties）**：サマリー テーブルにある情報が含まれます
  - **詳細（Details）**：サマリーテーブルで見つかった障害情報、発生数、変更セット、および選択した障害の元、以前、および最高の重大度レベルが含まれます。
- **トラブルシューティング（Troubleshooting）**：次のとおり、表示します。
  - **トラブルシューティング（Troubleshooting）**：障害の説明と推奨されるアクションを含むトラブルシューティング情報が含まれています。
  - **監査ログ（Audit log）**：障害が発生する前にユーザーが開始したイベントの履歴を表示できるツール。指定した分数ごとに履歴が一覧表示されます。ドロップダウン矢印をクリックして、分数を調整できます。

- **履歴（History）**：影響を受けるオブジェクトの履歴情報を表示します

アップリンク障害検出のためのポート トラッキングの有効化

このセクションでは、GUI、NX-OS CLI、および REST API を使用してポート トラッキングを有効にする方法について説明します。

ファブリックポートの障害検出のためのポートトラッキングポリシー

ファブリック ポートの障害検出は、ポート トラッキング システム設定で有効にすることができます。ポートトラッキングポリシーは、リーフスイッチとスパインスイッチ間のファブリック ポート、およびティア 1 リーフスイッチとティア 2 リーフスイッチ間のポートのステータスを監視します。有効なポート トラッキングポリシーがトリガーされると、リーフ スイッチは、EPGによって導入されたスイッチ上のすべてのアクセスインターフェイスをダウンさせます。

[ポート トラッキングがトリガーされたときに APIC ポートを含める（Include APIC ports when port tracking is triggered）] オプションを有効にした場合、リーフスイッチがすべてのファブリック ポートへの接続を失うと（つまり、ファブリック ポートが 0 になると）、ポート トラッキングは Cisco Application Policy Infrastructure Controller（APIC）ポートを無効にします。Cisco APIC がファブリックに対してデュアルまたはマルチホームの場合にのみ、この機能を有効にしてください。Cisco APIC ポートを停止すると、デュアルホームの Cisco APIC の場合にセカンダリ ポートに切り替えるのに役立ちます。

（注） ポートトラッキングの設定は、**[システム（System）]**>>**[システム設定（System Settings）]**>>**[ポート トラッキング（Port Tracking）]** で行えます。

ポート トラッキング ポリシーは、ポリシーをトリガーするファブリック ポート接続の数と、指定されたファブリック ポートの数を超えた後にリーフスイッチアクセス ポートをバックアップするための遅延タイマーを指定します。

次の例は、ポート トラッキング ポリシーの動作を示しています。

- ポート トラッキング ポリシーは、ポリシーをトリガーする各リーフ スイッチのアクティブなファブリック ポート接続のしきい値が 2 であると指定しています。
- ポート トラッキング ポリシーは、リーフ スイッチからスパイン スイッチへのアクティブなファブリック ポート接続の数が 2 に低下したときにトリガーされます。
- 各リーフスイッチは、そのファブリック ポート接続を監視し、ポリシーで指定されたしきい値に従ってポート トラッキング ポリシーをトリガーします。

- ファブリック ポート接続が復旧すると、リーフ スイッチは遅延タイマーの設定時間が経過するのを待ってから、アクセス ポートを復旧します。これにより、トラフィックがリーフスイッチ アクセス ポートで再開可能になる前に、ファブリックが再コンバージェンスする時間が与えられます。大規模なファブリックでは、遅延タイマーをより長い時間に設定する必要がある場合があります。

（注） このポリシーを構成するときは注意してください。ポート トラッキングをトリガーする、アクティブなスパイン ポートの数に関するポート トラッキング設定が高すぎる場合、すべてのリーフ スイッチ アクセス ポートがダウンします。

GUI を使用したポート トラッキングの構成

この手順では、GUIを使用してポート トラッキング機能を使用する方法について説明します。

ステップ 1 **[システム（System）]**メニューから、**[システム設定（System Settings）]**を選択します。

ステップ 2 ナビゲーションウィンドウから**[ポート トラッキング（Port Tracking）]**を選択します。

ステップ 3 **[ポート トラッキング状態（Port tracking state）]** の横にある **[オン（on）]** を選択して、ポート トラッキング機能をオンにします。

ステップ 4 プロパティのポート トラッキング状態の横にある **[オフ（off）]** を選択して、ポート トラッキング機能をオフにします。

ステップ 5 （任意） **[遅延復元タイマー（Delay restore timer）]**をデフォルト（120 秒）からリセットします。

ステップ 6 ポート トラッキングがトリガーされる前に稼働しているアクティブなスパイン リンクの最大数（0 ～ 12 の任意の構成値）を入力します。

ステップ 7 **[送信（Submit）]** をクリックして、目的のポート トラッキング構成をファブリック上のすべてのスイッチにプッシュします。

NX-OS CLI を使用したポート トラッキング

この手順では、NX-OS CLI を使用してポート トラッキング機能を使用する方法について説明します。

ステップ 1 次のように、ポート トラッキング機能をオンにします。

例：

```
apic1# show porttrack
Configuration
Admin State                  : on
Bringup Delay(s)             : 120
Bringdown # Fabric Links up  : 0
```

ステップ 2 次のように、ポート トラッキング機能をオフにします。

例：

```
apic1# show porttrack
Configuration
Admin State                  : off
Bringup Delay(s)             : 120
Bringdown # Fabric Links up  : 0
```

REST API を使用した ポート トラッキング

始める前に

この手順では、REST API を使用してポート トラッキング機能を使用する方法について説明します。

ステップ 1 次のように REST API を使用してポート トラッキング機能をオンにします（**admin state: on**）：

```
<polUni>
<infraInfra dn="uni/infra">
<infraPortTrackPol name="default" delay="5" minlinks="4" adminSt="on">

</infraPortTrackPol>
</infraInfra>
</polUni>
```

ステップ 2 次のように REST API を使用してポート トラッキング機能をオフにします（**admin state: off**）：

```
<polUni>
<infraInfra dn="uni/infra">
<infraPortTrackPol name="default" delay="5" minlinks="4" adminSt="off">

</infraPortTrackPol>
</infraInfra>
</polUni>
```

デバイスのモニタリングおよび管理用 SNMP の構成

このセクションでは、GUI を使用して SNMP を構成する方法について説明します。

SNMP について

Cisco Application Centric Infrastructure（ACI）は、管理情報ベース（MIB）と通知（トラップ）を含む広範な SNMPv1、v2、および v3 のサポートを提供します。SNMP 標準では、Cisco ACI ファブリックを管理しモニタリングする各 MIB をサポートするサードパーティ製アプリケーションを使用できます。

SNMPv3 はさらに広範なセキュリティ機能を提供します。各 SNMPv3 デバイスで SNMP サービスを有効または無効にするように選択できます。また、各デバイスで SNMP v1 および v2 要求の処理方法を設定できます。

5.1(1) リリース以降、SNMPv3 は Secure Hash Algorithm-2（SHA-2）認証タイプをサポートします。

SNMP の使用方法の詳細については、『*Cisco ACI MIB Quick Reference*』を参照してください。

Cisco ACI での SNMP アクセスのサポート

（注） Cisco Application Centric Infrastructure（ACI）でサポートされる MIB の完全なリストについては、 http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/mib/list/mib-support.html を参照してください。

Cisco ACI での SNMP サポートは次のとおりです。

- SNMP 読み取りクエリー（Get、Next、Bulk、Walk）は、リーフおよびスパイン スイッチと Cisco Application Policy Infrastructure Controller（APIC）によってサポートされます。
- SNMP 書き込みコマンド（Set）は、リーフおよびスパイン スイッチまたは Cisco APIC によってサポートされません。
- SNMP トラップ（v1、v2c、および v3）は、リーフおよびスパイン スイッチと Cisco APIC によってサポートされます。

（注） Cisco ACI は最大 10 個のトラップ レシーバをサポートします。

- SNMPv3 は、リーフおよびスパイン スイッチと Cisco APIC によってサポートされます。
- Cisco APIC IPv6 アドレスを使用した SNMP はサポートされていません。

表 *1 : Cisco APIC* リリースでの *SNMP* サポートの変更

リリース	説明
1.2(2)	SNMP トラップの宛先として IPv6 サポートを追加。
1.2(1)	Cisco APIC コントローラの SNMP サポートを追加。以前のリリースでは、リーフおよびスパイン スイッチについてのみ SNMP がサポートされています。

GUI による SNMP ポリシーの設定

この手順では、ACI スイッチの SNMP ポリシーを設定し、有効にします。

始める前に

SNMP 通信を有効にするには、以下の設定が必要です。

- アウトオブバンド コントラクトを設定して SNMP トラフィックを許可します。SNMP トラフィックは、通常、SNMP 要求に UDP ポート 161 を使用します。
- 'mgmt' テナントで APIC アウトオブバンド IP アドレスを設定します。アウトオブバンド アドレスは APIC セットアップ時に設定されますが、アウトオブバンド コントラクトを有効にするには 'mgmt' テナントでアドレスを明示的に設定する必要があります。

ステップ 1 メニュー バーで、[Fabric] をクリックします。

ステップ 2 サブメニュー バーで、[Fabric Policies] をクリックします。

ステップ 3 **[Navigation]** ペインで、**[Pod Policies]** を展開します。

ステップ 4 **[Pod Policies]** の下で **[Policies]** を展開します。

ステップ 5 **[SNMP]** を右クリックし、**[Create SNMP Policy]** を選択します。

新しい SNMP ポリシーを作成する代わりに、次の手順で示されるものと同じ方法で **[default]** ポリシー フィールドを編集できます。

ステップ 6 SNMP ポリシーのダイアログボックスで、次の操作を実行します。

a) **[Name]** フィールドに、SNMP ポリシーの名前を入力します。

b) **[Admin State]** フィールドで、**[Enabled]** を選択します。

c) （任意） **[SNMP v3 Users]** テーブルで **[+]** アイコンをクリックし、**名前**を入力して、ユーザの認証データを入力し、**[Update]** をクリックします。

この手順は SNMPv3 アクセスが必要な場合のみ実行します。

d) **[コミュニティ ポリシー（Community Policies）]**テーブルで**[+]**アイコンをクリックし、**[名前（Name）]** を入力して、**[更新（Update）]** をクリックします。

コミュニティポリシー名の最大長は32文字です。名前には、アンダースコア（_）、ハイフン（-）、またはピリオド（。）の文字、数字、および特殊文字のみを使用できます。名前に @ 記号を含めることはできません。

e) **[Trap Forward Servers]** テーブルで、**[+]**アイコンをクリックし、外部サーバの**[IP Address]**を入力し、**[Update]**をクリックします。

ステップ 7 必須:許可された SNMP 管理ステーションを設定するには、SNMP ポリシーのダイアログボックスで、次の操作を実行します。

a) **[Client Group Policies]** テーブルで **[+]** アイコンをクリックし、**[Create SNMP Client Group Profile]** ダイアログボックスを開きます。

b) **[Name]** フィールドに、SNMP クライアント グループのプロファイル名を入力します。

c) **[Associated Management EPG]** ドロップダウン リストから管理 EPG を選択します。
d) **[Client Entries]** テーブルで **[+]** アイコンをクリックします。
e) **[Name]** フィールドにクライアントの名前を入力し、**[Address]** のフィールドにクライアントの IP アドレスを入力して、**[Update]** をクリックします。

（注） SNMP 管理ステーションが SNMPv3 を使用して APIC と接続する場合、APIC は SNMP クライアント グループのプロファイルに指定されたクライアント IP アドレスを強制しません。SNMPv3 の場合、管理ステーションが **[Client Entries]** リストに含まれている必要がありますが、SNMPv3 クレデンシャルのみでアクセス可能なため、IP アドレスが一致している必要はありません。

ステップ 8 [OK] をクリックします。

ステップ 9 **[送信（Submit）]** をクリックします。

ステップ 10 **[Pod Policies]** の下で **[Policy Groups]** を展開して、ポリシー グループを選択するか、または **[Policy Groups]** を右クリックし、**[Create POD Policy Group]** を選択します。

新しいポッドポリシー グループを作成することも、既存のグループを使用することもできます。ポッド ポリシー グループには、SNMP ポリシーに加えて他のポッド ポリシーを含めることができます。

ステップ 11 ポッド ポリシー グループのダイアログボックスで、次の操作を実行します。

a) **[Name]** フィールドに、ポッド ポリシー グループの名前を入力します。
b) **[SNMP Policy]** ドロップダウン リストから、設定した SNMP ポリシーを選択して、**[Submit]** をクリックします。

ステップ 12 **[Pod Policies]** の下で **[Profiles]** を展開し、**[default]** をクリックします。

ステップ 13 **[Work]** ペインで、**[Fabric Policy Group]** ドロップダウン リストから、作成したポッド ポリシー グループを選択します。

ステップ 14 [送信（Submit）] をクリックします。

ステップ 15 [OK] をクリックします。

GUI による SNMP トラップ通知先の設定

この手順では、SNMP トラップ通知を受信する SNMP マネージャのホスト情報を設定します。

（注） ACI は最大 10 個のトラップ レシーバをサポートします。10 個より多く設定すると、一部では通知が受信されません。

ステップ 1 メニュー バーで、[Admin] をクリックします。

ステップ 2 サブメニュー バーで、[External Data Collectors] をクリックします。

ステップ 3 **[Navigation]** ペインで、**[Monitoring Destinations]** を展開します。

ステップ 4 **[SNMP]** を右クリックし、**[Create SNMP Monitoring Destination Group]** を選択します。

ステップ 5 **[Create SNMP Monitoring Destination Group]** ダイアログボックスで、次の操作を実行します。

a) **[Name]** フィールドに、SNMP 通知先の名前を入力し、**[Next]** をクリックします。

b) **[Create Destinations]** テーブルで **[+]** アイコンをクリックし、**[Create SNMP Trap Destination]** ダイアログボックスを開きます。

c) **[ホスト名/IP（Host Name/IP）]** フィールドに、IPv4 または IPv6 アドレスまたは宛先ホストの完全修飾ドメイン名を入力します。

d) 通知先の**ポート**番号と SNMP **バージョン**を選択します。

e) SNMP v1 または v2c 通知先の場合、**[Security Name]** として設定したコミュニティ名の 1 つを入力し、**[v3 Security Level]** として **[noauth]** を選択します。

SNMP v1 または v2c セキュリティ名の最大長は 32 文字です。名前には、アンダースコア（_）、ハイフン（-）、またはピリオド（。）の文字、数字、および特殊文字のみを使用できます。SNMP v2c の場合、@ 記号も使用できます。

f) SNMP v3 通知先の場合、**[Security Name]** として設定したユーザ名の 1 つを入力し、必要な **[v3 Security Level]** を選択します。

SNMP v3 セキュリティ名の最大長は 32 文字です。名前は大文字または小文字で始まる必要があり、文字、数字、およびアンダースコア（_）、ハイフン（-）、ピリオド（。）、または@記号の特殊文字のみを使用できます。

g) **[Management EPG]** ドロップダウン リストから管理 EPG を選択します。

h) [**OK**] をクリックします。

i) **[完了（Finish）]** をクリックします。

GUI による SNMP トラップ ソースの設定

この手順では、ファブリック内のソース オブジェクトを選択して有効にし、SNMP トラップ通知を生成します。

ステップ 1 メニュー バーで、[Fabric] をクリックします。

ステップ 2 サブメニュー バーで、[Fabric Policies] をクリックします。

ステップ 3 [Navigation] ペインで、[Monitoring Policies] を展開します。

共通ポリシー、**デフォルト** ポリシーで SNMP ソースを作成することも、または新しいモニタリング ポリシーを作成することもできます。

ステップ 4 必要なモニタリング ポリシーを展開し、[Callhome/SNMP/Syslog] を選択します。

[Common Policy] を選択する場合は、[Common Policy] を右クリックして、[Create SNMP Source] を選択し、そのダイアログボックスで次の手順に従ってください。

ステップ 5 [Work] ペインで、[Monitoring Object] ドロップダウン リストから [ALL] を選択します。

ステップ 6 [Source Type] ドロップダウン リストから、[SNMP] を選択します。

ステップ 7 テーブルで + アイコンをクリックし、[Create SNMP Source] ダイアログボックスを開きます。

ステップ 8 [Create SNMP Source] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、SNMP ポリシーの名前を入力します。

b) [Dest Group] ドロップダウン リストから、通知を送信する既存の宛先を選択するか、または [Create SNMP Monitoring Destination Group] を選択して、新しい宛先を作成します。

SNMP の通知先グループを作成する手順は、別項で説明します。

c) [Submit] をクリックします。

SNMP を使用したシステムのモニタリング

個々のホスト（APIC またはその他のホスト）をリモートでモニタし、特定のノードの状態を確認できます。

SNMP を使用してシステムの CPU とメモリの使用状況をチェックし、CPU のスパイクが発生しているかどうかを確認できます。SNMP（ネットワーク管理システム）は、SNMP クライアントを使用して APIC の情報にアクセスし、情報を取得します。

リモートでシステムにアクセスして、情報がネットワーク管理システムのコンテキストに属するものかどうかを確認し、CPU またはメモリの使用量が多すぎないか、またはシステムやパフォーマンスの問題が発生しているかどうかを調べることができます。問題の原因がわかると、システムの正常性をチェックし、メモリまたは CPU の使用量が多すぎないかどうかを確認できます。

詳細については、「*Cisco ACI MIB Quick Reference Manual*」を参照してください。

トラフィック モニタリングの SPAN の構成

このセクションでは、SPAN のガイドラインと制約事項をリストし、SPAN セッションの構成方法について説明します。

SPAN の概要

スイッチドポートアナライザ（SPAN）ユーティリティを使って、詳細なトラブルシューティングの実行または特定のアプリケーションホストからトラフィックのサンプルを取得し、プロアクティブなモニタリングと分析を行うことができます。

SPANは1つ以上のポート、VLAN、またはエンドポイントグループ（EPG）からのトラフィックをコピーし、ネットワークアナライザによる分析のためにコピーしたトラフィックを1つ以上の送信先に送信します。このプロセスはどの接続デバイスも中断せず、ハードウェア内で実施されるので不要な CPU 負荷を防ぎます。

SPANセッションはソースが受信したトラフィック（入力トラフィック）、ソースから送信したトラフィック（出力トラフィック）、またはその両方をモニタリングするように設定できま

す。デフォルトでは、SPAN はすべてのトラフィックをモニタリングしますが、選択したトラフィックだけをモニタリングするようにフィルタを設定できます。

テナントまたはスイッチで SPAN を設定できます。スイッチ上で設定する場合、SPAN をファブリック ポリシーまたはアクセス ポリシーとして設定できます。

APIC は、SPAN（ERSPAN）のカプセル化されたリモート拡張をサポートします。

リリース 4.1(1i) 以降、次の機能がサポートされるようになりました。

- 送信元とポートチャネルが同じスイッチ上でローカルである限り、宛先として静的ポートチャネルを使用した、ローカル SPAN に対するサポート。

（注） APIC リリース 4.1(1i) 以降を実行していて、宛先として静的ポート チャネルを設定した後、4.1(1i) より前のリリースにダウングレードすると、これが原因で SPAN セッションが管理者無効状態になります。この機能は、リリース 4.1.(1i) より前には利用できませんでした。機能への影響はありません。

- レイヤ 3 インターフェイス フィルタリングを使用して送信元 SPAN を設定するときに、レイヤ 3 インターフェイスの IP プレフィックスを含める必要がなくなりました。
- 1 つ以上のフィルタ エントリのグループであるフィルタ グループ設定のサポート。フィルタ グループを使用すれば、受信したパケットを SPAN を使用して分析する必要があるかどうかを判断するために使用される一致基準が指定できます。
- ASIC の入力での転送が原因でドロップされたパケットをキャプチャし、事前設定された SPAN 宛先に送信する SPAN-on-drop 機能。SPAN-on-drop 設定には、アクセス ポートを SPAN 送信元として使用するアクセス ドロップ、ファブリック ポートを SPAN 送信元として使用するファブリック ドロップ、およびノード上のすべてのポートを SPAN 送信元として使用するグローバル ドロップの 3 種類があります。SPAN-on-drop は、通常の SPAN を使用し (CLI、GUI、および REST API 経由) とトラブルシューティング SPAN を使用して (CLI および REST API のみを経由) 設定されます。この機能の設定の詳細については、*GUI* を使用した *SPAN* の設定、*NX-OS* スタイル *CLI* を使用した *SPAN* の設定、および *REST API* を使用した *SPAN* の設定を参照してください。

マルチノード SPAN

APIC のトラフィックのモニタリング ポリシーは、各アプリケーション グループのすべてのメンバーと彼らが接続する場所を追跡するために、適切な範囲にポリシーのスパンを広げることが可能です。メンバーが移動すると、APIC は新しいリーフにポリシーを自動的にプッシュします。たとえば、エンドポイントが新しいリーフ スイッチに VMotion により移動すると、スパンの設定は自動的に調整されます。

ACI ファブリックは、カプセル化リモート SPAN（ERSPAN）形式の次の 2 つの拡張をサポートします。

- アクセスまたはテナント SPAN：VLAN をフィルタとして使用するかどうかにかかわらず、リーフ スイッチのフロント パネル ポートに対して実行されます。リーフ スイッチの Broadcom Trident 2 ASIC は、ERSPAN タイプ 1 形式とはわずかに異なるバージョンをサポートします。上記で参照したドキュメントで定義されている ERSPAN タイプ 1 フォーマットとは、GRE ヘッダーが 4 バイトのみであり、シーケンス フィールドがないという点で異なっています。GRE ヘッダーは常に次のようにエンコードされます–0x000088be。0x88be は ERSPAN タイプ 2 を示していますが、フィールドの残りの 2 バイトにより、これは 4 バイトの GRE ヘッダーを持つ ERSPAN タイプ 1 パケットとして識別されます。
- ファブリック SPAN：リーフ スイッチの Northstar ASIC により、またはスパイン スイッチの Alpine ASIC により実行されます。これらの ASIC は ERSPAN タイプ 2 および 3 フォーマットをサポートしていますが、ACI ファブリックは現在、ファブリック SPAN の ERSPAN タイプ 2 のみをサポートしています。これについては、上記のベースライン ドキュメントに記載されています。

ERSPAN ヘッダーの説明については、次の URL にある IETF インターネット ドラフトを参照してください。https://tools.ietf.org/html/draft-foschiano-erspan-00

SPAN の注意事項と制約事項

（注） 多くのガイドラインと制約事項は、スイッチが第 1 世代スイッチか第 2 世代スイッチかによって異なります。スイッチの生成は次のように定義されます。

- 第 1 世代スイッチは、スイッチ名の末尾に「EX」、「FX」、「FX2」などのサフィックスがないことで識別されます（N9K-9312TX など）。
- 第 2 世代スイッチは、スイッチ名の末尾に「EX」、「FX」、「FX2」などのサフィックスが付いています。

- サポートされる SPAN のタイプはさまざまです。
  - 第 1 世代のスイッチの場合、テナントおよびアクセス SPAN は、カプセル化された SPAN（ERSPAN）タイプ I を使用します（Cisco Application Policy Infrastructure Controller（APIC）GUI のバージョン 1 オプション）。
  - 第 2 世代スイッチの場合、テナントおよびアクセス SPAN は、カプセル化された SPAN（ERSPAN）タイプ II（Cisco APIC GUI のバージョン 2 オプション）を使用します。
  - ファブリック SPAN は ERSPAN タイプ II を使用します。
- リリース 5.2(3) 以降、ERSPAN は IPv6 接続先をサポートしています。
- 6.0(3) リリースは、次の SPAN 制限がある Cisco N9K-C9808 スイッチをサポートします。
  - 出力（トランジット（Tx））SPAN はサポートされていません。
  - ドロップ時の SPAN はサポートされていません。

- 複数のセッションで同じ SPAN 送信元を使用することはできません。
- SPAN は、最大 343 バイトの MTU をサポートします。

- uSeg EPG または ESG は、SPAN 送信元 EPG として使用できません。これは、SPAN 送信元フィルタが VLAN ID に基づいているためです。したがって、エンドポイントが uSeg EPG または ESG に分類されている場合でも、その VLAN が SPAN 送信元 EPG の VLAN である場合、エンドポイントからのトラフィックはミラーリングされます。
- ERSPAN セッションを構成するときに、SPAN ソースに GOLF VRF インスタンス内のスパイン スイッチからの宛先とインターフェイスが含まれている場合、L3Out プレフィックスが間違った BGP ネクストホップで GOLF ルータに送信され、GOLF からその L3Out への接続が切断されます。
- SPAN 送信元として l3extLIfP のレイヤ 3 サブインターフェイスを指定することはできません。外部ソースからのトラフィックをモニタリングするためにはポート全体を使用します。
- FEX インターフェイスのローカル SPAN では、FEX インターフェイスは SPAN 送信元としてのみ使用でき、SPAN 宛先としては使用できません。
  - 第 1 世代スイッチでは、レイヤ 3 スイッチド トラフィックに対して Tx SPAN は機能しません。
  - 第 2 世代のスイッチでは、トラフィックがレイヤ 2 またはレイヤ 3 のどちらでスイッチングされているかにかかわらず、Tx SPAN は機能しません。

  Rx SPAN に制限はありません。

  FEX ファブリック ポートチャネル（NIF）の SPAN の場合、メンバー インターフェイスは第 1 世代リーフ スイッチの SPAN 送信元インターフェイスとしてサポートされます。

（注） 第 2 世代スイッチで FEX ファブリック ポートチャネル（NIF）メンバー インターフェイスを SPAN 送信元インターフェイスとして設定することもできますが、これは Cisco APIC リリース 4.1 より前のリリースではサポートされていません。

  ERSPAN ヘッダーについては、IETF の Internet Draft（https://tools.ietf.org/html/draft-foschiano-erspan-00）を参照してください。
- ERSPAN 宛先 IP アドレスは、エンドポイントとしてファブリックで学習する必要があります。
- SPAN は IPv6 トラフィックをサポートします。
- ポート チャネルまたは vPC の個別ポート メンバーは送信元として設定されます。ポート チャネル、vPC、または vPC コンポーネントを SPAN セッションの送信元として使用します。

- 宛先 EPG が削除されるか使用できない場合、ERSPAN 送信元グループで障害は発生しません。
- SPAN フィルタは、第 2 世代のリーフ スイッチでのみサポートされます。

  アクセス SPAN 送信元は、特定の時点で次のいずれかのフィルタのみをサポートします。

  - EPG
  - 外部ルーティング（L3Out）

- L3Out フィルタを使用してアクセス SPAN 送信元を展開する場合は、L3Out が一致するインターフェイスにも展開されていることを確認します。
  - L3Out がポートに展開されている場合、SPAN 送信元は同じポートに展開する必要があります。
  - L3Out が PC に展開されている場合、SPAN 送信元は同じ PC に展開する必要があります。
  - L3Out が vPC に展開されている場合、SPAN 送信元は同じ vPC に展開する必要があります。
- L3Out ルーテッド インターフェイスおよびルーテッド サブインターフェイスはポートまたは PC に導入できますが、L3Out SVI はポート、PC、または vPC に導入できます。L3Out フィルタを使用する SPAN 送信元は、それに応じて展開する必要があります。
- L3Out フィルタは、ファブリック SPAN またはテナント SPAN セッションではサポートされません。
- EPG ブリッジ ドメインの [L3 設定（L3 Configuration）] タブで正しい L3Out を選択する必要があります。そうしないと、基本的な L3Out のパケット フローが機能しません。
- カプセル化値は、ルーテッドサブインターフェイスおよびSVIには必須ですが、ルーテッドインターフェイスには適用されません。L3Outサブインターフェイスまたは SVI カプセル化値は、EPG カプセル化値とは異なる必要があります。

  SPAN セッション内で EPG フィルタが有効になっている場合、中継、つまり tx 方向のインターフェイスから送信される ARP パケットはスパンされません。
- 次の場合、SPAN フィルタはサポートされません。
  - ファブリック ポート
  - ファブリックおよびテナント SPAN セッション
  - スパイン スイッチ
- 公式にサポートされているよりも多くの L4 ポート範囲を追加しようとしても、L4 ポート範囲フィルタ エントリは追加されません。

- SPAN 送信元グループ レベルまたは個々の SPAN 送信元レベルで、サポートされているフィルタ エントリより多くのエントリを関連付けようとすると、SPANセッションは起動しません。
- 公式にサポートされているよりも多くのフィルタ エントリを追加または削除すると、削除されたフィルタ エントリは TCAM に残ります。
- アクティブな SPAN セッションの最大数や、SPAN フィルタ制限など、SPAN 関連の制限については、『*Verified Scalability Guide for Cisco ACI*』ドキュメントを参照してください。
- SPAN-on-drop 機能では、次の注意事項と制限事項が適用されます。
  - SPAN-on-drop 機能は、第 2 世代リーフ スイッチでサポートされます。
  - SPAN-on-drop 機能は、LUX ブロック内の転送ドロップがあるパケットのみをキャプチャします。これは、入力での転送ドロップ パケットをキャプチャします。SPAN-on-drop 機能は、BMX（バッファ）ドロップおよび RWX（出力）ドロップをキャプチャできません。
  - トラブルシューティング CLI を使用して SPAN-on-drop と Cisco APICを有効にして宛先として SPAN セッションを作成する場合、100 MBの データがキャプチャされるとセッションは無効になります。
  - モジュラ シャーシでは、SPAN-on-drop 機能はライン カードでドロップされたパケットに対してのみ機能します。ファブリック カードでドロップされたパケットはスパンされません。
  - SPAN-on-drop ACL と他の SPAN ACL はマージされません。SPAN-on-drop セッションが ACL ベースの SPAN とともにインターフェイスで設定されている場合、そのインターフェイスでドロップされたパケットは SPAN-on-drop セッションにのみ送信されます。
  - SPAN on drop と SPAN ACL を同じセッションで設定することはできません。
  - アクセスまたはファブリック ポート ドロップ セッションとグローバル ドロップ セッションが設定されている場合、アクセスまたはファブリック ポート ドロップセッションがグローバル ドロップ セッションよりも優先されます。
  - TCAM でサポートされるフィルタ エントリの数=（M * S1 * 1 + N * S2 * 2）+（S3 * 2）。これは、rx SPAN または tx SPAN に個別に適用されます。現在この式に従うと、tx または rx SPAN でサポートされる最大フィルタ エントリは各方向で 480 です（また、フィルタ グループ アソシエーション（S3 = 0 を意味する）なしで、16 個のポート範囲を含む他の送信元が設定されていない場合））。フィルタ エントリの数が最大許容数を超えると、障害が発生します。フィルタエントリでレイヤ4ポート範囲を指定できることに注意してください。ただし、16 個のレイヤ 4 ポートが単一のフィルタ エントリとしてハードウェアにプログラムされます。

（注）
- M = IPv4 フィルタの数
- S1 = IPv4 フィルタを使用した送信元の数
- N = IPv6 フィルタの数
- S2 = IPv6 フィルタを使用した送信元の数
- S3 =フィルタ グループが関連付けられていない送信元の数

- PC または vPC の LACP ポリシーで MAC ピニングを設定すると、PC メンバー ポートは LACP 個別ポートモードになり、PC は動作しません。したがって、このような PC での SPAN 送信元設定は失敗し、「No operating src / dst」障害が生成されます。MAC ピニング モードが設定されている場合、SPAN は個々のポートでのみ設定できます。
- Cisco Application Centric Infrastructure（ACI）リーフ スイッチで受信されたパケットは、スパン インターフェイスが入力インターフェイスと出力インターフェイスの両方で設定されている場合でも、一度だけスパンされます。
- ルーテッド外部 SPAN 送信元フィルタを使用すると、Tx 方向のユニキャストのみが表示されます。Rx 方向では、ユニキャスト、ブロードキャスト、およびマルチキャストを確認できます。
- L3Out フィルタは、送信マルチキャスト SPAN ではサポートされません。L3Out は、入力 ACL フィルタでは sclass / dclass の組み合わせとして表されるため、ユニキャスト トラフィックのみを照合できます。送信マルチキャスト トラフィックは、ポートおよびポート チャネルでのみスパンできます。
- ポート チャネル インターフェイスを SPAN 宛先として使用できるのは、-EX 以降のスイッチだけです。
- SPAN フィルタ（5 タプル フィルタ）が適用されている場合、同じ送信元インターフェイスで複数の SPAN セッションを設定することはできません。

  リーフ スイッチのローカル SPAN 宛先ポートは、着信トラフィックを予期しません。レイヤ 2 インターフェイス ポリシーを設定し、**VLAN 範囲**プロパティを**グローバル範囲**ではなく**ポート ローカル範囲**に設定することで、スイッチが着信 SPAN 宛先ポート トラフィックをドロップするようにできます。このポリシーを SPAN 宛先ポートに適用します。レイヤ2インターフェイスポリシーを設定するには、GUI で次の場所に移動します。**[ファブリック（Fabric）]**>**[アクセス ポリシー（Access Policies）]**>**[ポリシー（Policies）]**>**[インターフェイス（Interface）]** > **[L2 インターフェイス（L2 Interface）]**

  特定のパケットに SPAN を設定すると、SPAN はそのパケットに対して 1 回だけサポートされます。最初の SSN の Rx の SPAN によってトラフィックが選択された場合、2 番目の SSN の Tx の SPAN によってトラフィックが再度選択されることはありません。したがって、SPAN セッションの入力ポートと出力ポートが単一のスイッチ上にある場合、SPAN

セッションのキャプチャは一方向のみです。SPAN セッションは双方向トラフィックを表示できません。

- フィルタ グループに設定された SPAN ACL フィルタは、アクセス インターフェイスから出力されるブロードキャスト、不明ユニキャスト、およびマルチキャスト（BUM）トラフィックをフィルタリングしません。出力方向の SPAN ACL は、ユニキャスト IPv4 または IPv6 トラフィックに対してのみ機能します。

  SPAN宛先をローカル ポートとして設定する場合、EPGはそのインターフェイスに展開できません。

  リーフ スイッチでは、VRF フィルタを持つ SPAN 送信元は、VRF インスタンスの下のすべての通常のブリッジ ドメインとすべてのレイヤ 3 SVI にマッチします。

  スパイン スイッチでは、VRF を持つ SPAN 送信元は、設定された VRF VNID トラフィックのみにマッチします。また、ブリッジ ドメイン フィルタは、ブリッジ ドメイン VNID トラフィックのみにマッチします。

- 独自の SPAN 拡張フィルタ エントリを作成する場合、拡張フィルタ エントリの管理対象オブジェクトを識別するために、**__UI__AUTO__CONFIG__DEFAULT__EXTENDED__MO** をオブジェクト名として使用することはできません。

- 同じ速度の SPAN 接続先インターフェイスを使用します。SPAN セッションによってモニターされるトラフィックは、接続先ポートがオーバーサブスクライブされていないが、他の SPAN 接続先ポートの 1 つがオーバーサブスクライブされている場合でも、SPAN バッファのドロップが原因でトラフィック損失が発生する可能性があります。SPANトラフィック レートは、接続先インターフェイスの速度が異なる場合、およびそれらの 1 つがオーバーサブスクライブされている場合、最も遅いSPAN接続先インターフェイス速度に制限されます。

- 構成されているどの送信元インターフェイスよりも高いSPAN接続先インターフェイス速度を使用し、マイクロバーストに十分な余裕がある速度を選択します。クラウドスケール ASIC は、SPAN クラスのマイクロバースト モニタリング オプションを提供しません。

GUI を使用した SPAN の設定

Cisco APIC GUI を使用したテナント SPAN セッションの設定

SPAN は、スイッチまたはテナントで設定できます。このセクションでは、Cisco APIC GUI を使用して、複製された送信元パケットをリモート トラフィック アナライザに転送するようにテナントの SPAN ポリシーを設定する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。フィールドを理解し、有効な値を決定するには、ダイアログ ボックスの右上隅にあるヘルプアイコン（**?**）をクリックしてヘルプ ファイルを表示します。

ステップ 1 メニュー バーで、[Tenants] をクリックします。

ステップ 2 サブメニュー バーで、送信元エンドポイントを含むテナントをクリックします。

ステップ 3 **[ナビゲーション（Navigation）]**ペインでテナントを展開し、**[ポリシー（Policies）]**>**[トラブルシューティング（Troubleshooting）] を展開して、** > **[SPAN]** を展開します。

[SPAN] に表示される 2 つのノード：**[SPAN 宛先グループ（SPAN Destination Groups）]** と **[SPAN 送信元グループ（SPAN Source Groups）]**。

ステップ 4 **[ナビゲーション（Navigation）]**の下で **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Group）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 5 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスの必須フィールドに適切な値を入力します。

ステップ 6 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成]** ダイアログ ボックスを開きます。

ステップ 7 **[SPAN送信元の作成（Create SPAN Source）]**ダイアログボックスのフィールドに適切な値を入力します。

ステップ 8 SPAN送信元の作成が完了したら、**[OK]** をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 9 **[リモート場所の作成（Create Remote Location）]** ダイアログのフィールドに値を入力したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 送信先のトラフィック アナライザを使用して、SPAN 送信元 EPGからのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

APIC GUI を使用した SPAN フィルタ グループの設定

ステップ 1 メニューバーで**[ファブリック（Fabric）]**をクリックし、サブメニューバーで**[アクセス ポリシー（Access Policies）]**をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** を展開し、**[SPAN]** を展開します

ステップ 3 **[SPAN]** の下で **[SPAN フィルタ グループ（SPAN Filter Groups）]** を右クリックし、**[SPAN フィルタ グループの作成（Create SPAN Filter Group）]** を選択します。
[フィルタ グループの作成（Create Filter Group）] ダイアログボックスが表示されます。

ステップ 4 SPAN フィルタ グループの名前を入力します。**[フィルタ エントリ（Filter Entries）]** テーブルで、**[+]** をクリックし、次のフィールドに値を入力します。

- **[送信元 IP プレフィックス（Source IP Prefix）]** : *IP* アドレス/マスクの形式で送信元 IP アドレスを入力します。IPv4 アドレスおよびIPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- **[最初の送信元ポート（First Source Port）]** 最初の送信元レイヤー 4 ポートを入力します。このフィールドは、**[最後の送信元ポート（Last Source Port）]** フィールドとともに、送信元ポートをフィルタリ

ングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。

- **[最後の送信元ポート（Last Source Port）]** 最後の送信元レイヤー 4 ポートを入力します。このフィールドは、**[最初の送信元ポート（First Source Port）]** フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[宛先 IP プレフィックス（Destination IP Prefix）]** : *IP* アドレス/マスクの形式で宛先 IP アドレスを入力します。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- **[最初の宛先ポート（First Destination Port）]** : 最初の宛先レイヤー 4 ポートを入力します。このフィールドは、**[最後の宛先ポート（Last Destination Port）]** フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[最後の宛先ポート（Last Destination Port）]** : 最後の宛先レイヤー 4 ポートを入力します。このフィールドは、**[最初の宛先ポート（First Destination Port）]** フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[IP プロトコル（IP Protocol）]** : IP プロトコルを入力します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- **[拡張フィルタ エントリ（Extended Filter Entries）]** テーブルで、**[+]** をクリックし、次のフィールドに値を入力します。
  - **[名前（Name）]:** 拡張フィルタ エントリの名前を入力します。
  - **[最初の DSCP（DSCP From）]** : DSCP 値を入力します。このフィールドは、**[最後の DSCP（DSCP To）]** フィールドとともに、DSCP 値をフィルタリングする範囲を指定します。
  - **[最後ののDSCP（DSCP To）]** : DSCP 値を入力します。このフィールドは、**[最初の DSCP（DSCP From）]** フィールドとともに、DSCP 値をフィルタリングする範囲を指定します。
  - **[最初の Dot1P（Dot1P From）]** : Dot1P 値を入力します。このフィールドは、**[最後の Dot1P（Dot1P To）]** フィールドとともに、Dot1P 値をフィルタリングする範囲を指定します。
  - **[最後の Dot1P（Dot1P To）]** : Dot1P 値を入力します。このフィールドは、**[最初の Dot1P（Dot1P From）]** フィールドとともに、Dot1P 値をフィルタリングする範囲を指定します。

    送信元ポートと宛先ポートの範囲、または DSCP と Dot1P の範囲の値を指定できます。送信元ポートと宛先ポートの範囲、および DSCP と Dot1P の範囲の両方を指定すると、障害が表示されます。

    DSCP または Dot1P は、出力方向ではサポートされていません。方向として **[両方（Both）]** を選択した場合、DSCP または Dot1P のいずれかが入力方向のみでサポートされ、出力方向ではサポートされません。
  - **[TCP フラグ（TCP Flags）]** ドロップダウンリストで、**TCPフラグ**を選択します。

TCP フラグを設定できるのは、フィルタ グループのドロップダウン リストで **[未指定（Unspecified）]** または **[TCP]** を **[IP プロトコル（IP Protocol）]** として選択した場合だけです。

- **[パケット タイプ（Packet Type）]**：パケット タイプを選択します。**[ルート/スイッチ（Routed/Switched）]**、**[ルート（Routed）]**、または **[スイッチのみ（Switched Only）]** のいずれかを選択します。

ステップ 5 このフォームの各フィールドに適切な値を入力したら、**[更新（Update）]** をクリックし、**[送信（Submit）]** をクリックします。

NX-OS スタイルの CLI を使用した拡張フィルタによる SPAN フィルタの設定

次の例は、CLI を使用して SPAN フィルタと拡張フィルタを設定する方法を示しています。

CLI を使用して SPAN フィルタと拡張フィルタを設定するには：

例：

```
apic1(config-monitor-access-filtergrp-filter-extended-filters)# show run
# Command: show running-config monitor access filter-group filtergroup1 filter dstaddr 192.168.10.1
 srcaddr 192.168.10.100 extended-filters ext1
# Time: Wed May 11 11:25:23 2022
  monitor access filter-group filtergroup1
    filter srcaddr 192.168.10.100 dstaddr 192.168.10.1
      extended-filters ext1
        dscp from CS0 to 4
        dot1p from 1 to 5
        forwarding-type switched
        tcp-flag ack off
        tcp-flag fin off
        tcp-flag rst on
        exit
      exit
    exit
apic1#
```

REST API を使用した拡張フィルタによる SPAN フィルタの設定

次の例は、REST API を使用して SPAN フィルタを設定する方法を示しています。

Rest API を使用して SPAN フィルタを設定するには:

例：

```
URL: {{apic-host}}/api/node/mo/.xml
BODY:
<polUni>
   <infraInfra dn="uni/infra">
      <spanSrcGrp adminSt="enabled" descr="" dn="uni/infra/srcgrp-local1" nameAlias="" ownerKey=""

       ownerTag="">
```

```
            <spanRsSrcGrpToFilterGrp tDn="uni/infra/filtergrp-two" />
            <spanSrc descr="" dir="both" name="src1" nameAlias="" ownerKey="" ownerTag="">
            <spanRsSrcToPathEp tDn="topology/pod-1/paths-101/pathep-[eth1/15]" />
            </spanSrc>
            <spanSpanLbl descr="" name="dest1" nameAlias="" ownerKey="" ownerTag="" tag=
             "yellow-green" />
          </spanSrcGrp>
          <spanDestGrp annotation="" descr="" dn="uni/infra/destgrp-dest1" nameAlias="" ownerKey=""
           ownerTag="">
            <spanDest annotation="" descr="" name="destg" nameAlias="" ownerKey="" ownerTag="">
            <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-101/pathep-
             [eth1/7]" />
             </spanDest>
          </spanDestGrp>
          <spanFilterGrp name="two">
            <spanFilterEntry name="udp_two" ipProto="udp" srcAddr="1002::1/64" dstAddr="1001::1/64"
             srcPortFrom="1" srcPortTo="2" dstPortFrom="1" dstPortTo="2">
               <spanExtendedFltEntry name="arun1" dscpFrom="0" dscpTo="10" dot1pFrom="0" dot1pTo="7"

                tcpFlags="128" v6FlowLabel="1522" forwardingVal="switched" />
             </spanFilterEntry>
          </spanFilterGrp>
       </infraInfra>
</polUni>
```

APIC GUI を使用したアクセス SPAN ポリシーの設定

この手順では、Cisco APIC GUI を使用してアクセス SPAN ポリシーを設定します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

ステップ 1 メニュー バーで、**[ファブリック（Fabric）]** > **[アクセス ポリシー（Access Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Groups）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 4 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスを開いて、必須のフィールドに適切な値を入力します。

ステップ 6 [Create SPAN Source] ダイアログ ボックスで、[Add Source Access Paths] を展開して、ソース パスを指定します。

[送信元をパスに関連付ける（Associate Source to Path）] ダイアログ ボックスが表示されます。

ステップ 7 **[送信元をパスに関連付ける（Associate Source to Path）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 8 送信元とパスの関連付けが完了したら、**[OK]** をクリックします。

[SPAN 送信元の作成（Create SPAN Source）] ダイアログ ボックスに戻ります。

ステップ 9 SPAN 送信元の作成が完了したら、**[OK]** をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 10 SPAN 送信元グループの設定が完了したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

Cisco APIC GUI を使用したファブリック SPAN ポリシーの設定

このセクションでは、Cisco APIC GUI を使用してファブリック SPAN ポリシーを作成する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

ステップ 1 メニュー バーで、**[ファブリック（Fabric）]** > **[ファブリック ポリシー（Fabric Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Groups）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 4 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成]** ダイアログ ボックスを開きます。

ステップ 6 **[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 7 完了したら、[OK] をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 8 **[リモート場所の作成（Create Remote Location）]** ダイアログのフィールドに値を入力したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

APIC GUI を使用した外部アクセス用のレイヤ 3 EPG SPAN セッションの設定

この手順は、Cisco APIC GUI を使用して外部アクセス用のレイヤ 3 EPG SPAN ポリシーを設定する方法を示しています。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

ステップ 1 メニュー バーで、**[ファブリック（Fabric）]** > **[アクセス ポリシー（Access Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 送信元グループ（SPAN Source Groups）]** を右クリックし、**[SPAN 送信元グループの作成（Create SPAN Source Groups）]** を選択します。
[Create SPAN Source Group] ダイアログが表示されます。

ステップ 4 **[SPAN 送信元グループの作成（Create SPAN Source Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 **[フィルタ グループ（Filter Group）]** フィールドで、フィルタ グループを選択または作成します。

詳細については、APIC GUI を使用した SPAN フィルタ グループの設定 （65 ページ）を参照してください。

ステップ 6 **[送信元の作成（Create Sources）]** テーブルを展開し、**[SPAN 送信元の作成（Create SPAN Source）]** ダイアログ ボックスを開きいて、以下の操作を実行します。

a) 送信元ポリシーの**[名前（Name）]** を入力します。
b) トラフィック フローの **[方向（Direction）]** オプションを選択します。
c) (オプション) **[ドロップ パケットのスパニング（Span Drop Packets）]** チェックボックスをクリックしてチェックマークを付けます。オンにすると、SPAN-on-drop 機能が有効になります。
d) 外部アクセスの場合は、**[外部にルーティング（Routed Outside）]**（**[タイプ（Type）]** フィールド）をクリックします。

（注） 外部アクセスで**[外部にルーティング（Routed Outside）]** を選択した場合、**[名前（Name）]**、**[アドレス（Address）]**、および **[Encap]** フィールドが表示されて、**[L3 Outside]** を設定できるようになります。

e) **[送信元アクセス パスの追加（Add Source Access Paths）]** を展開して、送信元パスを指定します。

[送信元をパスに関連付ける（Associate Source to Path）] ダイアログ ボックスが表示されます。

f) **[送信元をパスに関連付ける（Associate Source to Path）]** ダイアログ ボックスのフィールドに適切な値を入力します。

g) 送信元とパスの関連付けが完了したら、**[OK]** をクリックします。

[SPAN 送信元の作成（Create SPAN Source）] ダイアログ ボックスに戻ります。

h) SPAN 送信元の作成が完了したら、**[OK]** をクリックします。

[SPAN 送信元グループの作成（Create VRF）] ダイアログ ボックスに戻ります。

ステップ 7 SPAN 送信元グループの設定が完了したら、**[送信（Submit）]** をクリックします。

次のタスク

SPAN 宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

Cisco APIC GUI を使用したアクセス SPAN ポリシーの宛先グループの設定

このセクションでは、Cisco APIC GUI を使用して、アクセス SPAN ポリシーの宛先グループを作成する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

SPAN宛先グループと送信元を作成すれば、SPAN宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

ステップ 1 メニュー バーで、**[ファブリック（Fabric）]** > **[アクセス ポリシー（Access Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）]** > **[トラブルシューティング（Troubleshooting）]** > **[SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 宛先グループ（SPAN Destination Groups）]** を右クリックして、**[SPAN 宛先グループの作成（Create SPAN Destination Groups）]** を選択します。
[Create SPAN Destination Group] ダイアログが表示されます。

ステップ 4 **[SPAN 宛先グループの作成（Create SPAN Destination Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 完了したら、[送信（Submit）] をクリックします。

宛先グループが作成されます。

Cisco APIC GUI を使用したファブリック SPAN ポリシーの宛先グループの設定

このセクションでは、Cisco APIC GUI を使用して、ファブリック SPAN ポリシーの宛先グループを作成する方法について説明します。設定手順では、1 つ以上の GUI ダイアログ ボックスのフィールドに値を入力する必要があります。

SPAN宛先グループと送信元を作成すれば、SPAN宛先のトラフィック アナライザを使用して、SPAN 送信元からのデータ パケットを観察し、パケット形式、アドレス、プロトコルおよびその他の情報を確認できます。

ステップ 1 メニュー バーで、**[ファブリック（Fabric）] > [ファブリック ポリシー（Fabric Policies）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで、**[ポリシー（Policies）] > [トラブルシューティング（Troubleshooting）] > [SPAN]** を展開します。

[SPAN] の下には、**[SPAN 送信元グループ（SPAN Source Groups）]**, **[SPAN フィルタ グループ（SPAN Filter Groups）]**、および **[SPAN 宛先グループ（SPAN Destination Groups）]** の 3 つのノードが表示されます。

ステップ 3 **[SPAN 宛先グループ（SPAN Destination Groups）]** を右クリックして、**[SPAN 宛先グループの作成（Create SPAN Destination Groups）]** を選択します。
[Create SPAN Destination Group] ダイアログが表示されます。

ステップ 4 **[SPAN 宛先グループの作成（Create SPAN Destination Group）]** ダイアログ ボックスのフィールドに適切な値を入力します。

ステップ 5 完了したら、[送信（Submit）] をクリックします。

宛先グループが作成されます。

次のタスク

まだ作成していない場合は、ファブリック SPAN ポリシーの送信元を設定します。

NX-OS Style CLI を使用した SPAN の設定

NX-OS スタイルの CLI を使用したアクセス モードでのローカル SPAN の設定

これは、アクセス リーフ ノードにローカルな従来の SPAN 設定です。1 つ以上のアクセス ポートまたはポート チャネルから発信されたトラフィックをモニタリングし、同じリーフ ノードにローカルな宛先ポートに送信できます。

ステップ 1 **configure terminal**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor access session** *session-name*

アクセス モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor access session mySession
```

ステップ 3 [**no**] **description** *text*

このアクセス モニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-access)# description "This is my SPAN session"
```

ステップ 4 [**no**] **destination interface ethernet** *slot/port* **leaf** *node-id*

宛先インターフェイスを指定します。宛先インターフェイスを FEX ポートにすることはできません。

例：

```
apic1(config-monitor-access)# destination interface ethernet 1/2 leaf 101
```

ステップ 5 [**no**] **source interface ethernet** {[*fex/*]*slot/port* | *port-range*} **leaf** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-access)# source interface ethernet 1/2 leaf 101
```

ステップ 6 **drop enable**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-access-source)# drop enable
```

ステップ 7 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# direction tx
```

ステップ 8 [**no**] **filter tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

モニタリングするトラフィックのフィルタ処理を行います。フィルタは、送信元ポート範囲ごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
```

ステップ 9 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-access-source)# exit
```

ステップ 10 [**no**] **destination interface port-channel** *port-channel-name-list* **leaf** *node-id*

宛先インターフェイスを指定します。宛先インターフェイスを FEX ポートにすることはできません。

（注） リリース 4.1(1) 以降、コマンド例に示すように、宛先インターフェイスとしてスタティック ポート チャネルを使用できるようになりました。

例：

```
apic1(config-monitor-access)# destination interface port-channel pc1 leaf 101
```

ステップ 11 [**no**] **source interface port-channel** *port-channel-name-list* **leaf** *node-id* [**fex** *fex-id*]

送信元インターフェイス ポート チャネルを指定します。

（トラフィックの方向とフィルタ設定を入力します。ここには表示されていません）。

例：

```
apic1(config-monitor-access)# source interface port-channel pc5 leaf 101
```

ステップ 12 [**no**] **filter tenant** *tenant-name* **l3out** *L3Out-name* **vlan** *interface-VLAN*

モニタリングするトラフィックのフィルタ処理を行います。フィルタは、送信元ポート範囲ごとに独立して設定できます。

（注） リリース 4.1(1) 以降、例に示すように、L3Out インターフェイス フィルタリングを設定するときに IP プレフィックスを指定する必要がなくなりました。

例：

```
apic1(config-monitor-access-source)# filter tenant t1 l3out l3out1 vlan 2820
```

ステップ 13 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-access)# no shut
```

例

この例は、ローカル アクセス モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-access)# description "This is my SPAN session"
apic1(config-monitor-access)# destination interface ethernet 1/2 leaf 101
apic1(config-monitor-access)# source interface ethernet 1/1 leaf 101
apic1(config-monitor-access)# drop enable
apic1(config-monitor-access-source)# direction tx
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access)# no shut
apic1(config-monitor-access)# show run
# Command: show running-config monitor access session mySession
# Time: Fri Nov 6 23:55:35 2015
  monitor access session mySession
    description "This is my SPAN session"
    destination interface eth 1/2 leaf 101
    source interface eth 1/1 leaf 101
      direction tx
      filter tenant t1 application app1 epg epg
      exit
    exit
```

NX-OS スタイルの CLI を使用した SPAN フィルタ グループの設定

次の手順では、SPAN フィルタ グループとフィルタ エントリを設定する方法について説明します。

ステップ 1 **configure**

グローバル設定モードを開始します。

例：

```
apic1# configure
```

ステップ 2 [**no**] **monitor access filter-group** *filtergroup-name*

アクセス モニタリング フィルタ グループ設定を作成します。

例：

```
apic1(config)# monitor access filter-group filtergroup1
```

ステップ 3 [**no**] **filter srcaddress** *source-address* **dstaddress** *destination-address* **srcport-from** *source-from-port* **srcport-to** *source-to-port* **dstport-from** *destination-from-port* **dstport-to** *destination-to-port* **ipproto** *IP-protocol*

フィルタ グループのフィルタ エントリを設定します。ここで、

- *source-address* は、*IP* アドレス/マスク 形式の送信元 IP アドレスです。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- *destination-address* は、*IP* アドレス/マスク形式の宛先 IP アドレスです。IPv4 アドレスおよび IPv6 アドレスのどちらもサポートされています。**0.0.0.0** の値は、このフィールドで **任意の** IPv4 アドレス エントリを指定するために、**::** の値は、**任意の** IPv6 アドレスエントリを指定するために使用します。
- *source-from-port* は、最初の送信元レイヤ 4 ポートです。このフィールドは、**srcport-to**フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *source-to-port* は、最後の送信元レイヤ 4 ポートです。このフィールドは、**srcport-from**フィールドとともに、送信元ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *destination-from-port* は、最初の宛先レイヤ 4 ポートです。このフィールドは、**dstport-to**フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *destination-to-port* は、最後の宛先レイヤ 4 ポートです。このフィールドは、**dstport-from**フィールドとともに、宛先ポートをフィルタリングするためのポート範囲を指定します。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。
- *IP-protocol* は IP プロトコルです。値 **0** は、このフィールドで**任意の**エントリを指定するために使用します。

例：

```
apic1(config-monitor-fltgrp)# filter srcaddress 1.1.1.0/24 dstaddress 0.0.0.0 srcport-from 0 srcport-to
 0 dstport-from 0 dstport-to 0 ipproto 20
```

ステップ 4 **exit**

アクセス モニター フィルタ グループ設定モードに戻ります。

例：

```
apic1(config-monitor-fltgrp)# exit
```

ステップ 5 **exit**

グローバル コンフィギュレーション モードを終了します。

例：

```
apic1(config)# exit
```

例

この例は、SPAN フィルタ グループとフィルタ エントリを設定する方法を示しています。

```
apic1# configure
apic1(config)# monitor access filter-group filtergroup1
apic1(config-monitor-fltgrp)# filter srcaddress 1.1.1.0/24 dstaddress 0.0.0.0 srcport-from
 0 srcport-to 0 dstport-from 0 dstport-to 0 ipproto 20
apic1(config-monitor-fltgrp)# exit
apic1(config)# exit
```

NX-OS スタイルの CLI を使用した SPAN フィルタ グループの関連付け

次の手順では、フィルタ グループを SPAN 送信元グループに関連付ける方法について説明します。

ステップ 1 **configure**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure
```

ステップ 2 [**no**] **monitor access session** *session-name*

アクセス モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor access session session1
```

ステップ 3 **filter-group** *filtergroup-name*

フィルタ グループを関連付けます。

例：

```
apic1(config-monitor-access)# filter-group filtergroup1
```

ステップ 4 **no filter-group**

必要に応じて、フィルタ グループの関連付けを解除します。

例：

```
apic1(config-monitor-access)# no filter-group
```

ステップ 5 [**no**] **source interface ethernet** {[*fex*/]*slot*/*port* | *port-range*} **leaf** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-access)# source interface ethernet 1/9 leaf 101
```

ステップ 6 **filter-group** *filtergroup-name*

フィルタ グループを SPAN 送信元に関連付けます。

例：

```
apic1(config-monitor-access-source)# filter-group filtergroup2
```

ステップ 7 **exit**

アクセス モニター フィルタ グループ設定モードに戻ります。

例：

```
apic1(config-monitor-access-source)# exit
```

ステップ 8 **no filter-group**

必要に応じて、SPAN 送信元からフィルタ グループの関連付けを解除します。

例：

```
apic1(config-monitor-access-source)# no filter-group
```

ステップ 9 **exit**

アクセス モニター フィルタ グループ設定モードに戻ります。

例：

```
apic1(config-monitor-access)# exit
```

ステップ 10 **exit**

グローバル コンフィギュレーション モードを終了します。

例：

```
apic1(config)# exit
```

例

この例は、フィルタ グループを関連付ける方法を示しています。

```
apic1# configure
apic1(config)# monitor access session session1
apic1(config-monitor-access)# filter-group filtergroup1
apic1(config-monitor-access)# source interface ethernet 1/9 leaf 101
apic1(config-monitor-access-source)# filter-group filtergroup2
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access-source)# no filter-group
apic1(config-monitor-access)# exit
apic1(config)# exit
```

NX-OS スタイルの CLI を使用したアクセス モードでの ERSPAN の設定

ACI ファブリックでは、アクセス モードの ERSPAN 設定を使用して、1 つ以上のリーフ ノードのアクセス ポート、ポート チャネル、および vPC から発信されたトラフィックを監視できます。

ERSPAN セッションの場合、宛先は常にエンドポイント グループ (EPG) で、これらはファブリック内のどこにでも展開できます。監視対象のトラフィックは、どこであれ、EPG が移動した場所である宛先に転送されます。

ステップ 1 **configure terminal**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor access session** *session-name*

アクセス モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor access session mySession
```

ステップ 3 [**no**] **description** *text*

このモニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-access)# description "This is my access ERSPAN session"
```

ステップ 4 [**no**] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーション モードを開始します。

例：

```
apic1(config-monitor-access)# destination tenant t1 application app1 epg epg1 destination-ip
192.0.20.123 source-ip-prefix 10.0.20.1
```

ステップ 5 [**no**] **erspan-id** *flow-id*

ERSPAN セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。

例：

```
apic1(config-monitor-access-dest)# erspan-id 100
```

ステップ 6 [**no**] **ip dscp** *dscp-code*

ERSPAN トラフィックのパケットの DiffServ コード ポイント（DSCP）値を設定します。指定できる範囲は 0 ～ 64 です。

例：

```
apic1(config-monitor-access-dest)# ip dscp 42
```

ステップ 7 [**no**] **ip ttl** *ttl-value*

ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。

例：

```
apic1(config-monitor-access-dest)# ip ttl 16
```

ステップ 8 [**no**] **mtu** *mtu-value*

ERSPAN セッションの最大伝送単位（MTU）サイズを設定します。指定できる範囲は 64 ～ 9216 バイトです。

例：

```
apic1(config-monitor-access-dest)# mtu 9216
```

ステップ 9 **exit**

モニター アクセス設定モードに戻ります。

例：

```
apic1(config-monitor-access-dest)#
```

ステップ 10 [**no**] **source interface ethernet** {[*fex/*]*slot/port* | *port-range*} **leaf** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-access)# source interface eth 1/2 leaf 101
```

ステップ 11 [**no**] **source interface port-channel** *port-channel-name-list* **leaf** *node-id* [**fex** *fex-id*]

送信元インターフェイスのポートチャネルを指定します。

例：

```
apic1(config-monitor-access)# source interface port-channel pc1 leaf 101
```

ステップ 12 [**no**] **source interface vpc** *vpc-name-list* **leaf** *node-id1 node-id2* [**fex** *fex-id1 fex-id2*]

送信元インターフェイス vPC を指定します。

例：

```
apic1(config-monitor-access)# source interface vpc pc1 leaf 101 102
```

ステップ 13 **drop enable**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-access-source)# drop enable
```

ステップ 14 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# direction tx
```

ステップ 15 [**no**] **filter tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

モニタリングするトラフィックのフィルタ処理を行います。フィルタは、送信元ポート範囲ごとに独立して設定できます。

例：

```
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
```

ステップ 16 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-access-source)# exit
```

ステップ 17 **[no] shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-access)# no shut
```

例

この例は、ERSPAN アクセス モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-access)# description "This is my access ERSPAN session"
apic1(config-monitor-access)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-access-dest)# erspan-id 100
apic1(config-monitor-access-dest)# ip dscp 42
apic1(config-monitor-access-dest)# ip ttl 16
apic1(config-monitor-access-dest)# mtu 9216
apic1(config-monitor-access-dest)# exit
apic1(config-monitor-access)# source interface eth 1/1 leaf 101
apic1(config-monitor-access-source)# direction tx
apic1(config-monitor-access-source)#drop enable
apic1(config-monitor-access-source)# filter tenant t1 application app1 epg epg1
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access)# no shut
apic1(config-monitor-access)# show run
# Command: show running-config monitor access session mySession
# Time: Fri Nov 6 23:55:35 2015
  monitor access session mySession
    description "This is my ERSPAN session"
    source interface eth 1/1 leaf 101
      direction tx
      filter tenant t1 application app1 epg epg1
      exit
    destination tenant t1 application app1 epg epg1 destination-ip 192.0.20.123
source-ip-prefix 10.0.20.1
      ip dscp 42
      ip ttl 16
      erspan-id 9216
      mtu 9216
      exit
    exit
```

この例は、モニタリング送信元としてポート チャネルを設定する方法を示しています。

```
apic1(config-monitor-access)# source interface port-channel pc3 leaf 105
```

この例は、モニタリング送信元として vPC の 1 つのレッグを設定する方法を示しています。

```
apic1(config-monitor-access)# source interface port-channel vpc3 leaf 105
```

次の例は、FEX 101 からのポートの範囲をモニタリング送信元として設定する方法を示しています。

```
apic1(config-monitor-access)# source interface eth 101/1/1-2 leaf 105
```

NX-OS スタイルの CLI を使用したファブリック モードでの ERSPAN の設定

ACI ファブリックでは、ファブリック モードの ERSPAN 設定を使用して、リーフ ノードまたはスパイン ノードの 1 つ以上のファブリック ポートから発信されたトラフィックをモニタリングできます。ローカル SPAN はファブリック モードではサポートされていません。

ERSPAN セッションの場合、宛先は常にエンドポイント グループ (EPG) で、これらはファブリック内のどこにでも展開できます。監視対象のトラフィックは、どこであれ、EPGが移動した場所である宛先に転送されます。ファブリック モードでは、ファブリック ポートのみが送信元として許可されますが、リーフ スイッチとスパイン スイッチの両方が許可されます。

ステップ 1 **configure terminal**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor fabric session** *session-name*

ファブリック モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor fabric session mySession
```

ステップ 3 [**no**] **description** *text*

このモニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"
```

ステップ 4 [**no**] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーション モードを開始します。

例：

```
apic1(config-monitor-fabric)# destination tenant t1 application app1 epg epg1 destination-ip
192.0.20.123 source-ip-prefix 10.0.20.1
```

ステップ 5 [**no**] **erspan-id** *flow-id*

ERSPAN セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。

例：

```
apic1(config-monitor-fabric-dest)# erspan-id 100
```

ステップ 6 [**no**] **ip dscp** *dscp-code*

ERSPAN トラフィックのパケットの DiffServ コード ポイント（DSCP）値を設定します。指定できる範囲は 0 ～ 64 です。

例：

```
apic1(config-monitor-fabric-dest)# ip dscp 42
```

ステップ 7 [**no**] **ip ttl** *ttl-value*

ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。

例：

```
apic1(config-monitor-fabric-dest)# ip ttl 16
```

ステップ 8 [**no**] **mtu** *mtu-value*

ERSPAN セッションの最大伝送単位（MTU）サイズを設定します。指定できる範囲は 64 ～ 9216 バイトです。

例：

```
apic1(config-monitor-fabric-dest)# mtu 9216
```

ステップ 9 **exit**

モニター アクセス設定モードに戻ります。

例：

```
apic1(config-monitor-fabric-dest)#
```

ステップ 10 [**no**] **source interface ethernet** {*slot/port* | *port-range*} **switch** *node-id*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-fabric)# source interface eth 1/2 switch 101
```

ステップ 11 **drop enable**

ASICでドロップされたすべてのパケットをキャプチャし、事前設定されたSPAN宛先に送信する、SPANオン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-fabric-source)# drop enable
```

ステップ 12 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-fabric-source)# direction tx
```

ステップ 13 [**no**] **filter tenant** *tenant-name* **bd** *bd-name*

ブリッジ ドメインでトラフィックをフィルタリングします。

例：

```
apic1(config-monitor-fabric-source)# filter tenant t1 bd bd1
```

ステップ 14 [**no**] **filter tenant** *tenant-name* **vrf** *vrf-name*

VRF でトラフィックをフィルタリングします。

例：

```
apic1(config-monitor-fabric-source)# filter tenant t1 vrf vrf1
```

ステップ 15 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-fabric-source)# exit
```

ステップ 16 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-fabric)# no shut
```

例

この例は、ERSPANファブリック モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor fabric session mySession
apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"
apic1(config-monitor-fabric)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-fabric-dest)# erspan-id 100
```

```
apic1(config-monitor-fabric-dest)# ip dscp 42
apic1(config-monitor-fabric-dest)# ip ttl 16
apic1(config-monitor-fabric-dest)# mtu 9216
apic1(config-monitor-fabric-dest)# exit
apic1(config-monitor-fabric)# source interface eth 1/1 switch 101
apic1(config-monitor-fabric-source)# drop enable
apic1(config-monitor-fabric-source)# direction tx
apic1(config-monitor-fabric-source)# filter tenant t1 bd bd1
apic1(config-monitor-fabric-source)# filter tenant t1 vrf vrf1
apic1(config-monitor-fabric-source)# exit
apic1(config-monitor-fabric)# no shut
```

NX-OS スタイルの CLI を使用したテナント モードでの ERSPAN の設定

ACI ファブリックでは、テナント モードの ERSPAN 設定を使用して、テナント内のエンドポイント グループから発信されたトラフィックをモニタリングできます。

テナント モードでは、送信元 EPG から発信されたトラフィックは、同じテナント内の宛先 EPG に送信されます。送信元または宛先の EPG がファブリック内で移動しても、トラフィックのモニタリングには影響しません。

ステップ 1 **configure terminal**

グローバル設定モードを開始します。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor tenant** *tenant-name* **session** *session-name*

テナント モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor tenant session mySession
```

ステップ 3 [**no**] **description** *text*

このアクセス モニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-tenant)# description "This is my tenant ERSPAN session"
```

ステップ 4 [**no**] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーション モードを開始します。

例：

```
apic1(config-monitor-tenant)# destination tenant t1 application app1 epg epg1 destination-ip
192.0.20.123 source-ip-prefix 10.0.20.1
```

ステップ 5 [**no**] **erspan-id** *flow-id*

ERSPAN セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。

例：

```
apic1(config-monitor-tenant-dest)# erspan-id 100
```

ステップ 6 [**no**] **ip dscp** *dscp-code*

ERSPAN トラフィックのパケットの DiffServ コード ポイント（DSCP）値を設定します。指定できる範囲は 0 ～ 64 です。

例：

```
apic1(config-monitor-tenant-dest)# ip dscp 42
```

ステップ 7 [**no**] **ip ttl** *ttl-value*

ERSPAN トラフィックの IP 存続可能時間（TTL）値を設定します。範囲は 1 ～ 255 です。

例：

```
apic1(config-monitor-tenant-dest)# ip ttl 16
```

ステップ 8 [**no**] **mtu** *mtu-value*

ERSPAN セッションの最大伝送単位（MTU）サイズを設定します。指定できる範囲は 64 ～ 9216 バイトです。

例：

```
apic1(config-monitor-tenant-dest)# mtu 9216
```

ステップ 9 **exit**

モニター アクセス設定モードに戻ります。

例：

```
apic1(config-monitor-tenant-dest)#
```

ステップ 10 [**no**] **source application** *application-name* **epg** *epg-name*

送信元インターフェイス ポートまたはポート範囲を指定します。

例：

```
apic1(config-monitor-tenant)# source application app2 epg epg5
```

ステップ 11 [**no**] **direction** {**rx** | **tx** | **both**}

モニタリングするトラフィックの方向を指定します。方向は、送信元ポートごとに独立して設定できます。

例：

```
apic1(config-monitor-tenant-source)# direction tx
```

ステップ 12 **exit**

アクセス モニタリング セッション設定モードに戻ります。

例：

```
apic1(config-monitor-tenant-source)# exit
```

ステップ 13 [**no**] **shutdown**

モニタリング セッションをディセーブル（またはイネーブル）にします。

例：

```
apic1(config-monitor-tenant)# no shut
```

例

この例は、ERSPAN テナント モニタリング セッションを設定する方法を示しています。

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-tenant)# description "This is my tenant ERSPAN session"
apic1(config-monitor-tenant)# destination tenant t1 application app1 epg epg1
destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-tenant-dest)# erspan-id 100
apic1(config-monitor-tenant-dest)# ip dscp 42
apic1(config-monitor-tenant-dest)# ip ttl 16
apic1(config-monitor-tenant-dest)# mtu 9216
apic1(config-monitor-tenant-dest)# exit
apic1(config-monitor-tenant)# source application app2 epg epg5
apic1(config-monitor-tenant-source)# direction tx
apic1(config-monitor-tenant-source)# exit
apic1(config-monitor-tenant)# no shut
```

NX-OS スタイルの CLI を使用したグローバル SPAN-On-Drop セッションの設定

このセクションでは、ノード上のすべてのポートを SPAN 送信元とするグローバル ドロップを作成する方法を示します。

ステップ 1 **configure terminal**

グローバル コンフィギュレーション モードを開始します。

例：

```
apic1# configure terminal
```

ステップ 2 [**no**] **monitor fabric session** *session-name*

ファブリック モニタリング セッション設定を作成します。

例：

```
apic1(config)# monitor fabric session Spine301-GD-SOD
```

ステップ 3 [**no**] **description** *text*

このモニタリング セッションの説明を追加します。テキストにスペースが含まれている場合は、単一引用符で囲む必要があります。

例：

```
apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"
```

ステップ 4 **source global-drop switch**

ASIC でドロップされたすべてのパケットをキャプチャし、事前設定された SPAN 宛先に送信する、SPAN オン ドロップ機能をイネーブルにします。

例：

```
apic1(config-monitor-fabric)# source global-drop switch
```

ステップ 5 [**no**] **destination tenant** *tenant-name***application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*

宛先インターフェイスをテナントとして指定し、宛先コンフィギュレーション モードを開始します。

例：

```
apic1(config-monitor-fabric-dest)# destination tenant ERSPAN application A1 epg E1 destination-ip
165.10.10.155 source-ip-prefix 22.22.22.22
```

例

次に、SPAN-on-Drop セッションを設定する例を示します。

```
apic1# configure terminal
apic1(config)# monitor fabric session Spine301-GD-SOD
apic1(config-monitor-fabric)# source global-drop switch
apic1(config-monitor-fabric)# destination tenant ERSPAN application A1 epg E1
destination-ip 179.10.10.179 source-ip-prefix 31.31.31.31
```

REST API を使用した SPAN の構成

REST API を使用した ERSPAN 宛先のファブリック宛先グループの設定

このセクションでは、REST API を使用して、ERSPAN 宛先のファブリック宛先グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

ERSPAN 宛先のファブリック宛先グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag="">
    <spanDest annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag="">
    <spanRsDestEpg annotation="" dscp="unspecified" finalIp="0.0.0.0" flowId="1" ip="179.10.10.179"

     mtu="1518"srcIpPrefix="20.20.20.2" tDn="uni/tn-ERSPAN/ap-A1/epg-E1" ttl="64" ver="ver2"
     verEnforced="no"/>
```

```
    </spanDest>
</spanDestGrp>
```

REST API を使用したグローバル ドロップ送信元グループの設定

このセクションでは、REST API を使用してグローバル ドロップ送信元グループを構成することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

グローバル ドロップ送信元グループを構成します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Spine-402-GD-SOD" nameAlias="">
    <spanSrc annotation="" descr="" dir="both" name="402" nameAlias=""  spanOnDrop="yes">
    <spanRsSrcToNode annotation="" tDn="topology/pod-1/node-402"/>
   </spanSrc><spanSpanLbl annotation="" descr="" name="402-dst-179" nameAlias=""  tag="yellow-green"/>
</spanSrcGrp>
```

REST API を使用した SPAN 宛先としてのリーフ ポートの設定

このセクションでは、REST API を使用してリーフ ポートを SPAN 宛先として設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

リーフ ポートを SPAN 宛先として設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest4" nameAlias="" ownerKey="" ownerTag="">
    <spanDest annotation="" descr="" name="Dest4" nameAlias="" ownerKey="" ownerTag="">
    <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-301/pathep-[eth1/18]"/>
    </spanDest>
</spanDestGrp>
```

REST API を使用した SPAN アクセス送信元グループの設定

このセクションでは、REST API を使用して SPAN アクセス ソース グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

SPAN アクセス送信元グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr=""  name="Test-Src2" nameAlias="" ownerKey=""
```

```
 ownerTag="">
    <spanSrc annotation="" descr="" dir="both" name="Src1" nameAlias="" ownerKey="" ownerTag=""
     spanOnDrop="yes">
    <spanRsSrcToPathEp annotation="" tDn="topology/pod-1/paths-301/pathep-[eth1/1]"/>
    </spanSrc>
    <spanSpanLbl annotation="" descr="" name="Dest1" nameAlias="" ownerKey="" ownerTag=""
     tag="yellow-green"/>
</spanSrcGrp>
```

REST API を使用した SPAN ファブリック送信元グループの設定

このセクションでは、REST API を使用して SPAN ファブリック送信元グループを設定することにより、REST API の使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

SPAN ファブリック送信元グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Test-Src2" nameAlias="" ownerKey=""
 ownerTag="">
    <spanSrc annotation="" descr="" dir="both" name="Src1" nameAlias="" ownerKey=""
     ownerTag="" spanOnDrop="yes">
    <spanRsSrcToPathEp annotation="" tDn="topology/pod-1/paths-301/pathep-[eth1/51]"/>
    </spanSrc>
    <spanSpanLbl annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag=""
     tag="yellow-green"/>
</spanSrcGrp>
```

REST API を使用した ERSPAN 宛先のアクセス宛先グループの設定

このセクションでは、REST API を使用して、ERSPAN 宛先のアクセス宛先グループを設定することにより、REST APIの使用方法を示します。使用可能なプロパティのリストについては、『*APIC* 管理情報モデル資料』を参照してください。

ERSPAN 宛先のアクセス宛先グループを設定します。

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest4" nameAlias="" ownerKey=""
  ownerTag="">
    <spanDest annotation="" descr="" name="Dest4" nameAlias="" ownerKey=""
     ownerTag="">
    <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-301/pathep-
     [eth1/18]"/>
    </spanDest>
</spanDestGrp>
```

統計の使用

統計は、観測しているオブジェクトのリアルタイムの測定値を提供し、傾向分析とトラブルシューティングを可能にします。統計収集は、継続的またはオンデマンドの収集用に構成でき、累計カウンタとゲージで収集できます。

ポリシーは、収集する統計の種類、間隔、実行するアクションを定義します。たとえば、入力VLANでドロップされたパケットのしきい値が毎秒1000を超える場合、ポリシーはEPG上で1つの障害を生成することができます。

統計データは、インターフェイス、VLAN、EPG、アプリケーションプロファイル、ACLルール、テナント、内部APICプロセスなどのさまざまなソースから収集されます。統計は、5分、15分、1時間、1日、1週間、1か月、1四半期、または1年のサンプリング間隔でデータを蓄積します。短い期間の間隔によって、長い間隔が与えられます。さまざまな統計情報プロパティを利用でき、最終値、累計、周期、変化のレート、トレンド、最大、最小と平均などがあります。収集/保持時間は構成できます。ポリシーは、統計をシステムの現在の状態から収集するか、履歴的に蓄積するか、またはその両方かを指定できます。たとえば、ポリシーは、履歴統計を1時間にわたって5分間隔で収集するように指定できます。1時間は移動ウィンドウです。1時間が経過すると、次の5分間の統計が追加され、一番最初の5分間に収集されたデータが放棄されます。

（注） 5分粒度のサンプル レコードの最大数は12サンプル（1時間の統計）に制限されます。他のすべてのサンプル間隔は、1,000 サンプル レコードに制限されています。たとえば、1時間の粒度統計は41日間まで保持できます。

GUIでの統計情報の表示

アプリケーション プロファイル、物理インターフェイス、ブリッジ ドメイン、ファブリック ノードなど、APIC GUIを使用して、多数のオブジェクトの統計情報を表示できます。GUIで統計情報を表示するには、**ナビゲーション**ペインでオブジェクトを選択し、[STATS]タブをクリックします。

インターフェイスの統計情報を表示する手順は、次のとおりです。

ステップ1 メニュー バーで、**[ファブリック（Fabric）] >[インベントリ（Inventory）]** を選択します。

ステップ2 **[ナビゲーション（navigation）]** のペインで、ポッドを選択します。

ステップ3 ポッドを展開し、スイッチを展開します。

ステップ4 **[ナビゲーション（Navigation）]** ペインで、**[インターフェイス（Interfaces）]** を展開し、**eth1/1** を選択します。

ステップ5 **[作業（Work）]** ペインで、**[STATS（統計）]** タブを選択します。

APIC はインターフェイス統計情報を表示します。

例

次のタスク

[作業（Work）] ペインの次のアイコンを使用して、APIC での統計情報の表示方法を管理できます。

- 更新（Refresh）：統計情報を手動で更新します。
- テーブル ビューの表示（Show Table View）：表とチャートの表示を切り替えます。
- 統計の開始または停止（Start or Stop Stats）：統計情報の自動更新を有効または無効にします。
- 統計の選択（Select Stats）：表示するカウンタとサンプルのインターバルを指定します。
- オブジェクトを XML としてダウンロード（Download Object as XML）：XML 形式でオブジェクトをダウンロードします。
- 測定タイプ（Measurement Type、歯車のアイコン）：統計情報の測定タイプを指定します。オプションとして累積値、定期値、平均値、傾向値があります。

スイッチの統計情報コマンド

次のコマンドを使って、ACI リーフ スイッチの統計情報を表示できます。

コマンド	目的
レガシー Cisco Nexus の **show**/**clear** コマンド	詳細については、*Cisco Nexus 9000* シリーズ *NX-OS* 構成ガイドを参照してください。

コマンド	目的
show platform internal counters port [*port_num* | **detail** | **nz** | {**internal** [**nz** | *int_port_num*]}]	スパイン ポート統計情報を表示します。 • *port_num*：スロットのない前面ポート番号。 • **detail**：SNMP、クラス、および転送の統計を返します。 • **nz**：ゼロ以外の値のみを表示します。 • **internal**：内部ポートの統計情報を表示します。 • *int_port_num*：内部論理ポート番号。たとえば、BCM-0/97 の場合は、97 と入力します。 （注） リンクがリセットされると、スイッチのカウンタがゼロになります。カウンタ リセットの条件には以下のものがあります。 • 偶発的なリンクのリセット • 手動によるポートの有効化（ポートが無効化された後）
show platform internal counters vlan [*hw_vlan_id*]	VLAN 統計情報を表示します。
show platform internal counters tep [*tunnel_id*]	TEP 統計情報を表示します。
show platform internal counters flow [*rule_id* | {**dump** [*asic inst*] | [**slice direction** | **index** *hw_index*]}]	フロー統計情報を表示します。
clear platform internal counters port [*port_num* | {**internal** [*int_port_num*]}]	ポート統計情報を消去します。
clear platform internal counters vlan [*hw_vlan_id*]	VLAN カウンタを消去します。
debug platform internal stats logging level *log_level*	デバッグ ロギング レベルを設定します。
debug platform internal stats logging {**err**|**trace**|**flow**}	デバッグのロギング タイプを設定します。

GUI を使用する統計情報しきい値の管理

ステップ 1 メニュー バーで、[Fabric] > [Fabric Policies] を選択します。

ステップ 2 **[ナビゲーション（Navigation）]** ペインで + をクリックし、**[モニタリング ポリシー（Monitoring Policies）]** を展開します。

ステップ 3 **[ナビゲーション（Navigation）]** ペインで、モニタリング ポリシー名（Default など）を展開します。

ステップ 4 **[統計収集ポリシー（Stats Collection Policies）]** をクリックします。

ステップ 5 **[統計収集ポリシー（Stats Collection Policies）]** ウィンドウで、しきい値を設定する **[モニタリング オブジェクト（Monitoring Object）]** および **[統計タイプ（Stat Type）]** を選択します。

ステップ 6 **[作業（Work）]** ペインで、**[構成しきい値（CONFIG THRESHOLDS）]** の下の + をアイコンをクリックします。

ステップ 7 **[コレクションのためのしきい値（THRESHOLDS FOR COLLECTION）]** ウィンドウで + をクリックし、しきい値を追加します。

ステップ 8 **[プロパティを選択（Choose a Property）]** ウィンドウで、統計タイプを選択します。

ステップ 9 **[統計しきい値を編集（EDIT STATS THRESHOLD）]** ウィンドウで、次のしきい値を指定します。

- 標準値（Normal Value）：カウンタの有効な値。
- しきい値の方向（Threshold Direction）：しきい値が最大値または最小値かどうかを示します。
- 上昇しきい値（Rising Thresholds）（クリティカル（Critical）、メジャー（Major）、マイナー（Minor）、警告（Warning））：値がしきい値を上回った場合にトリガーされます。
- 下降しきい値（Falling Threshold）（クリティカル（Critical）、メジャー（Major）、マイナー（Minor）、警告（Warning））： 値がしきい値を下回った場合にトリガーされます。

ステップ 10 上昇および下降しきい値の設定値、リセット値を指定できます。設定値はエラーがトリガーされるタイミングを指定します。リセット値はエラーが消去されるタイミングを指定します。

ステップ 11 しきい値を保存するには、**[送信する（SUBMIT）]** をクリックします。

ステップ 12 **[コレクションのためのしきい値（THRESHOLDS FOR COLLECTION）]** ウィンドウで、**[閉じる（CLOSE）]** をクリックします。

統計情報に関するトラブルシューティングのシナリオ

次の表に、Cisco APIC に共通する統計情報に関するトラブルシューティングのシナリオを要約します。

問題	ソリューション
APIC は、構成されたモニタリングポリシーを適用しません。	モニタリング ポリシーが適用されていても、APIC が統計情報の収集やトリガしきい値に対する操作など、対応するアクションを実行しないと問題が発生します。問題を解決するには、次の手順に従ってください。 • monPolDn が正しいモニタリング ポリシーを指していることを確認します。 • セレクタが正しく設定され、エラーがないことを確認します。 • テナントのオブジェクトの場合は、モニタリングポリシーとの関係を確認します。
構成した一部の統計情報が見つからない。	問題を解決するには、次の手順に従ってください。 • モニタリングポリシーおよび収集ポリシー内でデフォルトによって無効になっている統計情報を確認します。 • 収集ポリシーを確認し、統計情報がデフォルトで無効になっているか、または特定のインターバルで無効になっているかを識別します。 • 統計ポリシーを確認し、統計情報がデフォルトで無効になっているか、または特定のインターバルで無効になっているかを識別します。 （注） ファブリック ヘルスの統計情報を除き、5 分間の統計情報がスイッチに保存され、スイッチがリブートされると失われます。
統計情報や履歴を設定した期間保持できない。	問題を解決するには、次の手順に従ってください。 • 収集設定を確認してください。モニタリングポリシーの最上位レベルで設定されていると、特定のオブジェクトまたは統計タイプでは、統計情報が無効になる場合があります。 • モニタリングオブジェクトに割り当てられた収集ポリシーを確認します。ポリシーが存在するのを確認し、管理状態および履歴保持の値を確認します。 • 統計タイプが正しく構成されていることを確認します。

問題	ソリューション
構成されたインターバルにわたって保持されない統計情報がある。	構成が履歴記録サイズの最大値を超えていないかどうか確認します。制限は次のとおりです。 • 5 分間の細かさでのスイッチ統計情報は 12 サンプル（5 分間の細かさの統計情報の 1 時分）に限られています。 • 1000 サンプルの厳しい制限があります。たとえば、粒度 1 時間の統計情報は 41 日間まで保持できます。
エクスポートポリシーは構成されるが、APIC が統計情報をエクスポートしない。	問題を解決するには、次の手順に従ってください。 • 送信先ポリシーの状態オブジェクトを確認します。 • 統計をエクスポートするノードでエクスポートステータスのオブジェクトをチェックし、エクスポートステータスと詳細のプロパティを確認してください。集約された EPG 統計は APIC ノードから 15 分ごとにエクスポートされます。その他の統計は、送信元ノードから 5 分ごとにエクスポートされます。たとえば、EPG が 2 つのリーフ スイッチに展開され、EPG アグリゲーション パーツをエクスポートするように設定されている場合、それらのパーツは 5 分ごとにノードからエクスポートされます。 • 構成がエクスポートポリシーの最大数を超えていないかどうかを確認します。統計のエクスポートポリシーの最大数は、テナントの数とほぼ同じです。 （注） 各テナントは複数の統計エクスポートポリシーを持つことができ、複数のテナントが同じエクスポートポリシーを共有できますが、ポリシーの合計数はテナントの数とほぼ同数に制限されます。
5 分間統計が変動する	APIC システムは、約 10 秒ごとにサンプリングされた統計を 5 分ごとにレポートします。データが収集されるときにわずかな時間差があるため、5 分間で取得されるサンプルの数は異なる場合があります。その結果、統計情報が少し長い、または短い期間を表す場合があります。これは想定されている動作です。
一部の履歴統計情報が見つからない。	詳しくは、統計情報の消去を参照してください。

統計情報の消去

APIC とスイッチは次のように統計情報を消去します。

- スイッチ：スイッチは次のように統計情報を消去します。

- スイッチの 5 分間の統計情報は、5 分間カウンタ値が報告されないと消去されます。この状況はポリシーによってオブジェクトが削除される、または統計情報が無効化されるときに起こる場合があります。
- 統計が 1 時間以上欠落している場合、粒度の大きい統計はパージされます。これは、次の場合に発生する可能性があります。
  - 統計情報がポリシーによって無効化されている。
  - スイッチが 1 時間以上 APIC から切断されている。
- スイッチは削除されたオブジェクトの統計情報を 5 分後に消去します。オブジェクトがこの時間内に再作成されると、統計カウントは未変更のままになります。
- 無効化されたオブジェクト統計情報は 5 分後に削除されます。
- 統計情報レポートが 5 分間無効化されるなど、システム状態が変化すると、このスイッチによって統計情報が消去されます。

- APIC：APIC はインターフェイス、EPG、温度センサーと正常性統計情報を含むオブジェクトを 1 時間後に消去します。

Syslog のソースと宛先の指定

このセクションでは、syslog 宛先グループ、syslog ソースを作成する方法、および REST API を使用して syslog を NX-OS CLI 形式で表示できるようにする方法について説明します。

Syslog について

稼働中、シスコ アプリケーション セントリック インフラストラクチャ（ACI）システムでの障害またはイベントは、コンソール、ローカル ファイル、および別のシステム上のロギング サーバへのシステム ログ（syslog）の送信をトリガーできます。システム ログ メッセージには、通常、障害またはイベントに関する情報のサブセットが含まれます。システム ログ メッセージには、監査ログとセッション ログのエントリを含めることもできます。

（注） APIC およびファブリック ノードが生成できる syslog メッセージのリストについては、http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci_syslog/ACI_SysMsg.html を参照してください。

多くのシステム ログ メッセージは、ユーザが実行している処理、あるいはユーザが設定または管理しているオブジェクトに固有のものです。これらのメッセージには次のようなものがあります。

- 情報メッセージ。実行している処理のヘルプおよびヒントを提供します。

- 警告メッセージ。ユーザが設定または管理しているオブジェクト（ユーザ アカウントやサービス プロファイルなど）に関連するシステム エラーの情報を提供します。

システム ログ メッセージを受信してモニタするためには、syslog 宛先（コンソール、ローカル ファイル、または syslog サーバを実行している 1 つ以上のリモート ホスト）を指定する必要があります。また、コンソールに表示されるか、ファイルまたはホストによってキャプチャされるメッセージのシビラティ（重大度）の最小値を指定できます。syslog メッセージを受信するローカル ファイルは `/var/log/external/messages` です。

Syslog 送信元は、オブジェクト モニタリング ポリシーを適用できる任意のオブジェクトにすることができます。送信されるメッセージのシビラティ（重大度）の最小値、syslog メッセージに含める項目、および syslog の宛先を指定できます。

Syslog の表示形式を NX-OS スタイル形式に変更できます。

これらのシステム メッセージを生成する障害またはイベントの詳細は、『*Cisco APIC Faults, Events, and System Messages Management Guide*』で説明しています。システム ログ メッセージのリストについては『*Cisco ACI System Messages Reference Guide*』を参照してください。

（注） システム ログ メッセージは、必ずしもシステムに問題があることを示しているとは限りません。単に情報を通知するだけのメッセージもありますし、通信回線、内部ハードウェア、またはシステム ソフトウェアに関する問題点の診断に役立つメッセージもあります。

Syslog の宛先および宛先グループの作成

この手順では、ロギングおよび評価用の syslog データの宛先を設定します。syslog データは、コンソール、ローカル ファイル、または宛先グループ内の 1 つまたは複数の syslog サーバにエクスポートできます。

ステップ 1 メニュー バーで、[Admin] をクリックします。

ステップ 2 サブメニュー バーで、[External Data Collectors] をクリックします。

ステップ 3 [Navigation] ペインで、[Monitoring Destinations] を展開します。

ステップ 4 [Syslog] を右クリックし、[Create Syslog Monitoring Destination Group] を選択します。

ステップ 5 [Create Syslog Monitoring Destination Group] ダイアログボックスで、次の操作を実行します。

a) グループおよびプロファイルの [Name] フィールドに、モニタリングの宛先グループおよびプロファイルの名前を入力します。

b) グループおよびプロファイルの [Format] フィールドで、Syslog メッセージの形式を選択します。

デフォルトは [aci]、または RFC 5424 準拠のメッセージ形式ですが、NX-OS スタイル形式に設定することもできます。

c) グループおよびプロファイルの [Admin State] ドロップダウン リストで、[enabled] を選択します。

d) ローカル ファイルへの syslog メッセージの送信を有効にするには、[Local File Destination] の [Admin State] ドロップダウン リストから [enabled] を選択し、[Local File Destination] の [Severity] ドロップダウン リストからシビラティ（重大度）の最小値を選択します。

syslog メッセージを受信するローカル ファイルは `/var/log/external/messages` です。

e) コンソールへの syslog メッセージの送信を有効にするには、[Console Destination] の [Admin State] ドロップダウン リストから [enabled] を選択し、[Console Destination] の [Severity] ドロップダウン リストからシビラティ（重大度）の最小値を選択します。

f) [Next] をクリックします。

g) [Create Remote Destinations] 領域で、[+] をクリックしてリモート宛先を追加します。

注意 指定したDNSサーバがインバンド接続を介して到達可能に設定されている場合、リモートsyslog宛先のホスト名解決に失敗するリスクがあります。この問題を回避するには、IP アドレスを使用して syslog サーバを設定します。ホスト名を使用する場合は、アウトオブバンド インターフェイス経由で DNS サーバに到達できることを確認します。

ステップ 6 [Create Syslog Remote Destination] ダイアログボックスで、次の操作を実行します。

a) [Host] フィールドに、送信先ホストの IP アドレスまたは完全修飾ドメイン名を入力します。

b) （任意） [Name] フィールドに、宛先ホストの名前を入力します。

c) [Admin State] フィールドで、[enabled] オプション ボタンをクリックします。

d) （任意） 最小シビラティ（重大度）、**[シビラティ（重大度）（Severity）]**、**[ポート（Port）]** 番号、および syslog **[ファシリティ（Facility）]**を選択します。

[ファシリティ（Facility）] は、メッセージを生成したプロセスを示すためにオプションで使用できる番号で、受信側でのメッセージの処理方法を決定するために使用できます。

e) 5.2（3）以降のリリースでは、**[トランスポート（Transport）]** フィールドで、メッセージに使用するトランスポート プロトコルを選択します。

- リリース 5.2(4) より前のリリースでは、メッセージに使用するトランスポート プロトコルとして **tcp** または **udp** を選択します。
- 5.2(4)リリース以降では、メッセージに使用するトランスポートプロトコルのオプションとして、**ssl**も選択できるようになりました。この機能を使用すると、（クライアントとして機能している）ACI スイッチが、ロギングにセキュアな接続をサポートする（サーバーとして機能している）リモート Syslog サーバーに対してセキュアな暗号化されたアウトバウンド接続を確立できるようになります。認証と暗号化により、この機能では、セキュリティ保護されていないネットワーク上でもセキュアな通信を実現できます。

  メッセージに使用するトランスポート プロトコルとして **ssl** を選択した場合は、必要な SSL 証明書もアップロードする必要があることに注意してください。**[認証局の作成（Create Certificate Authority）]** ウィンドウに移動して、必要な SSL 証明書をアップロードできます。

  [管理（Admin）] > **[AAA]** > **[セキュリティ（Security）]** > **[公開キー管理（Public Key Management）]** > **[認証局（Certificate Authorities）]**を選択し、その後**[アクション（Actions）]** > **[認証局の作成（Create Certificate Authority）]**を選択します。

トランスポート プロトコルのデフォルト オプションは **udp** です。

f) [Management EPG] ドロップダウン リストから管理エンドポイント グループを選択します。
g) [OK] をクリックします。

ステップ 7 （任意）リモート宛先グループにリモート宛先を追加するには、もう一度 [+] をクリックし、[Create Syslog Remote Destination] ダイアログボックスの手順を繰り返します。

ステップ 8 **[完了（Finish）]** をクリックします。

Syslog 送信元の作成

Syslog 送信元は、オブジェクト モニタリング ポリシーを適用できる任意のオブジェクトにすることができます。

始める前に

syslog モニタリング宛先グループを作成します。

ステップ 1 メニュー バーおよびナビゲーション フレームから、関心領域の [Monitoring Policies] メニューに移動します。

テナント、ファブリック、およびアクセスのモニタリング ポリシーを設定できます。

ステップ 2 [Monitoring Policies] を展開し、モニタリング ポリシーを選択して展開します。

[Fabric] > [Fabric Policies] > [Monitoring Policies] > [Common Policy] の下に、基本モニタリング ポリシーがあります。このポリシーは、すべての障害とイベントに適用され、ファブリック内のすべてのノードとコントローラに自動的に導入されます。または、スコープが限定された既存のポリシーを指定することもできます。

ステップ 3 モニタリング ポリシーの下で、[Callhome/SNMP/Syslog] をクリックします。

ステップ 4 [Work] ペインで、[Source Type] ドロップダウン リストから [Syslog] を選択します。

ステップ 5 [Monitoring Object] リストから、モニタ対象の管理対象オブジェクトを選択します。

目的のオブジェクトがリストに表示されない場合は、次の手順に従います。

a) [Monitoring Object] ドロップダウン リストの右側にある [Edit] アイコンをクリックします。
b) [Select Monitoring Package] ドロップダウン リストから、オブジェクト クラス パッケージを選択します。
c) モニタ対象の各オブジェクトのチェックボックスをオンにします。
d) [Submit] をクリックします。

ステップ 6 テナント モニタリング ポリシーでは、[All] ではなく特定のオブジェクトを選択すると、[Scope] 選択が表示されます。

[Scope] フィールドで、オプション ボタンを選択して、このオブジェクトに関して送信するシステム ログ メッセージを指定します。

- [all]：このオブジェクトに関連するすべてのイベントと障害を送信します。

- [specific event]：このオブジェクトに関連する指定されたイベントのみを送信します。[Event] ドロップダウン リストからイベント ポリシーを選択します。
- [specific fault]：このオブジェクトに関連する指定された障害のみを送信します。[Fault] ドロップダウン リストから障害ポリシーを選択します。

ステップ 7 [+] をクリックして syslog 送信元を作成します。

ステップ 8 [Create Syslog Source] ダイアログボックスで、次の操作を実行します。

a) [Name] フィールドに、syslog 送信元の名前を入力します。
b) [Min Severity] ドロップダウン リストから、送信するシステム ログ メッセージのシビラティ（重大度）の最小値を選択します。
c) [Include] フィールドで、送信するメッセージ タイプのチェックボックスをオンにします。
d) [Dest Group] ドロップダウン リストから、システム ログ メッセージの送信先の syslog 宛先グループを選択します。
e) [Submit] をクリックします。

ステップ 9 （任意） syslog 送信元を追加するには、もう一度 [+] をクリックし、[Create Syslog Source] ダイアログボックスの手順を繰り返します。

REST API を使用した NX-OS CLI 形式での Syslog 表示の有効化

デフォルトで Syslog 形式は RFC 5424 に準拠しています。次の例のように、Syslog のデフォルト表示を NX-OS タイプ形式に変更できます。

```
apic1# moquery -c "syslogRemoteDest"
Total Objects shown: 1

# syslog.RemoteDest
host                : 172.23.49.77
adminState          : enabled
childAction         :
descr               :
dn                  : uni/fabric/slgroup-syslog-mpod/rdst-172.23.49.77
epgDn               :
format              : nxos
forwardingFacility  : local7
ip                  :
lcOwn               : local
modTs               : 2016-05-17T16:51:57.231-07:00
monPolDn            : uni/fabric/monfab-default
name                : syslog-dest
operState           : unknown
port                : 514
rn                  : rdst-172.23.49.77
severity            : information
status              :
uid                 : 15374
```

```
vrfId               : 0
vrfName             :
```

NX-OS タイプ形式で Syslog を表示できるようにするには、REST API を使用して次の手順を実行します。

ステップ 1 次の例に示すように、NX-OS タイプ形式での Syslog の表示を有効にします。

```
POST https://192.168.20.123/api/node/mo/uni/fabric.xml
<syslogGroup name="DestGrp77" format="nxos">
<syslogRemoteDest name="slRmtDest77" host="172.31.138.20" severity="debugging"/>
</syslogGroup>
```

syslogGroup は Syslog モニタリングの宛先グループ、**sysLogRemoteDest** は事前に設定した Syslog サーバの名前、**host** は事前に設定した Syslog サーバの IP アドレスです。

ステップ 2 次の例に示すように、Syslog 形式をデフォルトの RFC 5424 形式に戻します。

```
POST https://192.168.20.123/api/node/mo/uni/fabric.xml
<syslogGroup name="DestGrp77" format="aci">
<syslogRemoteDest name="slRmtDest77" host="172.31.138.20" severity="debugging"/>
</syslogGroup>
```

Traceroute を使用したパスの検出と接続性のテスト

このセクションでは、traceroute の注意事項と制限事項をリストし、エンドポイント間で traceroute を実行する方法について説明します。

トレースルートの概要

トレースルート ツールは、パケットが送信先に移動するときに実際に通るルートを検出するために使用されます。traceroute では、ホップごとに使用されるパスが識別され、双方向で各ホップにタイムスタンプが付けられます。traceroute を使用すると、発信元のデバイスと送信先に最も近いデバイスの間のパスに沿ってポート接続をテストできます。送信先に到達できない場合は、パス検出によってパスが障害ポイントまで追跡されます。

テナントのエンドポイントから開始されたトレースルートは、入力リーフのスイッチに表示される中間ホップとしてデフォルト ゲートウェイを示します。

トレースルートでは、次のようなさまざまなモードがサポートされています。

- エンドポイント間、リーフ間（トンネル エンドポイント、または TEP 間）
- エンドポイントから外部 IP
- 外部 IP からエンドポイント
- 外部 IP 間

トレースルートはファブリック全体のすべてのパスを検出し、外部エンドポイントの出口を検出します。パスが妨げられているかどうかを発見するのに役立ちます。

Windows および Linux トレースルートについて

traceroute コマンドを使用すると、パケットが通過した一連のホップを返すことで、特定の送信元からパケットが接続先に到達するまでのパスを判断できます。このユーティリティは、ホスト オペレーティング システム（Linux や Microsoft（MS）Windows）に付属しています。

送信元デバイス (ホスト、またはホストとして機能するルータなど) で **traceroute** *ip-address* コマンドを実行すると、指定された最大値まで増加する存続可能時間 (TTL) 値を持つ IP パケットが宛先に送信されます。ホップカウント。これはデフォルトで 30 です。通常、宛先へのパスにある各ルータは、これらのパケットを転送している間、TTL フィールドを 1 単位だけ減らします。パスの途中にあるルータが TTL = 1 のパケットを見つけると、インターネット制御メッセージ プロトコル (ICMP) の「時間超過」メッセージでソースに応答します。このメッセージは、パケットがその特定のルータをホップとして通過することを送信元に知らせます。

（注） 以下の Linux および Windows セクションで説明するように、さまざまなオペレーティング システムで **traceroute** コマンドを実装する方法にはいくつかの違いがあります。

Linux

最初のユーザー データグラム プロトコル（UDP）データグラム プローブの TTL は、1（または拡張 **traceroute** コマンドでユーザーが指定した最小 TTL）に設定されます。初期データグラム プローブの宛先 UDP ポートは 33434（または拡張 **traceroute** コマンド出力で指定されたとおり）に設定されています。拡張 **traceroute** コマンドは、通常の **traceroute** コマンドのバリエーションであり、TTL や宛先ポート番号などの **traceroute** 操作で使用されるパラメーターのデフォルト値を変更できます。初期データグラム プローブのソース UDP ポートはランダム化されており、論理演算子 OR と 0x8000 が含まれています（最小ソース ポートが 0x8000 であることを保証します）。これらの手順は、UDPデータグラムが起動されたときに何が起こるかを示しています。

（注） パラメータは構成可能です。この例は、n = 1 で始まり、n = 3 で終わります。

1. UDP データグラムは、TTL = 1、宛先 UDP ポート = 33434、および送信元ポートがランダム化された状態で配信されます。
2. UDP 宛先ポートが増分され、送信元 UDP ポートがランダム化され、2 番目のデータグラムが配信されます。
3. ステップ 2 は、最大 3 つのプローブに対して（または拡張 **traceroute** コマンド出力で要求される回数）繰り返されます。送信されたプローブごとに、宛先ホストへの段階的なパスを構築するために使用される「TTL 超過」メッセージを受信します。

4. ICMP の「時間超過」メッセージを受信すると、TTL が増分され、このサイクルが増分の宛先ポート番号で繰り返されます。次のいずれかのメッセージを受け取ることもできます。
   - ホストに到達したことを示す ICMP タイプ 3、コード 3 (「接続先到達不能」、「ポート到達不能」) メッセージ。
   - 「ホスト到達不能」、「ネット到達不能」、「最大TTL超過」、または「タイムアウト」タイプのメッセージ。これは、プローブが再送信されたことを意味します。

Cisco ルータは、ランダムな送信元ポートと増分の宛て先ポートを使用して UDP プローブ パケットを送信します (異なるプローブを区別するため)。Cisco ルータは、UDP/ICMP パケットを受信した送信元に ICMP メッセージ「時間超過」を送信します。

Linux **traceroute** コマンドは、Cisco ルータの実装に似ています。ただし、固定送信元ポートを使用します。**traceroute** コマンドの *-n* オプションは、ネーム サーバーへの要求を回避するために使用されます。

（注） UCS サーバーの CIMC コントローラは、UDP ベースの traceroute メッセージに応答しません。ICMP ベースの traceroute にのみ応答します。デフォルトでは、Windows traceroute は ICMP ベースのメッセージを送信します。Linux（および Mac）traceroute は、デフォルトで UDP ベースのメッセージを送信します。-I（大文字の i）オプションを使用すると、Linux（および Mac）の traceroute は ICMP ベースのメッセージを送信します。

Linux の traceroute がデフォルトであるため、ACI ネットワークのトラブルシューティングで traceroute を Cisco APIC に送信する必要がある場合は、Windows の traceroute を使用するか、ICMP ベースの traceroute を指定する必要があります。

Windows

MS Windows の **tracert** コマンドは、UDP データグラムの代わりに ICMP エコー要求データグラムをプローブとして使用します。ICMP エコー要求は、TTL を増分して起動され、上記と同じ動作が発生します。ICMP エコー要求データグラムを使用する意義は、最終ホップが宛先ホストからの ICMP 「到達不能」メッセージの応答に依存しないことです。代わりに、ICMP エコー応答メッセージに依存します。

コマンド構文は次のとおりです。

```
tracert [-d] [-h maximum_hops] [-j computer-list] [-w timeout] target_name
```

次の表に、コマンド パラメータについての説明を記載します。

表 2:

パラメータ	説明
-d	アドレスをコンピュータ名に解決しないように指定します。

パラメータ	説明
-h maximum_hops	ターゲットを検索する最大ホップ カウントを指定します。
-j computer-list	computer-list に沿ったルーズなソース ルートを指定します。
-w timeout	各応答のタイムアウトで指定されたミリ秒数を待機します。
target_name	ターゲット コンピュータの名前。

トレースルートの注意事項および制約事項

- トレースルートの送信元または宛先がエンドポイントである場合、そのエンドポイントはスタティックではなくダイナミックである必要があります。ダイナミック エンドポイント（fv:CEp）とは異なり、スタティック エンドポイント（fv:StCEp）にはトレースルートに必要な子オブジェクト（fv:RsCEpToPathEp）がありません。
- トレースルートは IPv6 の送信元と宛先で動作しますが、IPv4 アドレスと IPv6 アドレスを混在させて送信元 IP アドレスと宛先 IP アドレスを設定することはできません。
- トレースルート関連の制限については、『*Verified Scalability Guide for Cisco ACI*』 ドキュメントを参照してください。
- エンドポイントを新しい MAC アドレス（トレースルート ポリシーを設定する際に指定した MAC アドレスと異なる）の ToR スイッチに移動すると、トレースルート ポリシーでそのエンドポイントに「missing-target」と表示されます。この場合は、新しい MAC アドレスを指定して新しいトレースルート ポリシーを設定する必要があります。
- ポリシーベースのリダイレクト機能を含むフローに対してトレースルートを実行する場合、パケットがサービスデバイスからリーフスイッチに送信されるときに、リーフスイッチが存続時間（TTL）期限切れメッセージを送信元に伝えるために使用する IP アドレスは、必ずしもサービス デバイスのブリッジ ドメインのスイッチ仮想インターフェイス（SVI）の IP アドレスにはなりません。この動作は表面的なものであり、トラフィックが予期された経路をたどっていないことを示すものではありません。

エンドポイント 間での **traceroute** の実行

ステップ 1 メニュー バーで、[Tenants] をクリックします。
ステップ 2 サブメニュー バーで、送信元エンドポイントを含むテナントをクリックします。
ステップ 3 **[ナビゲーション]** ペインでテナントを展開し、**[ポリシー]** > **[トラブルシューティング]** を展開します。
ステップ 4 [Troubleshoot] で次のトレースルート ポリシーのいずれかを右クリックします。

- [Endpoint-to-Endpoint Traceroute Policies] を右クリックして [Create Endpoint-to-Endpoint Traceroute Policy] を選択する
- [Endpoint-to-External-IP Traceroute Policies] を右クリックして [Create Endpoint-to-External-IP Traceroute Policy] を選択する
- [External-IP-to-Endpoint Traceroute Policies] を右クリックして [Create External-IP-to-Endpoint Traceroute Policy] を選択する
- [External-IP-to-External-IP Traceroute Policies] を右クリックして [Create External-IP-to-External-IP Traceroute Policy] を選択する

ステップ 5 ダイアログボックスのフィールドに適切な値を入力し、[Submit] をクリックします。

（注） フィールドの説明については、ダイアログボックスの右上隅にあるヘルプ アイコン（[?]）をクリックしてください。

ステップ 6 **[Navigation]** ペインまたは **[Traceroute Policies]** テーブルで、traceroute ポリシーをクリックします。
トレースルート ポリシーが [Work] ペインに表示されます。

ステップ 7 [Work] ペインで [Operational] タブをクリックし、[Source Endpoints] タブ、[Results] タブの順にクリックします。

ステップ 8 **[Traceroute Results]** テーブルで、追跡に使用された単数または複数のパスを確認します。

（注）
- 複数のパスが、送信元ノードから宛先ノードへの移動に使用されている場合があります。
- **[Name]** 列など、1 つまたは複数の列の幅を広げると確認しやすくなります。

トラブルシューティング ウィザードの使用

トラブルシューティング ウィザードを使用すると、ネットワークの動作を理解して可視化できるため、問題が発生した場合にネットワークに関する懸念を緩和できます。たとえば、2 つのエンドポイントで断続的なパケット損失が発生しているが、その理由がわからない場合があります。トラブルシューティング ウィザードを使用すると、問題を評価できるため、この問題のある動作の原因であると思われる各マシンにログオンするのではなく、問題を効果的に解決できます。

このウィザードを使用すると、管理ユーザは、選択した送信元と接続先の特定の時間枠に発生する問題のトラブルシューティングを行うことができます。デバッグを実行する時間枠を定義でき、TAC に送信できるトラブルシューティング レポートを生成できます。

トラブルシューティング ウィザードの開始

トラブルシューティング ウィザードの使用を開始する前に、管理ユーザとしてログオンする必要があります。次に、送信元と接続先を指定し、トラブルシューティングセッションの時間枠を選択する必要があります。時間枠は、イベント、障害レコード、展開レコード、監査ログ、および統計を取得するために使用されます。

トラブルシューティングウィザードの画面をナビゲートするときに、いつでもスクリーンショットを撮ってプリンタに送信するか、画面の右上にある**[プリント（Print）]**アイコン（  ）をクリックして PDF として保存することができます。画面の表示を変更するために使用できるズームインおよびズームアウトアイコン（  ）もあります。

（注）

- **[レポートの生成（Generate Report）]** または **[送信（Submit）]** をクリックした後は、送信元と接続先を変更できません。入力した送信元と接続先の情報を変更する場合は、現在のセッションを削除して、新しいセッションを開始する必要があります。
- **[送信（Submit）]**をクリックした後は、ウィザードの最初のページで説明と時間枠を変更することはできません。
- トラブルシューティング ウィザードで静的 IP アドレス エンドポイントを使用することはできません。
- 指定するエンドポイントはすべて、EPG の下にある必要があります。

トラブルシューティング セッション情報を設定するには、次の手順を実行します。

ステップ 1 **[オペレーション（Operations）]**> > **[可視性とトラブルシューティング（Visibility & Troubleshooting）]** を選択します。

[可視性とトラブルシューティング（Visibility & Troubleshooting）] 画面が表示されます。

ステップ 2 **[セッション名（Session Name）]** フィールドで、ドロップダウンリストを使用して既存のトラブルシューティング セッションを選択するか、名前を入力して新しいセッションを作成します。

ステップ 3 **[セッション タイプ（Session Type）]** ドロップダウンリストから目的のセッション タイプを選択します。

- **[エンドポイントからエンドポイント（Endpoint to Endpoint）]**：送信元と接続先は両方とも内部エンドポイントです。

  同じテナントから送信元エンドポイントと接続先エンドポイントを選択する必要があります。そうしないと、このドキュメントで後述するように、トラブルシューティング機能の一部が影響を受ける可能性があります。このセッション タイプでは、両方のエンドポイントが同じリーフスイッチのセットに接続している場合、アトミック カウンタを使用できません。

- **[エンドポイントから外部 IP（Endpoint to External IP）]**：送信元は内部エンドポイントであり、接続先は外部 IP アドレスです。
- **[外部 IP からエンドポイント（External IP to Endpoint）]**：送信元は外部 IP アドレスであり、接続先は内部エンドポイントです。
- **[外部 IP から外部 IP（External IP to External IP）]**：送信元と接続先は両方とも外部 IP アドレスです。3.2(6) リリース以降、このタイプを選択できます。このセッション タイプでは、トレースルート、アトミック カウンタ、または遅延を使用できません。

ステップ 4 （任意）**[説明（Description）]** フィールドに説明を入力し、追加情報を入力します。

ステップ 5 **[送信元（Source）]** エリアに送信元情報を入力します。

- **[エンドポイントからエンドポイント（Endpoint to Endpoint））]** または**[エンドポイントから外部 IP（Endpoint to External IP）]** のセッション タイプを選択した場合は、MAC、IPv4、または IPv6 アドレス、または VM 名を入力して、**[検索（Search）]** をクリックします。

  セッション タイプが **[エンドポイントからエンドポイント（Endpoint to Endpoint）]** であり、両方のエンドポイントの MAC アドレスにそれらから学習された IP アドレスがない場合にのみ、MAC アドレスを入力できます。

  選択に役立つ詳細情報を含む 1 つ以上の行を表示するボックスが表示されます。各行は、入力した IP アドレス（**[IP]** 列）が特定のエンドポイント グループ（**[EPG]** 列）にあり、特定のテナント（**[テナント（Tenant）]** 列内）にある、特定のアプリケーション（**[アプリケーション（Application）]** 列）に属していることを示しています。リーフスイッチ番号、FEX 番号、およびポートの詳細は、**[学習した場所（Learned At）]** 列に表示されます。

- **[外部 IP からエンドポイント（External IP to Endpoint）]** のセッション タイプを選択した場合は、外部 IP アドレスを入力します。
- **[外部 IP から外部 IP（External IP to External IP）]** へのセッション タイプを選択した場合は、外部レイヤ 3 外部ネットワークの外部 IP アドレスと識別名を入力します。

ステップ 6 **接続先（Destination）]** エリアに接続先情報を入力します。

- **[エンドポイントからエンドポイント（Endpoint to Endpoint））]** または**[外部 IP からエンドポイント（External IP to Endpoint）]** のセッション タイプを選択した場合は、MAC、IPv4、または IPv6 アドレス、または VM 名を入力して、**[検索（Search）]** をクリックします。

  セッション タイプが **[エンドポイントからエンドポイント（Endpoint to Endpoint）]** であり、両方のエンドポイントの MAC アドレスにそれらから学習された IP アドレスがない場合にのみ、MAC アドレスを入力できます。

  選択に役立つ詳細情報を含む 1 つ以上の行を表示するボックスが表示されます。各行は、入力した IP アドレス（**[IP]** 列）が特定のエンドポイント グループ（**[EPG]** 列）にあり、特定のテナント（**[テナント（Tenant）]** 列内）にある、特定のアプリケーション（**[アプリケーション（Application）]** 列）に属していることを示しています。リーフスイッチ番号、FEX 番号、およびポートの詳細は、**[学習した場所（Learned At）]** 列に表示されます。

- **[エンドポイントから外部 IP（Endpoint to External IP）]** のセッション タイプを選択した場合は、外部 IP アドレスを入力します。
- **[外部 IP から外部 IP（External IP to External IP）]** へのセッション タイプを選択した場合は、外部レイヤ 3 外部ネットワークの外部 IP アドレスと識別名を入力します。

ステップ 7 **[タイム ウィンドウ（Time Window）]** エリアで、タイム ウィンドウを指定します。

[タイム ウィンドウ（Time Window）] は、過去の特定の時間枠に発生した問題をデバッグするために使用され、イベント、すべてのレコード、展開レコード、監査ログ、および統計を取得するために使用されます。2 つのウィンドウ セットがあります。1 つはすべてのレコード用で、もう 1 つは個々のリーフスイッチ（またはノード）用です。

デフォルトでは、**[最新（Latest Minutes）]** フィールドで指定した任意の分数に基づいて、ローリング タイム ウィンドウを指定できます。デフォルトは 240 分です。セッションには、セッションを作成した時刻より前に指定した過去（分）のデータが含まれます。

[固定時間を使用（Use fixed time）] ボックスにチェックを入れると、**[開始（From）]** および **[終了（To）]** フィールドでセッションの固定時間ウィンドウを指定できます。セッションには、**[開始（From）]** から **[終了（To）]** 時刻までのデータが含まれます。

ステップ 8 **[送信（Submit）]** をクリックして、トラブルシューティング セッションを開始します。

しばらくすると、トラブルシューティング セッションのトポロジ図が表示されます。

トラブルシューティング レポートの生成

トラブルシューティング レポートは、JSON、XML、PDF、HTML などのいくつかの形式で生成できます。形式を選択したら、レポートをダウンロードして（またはレポートのダウンロードをスケジュールして）、オフライン分析に使用するか、サポート ケースを作成できるように TAC に送信することができます。

トラブルシューティングに関するレポートを生成するには、次のようにします：

ステップ 1 画面の右下隅にある **[レポートの生成（GENERATE REPORT）]** をクリックします。
[レポート ジェネレータ（Report Generator）] ダイアログボックスが表示されます。

ステップ 2 [レポート形式（Report Format）] ドロップダウン メニューから出力フォーマット（**XML**、**HTML**、**JSON**、または **PDF**）を選択します。

ステップ 3 レポートのダウンロードをすぐに実行するようにスケジュールする場合は、[今すぐ送信（**Now**>**SUBMIT**）] をクリックします。
レポートが生成されると、レポートの入手先を示す**情報**ボックスが表示されます。

ステップ 4 レポートの生成を後でスケジュールするには、**[スケジューラを使用（Use a scheduler）]** > **[スケジューラ（Scheduler）]** ドロップダウン メニュークリックして、存在するスケジュールを選択するか、**[スケジューラを作成（Create Scheduler）]** をクリックして新しいスケジューラを作成します。。
[トリガ スケジュールの作成（CREATE TRIGGER SCHEDULE）] ダイアログが表示されます。

ステップ 5 **[名前（Name）]**、**[説明（Description）]** (オプション)、および **[スケジュール ウィンドウ（Schedule Windows）]** フィールドに情報を入力します。

（注） **[スケジューラ（SCHEDULER）]** の使用方法の詳細については、オンライン ヘルプを参照してください。

ステップ 6 [SUBMIT] をクリックします。

レポートの生成には、ファブリックのサイズと障害またはイベントの数に応じて、数分から最大 10 分かかります。レポートの生成中はステータス メッセージが表示されます。トラブルシューティング レポートを取得して表示するには、**[生成されたレポートを表示（SHOW GENERATED REPORTS）]** をクリックします。

[必要な認証（Authentication Required）] ウィンドウで、サーバーの資格情報 **([ユーザー名（User Name）]**と**[パスワード（Password）])** を入力します。次に、トラブルシューティング レポートがシステムにローカルにダウンロードされます。

[すべてのレポート（ALL REPORTS）] ウィンドウが表示され、今、トリガしたものを含む、生成されたすべてのレポートのリストが表示されます。そこから、選択した出力ファイル形式に応じて、リンクをクリックしてレポートをダウンロードするか、すぐに表示することができます (たとえば、ファイルが PDF の場合、ブラウザですぐに開くことができます)。

トラブルシューティング ウィザードのトポロジについて

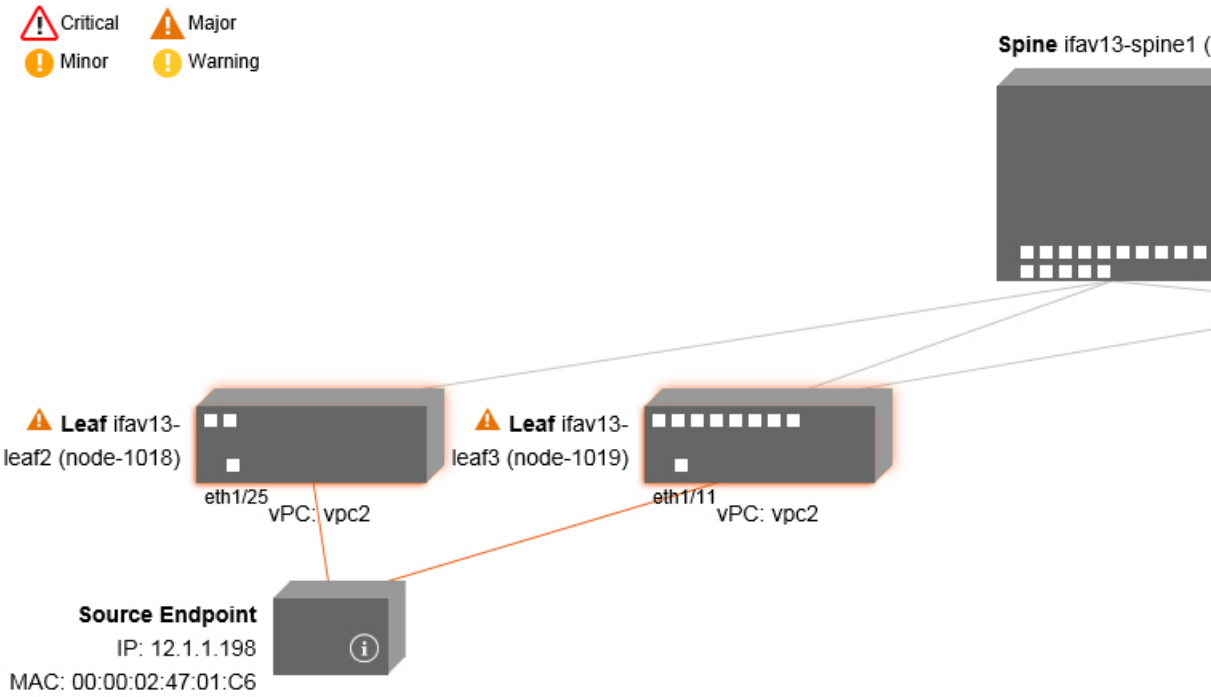
このセクションでは、トラブルシューティング ウィザードのトポロジについて説明します。トポロジは、送信元と接続先がどのようにファブリックに接続されているか、送信元から接続先までのネットワーク パス、および中間スイッチが何であるかを示しています。

次のウィザード トポロジ ダイアグラムに示すように、ソースはトポロジの左側に表示され、接続先は右側に表示されます。

（注） このウィザード トポロジには、送信元から接続先へのトラフィックに関係するデバイスのリーフスイッチ、スパイン スイッチ、および FEX のみが表示されます。ただし、他の多くのリーフスイッチ（数十または数百のリーフスイッチと他の多くのスパイン スイッチ）が存在する場合があります。

このトポロジには、リンク、ポート、およびデバイスも表示されます。 アイコンにカーソルを合わせると)、送信元または接続先が属するテナント、それが属するアプリケーション、使用しているトラフィックのカプセル化（VLAN など）が表示されます。

画面の左側に色の凡例があり（次のように表示されます）、トトポロジ図の各色に関連付けられたシビラティ（重大度）レベル（たとえば、クリティカルとマイナー）を説明します。



トポロジ内のボックスやポートなどの項目にカーソルを合わせると、より詳細な情報が表示されます。ポートまたはリンクに色が付いている場合は、トラブルシューティングが必要な問題があることを意味します。たとえば、色が赤またはオレンジの場合、これはポートまたはリンクに障害があることを示しています。色が白の場合、欠陥はありません。リンクで円の中に数字がある場合は、同じ2つのノード間の並列リンクの数が、円の色で示されるシビラティ（重大度）の障害の影響を受けていることを示します。ポートにカーソルを合わせると、送信元に接続されているポートを確認できます。

リーフスイッチを右クリックすると、スイッチのコンソールにアクセスできます。そのデバイスにログインできるポップアップ ウィンドウが表示されます。

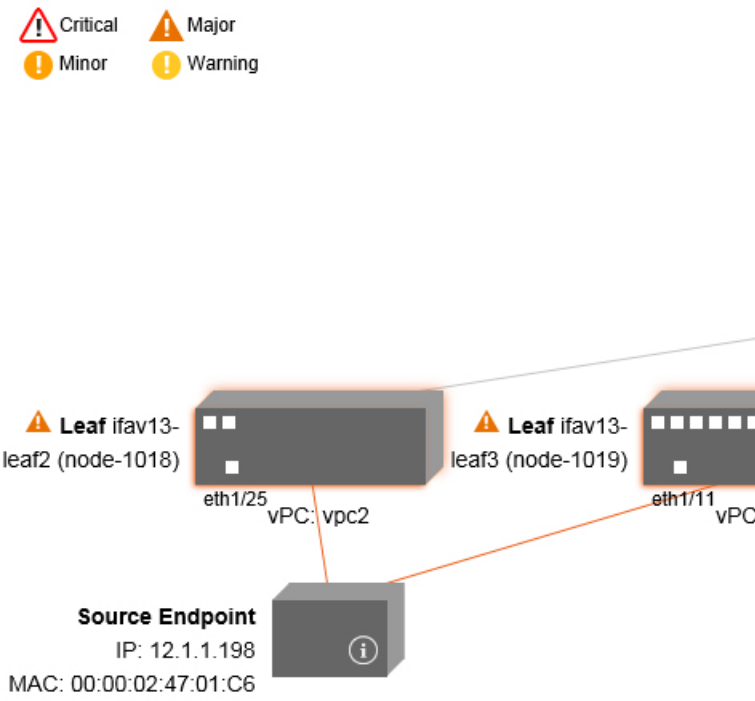
（注）

- レイヤ4からレイヤ7のサービス（ファイアウォールとロードバランサ）がある場合、それらもトポロジに表示されます。
- ロードバランサを使用するトポロジの場合、接続先は仮想 IP（VIP）アドレスであることが想定されます。
- 送信元またはターゲットが ESX サーバーの背後にある場合、ESX はトポロジに表示されます。

障害トラブルシューティング画面の使用

この手順では、障害トラブルシューティング ウィザードの使用方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	**[ナビゲーション（Navigation）]** ペインで **[障害（Faults）]** をクリックして、**[障害（Faults）]** トラブルシューティング画面の使用を開始します。	**[障害 （Faults）]** 画面には、以前に選択した送信元と接続先を接続するトポロジと、見つかった障害が表示されます。指定された通信の障害のみが表示されます。障害がある場合は常に、重大度を伝えるために特定の色で強調表示されます。画面上部の色の凡例を参照して、各色に関連付けられた重大度レベルを把握してください。白いボックスは、その特定の領域にはトラブルシューティング対象の問題がないことを示しています。 このトポロジには、トラブルシューティング セッションに関連するリーフ スイッチ、スパイン スイッチ、およびFEX も表示されます。リーフ スイッチ、スパイン スイッチ、FEX などの項目にカーソルを合わせるか、障害をクリックすると、分析のためのより詳細な情報が表示されます。 
ステップ 2	障害をクリックすると、分析のためのより詳細な情報を含む **[ドロップ統計（Drop Stats）]**、**[コントラクト ドロップ（Contract Drops）]**、および **[トラフィック統計（Traffic Stats）]** タブのあるダイアログ ボックスが表示されます。	

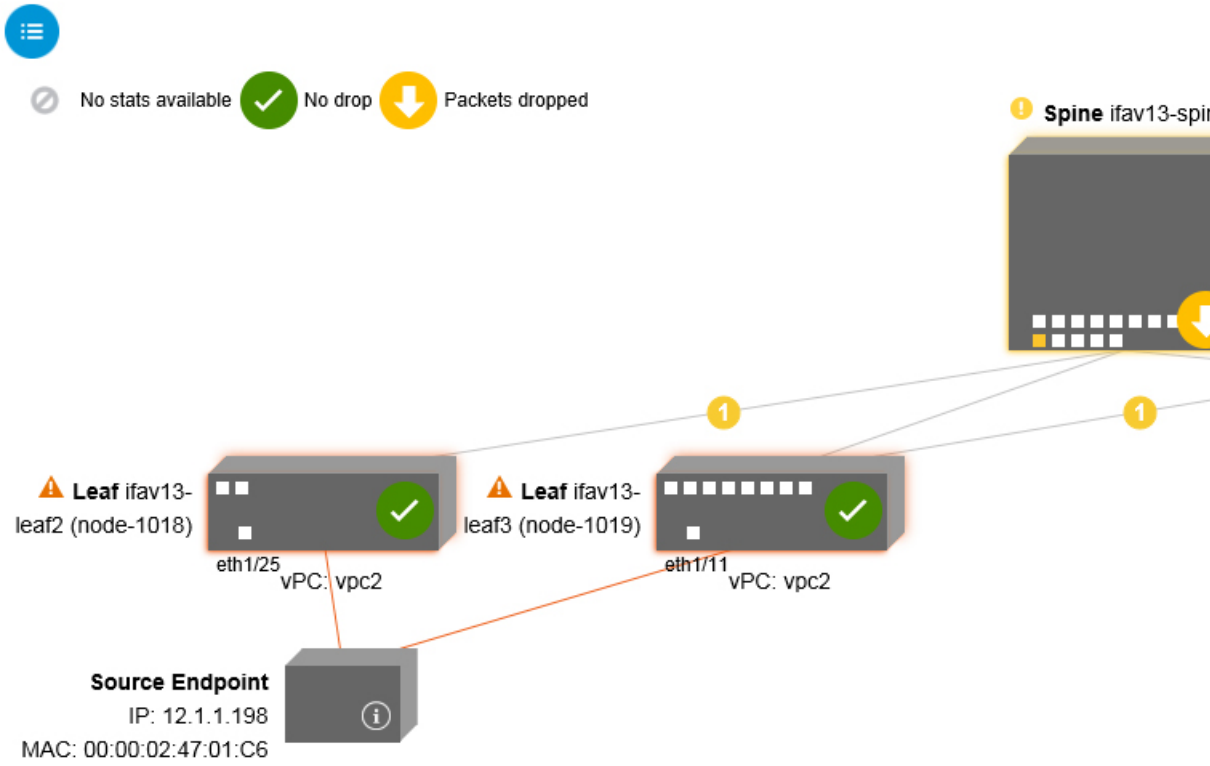
関連トピック

ドロップ/統計トラブルシューティング画面の使用 （113 ページ）

ドロップ/統計トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[ドロップ/統計（Drop/Stats）]** をクリックして、**[ドロップ/統計（Drop/Stats）]**のトラブルシューティング画面の使用を開始します。

[ドロップ/統計（Drop/Stats）] ウィンドウには、ドロップからのすべての統計情報を含むトポロジが表示されるため、ドロップが存在するかどうかを明確に確認できます。ドロップ画像をクリックすると、分析のための詳細情報が表示されます。



ドロップ画像をクリックすると、**[ドロップ/統計（Drop/Stats）]** 画面の上部に 3 つのタブがあり、表示される統計はその特定のリーフまたはスイッチにローカライズされます。

3 つの統計タブは次のとおりです。

- **[ドロップ統計（DROP STATS）]**

  このタブには、ドロップカウンタの統計が表示されます。さまざまなレベルでドロップされるパケットがここに表示されます。

（注） デフォルトでは、値がゼロのカウンタは非表示になっていますが、ユーザーはすべての値を表示するように設定できます。

- **[コントラクト ドロップ（CONTRACT DROPS）]**

  このタブには、発生したコントラクトドロップのリストが表示されます。これは個々のパケット ログ（ACL ログ）です。**送信元インターフェイス（Source Interface）、送信元 IP アドレス（Source IP address）、送信元ポート（Source Port）、宛先 IP アドレス（Destination IP address）、宛先ポート（Destination Port）**と**プロトコル（Protocol）**などの各パケットの情報が表示されます。

（注） すべてのパケットがここに表示されるわけではありません。

- **[トラフィック 統計情報（TRAFFIC STATS）]**

  このタブには、進行中のトラフィックを示す統計が表示されます。これらは、転送されたパケットの数です。

（注） デフォルトでは、値がゼロのカウンタは非表示になっていますが、ユーザーはすべての値を表示するように設定できます。

画面の左上隅にある [すべて] アイコン（  ）をクリックして、すべての管理対象オブジェクトのすべての統計を一度に表示することもできます。

ゼロまたはゼロ以外のドロップを選択するオプションもあります。**[値がゼロの統計を表示（Show stats with zero values）]**のチェックボックス（画面の左上隅）をオンにすると、既存のすべてのドロップを表示できます。**時間（Time）、影響を受けたオブジェクト（Affected Object）、統計（Stats）**、および**値（Value）**のフィールドには、すべてのゼロ値のデータが入力されます。

[ゼロ値の統計を表示（Show stats with zero values）] ボックスをチェックしない場合、ゼロ以外のドロップで結果が表示されます。

（注） **[すべて（All）]**アイコンをクリックした場合も、同じロジックが適用されます。3 つすべてのタブ（**[ドロップ統計（DROP STATS）]**、**[契約ドロップ（CONTRACT DROPS）]**、および**[トラフィック統計（TRAFFIC STATS）]**）も使用でき、同じタイプの情報が表示されます。

関連トピック

コントラクト トラブルシューティング画面の使用 （115 ページ）

コントラクト トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[コントラクト（Contracts）]** をクリックして、**[コントラクト（Contracts）]** トラブルシューティング画面の使用を開始します。

[コントラクト（Contracts）] トラブルシューティング画面には、送信元から宛先、および宛先から送信元に適用可能なコントラクトが表示されます。

青いテーブルの見出しの各行は、フィルタを示しています。各フィルタの下には、特定のリーフまたはスイッチの複数のフィルタ エントリ（**プロトコル、L4 発信元、L4 宛先、TCP フラグ、アクション、ノード**、および**ヒット**）を示す複数の行があります。

証明書アイコンにカーソルを合わせると、コントラクト名とコントラクトフィルタ名が表示されます。青いテーブルの各見出し行（またはフィルタ）の右側に表示されるテキストは、コントラクトのタイプを示します。次に例を示します。

- Epg から Epg
- BD 許可
- あらゆる状況に対応
- コンテキスト拒否

これらのコントラクトは、送信元から宛先へ、および宛先から送信元へと分類されます。

（注） 各フィルタに表示されるヒットは累積的です（つまり、特定のリーフごとに、そのコントラクト ヒット、コントラクト フィルタ、またはルールの合計ヒットが表示されます）。統計は 1 分ごとに自動的に更新されます。

情報（  ）アイコンにカーソルを合わせると、ポリシー情報を取得できます。また、参照されている EPG を確認することもできます。

（注） エンドポイント間にコントラクトがない場合、これは**[コントラクト データがありません（There is no contract）]** ポップアップで示されます。

関連トピック

イベントのトラブルシューティング画面の使用 （115 ページ）

イベントのトラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[イベントと監査（Events and Audits）]** をクリックして、**[イベントと監査（Events and Audits）]** トラブルシューティング画面の使用を開始します。

個々のリーフまたはスパインスイッチをクリックすると、その個々のイベントに関するより詳細な情報を表示できます。

[イベント（EVENTS）] と **[展開記録（DEPLOYMENT RECORDS）]** の 2 つのタブを使用できます。

- **[イベント（EVENTS）]** は、システム（物理インターフェースや VLANS など）で発生した変更のイベントレコードを表示します。特定のリーフごとに個別のイベントがリストされています。これらのイベントは、**重大度（Severity）**、**影響を受けるオブジェクト（Affected Object）**、**作成時間（Creation Time）**、**原因（ Cause）**、および **説明（Description）** に基づいて並べ替えることができます。
- **[展開記録（DEPLOYMENT RECORDS）]** は、物理インターフェイス、VLAN、VXLAN、および L3 CTX でのポリシーの展開を示しています。これらのレコードは、epg のために VLAN がリーフに配置された時刻を示しています。

[すべての変更（All Changes）] 画面の **[すべて（All）]** アイコン（  ）をクリックすると、指定した時間間隔（またはトラブルシューティングセッション）中に発生した変更を示すすべてのイベントを表示できます。

[すべての変更（All Changes）] 画面には、次の 3 つのタブがあります。

- **[監査（AUDITS）]**

  監査にはリーフ アソシエーションがないため、**[すべての変更（All Changes）]** 画面でのみ使用できます。

- **[イベント（EVENTS）]**（上記）
- **[展開記録（DEPLOYMENT RECORDS）]**（上記）

関連トピック

Traceroute トラブルシューティング画面の使用 （116 ページ）

Traceroute トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[Traceroute]** をクリックして、**[Traceroute]** トラブルシューティング画面の使用を開始します。

トラブルシューティングのために traceroute を作成して実行するには、次の手順を実行します。

1. **[Traceroute]** ダイアログボックスで、**[接続先ポート（Destination Port）]** ドロップダウンリストで、接続先ポートを選択します。
2. **[プロトコル（Protocol）]** プルダウン メニューからプロトコルを選択します。サポートされているオプションは次のとおりです。
   - **icmp**：このプロトコルは一方向であり、ソース リーフから接続先エンドポイントのみへの traceroute を実行します。
   - **tcp** ：このプロトコルも双方向です（**udp** プロトコルについての説明を参照してください）。

- **udp** ：このプロトコルは双方向であり、ソース リーフから接続先エンドポイントへの traceroute を実行し、次に接続先リーフからソース エンドポイントへの traceroute を実行します。

（注） IPv4 だけが UDP、TCP、および ICMP プロトコルをサポートします。IPv6 の場合、UDP のみがサポートされます。

3. traceroute を作成したら、**[再生（Play）]**（または Start）ボタンをクリックして traceroute を開始します。

（注） **[再生（Play）]** ボタンを押すと、システム上にポリシーが作成され、**警告**メッセージが表示されます。

4. **[OK]** をクリックして続行すると、traceroute の実行が開始されます。
5. **[停止（Stop）]** ボタンをクリックして、traceroute を終了します。

（注） **[停止（Stop）]** ボタンを押すと、ポリシーがシステムから削除されます。

traceroute が完了すると、起動された場所と結果が表示されます。**[Traceroute の結果（Traceroute Results）]** の隣には、traceroute が起動された場所（ソースから接続先へ、または接続先からソースへ）を示すプルダウン メニューがあります。

結果は、**実行時間、Traceroute ステータス、接続先ポート**、および**プロトコル**の情報を含む **[Traceroute]** ダイアログにも表示されます。

結果は、緑と赤の矢印で表されます。緑の矢印は、traceroute プローブに応答したパス内の各ノードを表すために使用されます。赤い矢印の始点は、トレースルートプローブに応答した最後のノードであるため、パスが終了する場所を表します。ユーザーは traceroute を起動する方向を選択しません。traceroute は常にセッションに対して開始されます。セッションが次の場合：

- EP から外部 IP または外部 IP から EP の場合、traceroute は常に EP から外部 IP に起動されます。
- EP から EP でありプロトコルが ICMP である場合、traceroute は常に送信元から接続先へ起動されます。
- EP から EP でありプロトコルが UDP/TCP である場合、traceroute は常に双方向です。

（注）

- **[Traceroute の結果（Traceroute Results）]**ドロップダウン メニューを使用して、上記のシナリオ #3 の各方向の結果を表示/視覚化できます。シナリオ #1 と #2 では、常にグレー表示です。
- **[Traceroute ステータス（Traceroute Status）]**が未完了と表示される場合、これは、データの一部が戻ってくるのをまだ待っていることを意味します。**[Traceroute ステータス（Traceroute Status）]** が**完了**の場合、実際に完了しています。

関連トピック

アトミック カウンタ トラブルシューティング画面の使用 （118 ページ）

アトミック カウンタ トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインの **[アトミック カウンタ（Atomic Counter）]** をクリックして、**[アトミック カウンタ（Atomic Counter）]** のトラブルシューティング画面の使用を開始します。

[アトミック カウンタ（Atomic Counter）] 画面は、送信元と接続先の情報を取得し、それに基づいてカウンタ ポリシーを作成するために使用されます。2 つのエンドポイント間にアトミック カウンタ ポリシーを作成し、ソースから宛先、および宛先からソースに行き来するトラフィックを監視できます。通過するトラフィックの量を判断でき、特に、送信元と宛先のリーフ間で異常（ドロップまたは超過パケット）が報告されているかどうかを判断できます。

画面の上部に **[再生（Play）]**（または [開始]（Start））および **[停止（Stop）]** ボタンがあるため、いつでもアトミック カウンタ ポリシーを開始または停止でき、送信されているパケットをカウントできます。

（注） **[再生（Play）]**ボタンを押すと、システム上にポリシーが作成され、パケットカウンターが開始されます。**[停止（Stop）]** ボタンを押すと、ポリシーがシステムから削除されます。

結果は 2 つの異なる形式で表示されます。要約を含む短い形式と、長い形式です（**[展開（Expand）]**ボタンをクリックします）。簡易形式と展開形式の両方で、両方の方向を表示できます。展開形式では、累積カウントと最新の 30 秒間隔ごとのカウントが表示されます。簡易形式では、累積および最後の間隔のカウントのみが表示されます。

関連トピック

SPAN トラブルシューティング画面の使用 （118 ページ）

SPAN トラブルシューティング画面の使用

[ナビゲーション（Navigation）] ペインで **[SPAN]** をクリックして、**SPAN** トラブルシューティング画面の使用を開始します。

この画面を使用して、双方向トラフィックをスパン（またはミラーリング）して、アナライザにリダイレクトできます。SPAN セッションでは、コピーを作成してアナライザに送信します。

このコピーは特定のホスト（アナライザーの IP アドレス）に送信され、Wireshark などのソフトウェア ツールを使用してパケットを表示できます。セッション情報には、送信元と宛先の情報、セッション タイプ、およびタイムスタンプの範囲があります。

（注） **[再生（Play）]** ボタンを押すと、システム上にポリシーが作成されます。**[停止（Stop）]** ボタンを押すと、ポリシーがシステムから削除されます。

（注） トラブルシューティング ウィザードの CLI コマンドのリストについては、*Cisco APIC* コマンドライン インターフェイス ユーザー ガイドを参照してください。

Cisco APIC トラブルシューティング CLI を使用して SPAN セッションを作成する

このセクションでは、Cisco APIC トラブルシューティング CLI を使用して SPAN セッションを作成する方法を示します。

ステップ 1 **troubleshoot node session** <*session_name*> **nodename** <*node_id*>

ノードレベルのセッション（グローバル ドロップ）を作成するには：

例：

```
apic1(config)# troubleshoot node session 301-GD-APIC nodeid 301
```

ステップ 2 **troubleshoot node session** <*session_name*> **nodename** <*node_id*> **interface ethernet** <*interface*>

インターフェイス レベルのセッションを作成するには：

例：

```
apic1(config)# troubleshoot node session 301-GD-APIC nodeid 301 interface eth1/3
```

ステップ 3 **troubleshoot node session** <*session_name*> **monitor destination** *apic_ip* **srcipprefix** <*ip_prefix*> **drop enable erspan-id**[optional]

宛先を Cisco APIC として指定し、ドロップ時に SPAN を有効にするには：

例：

```
apic1(config)# troubleshoot node  session 301-GD-APIC monitor destination apic srcipprefix 13.13.13.13
 drop enable
```

ステップ 4 **troubleshoot node session** <*session_name*> **monitor destination tenant** *tenant* **application** <*app*> **destip** <*dest_ip*>**srcipprefix**<*ip_prefix*>**drop enable erspan-id**[optional]

ERSPAN 宛先を指定し、ドロップ時に SPAN を有効にするには：

例：

```
apic1(config)# troubleshoot node  session 301-GD-APIC monitor destination tenant ERSPAN application
 A1 epg E1 destip 179.10.10.179 srcipprefix 31.31.13.31 drop enable
```

宛先として設定されているときに Cisco APIC で SPAN-on-drop パケットを確認するには：

1. SPAN-on-drop セッションを無効にします：

   ```
   apic1(config)# no troubleshoot node session 301-GD-APIC monitor
   ```

2. drop-stats ディレクトリに移動し、DropPackets_*.pcap ファイルを確認します（**/data2/techsupport/troubleshoot/node/Session_name/span_capture/drop-stats/DropPackets_*.pcap**）。

L4 ～ L7 サービス検証済みシナリオ

トラブルシューティング ウィザードを使用すると、ユーザーは 2 つのエンドポイントを指定し、それらのエンドポイント間の対応するトポロジを表示できます。トポロジ内の 2 つのエンドポイント間に L4 ～ L7 サービスが存在する場合、これらも表示できます。

このセクションでは、このリリースで検証された L4 から L7 のシナリオについて説明します。L4 ～ L7 サービス内では、トポロジの数が非常に多いため、ファイアウォール、ロードバランサ、およびそれぞれの組み合わせのため、さまざまな構成が使用される可能性があります。トポロジ内の 2 つのエンドポイント間にファイアウォールが存在する場合、トラブルシューティング ウィザードはファイアウォール データとファイアウォールからリーフへの接続を取得します。2 つのエンドポイント間にロードバランサーが存在する場合、ロードバランサーまでの情報を取得して表示できます（サーバーまでは表示できません）。

次の表は、トラブルシューティング ウィザードで検証された L4 ～ L7 サービス シナリオを示しています。

シナリオ	1	2	3	4	5	6
ノード数	1	1	2	1	1	2
デバイス	GoTo FW（vrf分割）	GoTo SLB	GoTo、GoTo FW、SLB	FW-GoThrough	SLB-GoTo	FW、SLB（GoThrough、GoTo）
アーム数	2	2	2	2	2	2
コンシューマ	EPG	EPG	EPG	L3Out	L3Out	L3Out
プロバイダー	EPG	EPG	EPG	EPG	EPG	EPG
デバイスタイプ	VM	VM	VM	物理	物理	物理
コントラクトの適用範囲	tenant	コンテキスト	コンテキスト	コンテキスト	コンテキスト	グローバル
コネクタ モード	L2	L2	L2、L2	L3、L2	L3	L3/L2、L3

シナリオ	1	2	3	4	5	6
サービスアタッチ	BSW	BSW	DL / PC	通常のポート	vPC	通常のポート
クライアントアタッチ	FEX	FEX	FEX	通常のポート	通常のポート	通常のポート
サーバーアタッチ	vPC	vPC	vPC	通常のポート	通常のポート	通常のポート

エンドポイントからエンドポイントへの接続 API のリスト

以下は、EP から EP への（エンドポイント間）接続で使用可能なトラブルシューティング ウィザード API のリストです。

インタラクティブ API

エンドポイント（ep）からエンドポイントへの対話型トラブルシューティング セッションを作成するには、**interactive** API を使用します。モジュール名は **troubleshoot.eptoeputils.topo** で、関数は **getTopo** です。対話型 API に必要な引数 **(req_args)** は **- session** です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

createsession API

エンドポイント（ep）からエンドポイントへのトラブルシューティングセッションを作成するには、**createsession** API を使用します。モジュール名は **troubleshoot.eptoeputils.session** で、関数は **createSession** です。

createsession API の必須引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
-action	traceroute/atomiccounter の start/stop/status など
- スケジューラ	
- srctenant	送信元エンドポイントのテナントの名前
- srcapp	送信元エンドポイントのアプリの名前
- srcepg	送信元エンドポイントのエンドポイント グループの名前

- dsttenant	宛先エンドポイントのテナントの名前
- dstapp	宛先エンドポイントのアプリの名前
- dstepg	宛先エンドポイントのエンドポイント グループの名前
- mode	内部で使用

変更セッション API

エンドポイント（ep）セッションからエンドポイントのトラブルシューティングセッションに変更するには、**modifysession** API を使用します。モジュール名は **troubleshoot.eptoeputils.topo** で、関数は **modifySession** です。

modifysession API に必要な引数（**req_args**）は、**- session**（セッション名）および**- mode** です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明

アトミックカウンタ API

エンドポイント（ep）からエンドポイントへのアトミック カウンタ セッションを作成するには、**atomiccounter API** を使用します。モジュール名は**troubleshoot.eptoeputils.atomiccounter** で、関数は**manageAtomicCounterPols**です。

atomiccounter API に必要な引数 (**req_args**) は次のとおりです。

- - session
- - アクション
- - モード

（注） atomiccounter API にはオプションの引数（**opt_args**）はありません。

traceroute API

APIを使用してエンドポイント（ep）からエンドポイントのトレースルートセッションを作成するには、**raceroute** API を使用します。モジュール名は **troubleshoot.eptoeputils.traceroute** で、関数は **manageTracerouteePols** です。

traceroute API に必要な引数（**req_args**）には、次のものがあります。

- - session（セッション名）
- - action（start/stop/status）
- - mode

構文の説明

オプションの引数（**opt_args**）	説明
- protocol	プロトコル名
- dstport	宛先ポート名

span API

エンドポイント（ep）からエンドポイントまでのスパンのトラブルシューティングセッションを作成するには、**span** API を使用します。モジュール名は **troubleshoot.eptoeputils.span** で、関数は **monitor** です。

span API に必要な引数（**req_args**）は、以下のものを含みます。

- - session（セッション名）
- - action（start/stop/status）

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス

- starttime	トラブルシューティングセッションの開始時刻
- endtime	トラブルシューティングセッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティングセッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- srctenant	送信元エンドポイントのテナントの名前
- srcapp	送信元エンドポイントのアプリの名前
- srcepg	送信元エンドポイントのエンドポイントグループの名前
- dsttenant	宛先エンドポイントのテナントの名前
- dstapp	宛先エンドポイントのアプリの名前
- dstepg	宛先エンドポイントのエンドポイントグループの名前
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

generatereport API

API を使用してトラブルシューティング レポートを生成するには、**generatereport** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **generateReport** です。

generatereport API に必要な引数（**req_args**）は、 **- session** (セッション名) および **- mode** です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- include	Obsolete
- format	生成するレポートのフォーマット

スケジュールレポート API

API を使用してトラブルシュート レポートの生成をスケジュールするには、**schedulereport API** を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **scheduleReport** です。schedulereport API に必要な引数（**req_args**）は **- session** です。

schedulereport API に必要な引数（**req_args**）には、以下のものが含まれます。

- - session（セッション名）
- - scheduler（スケジューラ名）
- - mode

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- include	Obsolete
- format	生成するレポートのフォーマット
- action	traceroute / atomiccounter の開始 / 停止 / ステータスなど

getreportstatus API

API を使用して生成されたレポートのステータスを取得するには、**getreportstatus** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **getStatus** です。

getreportstatus API に必要な引数（**req_args** ）は次のとおりです。

- - session（セッション名）
- - sessionurl（セッション URL）

- - mode

（注） getreportstatus API にはオプションの引数（**opt_args**）はありません。

getreportslist API

API を使用して生成されたレポートのリストを取得するには、**getreportslist** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **getReportsList** です。

getreportslist API に必要な引数（**req_args**）は、 **- session** (セッション名) および **- mode**です。

（注） getreportslist API には、オプションの引数（**opt_args**）はありません。

getsessionslist API

API を使用してトラブルシューティング セッションのリストを取得するには、**getsessionslist** API を使用します。モジュール名は **troubleshoot.eptoeputils.session** で、機能は **getSessions** です。

getsessionlist API の必須引数（**req_args**）は **- mode** です。

（注） getsessionlist API には、オプションの引数（**opt_args**）はありません。

getsessiondetail API

API を使用してトラブルシューティング セッションに関する特定の詳細を取得するには、**getsessiondetai**l API を使用します。モジュール名は **troubleshoot.eptoeputils.session** で、関数は **getSessionDetail** です。

getsessiondetail API に必要な引数（**req_args**）は、**- session**（セッション名）および **- mode** です。

（注） getsessiondetail API にはオプションの引数（**opt_args**）はありません。

deletesession API

API を使用して特定のトラブルシューティング セッションを削除するには、**deletesession** API を使用します。モジュール名は **troubleshoot.eptoeputils.session** で、機能は **deleteSession** です。

deletesession API の必須引数（**req_args**）は **- session** (セッション名) です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
- action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

clearreports API

API を使用して生成されたレポートのリストをクリアするには、**clearreports** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **clearReports** です。

clearreports API に必要な引数（**req_args**）は、 **- session**（セッション名) および **- mode** です。

（注） clearreports API にはオプションの引数（**opt_args**）はありません。

コントラクト API

API を使用してコントラクト情報を取得するには、**contracts** API を使用します。モジュール名は**troubleshoot.eptoeputils.contracts**で、関数は **getContracts** です。

contract API に必要な引数 （**req_args**）は、**- session**（セッション名）と **- mode** です。

contract API にはオプションの引数（**opt_args**）はありません。

エンドポイントからレイヤ 3 外部接続の API リスト

以下は、EPからEPへの（エンドポイント間）接続で使用可能なトラブルシューティング ウィザード API のリストです。

• コントラクト API （141 ページ）
• ratelimit API （142 ページ）
• 13ext API （143 ページ）

インタラクティブ API

エンドポイント（ep）からレイヤ3（L3）への外部対話型トラブルシューティングセッションを作成するには、**interactive** API を使用します。モジュール名は**troubleshoot.epextutils.epext_topo** で、関数は**getTopo**です。対話型 API に必要な引数 (**req_args**) は、**- session**、**- include**、および **- mode** です。

次の表にオプションの引数（**opt_args**）が表示されています：

構文の説明

オプションの引数（opt_args）	説明
- refresh	

createsession API

API を使用してエンドポイント（Ep）からレイヤ 3（L3）への外部トラブルシューティングセッションを作成するには、**createsession** API を使用します。モジュール名は **troubleshoot.epextutils.epextsession** で、関数は **createSession** です。createsession API の必須引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻

- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

変更セッション API

エンドポイント（Ep）をレイヤ 3（L3）の外部トラブルシューティング セッションに変更するには、**modifysession API** を使用します。モジュール名は **troubleshoot.epextutils.epextsession** で、関数は **modifySession** です。modifysession API の必須引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス

- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

アトミックカウンタ API

エンドポイント（ep）からエンドポイントへのアトミック カウンタ セッションを作成するには、**atomiccounter API** を使用します。モジュール名は **troubleshoot.epextutils.epext_ac** で、関数は **manageAtomicCounterPols** です。

atomiccounter API に必要な引数（**req_args**）は次のとおりです。

- - session（セッション名）
- - action（start/stop/status）

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名

- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- ui	内部で使用（無視）
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

traceroute API

API を使用してレイヤ 3 外部 traceroute トラブルシューティング セッションへのエンドポイント（ep）を作成するには、**traceroute** API を使用します。モジュール名は **troubleshoot.epextutils.epext_traceroute** で、関数は **manageTraceroutePols** です。

traceroute API に必要な引数（**req_args**）には、次のものがあります。

- - session（セッション名）
- - action（start/stop/status）

構文の説明

オプションの引数（**opt_args**）	説明
- protocol	プロトコル名
- dstport	宛先ポート名
- srcep	送信元エンドポイント
- dstep	宛先エンドポイント

- srcip	送信元 IP アドレス
- dstip	宛先 IP アドレス
- srcextip	送信元外部 IP アドレス
- dstIp	接続先外部 IP アドレス
- ui	内部で使用（無視）
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

span API

エンドポイント（Ep）からレイヤー 3（L3）への外部スパンのトラブルシューティング セッションを作成するには、**span** API を使用します。モジュール名は **troubleshoot.epextutils.epext_span**で、関数は **monitor** です。

span API に必要な引数（**req_args**）は、以下のものを含みます。

- - session（セッション名）
- - action（start/stop/status）
- - mode

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- ポートリスト	ポートのリスト
- dstapic	接続先 APIC
- srcipprefix	送信元エンドポイントのIPアドレスプレフィックス
- flowid	[フローID（Flow ID）]
- dstepg	接続先 エンドポイント グループ
- dstip	接続先エンドポイント IP アドレス
- analyser	???
- desttype	宛先タイプ（Destination type）

- spansrcports	スパン ソース ポート

generatereport API

API を使用してトラブルシューティング レポートを生成するには、**generatereport** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **generateReport** です。

generatereport API に必要な引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所

-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

スケジュールレポート API

API を使用してトラブルシュート レポートの生成をスケジュールするには、**schedulereport API** を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **scheduleReport** です。schedulereport API に必要な引数（**req_args**）は **- session** です。

schedulereport API に必要な引数（**req_args**）には、以下のものが含まれます。

- - session（セッション名）
- - scheduler（スケジューラ名）

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント
- dstep	宛先エンドポイント
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- srcepid	Obsolete

- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
-action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

getreportstatus API

API を使用して生成されたレポートのステータスを取得するには、**getreportstatus** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **getStatus** です。

getreportstatus API に必要な引数（**req_args** ）は次のとおりです。

- - session（セッション名）
- - sessionurl（セッション URL）
- - mode

（注） getreportstatus API にはオプションの引数（**opt_args**）はありません。

getreportslist API

API を使用して生成されたレポートのリストを取得するには、**getreportslist** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **getReportsList** です。

getreportslist API に必要な引数（**req_args**）は、 **- session** (セッション名) および **- mode**です。

（注） getreportslist API には、オプションの引数（**opt_args**）はありません。

getsessionslist API

API を使用してトラブルシューティング セッションのリストを取得するには、**getsessionslist** API を使用します。モジュール名は**troubleshoot.epextutils.epextsession**で、関数は**getSessions**です。

（注） この API には必須の引数はありません。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- session	セッション名
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）

- sessionurl	レポートの場所
- action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

getsessiondetail API

API を使用してトラブルシューティング セッションに関する特定の詳細を取得するには、**getsessiondetail** API を使用します。モジュール名は **troubleshoot.epextutils.session** で、関数は **getSessionDetail** です。getsessiondetail API の必須引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- description	セッションについての説明
- scheduler	レポート生成のスケジューラ名
- srcepid	Obsolete
- dstepid	Obsolete

- include	Obsolete
- format	生成するレポートのフォーマット
- ui	内部で使用（無視）
- sessionurl	レポートの場所
- action	traceroute/atomiccounter の start/stop/status など
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

deletesession API

API を使用して特定のトラブルシューティング セッションを削除するには、**deletesession** API を使用します。モジュール名は **troubleshoot.epextutils.epextsession** で、関数は **deleteSession** です。

deletesession API に必要な引数（**req_args**）は、**- session**（セッション名）および **- mode** です。

（注） deletesession API にはオプションの引数（**opt_args**）はありません。

clearreports API

API を使用して生成されたレポートのリストをクリアするには、**clearreports** API を使用します。モジュール名は **troubleshoot.eptoeputils.report** で、関数は **clearReports** です。

clearreports API に必要な引数（**req_args**）は、**- session**（セッション名) および **- mode** です。

（注） clearreports API にはオプションの引数（**opt_args**）はありません。

コントラクト API

API を使用してコントラクト情報を取得するには、**contracts** API を使用します。モジュール名は **troubleshoot.epextutils.epext_contracts** で、関数は **getContracts** です。contract API に必要な引数（**req_args**）は **- session**（セッション名）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- epext	エンドポイントから外部へ
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用
- ui	内部で使用（無視）

ratelimit API

このセクションでは、**ratelimit** API に関する情報を提供します。モジュール名は **troubleshoot.eptoeputils.ratelimit** で、関数は **control** です。ratelimit API に必要な引数（**req_args**）は **- action**（start/stop/status）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（opt_args）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス

- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス
- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- epext	エンドポイントから外部へ
- mode	内部で使用
- _dc	内部で使用
- ctx	内部で使用

13ext API

このセクションでは、**13ext** API に関する情報を提供します。モジュール名は **troubleshoot.epextutils.l3ext** で、関数は **execute** です。13ext API に必要な引数（**req_args**）は **-action**（start/stop/status）です。

次の表に、オプションの引数（**opt_args**）とそれぞれの説明を示します。

構文の説明

オプションの引数（**opt_args**）	説明
- srcep	送信元エンドポイント名
- dstep	接続先 エンドポイント名
- srcip	送信元 エンドポイントの IP アドレス
- dstip	接続先エンドポイント IP アドレス
- srcmac	送信元エンドポイント MAC
- dstmac	接続先 エンドポイント MAC
- srcextip	L3 外部送信元 IP アドレス
- dstextip	L3 外部接続先 IP アドレス

- starttime	トラブルシューティング セッションの開始時刻
- endtime	トラブルシューティング セッションの終了時刻
- latestmin	開始時刻から開始するトラブルシューティング セッションの時間枠（分単位）
- epext	エンドポイントから外部へ
- mode	内部で使用

設定の同期の問題の確認

Cisco Application Centric Infrastructure（APIC）で要求（構成の変更など）を行うと、通常、変更が行われたことがすぐにわかります。ただし、Cisco APICで問題が発生した場合は、GUI でチェックして、まだ有効になっていないユーザー設定可能なオブジェクトに関連するトランザクションがあるかどうかを確認できます。パネルの情報を使用して、デバッグに役立てることができます。

Cisco APIC GUI の **[解決を保留中の構成オブジェクト（Configuration Objects Pending Resolution）]** パネルには、遅れているものがあるかどうかが表示されます。

始める前に

ステップ 1 Cisco APIC にログインします。

ステップ 2 画面の右上にある設定アイコン（歯車の記号）をクリックし、**[構成の同期の問題（Config Sync Issues）]** を選択します。

ステップ 3 **[解決を保留中の構成オブジェクト（Configuration Objects Pending Resolution）]** パネルで、テーブルに何かがリストされていないか確認します。

テーブルにエントリがない場合、同期の問題はありません。

ステップ 4 エントリがある場合は、テーブルの情報をキャプチャし、デバッグまたはシスコ サポートとの連携に使用します。

ユーザー アクティビティの表示

Cisco APIC セットアップの変更に気付いた場合、管理者は **[ユーザー アクティビティ（User Activities）]**機能を使用して、ユーザーが実行したアクションの2週間の履歴を表示できます。履歴データには、アクションが発生したときのタイムスタンプ、アクションを実行したユーザー、ユーザーが実行したアクション、影響を受けるオブジェクト、および説明が含まれます。

ユーザー アクティビティへのアクセス

[ユーザー アクティビティ（User Activies）] ウィンドウでは、Cisco APIC GUI で実行されたユーザー アクティビティの 2 週間の履歴を表示できます。

ステップ 1 メニュー バーから、**[システム（System）]** > **[アクティブ セッション（Active Sessions）]** を選択します。

[アクティブ セッション（Active Session）] ウィンドウが表示されます。

ステップ 2 アクティブなセッションを右クリックし、**[ユーザー アクティビティ（User Activies）]** を選択します。

ユーザー アクティビティのリストが表示されます。

（注） フィールドの説明については、**[アクティブ セッション（Active Session）]**ウィンドウの右上隅のヘルプ アイコンをクリックして、ヘルプ ファイルを表示してください。

ステップ 3 ドロップダウン メニューの **[最後のアクション（Actions in the last）]** をクリックして、ユーザー アクティビティを表示する履歴を選択します。

組み込み論理アナライザ モジュール

組み込み論理アナライザ モジュールについて

ELAM（組み込み論理アナライザ モジュール）は、シスコ ASIC の内部を調べ、パケットの転送方法を理解するためのエンジニアリング ツールです。ELAMは、転送パイプラインの中に組み込まれていて、パフォーマンスとコントロール プレーン リソースに影響を及ぼさずにリアルタイムでパケットをキャプチャできます。ELAM は、次の機能を実行できます。

- パケットがフォワーディング エンジンに到達したかどうかを判断する
- 受信したパケットのポートと VLAN を指定する
- パケットを表示する（レイヤ 2 からレイヤ 4 のデータ）
- パケットが送信された場所で変更されたかどうかを確認する

モジュラ スイッチの簡略簡略出力での ELAM レポートの生成

Cisco Application Policy Infrastructure Controller（APIC）4.2(1) リリースでは、人間が読める簡略化された ELAM 出力が導入されました。簡略出力をサポートするのは、EX、FX か FX2 がスイッチ名の最後にあるスイッチ モデルだけです。モジュラ スイッチでは、次の手順に従います。

ステップ 1 ELAM ツールを実行して、パケット転送情報を収集します。正確なコマンドとパラメータは、ハードウェアによって異なります。

ステップ 2 **ereport** コマンドを実行して、オリジナル形式と簡略形式のパケット転送情報 ELAM レポートを作成します。

例：

```
module-1(DBG-elam-el6)# ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT

===============================================================
                    Trigger/Basic Information
===============================================================
ELAM Report File    : /tmp/logs/elam_2019-09-04-51m-13h-30s.txt
.
.
.

module-1(DBG-elam-el6)# exit
module-1(DBG-elam)# exit
module-1# exit

apic1-leaf11# cd /tmp/logs
apic1-leaf11# ls | grep elam
elam_2019-09-04-51m-13h-30s.txt
pretty_elam_2019-09-04-51m-13h-30s.txt
apic1-leaf11#
```

ELAM は、出力ファイルを /tmp/logs/ ディレクトリに保存します。この例では、elam_2019-09-04-51m-13h-30s.txt ファイルがオリジナル形式の ELAM レポートで、pretty_elam_2019-09-04-51m-13h-30s.txt ファイルが簡略形式の ELAM レポートです。ただし、このままでは簡略形式のファイルは空になります。簡略形式でレポートを取得するには、追加の手順を実行する必要があります。

ステップ 3 オリジナル形式の ELAM レポートをスーパーバイザの /bootflash ディレクトリにアップロードします。

この例では、このレポートは elam_2019-09-04-51m-13h-30s.txt ファイルです。

ステップ 4 管理者として スーパーバイザ にログインします。

ステップ 5 /tmp、または管理ユーザーが書き込み権限を持つ任意のディレクトリに移動します。

例：

```
# cd /tmp
```

ステップ 6 オリジナル形式の ELAM レポートに対し、**decode_elam_parser** コマンドを実行します。

例：

```
# decode_elam_parser /bootflash/elam_2019-09-04-51m-13h-30s.txt
```

decode_elam_parser コマンドは、簡略出力ファイルを現在のディレクトリに保存します。

固定フォームファクタースイッチの簡易出力でのELAMレポートの生成

Cisco Application Policy Infrastructure Controller（APIC）4.2(1) リリースでは、人間が読める簡略化された ELAM 出力が導入されました。簡略出力をサポートするのは、EX、FX か FX2 がスイッチ名の最後にあるスイッチ モデルだけです。固定フォーム ファクタのリーフ スイッチとスパイン スイッチには、次の手順を使用します。

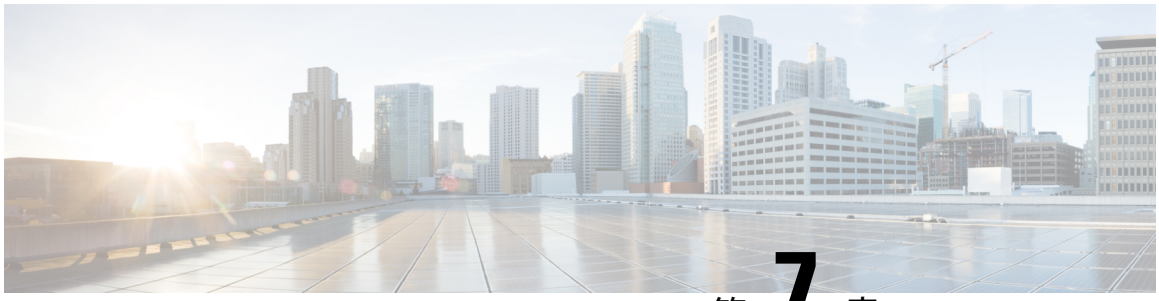
ステップ 1 ELAM ツールを実行して、パケット転送情報を収集します。正確なコマンドとパラメータは、ハードウェアによって異なります。

ステップ 2 **ereport** コマンドを実行して、オリジナル形式と簡略形式のパケット転送情報 ELAM レポートを作成します。

例：

```
module-1(DBG-elam-insel6)# ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT

================================================================
                    Trigger/Basic Information
================================================================
ELAM Report File    : /tmp/logs/elam_2019-09-04-51m-13h-30s.txt
.
.
.

module-1(DBG-elam-insel6)# exit
module-1(DBG-elam)# exit
module-1# exit

apic1-leaf11# cd /tmp/logs
apic1-leaf11# ls | grep elam
elam_2019-09-04-51m-13h-30s.txt
pretty_elam_2019-09-04-51m-13h-30s.txt
apic1-leaf11#
```

ELAM は、出力ファイルを `/tmp/logs/` ディレクトリに保存します。この例では、`elam_2019-09-04-51m-13h-30s.txt` ファイルがオリジナル形式の ELAM レポートで、`pretty_elam_2019-09-04-51m-13h-30s.txt` ファイルが簡略形式の ELAM レポートです。



第 7 章

GUIからの無効なインターフェイスおよび廃止されたスイッチの手動での削除

ファブリック ポートがシャットダウンされてから再びアップされるシナリオでは、ポート エントリが GUI で無効のままになる可能性があります。これが発生した場合、ポートで操作を実行できません。これを解決するには、ポートを GUI から手動で削除する必要があります。

GUI からの無効なインターフェイスおよび廃止されたスイッチの手動での削除

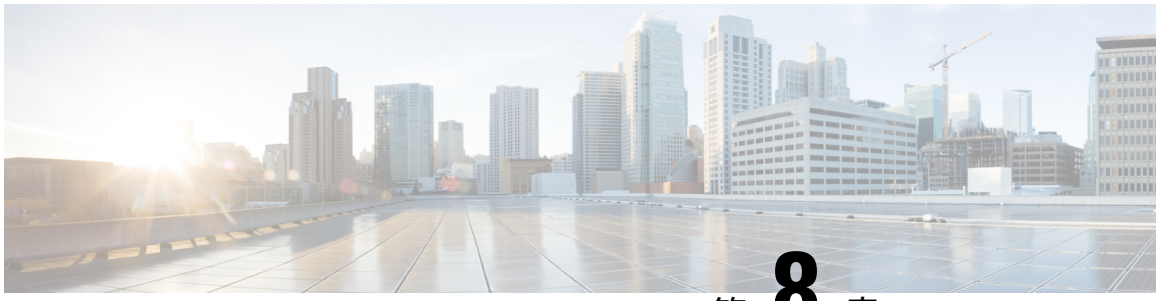
ファブリック ポートがシャットダウンされてから再びアップされるシナリオでは、ポート エントリが GUI で無効のままになる可能性があります。これが発生した場合、ポートで操作を実行できません。これを解決するには、ポートを GUI から手動で削除する必要があります。

ステップ 1 **[ファブリック（Fabric）]** タブで、**[インベントリ（Inventory）]** をクリックします。

ステップ 2 **[ナビゲーション（Navigation）]**ペインで、**[インターフェイスと廃止されたスイッチを無効にする（Disabled Interfaces and Decommissioned Switches）]** をクリックします。
無効になっているインターフェイスと廃止されたスイッチのリストが、**[作業（Work）]**ペインの要約テーブルに表示されます。

ステップ 3 **[作業（Work）]** ペインで、削除するインターフェイスまたはスイッチを右クリックし、**[削除（Delete）]** を選択します。

GUI からの無効なインターフェイスおよび廃止されたスイッチの手動での削除



第 8 章

スイッチのデコミッションおよび再コミッション

この章は、次の内容で構成されています。

スイッチのデコミッションおよび再コミッション

ポッドのすべてのノードをデコミッションし、再コミッションするには、この手順を実行します。この使用例の 1 つは、ノード ID をより論理的でスケーラブルな番号付け規則に変更することです。

ステップ 1 ノードごとに次の手順に従って、ポッド内のノードをデコミッションします。

a) **[ファブリック（Fabric）] > [インベントリ（Inventory）]** に移動し、**Pod** を展開します。

b) スイッチを選択して右クリックし、**[コントローラから削除（Remove from Controller）]** を選択します。

c) アクションを確認し、**[OK]** をクリックします。

プロセスにはおよそ 10 分ほどかかります。ノードは自動的にワイプされ、リロードされます。さらに、ノード構成がコントローラから削除されます。

d) 廃止されたノードにポート プロファイル機能が展開されている場合、一部のポート構成は残りの構成とともに削除されません。ポートをデフォルト状態に戻すには、デコミッション後に手動で構成を削除する必要があります。これを行うにはスイッチにログインし、**setup-clean-config.sh** スクリプトを実行し、実行されるまで待ちます。それから、**リロード** コマンドを入力します。

ステップ 2 すべてのスイッチがポッドから廃止されたら、それらがすべて物理的に接続され、目的の構成で起動されていることを確認します。

ステップ 3 次のアクションを実行して、各ノードを再稼働させます。

（注） ポート プロファイルが構成されたノードを新しいノードとして再コミッショ二ングさせる前に、**setup-clean-config.sh** スクリプトを実行して、ポート設定をデフォルト構成に復元する必要があります。

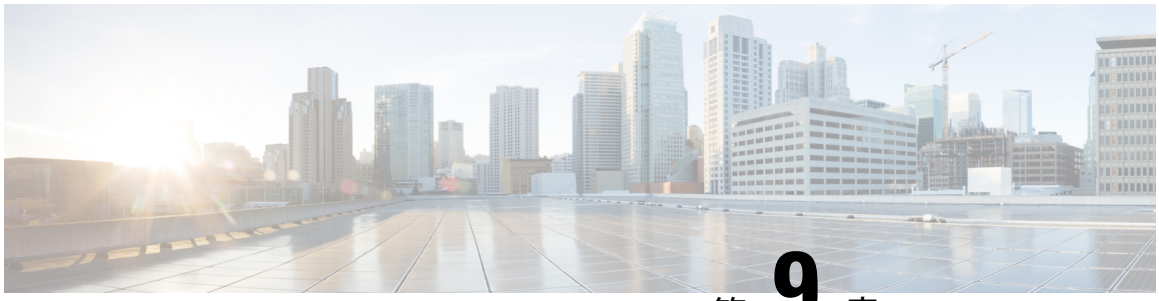
スイッチのデコミッションおよび再コミッション

a) **[ファブリック（Fabric）]**>**[インベントリ（Inventory）]**に移動し、**[クイック スタート（Quick Start）]**を展開し、**[ノードまたはポッドのセットアップ（Node or Pod Setup）** をクリックします。
b) **[セットアップ ノード（Setup Node）]** をクリックします。
c) **[ポッド ID（Pod ID）]** フィールドで、ポッド ID を選択します。
d) [+] をクリックして、**[ノード（Nodes）]** テーブルを開きます。
e) スイッチのノード ID、シリアル番号、スイッチ名、TEP プール ID、およびロール（**リーフ**または**スパイン**）を入力します。
f) [Update] をクリックします。

ステップ 4 **[ファブリック（Fabric）]** > **[インベントリ（Inventory）]** > **[ファブリック メンバーシップ（ Fabric Membership）]** に移動して、ノードがすべて設定されていることを確認します。

次のタスク

ポッドがマルチポッドトポロジ内のポッドの 1 つである場合は、このポッドとノード用にマルチポッドを再構成します。詳細については、『*Cisco APIC Layer 3 Networking* 構成ガイド』「マルチポッド」を参照してください。



第 9 章

エンドポイント接続の問題のトラブルシューティング手順

この章では、Cisco APIC ツールを使用してエンドポイント接続の問題をトラブルシューティングする手順を示し、エンドポイントとトンネルインターフェイスの動作ステータスを検査する手順が含まれており、SFP モジュールを接続する方法について説明します。

この章は、次の項で構成されています。

エンドポイント接続のトラブルシューティング

ステップ 1 各エンドポイントの動作ステータスを調べます。

動作ステータスにはエンドポイントのエラーや設定ミスが示されます。詳細は、

エンドポイント ステータスの検査 （154 ページ） を

ステップ 2 トンネル インターフェイスのステータスを調べます。

動作ステータスにはトンネルのエラーや設定ミスが示されます。「トンネル インターフェイス ステータスの検査 （155 ページ）」を参照してください。

ステップ 3 エンドポイント グループ（EPG）間で traceroute を実行します。

トレースルートでは、スパイン ノードなどの中間ノード、およびエンドポイント間の問題が明らかになります。「エンドポイント 間での traceroute の実行 （105 ページ）」を参照してください。

ステップ 4 エンドポイントのアトミック カウンタを構成します。

アトミック カウンタは、発信元エンドポイントがパケットを送信しているか、また送信先エンドポイントがパケットを受信しているか、そして受信されたパケット数が送信されたパケット数に等しいかどうかを確認します。「アトミック カウンタの構成 （40 ページ）」を参照してください。

ステップ 5 各 EPG でコントラクトを調べます。

各 EPG でのコントラクトを調べ、EPG 間でのトラフィックの流れが許可されているかを確認します。テストとして一時的にコントラクトを開き、無制限のトラフィックを許可することができます。

ステップ 6 発信元パケットをモニタリング ノードに転送するようにスパン ポリシーを構成します。

モニタリング ノードのパケット アナライザが誤ったアドレスやプロトコルなどのパケットの問題を示します。「Cisco APIC GUI を使用したテナント SPAN セッションの設定 （64 ページ）」を参照してください。

エンドポイントおよびトンネルインターフェイス ステータスの検査

このセクションでは、エンドポイントとトンネルインターフェイスの動作ステータスを検査する方法について説明します。これらの手順を実行すると、エンドポイントとトンネルインターフェイスの障害または構成ミスを明らかにすることができます。

エンドポイント ステータスの検査

ステップ 1 メニュー バーで、[Tenants] をクリックします。

ステップ 2 サブメニュー バーで、送信元エンドポイントを含むテナントをクリックします。

ステップ 3 **[ナビゲーション（Navigation）]** ペインでテナントを拡張し、**[アプリケーション プロファイル（Application Profiles）]** を拡張して、エンドポイントが含まれるアプリケーション プロファイルを拡張します。

ステップ 4 **[アプリケーション EPG（Application EPGs）]** を展開し、確認する EPG をクリックします。

ステップ 5 **[作業（Work）]** ペインで、**[エンドポイント（Endpoint）]** テーブルのエンドポイントのリストから送信元エンドポイントをダブルクリックし、**[クライアント エンド ポイント（Client End Point）]** ダイアログボックスを開きます。

ステップ 6 **[クライアント エンド ポイント（Client End Point）]** ダイアログボックスで、エンドポイントのプロパティを確認し、**[操作性（Operational）]** タブをクリックします。

ステップ 7 **[操作性（Operational）]** タブで、健全性、ステータスおよび障害情報を表示します。

[ステータス（Status）] テーブルで、変更、イベント、またはエラーなどのエントリがある項目をクリックします。

ステップ 8 **[クライアント エンド ポイント（Client End Point）]** ダイアログボックスを閉じます。

ステップ 9 **[エンドポイント（Endpoint）]** テーブルでエンドポイントの **[インターフェイス（Interface）]** エントリを表示し、ノードとトンネル ID をメモに記録します。

ステップ 10 送信先エンドポイントでこの手順を繰り返します。

（注） ファブリック内の 2 つのリーフ スイッチの背後に展開された 2 つのマイクロセグメント EPG の IP アドレス間で、双方向のトラフィックが中断されることがあります。これは、マイクロセグメント EPG からベース EPG への構成変更により、IP アドレスが移行しているときに発生する可能性があります。または逆に、双方向トラフィックの実行中に 2 つの異なるリーフ スイッチで同時に発生する可能性があります。この場合、各リモート エンドポイントのポリシー タグは引き続き以前の EPG を指します。

回避策：スイッチのリモートエンドポイントを手動でクリアするか、リモートエンドポイントが期限切れになるのを待ちます。エンドポイントをクリアするには、各スイッチの CLI にログオンし、適切なオプションを指定して **clear system internal epm endpoint** コマンドを入力します。たとえば、エンドポイントが IP アドレスに基づいている場合は、**clear system internal epm endpoint key vrf** *vrf_name* **{ip | ipv6}** *ip-address* と入力します。その後、エンドポイントは正しいポリシー タグで再学習されます。

トンネル インターフェイス ステータスの検査

この手順では、トンネル インターフェイスの動作ステータスを調べる方法を示します。

ステップ 1 メニュー バーで、[Fabric] をクリックします。

ステップ 2 サブメニュー バーで、[Inventory] をクリックします。

ステップ 3 **[ナビゲーション（Navigation）]** ペインでポッドを拡張し、発信元エンドポイント インターフェイスのノード ID を拡張します。

ステップ 4 ノードの下で **[インターフェイス（Interfaces）]** を拡張し、**[トンネル インターフェイス（Tunnel Interfaces）]** を拡張して、発信元エンドポイント インターフェイス のトンネル ID をクリックします。

ステップ 5 **[作業（Work）]** ペインで、トンネル インターフェイスのプロパティを確認し、**[操作（Operational）]** タブをクリックします。

ステップ 6 **[操作性（Operational）]** タブで、健全性、ステータスおよび障害情報を表示します。

[ステータス（Status）] テーブルで、変更、イベント、またはエラーなどのエントリがある項目をクリックします。

ステップ 7 送信先エンドポイント インターフェイスでこの手順を繰り返します。

SFP モジュールの節ゾック

SFPモジュールを新しいカードに接続するときは、モジュールがカードと通信するためのリンク速度ポリシーを作成する必要があります。リンク速度ポリシーを作成するには、次のステップに従います。

ステップ 1 リンク速度を指定するインターフェイス ポリシーを作成します。

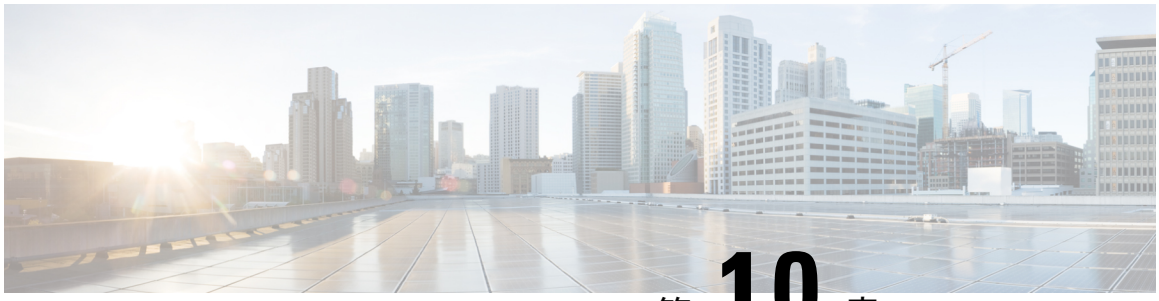
例：

```
<fabricHIfPol name="SpeedPol" speed="1G"/>
```

ステップ 2 インターフェイス ポリシー グループ内のリンク速度ポリシーを参照します。

例：

```
<infraAccPortGrp name="myGroup">
   <infraRsHIfPol tnFabricHIfPolName="SpeedPol"/>
</infraAccPortGrp>
```



第 10 章

EVPN タイプ 2 ルート アドバタイズメントのトラブルシューティング

DCIG への EVPN タイプ 2 ルート配布のトラブルシューティング

EVPN トポロジでのトラフィック転送を最適化するために、ファブリック スパインを有効にして、BGP EVPN タイプ 5（IP プレフィックス）ルートの形式のパブリック BD サブネットとともに、EVPN タイプ 2（MAC-IP）ルートを使用してホスト ルートをデータセンター インターコネクト ゲートウェイ（DCIG）に配布できます。これは、HostLeak オブジェクトを使用して有効にします。ルート配布で問題が発生した場合は、このトピックの手順を使用してトラブルシューティングを行ってください。

手順の概要

1. スパイン スイッチ CLI で次のようなコマンドを入力して、問題の VRF-AF で HostLeak オブジェクトが有効になっていることを確認します。
2. スパイン スイッチ CLI で次のようなコマンドを入力して、config-MO が BGP によって正常に処理されたことを確認します。
3. パブリック BD サブネットが EVPN タイプ 5 ルートとして DCIG にアドバタイズされていることを確認します。
4. EVPN ピアにアドバタイズされたホスト ルートが EVPN タイプ 2 MAC-IP ルートであったかどうかを確認します。
5. DCIG デバイスで次のようなコマンドを入力して、EVPN ピア（DCIG）が正しいタイプ 2 MAC-IP ルートを受信し、ホスト ルートが特定の VRF に正常にインポートされたことを確認します（DCIG が以下の例の Cisco ASR 9000 スイッチ）：

手順の詳細

ステップ 1 スパイン スイッチ CLI で次のようなコマンドを入力して、問題の VRF-AF で HostLeak オブジェクトが有効になっていることを確認します。

例：

```
spine1# ls /mit/sys/bgp/inst/dom-apple/af-ipv4-ucast/
ctrl-l2vpn-evpn  ctrl-vpnv4-ucast  hostleak  summary
```

ステップ 2 スパイン スイッチ CLI で次のようなコマンドを入力して、config-MO が BGP によって正常に処理されたことを確認します。

例：

```
spine1# show bgp process vrf apple
```

出力は次のようになります。

```
 Information for address family IPv4 Unicast in VRF apple
    Table Id                   : 0
    Table state                : UP
    Table refcount             : 3
    Peers      Active-peers    Routes     Paths      Networks   Aggregates
    0          0               0          0          0          0

    Redistribution
        None

    Wait for IGP convergence is not configured
    GOLF EVPN MAC-IP route is enabled
    EVPN network next-hop 192.41.1.1
    EVPN network route-map map_pfxleakctrl_v4
    Import route-map rtctrlmap-apple-v4
    EVPN import route-map rtctrlmap-evpn-apple-v4
```

ステップ 3 パブリック BD サブネットが EVPN タイプ 5 ルートとして DCIG にアドバタイズされていることを確認します。

例：

```
spine1# show bgp l2vpn evpn 10.6.0.0 vrf overlay-1
Route Distinguisher: 192.41.1.5:4123    (L3VNI 2097154)
BGP routing table entry for [5]:[0]:[0]:[16]:[10.6.0.0]:[0.0.0.0]/224, version 2088
Paths: (1 available, best #1)
Flags: (0x000002 00000000) on xmit-list, is not in rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: local 0x4000008c 0x0 ref 1, path is valid, is best path
  AS-Path: NONE, path locally originated
    192.41.1.1 (metric 0) from 0.0.0.0 (192.41.1.5)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 2097154
      Community: 1234:444
      Extcommunity:
          RT:1234:5101
          4BYTEAS-GENERIC:T:1234:444

  Path-id 1 advertised to peers:
    50.41.50.1
```

パス タイプ エントリで、**ref 1** は、1 つのルートが送信されたことを示します。

ステップ 4 EVPN ピアにアドバタイズされたホスト ルートが EVPN タイプ 2 MAC-IP ルートであったかどうかを確認します。

例：

```
spine1# show bgp l2vpn evpn 10.6.41.1 vrf overlay-1
Route Distinguisher: 10.10.41.2:100    (L2VNI 100)
BGP routing table entry for [2]:[0]:[2097154]:[48]:[0200.0000.0002]:[32]:[10.6.41
.1]/272, version 1146
Shared RD: 192.41.1.5:4123    (L3VNI 2097154)
Paths: (1 available, best #1)
Flags: (0x00010a 00000000) on xmit-list, is not in rib/evpn
Multipath: eBGP iBGP

  Advertised path-id 1
  Path type: local 0x4000008c 0x0 ref 0, path is valid, is best path
  AS-Path: NONE, path locally originated
  EVPN network: [5]:[0]:[0]:[16]:[10.6.0.0]:[0.0.0.0] (VRF apple)
    10.10.41.2 (metric 0) from 0.0.0.0 (192.41.1.5)
      Origin IGP, MED not set, localpref 100, weight 32768
      Received label 2097154 2097154
      Extcommunity:
          RT:1234:16777216

 Path-id 1 advertised to peers:
    50.41.50.1
```

共有 RD 行は、EVPN タイプ 2 ルートと BD サブネットによって共有される RD/VNI を示します。

EVPN ネットワーク行は、BD-Subnet の EVPN タイプ 5 ルートを示しています。

ピアにアドバタイズされたパス ID は、EVPN ピアにアドバタイズされたパスを示します。

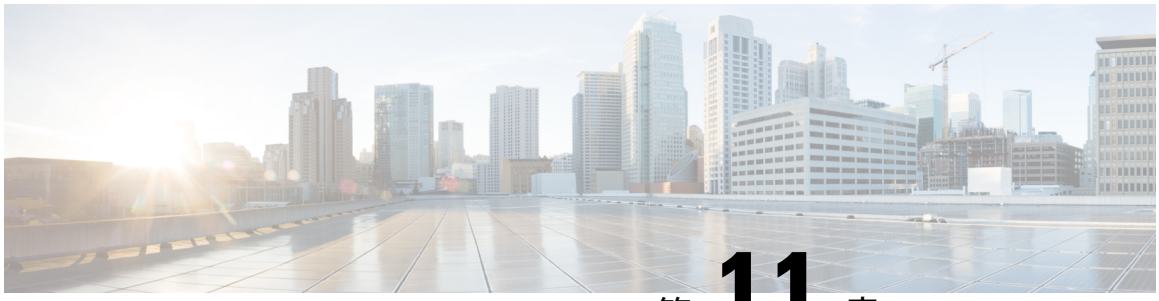
ステップ 5 DCIG デバイスで次のようなコマンドを入力して、EVPN ピア（DCIG）が正しいタイプ 2 MAC-IP ルートを受信し、ホスト ルートが特定の VRF に正常にインポートされたことを確認します（DCIG が以下の例の Cisco ASR 9000 スイッチ）：

例：

```
RP/0/RSP0/CPU0:asr9k#show bgp vrf apple-2887482362-8-1 10.6.41.1
Tue Sep  6 23:38:50.034 UTC
BGP routing table entry for 10.6.41.1/32, Route Distinguisher: 44.55.66.77:51
Versions:
  Process           bRIB/RIB  SendTblVer
  Speaker               2088        2088
Last Modified: Feb 21 08:30:36.850 for 28w2d
Paths: (1 available, best #1)
  Not advertised to any peer
  Path #1: Received by speaker 0
  Not advertised to any peer
  Local
    192.41.1.1 (metric 42) from 10.10.41.1 (192.41.1.5)
      Received Label 2097154
      Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported
      Received Path ID 0, Local Path ID 1, version 2088
      Community: 1234:444
      Extended community: 0x0204:1234:444 Encapsulation Type:8 Router
MAC:0200.c029.0101 RT:1234:5101
      RIB RNH: table_id 0xe0000190, Encap 8, VNI 2097154, MAC Address: 0200.c029.0101,
IP Address: 192.41.1.1, IP table_id 0x00000000
```

```
     Source AFI: L2VPN EVPN, Source VRF: default,
Source Route Distinguisher: 192.41.1.5:4123
```

この出力では、受信した RD、ネクスト ホップ、および属性は、タイプ 2 ルートと BD サブネットで同じです。



第 11 章

ファブリックの再構築の実行

この章では、ファブリックを再構築する方法について説明します。

ファブリックの再構築

⚠

注意 この手順は非常に混乱を招きます。既存のファブリックを取り除き、新しいファブリックを作り直します。

この手順により、ファブリックを再構築（再初期化）できます。これは、次のいずれかの理由で必要になる場合があります。

- TEP IP を変更するには
- インフラ VLAN を変更するには
- ファブリック名を変更するには
- TAC トラブルシューティング タスクを実行するには

APIC を削除すると、それらの構成が消去され、スタートアップ スクリプトでそれらが表示されます。APIC でこれを実行する順序は任意ですが、すべて（ファブリック内のすべてのリーフとスパイン）で手順を実行するようにしてください。

始める前に

以下が所定の場所に準備されていることを確認します。

- 定期的にスケジュールされた構成のバックアップ
- リーフとスパインへのコンソールアクセス
- KVM コンソール アクセスに必要な構成済みの到達可能な CIMC
- Java の問題なし

ステップ 1 現在の構成を保持したい場合は、構成のエクスポートを実行できます。詳細については、『*Cisco ACI Configuration Files：Import and Export*』の文書 https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html を参照してください。

ステップ 2 KVM コンソールに接続し、次のコマンドを入力して、APIC の設定を消去します。

a) **>acidiag touch clean**
b) **>acidiag touch setup**
c) **>acidiag reboot**

各ノードがファブリック検出モードで起動し、以前に構成されたファブリックの一部ではないことを確認します。

（注） スタートアップ スクリプトで APIC を起動しないため、 **acidiag touch** コマンドだけはこの手順では役に立ちません。

注意 以前のすべてのファブリック構成が削除されていることを確認することが非常に重要です。単一のノードに以前のファブリック構成が存在する場合でも、ファブリックを再構築することはできません。

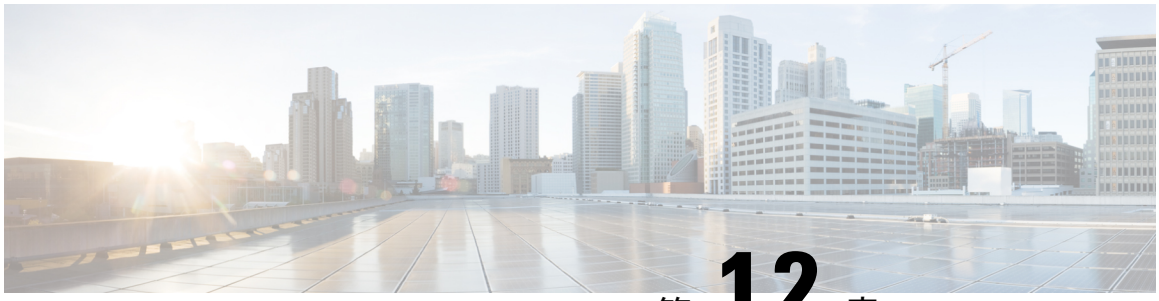
ステップ 3 以前の構成がすべて削除されたら、すべての APIC のスタートアップ スクリプトを実行します。この時点で、上記の値、TEP、TEP Vlan、および/またはファブリック名のいずれかを変更できます。これらがすべての APIC で一貫していることを確認してください。詳細については、『*Cisco APIC Getting Started Guide*』の https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html を参照してください。

ステップ 4 ファブリック ノードをクリーン リブートするには、各ファブリック ノードにログインし、次を実行します。

a) **>setup-clean-config.sh**
b) **>reload**

ステップ 5 apic1 にログインし、構成のインポートを実行します。詳細については、『*Cisco ACI Configuration Files：Import and Export*』の文書 https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html を参照してください。

ステップ 6 ファブリックが以前のファブリック登録ポリシーを使用してノード上でファブリックを再構築するようになったため、数分間待ちます。（ファブリックのサイズによっては、この作業に時間がかかる場合があります。）



第 12 章

IP bエース EPG 構成の確認

作成できるエンドポイント グループ（EPG）には、アプリケーション EPG と IP ベースの EPG の 2 種類があります。IP ベースの EPG は、マイクロセグメント EPG であるという点で通常のアプリケーション EPG とは異なります。この章では、GUI またはスイッチ コマンドを使用して、IP ベースの EPG 構成が IP ベースとして正しく分類されていることを確認する方法について説明します。

この章は、次の項で構成されています。

GUI を使用した IP ベースの EPG 構成の確認

この手順では、GUI および Visore ツールを使用して IP ベースの EPG が正しく構成されていることを確認する方法について説明します。

ステップ 1 作成した IP ベースの EPG が GUI の **uSeg EPGs** フォルダーの下に表示されていることを確認します（次のスクリーン キャプチャを参照）。
REST API を使用して作成された「IP」という名前の uSeg EPG の下にリストされている 1 つの IP ベースの EPG があることに注意してください。

ステップ 2 各 EPG IP（IP ベースの EPG）の EPG - IP プロパティ画面（右側のウィンドウ ペイン）で情報が正しいことを確認します。
画面の下部に表示される IP ベースの EPG と IP アドレスのリストに注意してください。

ステップ 3 Web ブラウザから、APIC の IP アドレスに続けて「/visore.html」を入力します。Visore は、EPG など、システム内のすべてのオブジェクトを表示できるツールです。Visore を使用して、IP ベースの EPG が正しく構成されていることを確認できます。Visore の詳細については、『アプリケーション ポリシー インフラストラクチャ コントローラ *Visore* ツールの紹介』を参照してください。

ステップ 4 ユーザー名とパスワードを入力し、**[ログイン（Login）]** をクリックします。

ステップ 5 **クラスまたは DN** の隣のフィールド（たとえば、「fvAEPg」）にクラスの名前を入力して、GUI で確認した IP ベースの EPG のクエリを実行します。

（注） これは、APIC の観点からのビューです。上記の「示されるオブジェクトの総数（Total objects shown）」が「3」であることがわかります。これは、スイッチにダウンロードされた 3 つの EPG があることを意味します。以前 GUI に「IP」としてリストされていた IP ベースの EPG が、「dn」の隣に表示されていることがわかります。また、「isAttrBasedEPg」の横に「yes」と表示されていることにも注意してください。これは、これが IP ベースの EPG として適切に構成されたことを意味します。アプリケーション EPG と IP ベースの EPG の両方を含む、すべてのオブジェクトが Visore を使用して正常に設定されていることを確認できます。

ステップ 6 スイッチ側から見た図です。スイッチで、fvEpP クラスのクエリを実行して EPG を表示し、「crtrnEnabled」属性を確認できます。IP ベースの EPG の場合は「yes」に設定されます。
この EPG の下で、EPG の子が IP アドレスとともに表示されていることを確認して、適切な構成を確保します。構成された IP アドレスごとに、スイッチがトラフィックの分類に使用する 1 つのオブジェクト（「l3IpCktEp」という名前）があります。構成が完了すると、パケットが到着すると、スイッチはこれらのオブジェクトを使用して分類します。

ステップ 7 構成したすべてのエンドポイントと IP アドレスの pcTag が一致することを確認します。すべての EPG には pcTag があります。構成した IP アドレスと一致するすべてのエンドポイントは、この pcTag に分類されます。すべてのエンドポイントには、クラスクエリを実行できる IP アドレスがあります。トラブルシューティングを行うときは、これらのエンドポイント（サーバー）がこの IP ベースの EPG に正しく分類されているかどうかを確認する必要があります。（pcTags は IP ベースの EPG に一致する必要があります。）

スイッチ コマンドを使用した IP-EPG 構成の確認

この手順では、スイッチ コマンドを使用して IP-EPG (「IpCkt」)構成定を確認する方法について説明します。

ステップ 1 リーフにログインします。

ステップ 2 /mit/sys ディレクトリに移動します。

ステップ 3 /mit/sys ディレクトリで、ctx（vrf コンテキスト ディレクトリ）を見つけます。

ステップ 4 VRF cts ディレクトリで、IpCkt が構成されている特定の BD ディレクトリに移動します。
IpCkt が表示されます。

（注） 「IpCkt」と「IP-EPG」は、このドキュメントでは同じ意味で使用されます。

ステップ 5 ディレクトリに移動すると、「猫の概要」に IpCkt に関する情報が表示されます。

ステップ 6 サマリーの「operSt」に「サポートされていない」と表示されていないことを確認してください。

ステップ 7 IpCkt が構成されている BD に対応する VLAN ID を見つけます。

（注） VLAN ID は、**show vlan internal bd-info** コマンドのいずれか、または **show system internal epm vlan all** コマンドで見つけることができます。

ステップ 8 BD の VLAN ID を見つけたら、**show system internal epm <vlan-id> detail** を発行します。

ここで、特定の sclass で構成されたすべての IpCkts を表示できるはずです。（/mit/sys ディレクトリに表示されるものと一致する必要があります。）

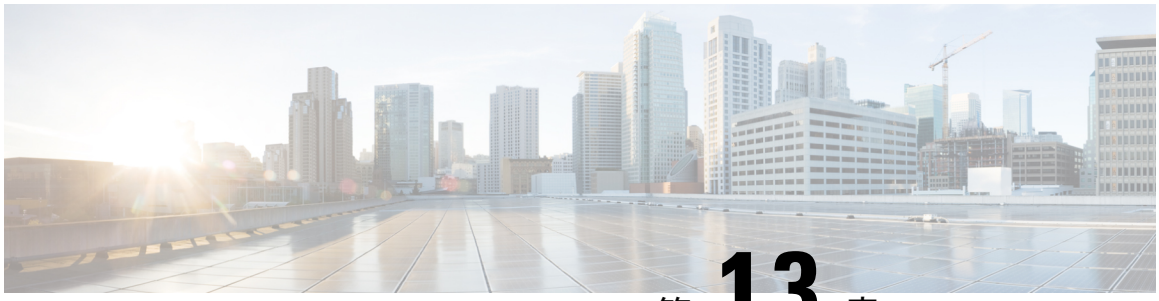
ステップ 9 vsh で実行した手順を vsh_lc に対して繰り返します。

ステップ 10 BD の IpCtk に一致する IP を使用して、**show system internal epm endp ip <a.b.c.d>** を介してトラフィックを送信します。学習した IP に「sclass」の IP フラグと特定の sclass 値があることを確認できます。

ステップ 11 vsh で実行した手順を vsh_lc に対して繰り返します。

この手順で使用するスイッチ トラブルシューティング コマンドのリスト:

```
Cd /mits/sys/ctx-vxlan…/bd-vxlan…
     - cat summary
Vsh -c “show system internal epm vlan all” or
Vsh -c “show vlan internal bd-info”
Vsh -c “show system internal epm vlan <vlan-id> detail”
Vsh -c “show system internal epm endp ip <a.b.c.d>"
Vsh_lc -c “show system internal epm vlan all” or
Vsh_lc -c “show vlan internal bd-info”
Vsh_lc -c “show system internal epm vlan <vlan-id> detail”
vsh_lc -c “show system internal epm endp ip <a.b.c.d>”
vsh_lc -c “show system internal epm epg”
```



第 13 章

切断されたリーフの復元

リーフにプッシュされた構成が原因で、リーフ上のすべてのファブリック インターフェイス（リーフをスパインに接続するインターフェイス）が無効になっている場合、リーフへの接続は永久に失われ、リーフはファブリック内で非アクティブになります。接続が失われたため、構成をリーフにプッシュしようとしても機能しません。この章では、切断されたリーフを回復する方法について説明します。

NX-OS-Style CLI を使用した切断されたリーフの復元

この手順では、Cisco Application Policy Infrastructure Controller（APIC）NX-OS スタイルの CLI を使用してファブリック インターフェイスを有効にします。REST API コールを実行できる外部ツールがない場合は、この手順を使用します。

（注） この手順では、1/31 がスパイン スイッチに接続するリーフ スイッチ ポートの 1 つであることを前提としています。

ステップ 1 Cisco APIC NX-OS-style CLI を使用して、ブロック リスト ポリシーを削除します。

例：

```
apic1# podId='1'
apic1# nodeId='103'
apic1# interface='eth1/31'
apic1# icurl -sX POST 'http://127.0.0.1:7777/api/mo/.json' -d '{"fabricRsOosPath":{"attributes":

{"dn":"uni/fabric/outofsvc/rsoosPath-[topology/pod-'$podId'/paths-'$nodeId'/pathep-['$interface']]","status":"deleted"}}}'
```

ステップ 2 リーフ スイッチまたはスパイン スイッチの CLI を使用して、サービス中のポートを設定して、リーフ スイッチのポートを起動します。

例：

```
switch1# podId='1'
switch1# nodeId='103'
switch1# interface='eth1/31'
switch1# icurl -X POST
'http://127.0.0.1:7777/api/node/mo/topology/pod-'$podId'/node-'$nodeId'/sys/action.json'
  -d
'{"actionLSubj":{"attributes":{"oDn":"sys/phys-['$interface']"},"children":[{"l1EthIfSetInServiceLTask":

  {"attributes":{"adminSt":"start"}}}]}}'
```

REST API を使用した切断されたリーフの復元

切断されたリーフ スイッチを復元するには、次のプロセスを使用して、ファブリック インターフェイスの少なくとも 1 つを有効にする必要があります。残りのインターフェイスは、GUI、REST API、または CLI を使用して有効にできます。

最初のインターフェイスを有効にするには、REST API を使用してポリシーを投稿し、投稿されたポリシーを削除し、ファブリック ポートをアウト オブ サービスにします。次のように、ポリシーをリーフ スイッチにポストして、アウト オブ サービスのポートをインサービスにすることができます。

（注） この手順では、1/49 がスパイン スイッチに接続するリーフ スイッチ ポートの 1 つであることを前提としています。

ステップ 1 REST API を使用して、Cisco APIC からブロック リスト ポリシーをクリアします。

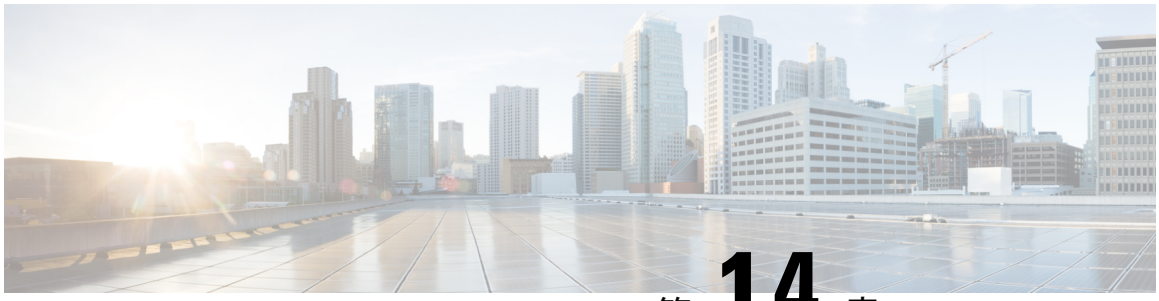
例：

```
$APIC_Address/api/policymgr/mo/.xml
<polUni>
    <fabricInst>
        <fabricOOServicePol>
            <fabricRsOosPath tDn="topology/pod-1/paths-$LEAF_Id/pathep-[eth1/49]" lc="blacklist"
status ="deleted"/>
        </fabricOOServicePol>
    </fabricInst>
</polUni>
```

ステップ 2 ローカル タスクをノード自体にポストし、**l1EthIfSetInServiceLTask** を使用して必要なインターフェイスを起動します。

例：

```
$LEAF_Address/api/node/mo/topology/pod-1/node-$LEAF_Id/sys/action.xml
<actionLSubj oDn="sys/phys-[eth1/49]">
    <l1EthIfSetInServiceLTask adminSt='start'/>
</actionLSubj>
```



第 14 章

ループバック障害のトラブルシューティング

障害の発生したライン カードの識別

このセクションでは、ループバック障害が発生したときに、障害が発生したライン カードを特定する方法について説明します。

始める前に

ファブリック ノードのオンデマンド TechSupport ポリシーを作成しておく必要があります。オンデマンド TechSupport ポリシーをまだ作成していない場合は、*Cisco APIC* ベーシック コンフィギュレーション ガイド の「GUI を使用したオンデマンド テクニカル サポート ファイルの送信」セクションを参照してください。

ステップ 1 ファブリック ノードのオンデマンド TechSupport ポリシーのログの場所ファイルを収集します。収集を開始するには：

a) メニュー バーで、[Admin] をクリックします。

b) サブメニュー バーで、[Import/Export] をクリックします。

c) **[ナビゲーション（Navigation）]** ペインで、**[ポリシーのエクスポート（Export Policies）]** を展開し、ファブリック ノードのオンデマンド TechSupport ポリシーを右クリックします。
オプションのリストが表示されます。

d) **[Tech サポートの収集（Collect Tech Supports）]** を選択します。
[Tech サポートの収集（Collect Tech Supports）] ダイアログ ボックスが表示されます。

e) **[Tech サポートの収集（Collect Tech Supports）]** ダイアログ ボックスで、**[はい（Yes）]** をクリックして、テクニカル サポート情報の収集を開始します。

ステップ 2 ファブリック ノードのオンデマンド TechSupport ポリシーのログの場所ファイルをダウンロードします。ログの場所ファイルをダウンロードするには：

a) **[作業（Work）]** ペインの [オンデマンド TechSupport ポリシー（On-Demand TechSupport policy）] ウィンドウから、**[操作性（Operational）]** タブをクリックします。
[オンデマンド TechSupport ポリシー（On-Demand TechSupport policy）] ウィンドウに、**[ログの場所（Logs Location）]** 列を含むいくつかの列とともに概要テーブルが表示されます。
b) **[ログの場所（Logs Location）]** 列の URL をクリックします。

ステップ 3 ログの場所ファイル内で、/var/sysmgr/tmp_logs/ ディレクトリに移動し、svc_ifc_techsup_nxos.tar ファイルを解凍します。

```
-bash-4.1$ tar xopf svc_ifc_techsup_nxos.tar
```

show_tech_info ディレクトリが作成されます。

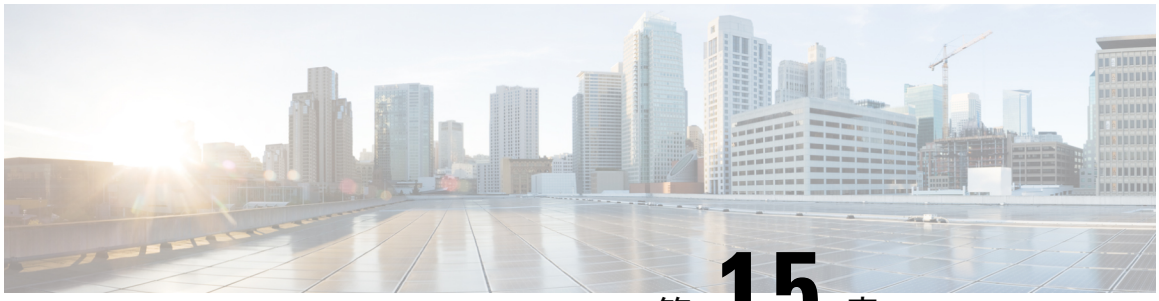
ステップ 4 **zgrep "fclc-conn failed" show-tech-sup-output.gz | less** を実行します。

```
-bash-4.1$ zgrep "fclc-conn failed" show-tech-sup-output.gz | less
[103] diag_port_lb_fail_module: Bringing down the module 25 for Loopback test failed. Packets possibly
 lost on the switch SPINE or LC fabric (fclc-conn failed)
[103] diag_port_lb_fail_module: Bringing down the module 24 for Loopback test failed. Packets possibly
 lost on the switch SPINE or LC fabric (fclc-conn failed)
```

（注） **fclc-conn failed** メッセージは、ラインカードの障害を示しています。

ステップ 5 現在障害が発生しているファブリック カードの電源を入れ直し、ファブリック カードがオンラインになることを確認します。

ステップ 6 ファブリック カードがオンラインにならない場合、またはファブリック カードが再びオフラインになった後、すぐに diag_port_lb.log ファイルを収集して、そのファイルを TAC チームに送信します。diag_port_lb.log ファイルは、ログの場所ファイルの /var/sysmgr/tmp_logs/ ディレクトリにあります。



第 15 章

PIMインターフェイスが作成されなかった理由の判別

PIM インターフェイス（pim:if）は、L3Out インターフェイス（L3Out SVI インターフェイスはサポートされていないことに注意）、マルチキャスト トンネル インターフェイス（VRF ごと）、PIM 対応のパーベイシブ BD に対応する SVI インターフェイス、および境界リーフ上のループバック インターフェイス (それぞれVRF ごと）に作成されます。

この章には、pim:ifが作成されていない場合のトラブルシューティング情報が含まれています。PIM の詳細については、『*Cisco ACI* および *Cisco ACI* のレイヤ *3* マルチキャスト』および『*Cisco* アプリケーション セントリック インフラストラクチャ基礎』ガイドを参照してください。

この章は、次の項で構成されています。

PIM インターフェイスが L3Out インターフェイス用に作成されていない

L3Out インターフェース用に PIM インターフェース（pim:If）が作成されていない場合は、以下を確認してください。

1. PIM が L3Out で有効になっています。PIM が無効になっている場合は、有効にします。
2. コンテナ L3Out で PIM が有効になっている場合は、マルチキャスト l3ext:InstP がプレフィックス名として「__int_」で作成されていることを確認します。このマルチキャスト l3ext:InstP は、L3Out PIM ポリシーをスイッチに展開するために使用されます。L3Out ごとに 1 つのマルチキャスト l3ext:InstP が必要です。

（注）
- マルチキャスト l3ext:InstP が IFC に存在する場合、対応する fv:RtdEpP が作成され、その L3Out にインターフェイスがある各スイッチに展開されているかどうかを確認できます。
- PIM の L3Out SVI インターフェイスはサポートしていません。

PIM インターフェイスがマルチキャスト トンネル インターフェイス用に作成されていない

マルチキャストトンネルインターフェイス（tunnel:If）に対してPIMインターフェース（pim:if）が作成されていない場合は、以下を確認してください。

1. 対応するトンネル:If が作成されました。

（注） tunnel:If のタイプは「underlay-mcast」である必要があります。

2. 各 mcast 対応 VRF は、mcast トンネルを作成しています。
3. tunnel:If の宛先 IP フィールドには、有効な GIPO アドレスが入力されています。
4. tunnel:If に有効な GIPO アドレスが入力されていない場合は、IFC の pim:CtxP とスイッチの pim:CtxDef をチェックして、GIPO が正しく割り当てられていることを確認します。
5. トンネルの送信元 IP:If には、BL の場合は L3Out のループバック アドレス、NBL の場合は「127.0.0.100」があります。

PIM インターフェイスがマルチキャスト対応ブリッジ ドメインに作成されない

マルチキャスト対応のブリッジ ドメイン（BD）に対して PIM インターフェイス（pim:if）が作成されていない場合は、次のことを確認します。

1. 対応する BD または対応する Ctx で PIM が有効になっています。
2. 対応する BD が普及しています。
3. 普及している BD ベースの pim:If は、デフォルトのパラメータを受け取ります。

（注） igmp snooping との相互作用については、普及 BD で PIM が有効になっている場合、対応する igmpsnoop:If に対してルーティング ビットが自動的に有効になっている必要があります。

第 16 章

ポート セキュリティのインストール

この章では、Visore を使用して APIC およびリーフ スイッチでポート セキュリティのインストールを確認する方法と、Cisco NX-OS スタイルの CLI を使用してハードウェアでポート セキュリティがプログラムされていることを確認する方法について説明します。ポートセキュリティの構成については、「*Cisco* ポート セキュリティ」のドキュメントを参照してください。

この章は、次の項で構成されています。

Visore を使用したポート セキュリティのインストールの確認

ステップ 1 Cisco APIC で、Visore の l2PortSecurityPol クラスのクエリを実行して、ポート セキュリティ ポリシーのインストールを確認します。

ステップ 2 リーフ スイッチで、Visore で l2PortSecurityPolDef のクエリを実行して、具体的なオブジェクトがインターフェイスに存在することを確認します。
ポート セキュリティが Cisco APIC およびリーフ スイッチにインストールされていることを確認したら、Cisco NX-OS CLI を使用して、ポート セキュリティがハードウェアにプログラムされていることを確認します。

Cisco NX-OS CLI を使用したハードウェア ポート セキュリティ設置の確認

ステップ 1 次のように、スイッチ インターフェイスのポート セキュリティ ステータスを表示します。

例：

```
switch# show system internal epm interface ethernet 1/35 det
name : Ethernet1/35 ::: if index : 0x1a022000 ::: state : UP
vPC : No ::: EPT : 0x0
MAC Limit : 8 ::: Learn Disable : No ::: PortSecurity Action : Protect
VLANs : 4-23
Endpoint count : 5
Active Endpoint count : 5

switch# show system internal epm interface port-channel 1 det

name : port-channel1 ::: if index : 0x16000000 ::: state : UP
vPC : No ::: EPT : 0x0
MAC Limit : 6 ::: Learn Disable : No ::: PortSecurity Action : Protect
VLANs :
Endpoint count : 0
Active Endpoint count : 0
Number of member ports : 1
Interface : Ethernet1/34    /0x1a021000
::::
```

ステップ 2 次のように、モジュール インターフェイスのポート セキュリティ ステータスを表示します。

例：

```
module-1# show system internal epmc interface ethernet 1/35 det
if index : 0x1a022000 ::: name : Ethernet1/35 ::: tun_ip = 0.0.0.0
MAC limit : 8 ::: is_learn_disable : No ::: MAC limit action: Protect
pc if index : 0 ::: name :
is_vpc_fc FALSE  ::: num_mem_ports : 0
 interface state : up
Endpoint count : 5
EPT : 0


module-1# show system internal epmc interface port-channel 1 det
if index : 0x16000000 ::: name : port-channel1 ::: tun_ip = 0.0.0.0
MAC limit :  6 ::: is_learn_disable : No ::: MAC limit action: Protect
pc if index : 0 ::: name :
is_vpc_fc FALSE  ::: num_mem_ports : 1
 interface state : up
Endpoint count : 0
EPT : 0
::::
```

ステップ 3 次のように、リーフ スイッチのポート セキュリティ ステータスを表示します。

例：

```
swtb15-leaf2# show system internal epm interface ethernet 1/35 det

name : Ethernet1/35 ::: if index : 0x1a022000 ::: state : UP
vPC : No ::: EPT : 0x0
MAC Limit : 5 ::: Learn Disable : Yes ::: PortSecurity Action : Protect
VLANs : 4-23
Endpoint count : 5
Active Endpoint count : 5
::::
```

ステップ 4 モジュール インターフェイスの MAC 制限を次のように確認します。

例：

```
module-1# show system internal eltmc info interface port-channel1 | grep mac_limit
   mac_limit_reached:              0   :::       mac_limit:              8
port_sec_feature_set:              1   ::: mac_limit_action:              1
```

例：

```
module-1# show system internal eltmc info interface ethernet 1/35 | grep mac_limit
   mac_limit_reached:              0   :::       mac_limit:              8
port_sec_feature_set:              1   ::: mac_limit_action:              1
```

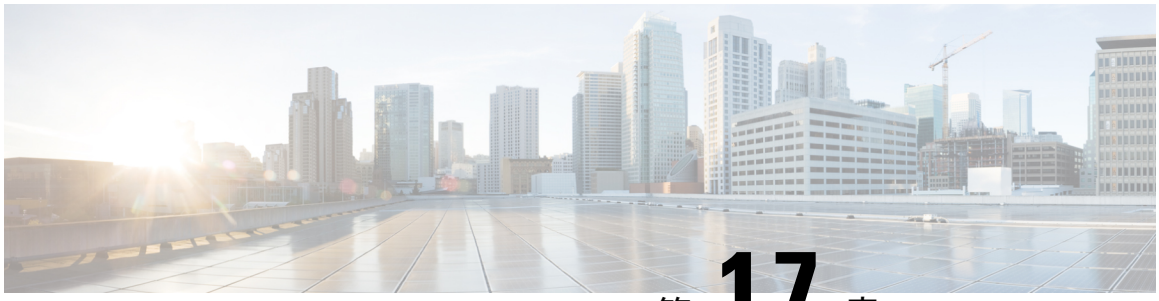
ステップ 5 モジュールのポート セキュリティ ステータスを表示し、次のように MAC 制限を確認します。

例：

```
module-1#  show system internal epmc interface ethernet 1/35 det
if index : 0x1a022000 ::: name : Ethernet1/35 ::: tun_ip = 0.0.0.0
MAC limit : 5 ::: is_learn_disable : Yes ::: MAC limit action: Protect
pc if index : 0 ::: name :
is_vpc_fc FALSE  ::: num_mem_ports : 0
 interface state : up
Endpoint count : 5
EPT : 0
::::
```

例：

```
module-1# show system internal eltmc info interface ethernet 1/35 | grep mac_limit
   mac_limit_reached:              1   :::       mac_limit:              5
port_sec_feature_set:              1   ::: mac_limit_action:              1
module-1# exit
```



第 17 章

QoS ポリシーのトラブルシューティング

このセクションでは、QoS ポリシーをトラブルシューティングするためのソリューションを提供します。

Cisco APIC QoS ポリシーのトラブルシューティング

次のセクションは、Cisco APIC QoS の一般的なトラブルシューティング シナリオをまとめたものです。

構成された QoS ポリシーを更新できません

1. 次の API を呼び出して、`qospDscpRule` がリーフに存在することを確認します。

   ```
   GET https://192.0.20.123/api/node/class/qospDscpRule.xml
   ```

2. QoS ルールが正確に構成され、ポリシーが接続されている EPG ID に関連付けられていることを確認してください。

   次の NX-OS スタイルの CLI コマンドを使用して、構成を確認します。

   ```
   leaf1# show vlan
   leaf1# show system internal aclqos qos policy detail

   apic1# show running-config tenant tenant-name policy-map type qos
   custom-qos-policy-name
   apic1# show running-config tenant tenant-name application application-name
    epg epg-name
   ```

CLI を使用して QoS インターフェイス統計情報を表示します

CLI は [詳細（detail）] オプションを使用しない場合、QoS クラス（level1、leve2、level3、level4、level5、level6、および policy-plane）の eth1/1 の統計のみを表示します。

```
NXOS ibash cli: tor-leaf1# show queuing interface ethernet 1/1 [detail]
```

インターフェイスのコントロールプレーンおよびスパン クラスの統計情報を表示する場合は、CLI を [詳細（detail）] オプションとともに使用する必要があります。

例：ファブリック 107 show queuing インターフェイス イーサネット 1/1 詳細

```
APIC CLI:
swtb123-ifc1# fabric node_id show queuing interface ethernet 1/1
```

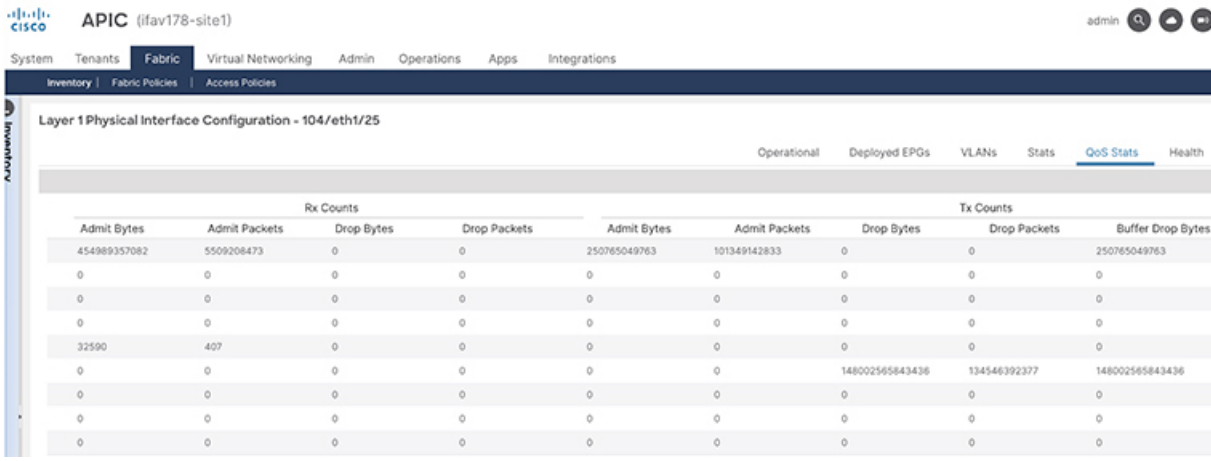
予想される出力は次のとおりです。

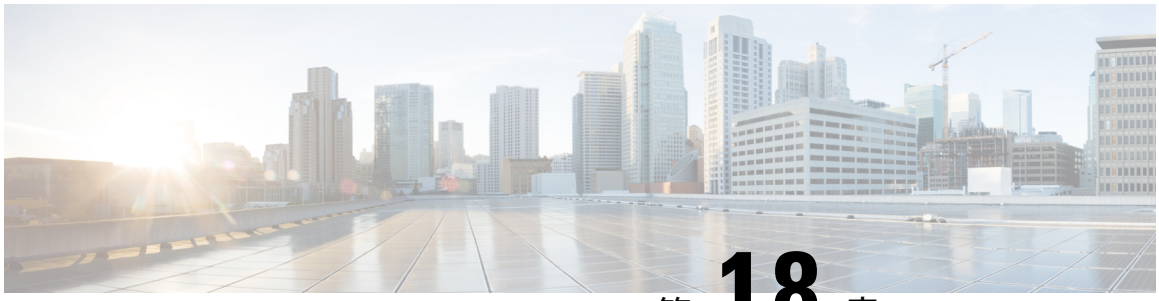
```
swtb95-leaf1# show queuing interface ethernet 1/31
====================================
Queuing stats for ethernet 1/31
====================================
Qos Class level3
====================================
Rx Admit Pkts : 0 Tx Admit Pkts : 0
Rx Admit Bytes: 0 Tx Admit Bytes: 0
Rx Drop Pkts : 0 Tx Drop Pkts : 0
Rx Drop Bytes : 0 Tx Drop Bytes : 0
====================================
Qos Class level2
====================================
Rx Admit Pkts : 0 Tx Admit Pkts : 0
Rx Admit Bytes: 0 Tx Admit Bytes: 0
Rx Drop Pkts : 0 Tx Drop Pkts : 0
Rx Drop Bytes : 0 Tx Drop Bytes : 0
====================================
Qos Class level1
====================================
Rx Admit Pkts : 0 Tx Admit Pkts : 0
Rx Admit Bytes: 0 Tx Admit Bytes: 0
Rx Drop Pkts : 0 Tx Drop Pkts : 0
Rx Drop Bytes : 0 Tx Drop Bytes : 0
====================================
Qos Class level6
====================================
Rx Admit Pkts : 0 Tx Admit Pkts : 401309848
Rx Admit Bytes: 0 Tx Admit Bytes: 47354562064
Rx Drop Pkts : 0 Tx Drop Pkts : 2066740320
Rx Drop Bytes : 0 Tx Drop Bytes : 140538341760
```

APIC GUI を使用して QoS インターフェイス統計情報を表示します

APIC GUI を使用して、QoS 統計を表示します。

[ファブリック（Fabric）]->[インベントリ（Inventory）]>[ポッド番号（Pod Number）] > [ノード ホスト名（Node Hostname）] > 物理インターフェイス（**[Physical Interfaces**）**] > [インターフェイス（Interface）] -> [QoS 統計情報（QoS Stats）]** に移動して、QoS 統計を表示します。


APIC (ifav178-site1)
admin
System
Tenants
Fabric
Virtual Networking
Admin
Operations
Apps
Integrations
Inventory | Fabric Policies | Access Policies
Inventory
Layer 1 Physical Interface Configuration - 104/eth1/25
Operational
Deployed EPGs
VLANs
Stats
QoS Stats
Health
Rx Counts
Tx Counts
Admit Bytes
Admit Packets
Drop Bytes
Drop Packets
Admit Bytes
Admit Packets
Drop Bytes
Drop Packets
Buffer Drop Bytes
454989357082 5509208473 0 0 250765049763 101349142833 0 0 250765049763
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
32590 407 0 0 0 0 0 0 0
0 0 0 0 0 0 148002565843436 134546392377 148002565843436
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0



第 18 章

サポートされている SSL 暗号の決定

この章では、サポートされている SSL 暗号を決定する方法について説明します。

SSL 暗号について

Cisco Application Centric Infrastructure（ACI）Representational State Transfer（REST）アプリケーションプログラミングインターフェイス（API）は、ソリューションがデビューした日から、HTTPS/SSL/TLS サポートがますます厳しくなる最近のバージョンへと進化を遂げました。このドキュメントは、Cisco ACI REST API での HTTPS、SSL、および TLS サポートの進化について説明し、クライアントが REST API を安全に利用するために必要なものに関するガイドを顧客に提供することを目的としています。

HTTPS は、Secure Socket Layers（SSL）または Transport Layer Security（TLS）のいずれかを利用して、HTTP セッションの安全な接続を形成するプロトコルです。SSL または TLS は、クライアントと HTTP サーバ間のトラフィックを暗号化するために使用されます。さらに、HTTPS をサポートするサーバーには、サーバーの信頼性を検証するためにクライアントが通常使用できる証明書があります。これは、サーバーで認証するクライアントの反対です。この場合、サーバーは「私は server_xyz です。それを証明する証明書はここにあります」と言っています。その後、クライアントはその証明書を利用して、サーバーが「server_xyz」であることを確認できます。

SSL/TLS には、SSL または TLS プロトコルの固有のセキュリティだけでなく、各プロトコルで使用可能なサポートされている暗号化方式も関係する、他の重要な側面があります。SSL は、SSLv1、SSLv2、SSLv3 の 3 回の反復を経て、現在ではすべて安全ではないと見なされています。TLS は、TLSv1、TLSv1.1、および TLSv1.2 の 3 つの反復を経ており、そのうち TLSv1.1 と TLSv1.2 のみが「安全」と見なされています。理想的には、クライアントは利用可能な最高の TLS バージョンを利用し、サーバーは TLSv1.1 と TLSv1.2 のみをサポートする必要があります。ただし、ほとんどのサーバは、古いクライアントに対して TLSv1 を保持する必要があります。

ほぼすべての最新のブラウザで、TLSv1.1 と TLSv1.2 の両方をサポートしています。ただし、HTTPS を使用するクライアントはブラウザではない場合があります。クライアントは、Web

サーバーと通信し、HTTPS/TLS をネゴシエートする必要がある Java アプリケーションまたは Python スクリプトである場合があります。このような状況では、何をどこでサポートするかという問題がより重要になります。

CLI を使用してサポートされている SSL 暗号を判別する

始める前に

このセクションでは、CLI を使用して、サポートされている SSL 暗号を判別する方法について説明します。

ステップ 1 次に示されているように、OpenSSL 環境でサポートされている暗号を取得します。

例：

```
openssl ciphers 'ALL:eNULL'
```

ステップ 2 次に示されているように、sed またはその他のツールを使用して暗号を分離します。

例：

```
openssl ciphers 'ALL:eNULL' |  sed -e 's/:/\n/g'
```

ステップ 3 次のように、暗号をループし、APIC をポーリングして、サポートされている暗号を確認します。

例：

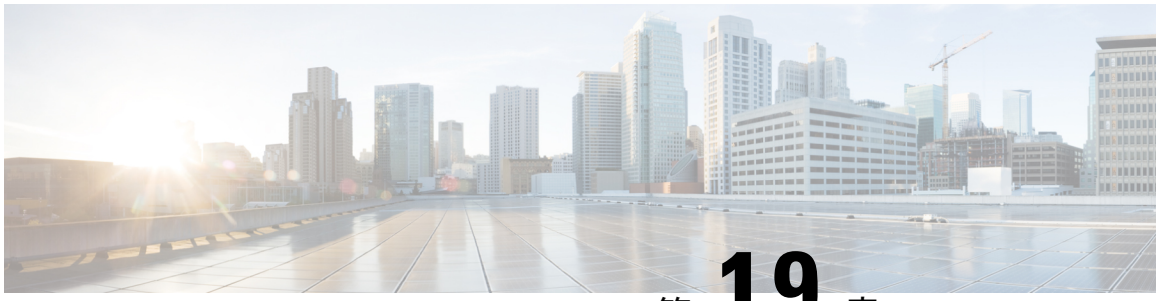
```
openssl s_client -cipher '<some cipher to test>' -connect <apic ipaddress>:<ssl port, usually 443>
```

次の暗号の例を参照してください。

例：

```
openssl s_client -cipher 'ECDHE-ECDSA-AES128-GCM-SHA256' -connect 10.1.1.14:443
```

（注）　応答に CONNECTED が含まれている場合、その暗号はサポートされています。



第 19 章

不要な _ui_ オブジェクトの削除

注意 APIC の基本 GUI を使用して行われた変更を拡張 GUI で表示することはできますが、変更を加えることはできません。また、拡張 GUI で行われた変更を基本 GUI で表示することはできません。基本 GUI と NX-OS スタイルの CLI は常に同期されるため、NX-OS スタイルの CLI から行った変更は基本 GUI に表示され、基本 GUI で行った変更は NX-OS スタイルの CLI に表示されます。ただし拡張 GUI と NX-OS スタイルの CLI の間ではこのような同期が行われません。次の例を参照してください。

- 基本 GUI モードと拡張 GUI モードを混在させないでください。拡張モードを使用して 2 つのポートにインターフェイスポリシーを適用し、次に基本モードを使用していずれかのポートの設定を変更すると、変更内容が両方のポートに適用される可能性があります。
- APIC でインターフェイスごとの設定を行う際に、拡張 GUI と CLI を混在させないでください。GUI で行われた設定が、NX-OS CLI では部分的にしか機能しない可能性があります。

  たとえば、GUI の **[Tenants]** > ***[tenant-name]*** > **[Application Profiles]** > ***[application-profile-name]*** > **[Application EPGs]** > ***[EPG-name]*** > **[Static Ports]** > **[Deploy Static EPG on PC, VPC, or Interface]** でスイッチ ポートを設定したと仮定します。

  次に NX-OS スタイルの CLI で show running-config コマンドを使用すると、以下のような出力を受信します。

  ```
  leaf 102
  interface ethernet 1/15
  switchport trunk allowed vlan 201 tenant t1 application ap1 epg ep1
   exit
   exit
  ```

  NX-OS スタイルの CLI でこれらのコマンドを使用してスタティック ポートを設定すると、次のエラーが発生します。

  ```
  apic1(config)# leaf 102
  apic1(config-leaf)# interface ethernet 1/15
  apic1(config-leaf-if)# switchport trunk allowed vlan 201 tenant t1 application ap1
  epg ep1
  No vlan-domain associated to node 102 interface ethernet1/15 encap vlan-201
  ```

  これは、CLI に APIC GUI では実行されない検証があることが原因です。show running-config コマンドによって出力されたコマンドが NX-OS CLI で機能するためには、VLAN ドメインが事前に設定されている必要があります。設定の順序は GUI に適用されません。
- 拡張 GUI を使用する前に、基本 GUI または NX-OS CLI によって変更を加えないでください。変更を加えてしまうと、名前の先頭に _ui_ が付加されたオブジェクトが意図せず作成される場合があります。このオブジェクトは拡張 GUI で変更または削除できません。

高度な GUI を使用する前に、基本 GUI または NX-OS CLI を変更する場合、これは意図せずにオブジェクトが作成され（名前に_ui_ が付加される）、高度な GUI で変更または削除できなくなる場合があります。

このようなオブジェクトを削除する手順については、REST API を使用した不要な _ui_ オブジェクトの削除 （185 ページ）を参照してください。

REST API を使用した不要な _ui_ オブジェクトの削除

Cisco APIC GUI を使用する前に Cisco NX OS スタイル CLI で変更を行い、名前の先頭に _ui_ が付加されたオブジェクトが表示された場合は、API に対して次を含む REST API 要求を実行することでこれらのオブジェクトを削除できます。

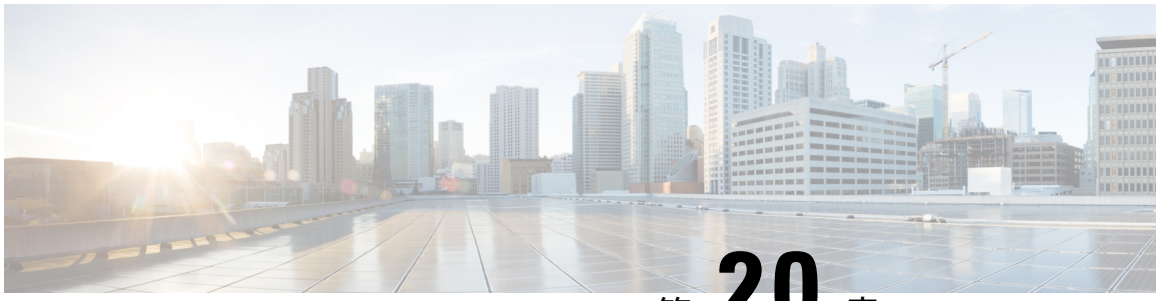
- クラス名（例：**infraAccPortGrp**）
- Dn 属性（例：**dn="uni/infra/funcprof/accportgrp-__ui_l101_eth1--31"**
- **status="deleted"** に設定したステータス属性

次の手順で API に POST を実行します。

ステップ 1 削除するオブジェクトへの書き込みアクセス権を持つユーザ アカウントにログインします。

ステップ 2 API に次の例のような POST を送信します。

```
POST https://192.168.20.123/api/mo/uni.xml
Payload:<infraAccPortGrp dn="uni/infra/funcprof/accportgrp-__ui_l101_eth1--31" status="deleted"/>
```



第 20 章

マルチポッドおよびマルチキャストの問題のトラブルシューティング

この章は、次の項で構成されています。

マルチサイトとマルチポッドのトラブルシューティング

このセクションでは、マルチサイトおよびマルチポッドをトラブルシューティングする方法を説明します。

エラー：400

次のエラーが表示される場合

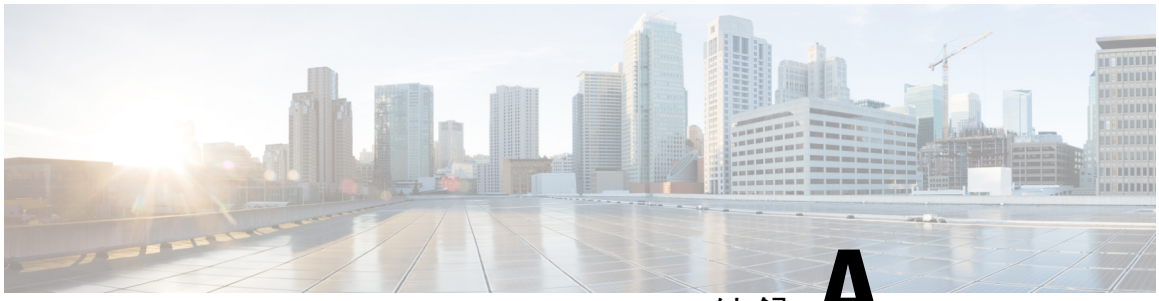
```
Error:400 - Invalid Configuration Following Intersite Spines are not configured as Mpod
 Spines: 1202
```

既存のすべてのスパインに対してファブリック外部接続を有効にする必要があります。新しいスパインを追加する場合は、**Setup Multipod** GUI ウィザードを使用します。

この問題を解決するには 2 つの方法があります。

- 外部ルーティング ネットワークの下ですべてのスパインを有効にします。
  - APIC GUI のメニュー バーで、**[テナント（Teant）]** > **[インフラ（infra）]** をクリックします。
  - **[Navigation（ナビゲーション）]** ペインで、**[ネットワーキング（Networking）]** > **[外部ルーテッド ネットワーク（External Routed Networks）]**を展開し、外部ルーテッド ネットワークを右クリックして、**[ファブリック外部接続を有効にする（Enable Fabric External Connectivity）]** を選択します。
- 外部ルーテッド ネットワークの下に新しいスパインを追加します。
  - APIC GUI のメニュー バーで、**[ファブリック（Fabric）]** をクリックします。

- **[ナビゲーション（Navigation）]** ペインで、**[クイック スタート（Quick Start）]** > **[ノードまたはポッド セットアップ（Node or Pod Setup）]** > **[マルチポッドのセットアップ（Setup Multipod）]** を展開し、マルチポッド セットアップを完了します。



付録 A

acidiag コマンド

Cisco APIC でのトラブルシューティング操作では、**acidiag** コマンドを使用します。

注意 このコマンドは、ACIの日常的な操作を目的としたものではありません。コマンドのすべての形式は、非常に混乱を招く可能性があり、適切に使用しないとネットワークに重大な問題が発生する場合があります。実行する前に、ファブリックへの完全な影響を理解してください。

クラスタ コマンド

```
acidiag

acidiag avread

acidiag fnvread

acidiag fnvreadex
```

acidiag コマンド

構文の説明

オプション	機能
avread	クラスタ内の APIC を表示します。avread の出力は次のとおりです。 • Cluster of：動作するクラスタのサイズ • out of target：必要なクラスタ サイズ • active=：APIC が到達可能かどうかを示します • health=：全体的な APIC の正常性の概要。正常性スコアが低下しているサービスを表示します。 • chassisID=：所定の APIC に対する既知のシャーシ ID。 （注） 現在クラスタにない APIC については、ピア シャーシ ID が正しくない可能性があります。
bootcurr	次回の起動時に、APIC システムは Linux パーティション内の現在の APIC イメージを起動します。このオプションは、通常は使用されません。
bootother	次回の起動時に、APIC システムは Linux パーティションの以前の APIC イメージを起動します。このオプションは、通常は使用されません。
bond0test	リーフへの APIC 接続の中断テスト。これは、シスコの内部テスト目的でのみ使用されます。それ以外では、ファブリックへの APIC 接続で問題が発生する可能性があります。
fnvread	ファブリックに登録されているスイッチ ノードのアドレスと状態を表示します。
fnvreadex	ファブリックに登録されているスイッチのノードの追加情報を表示します。
linkflap	指定された APIC インターフェイスを停止およびバックアップします。

オプション	機能
preservelogs	APIC は現在のログをアーカイブします。通常の再起動中に、これは自動的に発生します。このオプションは、ハード リブートの前に使用できます。
run	使用可能な 2 つのオプションは、iptables-list と lldptool です。iptables-list は、管理テナント コントラクトによって制御される Linux iptables を表示するために使用されます。 lldptool は、APIC によって送受信される lldp 情報を表示するために使用されます。
rvread	データ レイヤの状態を要約します。出力には、各サービスのデータ レイヤの状態の概要が表示されます。シャード ビューには、レプリカが昇順で表示されます。
acidiag rvread *service*	すべてのレプリカのすべてのシャードでのサービスのデータ レイヤの状態を表示します。 （注） 例については、例 （195 ページ）を参照してください。
acidiag rvread *service shard*	すべてのレプリカの特定のシャードでのサービスのデータ レイヤの状態を表示します。 （注） 例については、例 （195 ページ）を参照してください。
acidiag rvread *service shard replica*	特定のシャードとレプリカでのサービスのデータ レイヤの状態を表示します。 （注） 例については、例 （195 ページ）を参照してください。
validateimage	イメージをファームウェア リポジトリにロードする前に、イメージを検証できます。この関数は、リポジトリに追加されるイメージのプロセスの通常の一部として実行されることに注意してください。
validateenginxconf	APIC で生成された nginx 構成ファイルを検証して、nginx がその構成ファイルで起動できることを確認します。これは、nginx Web サーバーが APIC で実行されていない場合のデバッグでの使用を目的としています。

サービス ID

次の表にリストされているサービス ID は、**man acidiag** コマンドを入力するときにも表示されます。

表 *3:* サービス *ID*

サービス	ID
cliD	1
コントローラ	2
eventmgr	3
extXMLApi	4
ポリシー要素	5
policymgr	6
リーダー	7
AE	8
topomgr	9
observer	10
dbgr	11
observerelem	12
dbgrelem	13
vmmmgr	14
nxosmock	15
bootmgr	16
appliancedirector	17
adrelay	18 日
ospaagent	19
vleafelem	20
dhcpd	21
scripthandler	22
idmgr	23
ospaelem	24

サービス	ID
osh	25
opflexagent	26
opflexelem	27
confelem	28
vtap	29
snmpd	30
opflexp	31
分析	32
policydist	33
plghandler	34
domainmgr	35
licensemgr	36
なし	37
platformmgr	38
edmgr	39

表 *4:* データの状態

州	ID
コマトーセ	0
NEWLY_BORN	1
不明ファイル	2
DATA_LAYER_DIVERGED	11
DATA_LAYER_DEGRADED_LEADERSHIP	12
DATA_LAYER_ENTIRELY_DIVERGED	111
DATA_LAYER_PARTIALLY_DIVERGED	112
DATA_LAYER_ENTIRELY_DEGRADED_LEADERSHIP	121
DATA_LAYER_PARTIALLY_DEGRADED_LEADERSHIP	122
FULLY_FIT	255

システムのキーワード

```
acidiag [{start|stop|restart}] [{mgmt|xinetd}]

acidiag installer  -u imageurl -c

acidiag reboot

acidiag  touch [{clean|setup}]

acidiag verifyapic
```

構文の説明

オプション	機能
-c	クリーン インストールを指定します
-u	APIC イメージの URL を指定します。
imageurl	APIC イメージを指定します。
installer	APIC に新しいイメージをインストールします。-cでクリーンインストールを実行します。
mgmt	上のすべてのサービスを指定します。APIC
reboot	APICを再起動します。
restart	APIC でサービスを再起動します。
start	APIC でサービスを開始します。
stop	APIC でサービスを停止します。
touch [clean | setup]	APIC の構成をリセットします。 • **clean** オプションは、APIC ネットワーク構成（ファブリック名、IP アドレス、ログインなど）を保持しますが、すべてのポリシー データを削除します。 • **setup** オプションは、ポリシー データと APIC ネットワーク構成の両方を削除します。
verifyapic	APIC ソフトウェアのバージョンを表示します。
xinetd	ssh および telnet デーモンを制御する xinetd（拡張インターネット デーモン）サービスを指定します。6.0(2) リリース以降、telnet はサポートされていません。

診断キーワード

```
acidiag crashsuspecttracker

acidiag dbgtoken

acidiag version
```

構文の説明

オプション	機能
crashsuspecttracker	クラッシュを示すサービスまたはデータのサブセットの状態を追跡します。
dbgtoken	root パスワードの生成に使用するトークンを生成します。これは、必要な場合には、TAC と連携しながら、その指示どおりに使用してください。
version	APIC ISO ソフトウェアのバージョンを表示します。

例

次に、**acidiag** コマンドの使用例を示します。

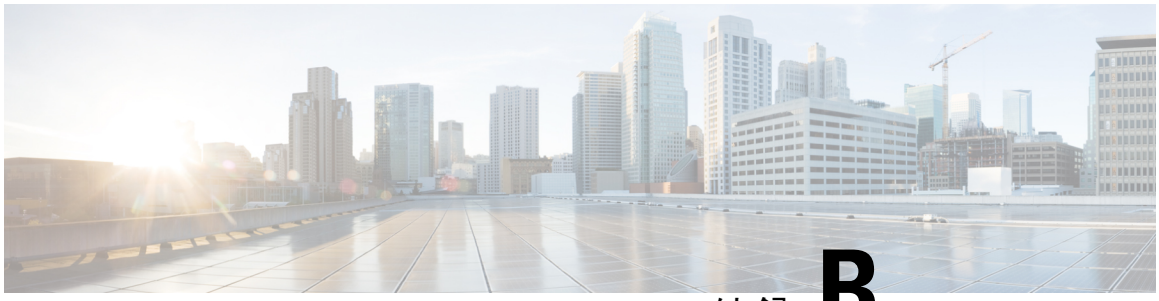
```
apic1# acidiag version 2.2.1o


apic1# acidiag verifyapic
openssl_check: certificate details
subject= CN=ABC12345678,serialNumber=PID:APIC-SERVER-L1 SN:ABC12345678
issuer= CN=Cisco Manufacturing CA,O=Cisco Systems
notBefore=Sep 28 17:17:42 2016 GMT
notAfter=Sep 28 17:27:42 2026 GMT
openssl_check: passed
ssh_check: passed
all_checks: passed


apic1# acidiag avread
Local appliance ID=1 ADDRESS=10.0.0.1 TEP ADDRESS=10.0.0.0/16 ROUTABLE IP ADDRESS=0.0.0.0
 CHASSIS_ID=1009f750-adab-11e9-a044-8dbd212cd556
Cluster of 7 lm(t):1(2019-08-08T01:02:17.961-07:00) appliances (out of targeted 7
lm(t):7(2019-08-08T03:50:57.240-07:00)) with FABRIC_DOMAIN name=ACI Fabric1 set to
version=apic-4.2(0.235j) lm(t):1(2019-08-17T01:09:16.413-07:00); discoveryMode=PERMISSIVE
 lm(t):0(1969-12-31T17:00:00.007-07:00); drrMode=OFF
lm(t):0(1969-12-31T17:00:00.007-07:00); kafkaMode=OFF
lm(t):0(1969-12-31T17:00:00.007-07:00)
        appliance id=1  address=10.0.0.1 lm(t):1(2019-08-08T01:02:08.544-07:00) tep
address=10.0.0.0/16 lm(t):1(2019-08-08T01:02:08.544-07:00) routable address=0.0.0.0
lm(t):1(zeroTime) oob address=172.23.96.10/21 lm(t):1(2019-08-08T01:02:18.218-07:00)
version=4.2(0.235j) lm(t):1(2019-08-15T15:22:00.158-07:00)
chassisId=1009f750-adab-11e9-a044-8dbd212cd556 lm(t):1(2019-08-15T15:22:00.158-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X7F lm(t):1(2019-08-17T01:13:46.997-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
```

acidiag コマンド

```
cntrlSbst=(APPROVED, FCH1748V0SZ) lm(t):1(2019-08-15T15:22:00.158-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):1(2019-08-08T01:02:08.544-07:00) commissioned=YES lm(t):1(zeroTime) registered=YES
 lm(t):1(2019-08-08T01:02:08.544-07:00) standby=NO lm(t):1(2019-08-08T01:02:08.544-07:00)
 DRR=NO lm(t):0(zeroTime) apicX=NO lm(t):1(2019-08-08T01:02:08.544-07:00) virtual=NO
lm(t):1(2019-08-08T01:02:08.544-07:00) active=YES(2019-08-08T01:02:08.544-07:00)
health=(applnc:255 lm(t):1(2019-08-17T01:39:26.296-07:00) svc's)
        appliance id=2  address=10.0.0.2 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):2(2019-07-23T17:51:38.997-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.96.11/21 lm(t):1(2019-08-18T23:14:28.720-07:00)
version=4.2(0.235j) lm(t):2(2019-08-15T15:22:00.300-07:00)
chassisId=694e6a98-adac-11e9-ad79-d1f60e3ee822 lm(t):2(2019-08-15T15:22:00.300-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X2 lm(t):2(2019-08-14T07:55:10.074-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
cntrlSbst=(APPROVED, FCH1748V0MS) lm(t):2(2019-08-15T15:22:00.300-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):2(2019-08-08T01:42:03.670-07:00) commissioned=YES
lm(t):1(2019-08-08T01:02:17.961-07:00) registered=YES
lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO lm(t):2(2019-08-08T01:42:03.670-07:00)
 DRR=NO lm(t):1(2019-08-08T01:02:17.961-07:00) apicX=NO
lm(t):2(2019-08-08T01:42:03.670-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:02:32.983-07:00) health=(applnc:255
lm(t):2(2019-08-17T01:32:51.454-07:00) svc's)
        appliance id=3  address=10.0.0.3 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):3(2019-07-23T19:05:56.405-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.96.12/21 lm(t):1(2019-08-18T23:14:28.721-07:00)
version=4.2(0.235j) lm(t):3(2019-08-15T15:21:59.893-07:00)
chassisId=1f98b916-adb7-11e9-a6f8-abe00a04e8e6 lm(t):3(2019-08-15T15:21:59.893-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X4 lm(t):3(2019-08-14T07:55:22.256-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
cntrlSbst=(APPROVED, FCH1930V1X6) lm(t):3(2019-08-15T15:21:59.893-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):3(2019-08-08T02:15:20.560-07:00) commissioned=YES
lm(t):2(2019-08-08T01:42:15.337-07:00) registered=YES
lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO lm(t):3(2019-08-08T02:15:20.560-07:00)
 DRR=NO lm(t):2(2019-08-08T01:42:15.337-07:00) apicX=NO
lm(t):3(2019-08-08T02:15:20.560-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:02:33.182-07:00) health=(applnc:255
lm(t):3(2019-08-15T16:08:46.119-07:00) svc's)
        appliance id=4  address=10.0.0.4 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):4(2019-07-23T17:46:15.545-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.97.231/21 lm(t):1(2019-08-18T23:14:28.717-07:00)
version=4.2(0.235j) lm(t):4(2019-08-15T15:22:00.669-07:00)
chassisId=3a7f38aa-adac-11e9-8869-a9e520cdc042 lm(t):4(2019-08-15T15:22:00.669-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X8 lm(t):4(2019-08-14T07:54:59.490-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
cntrlSbst=(APPROVED, FCH1902V1WW) lm(t):4(2019-08-15T15:22:00.669-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):4(2019-08-08T02:40:09.610-07:00) commissioned=YES
lm(t):3(2019-08-08T02:15:32.613-07:00) registered=YES
lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO lm(t):4(2019-08-08T02:40:09.610-07:00)
 DRR=NO lm(t):3(2019-08-08T02:15:32.613-07:00) apicX=NO
lm(t):4(2019-08-08T02:40:09.610-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-15T15:21:59.914-07:00) health=(applnc:255
lm(t):4(2019-08-17T01:39:26.477-07:00) svc's)
```

```
        appliance id=5  address=10.0.0.5 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):5(2019-07-23T19:05:11.089-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.97.232/21 lm(t):1(2019-08-18T23:14:28.723-07:00)
version=4.2(0.235j) lm(t):5(2019-08-15T15:22:00.248-07:00)
chassisId=35428666-adb7-11e9-a315-1d7671b518b3 lm(t):5(2019-08-15T15:22:00.248-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X10 lm(t):5(2019-08-14T07:55:19.573-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
cntrlSbst=(APPROVED, FCH1902V1EG) lm(t):5(2019-08-15T15:22:00.248-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):5(2019-08-08T03:03:50.338-07:00) commissioned=YES
lm(t):4(2019-08-08T02:40:15.939-07:00) registered=YES
lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO lm(t):5(2019-08-08T03:03:50.338-07:00)
 DRR=NO lm(t):4(2019-08-08T02:40:15.939-07:00) apicX=NO
lm(t):5(2019-08-08T03:03:50.338-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-15T15:21:59.756-07:00) health=(applnc:255
lm(t):5(2019-08-17T01:32:43.730-07:00) svc's)
        appliance id=6  address=10.0.0.6 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):6(2019-07-23T19:39:41.972-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.31.170.230/21 lm(t):1(2019-08-18T23:14:28.727-07:00)
version=4.2(0.235j) lm(t):6(2019-08-15T15:22:00.562-07:00)
chassisId=066c943a-adbc-11e9-bbed-257398025731 lm(t):6(2019-08-15T15:22:00.562-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X20 lm(t):6(2019-08-14T07:55:20.053-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.820-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.821-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.821-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.821-07:00)
cntrlSbst=(APPROVED, WZP22350JFT) lm(t):6(2019-08-15T15:22:00.562-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=9
lm(t):6(2019-08-08T03:28:11.246-07:00) commissioned=YES
lm(t):5(2019-08-08T03:03:57.387-07:00) registered=YES
lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO lm(t):6(2019-08-08T03:28:11.246-07:00)
 DRR=NO lm(t):5(2019-08-08T03:03:57.387-07:00) apicX=NO
lm(t):6(2019-08-08T03:28:11.246-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:30:37.663-07:00) health=(applnc:255
lm(t):6(2019-08-15T15:57:05.128-07:00) svc's)
        appliance id=7  address=10.0.0.7 lm(t):7(2019-08-08T03:50:48.149-07:00) tep
address=10.0.0.0/16 lm(t):7(2019-07-24T15:24:19.988-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.31.172.157/21 lm(t):1(2019-08-18T23:14:28.722-07:00)
version=4.2(0.235j) lm(t):7(2019-08-15T15:22:00.539-07:00)
chassisId=859be4ae-ae61-11e9-9840-7d9d67698989 lm(t):7(2019-08-15T15:22:00.539-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X40 lm(t):7(2019-08-14T07:55:23.872-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
cntrlSbst=(APPROVED, FCH2051V116) lm(t):7(2019-08-15T15:22:00.539-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=10
lm(t):7(2019-08-08T03:50:48.149-07:00) commissioned=YES
lm(t):6(2019-08-08T03:28:16.727-07:00) registered=YES
lm(t):6(2019-07-24T15:27:25.518-07:00) standby=NO lm(t):7(2019-08-08T03:50:48.149-07:00)
 DRR=NO lm(t):6(2019-08-08T03:28:16.727-07:00) apicX=NO
lm(t):7(2019-08-08T03:50:48.149-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:30:45.488-07:00) health=(applnc:255
lm(t):7(2019-08-17T01:39:26.549-07:00) svc's)
--------------------------------------------
clusterTime=<diff=2817 common=2019-08-19T15:33:55.929-07:00
local=2019-08-19T15:33:53.112-07:00 pF=<displForm=0 offsSt=0 offsVlu=-25200
lm(t):7(2019-08-08T03:50:55.925-07:00)>>
--------------------------------------------
```

```
apic1# acidiag rvread 6 3 1
(6,3,1)  st:6 lm(t):3(2014-10-16T08:48:20.238+00:00) le: reSt:LEADER voGr:0 cuTerm:0x19
 lCoTe:0x18
   lCoIn:0x1800000000001b2a veFiSt:0x31 veFiEn:0x31 lm(t):3(2014-10-16T08:48:20.120+00:00)

    lastUpdt 2014-10-16T09:07:00.214+00:00
---------------------------------------------
clusterTime=<diff=65247252 common=2014-10-16T09:07:01.837+00:00
local=2014-10-15T14:59:34.585+00:00
    pF=<displForm=0 offsSt=0 offsVlu=0 lm(t):3(2014-10-16T04:50:08.714+00:00)>>


apic1# acidiag rvread 6 3
(6,3,1)  st:6 lm(t):3(2014-10-16T08:48:20.238+00:00) le: reSt:LEADER voGr:0 cuTerm:0x19
 lCoTe:0x18
   lCoIn:0x1800000000001b2a veFiSt:0x31 veFiEn:0x31 lm(t):3(2014-10-16T08:48:20.120+00:00)

    lastUpdt 2014-10-16T09:08:30.240+00:00
(6,3,2)  st:6 lm(t):1(2014-10-16T08:47:25.323+00:00) le: reSt:FOLLOWER voGr:0 cuTerm:0x19
 lCoTe:0x18
   lCoIn:0x1800000000001b2a veFiSt:0x49 veFiEn:0x49 lm(t):1(2014-10-16T08:48:20.384+00:00)
 lp: clSt:2
   lm(t):1(2014-10-16T08:47:03.286+00:00) dbSt:2 lm(t):1(2014-10-16T08:47:02.143+00:00)
 stMmt:1
    lm(t):0(zeroTime) dbCrTs:2014-10-16T08:47:02.143+00:00 lastUpdt
2014-10-16T08:48:20.384+00:00
(6,3,3)  st:6 lm(t):2(2014-10-16T08:47:13.576+00:00) le: reSt:FOLLOWER voGr:0 cuTerm:0x19
 lCoTe:0x18
   lCoIn:0x1800000000001b2a veFiSt:0x43 veFiEn:0x43 lm(t):2(2014-10-16T08:48:20.376+00:00)

    lastUpdt 2014-10-16T09:08:30.240+00:00
---------------------------------------------
clusterTime=<diff=65247251 common=2014-10-16T09:08:30.445+00:00
local=2014-10-15T15:01:03.194+00:00
    pF=<displForm=0 offsSt=0 offsVlu=0 lm(t):3(2014-10-16T04:50:08.714+00:00)>>
```



付録 B

トラブルシューティングのためのエクスポート ポリシーの構成

エクスポートポリシーでは、エクスポートの統計情報、技術サポート収集、障害、イベントをエクスポートし、ファブリックから外部ホストにコア ファイルとデバッグ データを処理できます（APIC およびスイッチ）。

ファイルのエクスポートについて

管理者は、APIC 内で、コア ファイルとデバッグ データを処理するために、統計情報、テクニカル サポートの収集、障害およびイベントをファブリック (APIC およびスイッチ) から外部ホストにエクスポートするようエクスポート ポリシーを設定できます。エクスポートは XML、JSON、Web ソケット、Secure Copy Protocol (SCP) 、HTTP などのさまざまな形式にできます。ストリーミング、定期的、またはオンデマンドの各形式でエクスポートを登録できます。

管理者は、転送プロトコル、圧縮アルゴリズム、転送の頻度などポリシーの詳細を設定できます。ポリシーは、AAA を使用して認証されたユーザによって設定できます。実際の転送のセキュリティ メカニズムは、ユーザ名とパスワードに基づいています。内部的に、ポリシー要素はデータのトリガーを処理します。

ファイルのエクスポートに関するガイドラインと制約事項

- HTTP エクスポートとストリーミング API 形式は、統計情報の場合にのみサポートされます。コアおよびテクニカル サポート データはサポートされていません。
- エクスポートされるファイルの宛先 IP アドレスは、IPv6 アドレスであってはなりません。

- 5つを超えるノードからのテクニカル サポートを同時にトリガーしないでください。特にCisco Application Policy Infrastructure Controller（APIC）にエクスポートする場合、または帯域幅とコンピューティング リソースが不十分な外部サーバにエクスポートする場合は、トリガーを実行しないでください。
- ファブリック内のすべてのノードからテクニカル サポートを定期的に収集するには、複数のポリシーを作成する必要があります。各ポリシーは、ノードのサブセットをカバーする必要があり、時間をずらしてトリガーされるようにスケジュールします（少なくとも 30 分離す）。
- Cisco APIC の同じノードに対して複数のテクニカル サポート ポリシーをスケジュールしないでください。同じノードで複数のテクニカル サポート ポリシーのインスタンスを同時に実行すると、Cisco APIC が大量に消費されたり、 CPU サイクルやその他のリソースが切り替えられたりする可能性があります。
- メンテナンス モードになっているノードについては、オンデマンド テクニカル サポート ポリシーではなく、通常のテクニカル サポート ポリシーを使用することをお勧めします。
- メンテナンス モードのノードに対する進行中のテクニカル サポートのステータスは、Cisco APIC GUI の **[管理（Admin）]** > **[テクニカル サポート（Tech Support）]** > ***[policy_name]*** > **[操作（Operational）]** > **[ステータス（Status）]** セクションでは使用できません。テクニカル サポート ポリシーの **[コントローラへのエクスポート（Export to Controller）]** または **[エクスポート先（Export Destination）]** に基づいて、コントローラ（`/data/techsupport`）または宛先サーバを確認し、テクニカル サポートがキャプチャされていることを確認できます。
- Cisco APIC からのテクニカル サポートの収集は、リーフ スイッチ上のコアがビジー状態の場合にはタイムアウトすることがあります。BGP などのルーティング プロセスや HAL などのプラットフォーム プロセスが CPU を占有すると、コアがビジーになる可能性があります。テクニカル サポートの収集がタイムアウトした場合は、CPU 使用率を調べて、CPU 占有が発生しているかどうかを確認します。そのような場合には、リーフ スイッチのテクニカル サポートを直接収集すれば、タイムアウトの問題を回避できます。

バックアップのリモート ロケーションの構成

GUI を使用したリモート ロケーションの設定

この手順では、APIC GUI を使用してリモート ロケーションを作成する方法について説明します。

ステップ 1 メニュー バーで、[ADMIN] > [Import/Export] の順に選択します。

ステップ 2 ナビゲーション ペインで、[Remote Locations] を右クリックして [Create Remote Location] を選択します。[Create Remote Location] ダイアログが表示されます。

ステップ 3 [Create Remote Location] ダイアログのフィールドに適切な値を入力します。

（注） フィールドの説明については、[i]アイコンをクリックするとヘルプファイルが表示されます。

ステップ 4 [Create Remote Location] ダイアログのフィールドに値を入力したら、[Submit] をクリックします。
これで、データをバックアップするためのリモート ロケーションが作成されました。

REST API を使用したリモート ロケーションの設定

この手順では、REST API を使用してリモート ロケーションを作成する方法について説明します。

```
<fileRemotePath name="local" host="host or ip" protocol="ftp|scp|sftp" remotePath="path
 to folder" userName="uname" userPasswd="pwd" />
```

NX-OS スタイルの CLI を使用したリモート ロケーションの設定

ACI ファブリックでは、techsupport またはコンフィギュレーション ファイルをエクスポートする 1 つ以上のリモート宛先を設定できます。

手順の概要

1. **configure**
2. [**no**] **remote path** *remote-path-name*
3. **user** *username*
4. **path** {**ftp** | **scp** | **sftp**} *host*[*:port*] [**remote-directory**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `apic1# configure`	グローバル コンフィギュレーション モードを開始します。
ステップ 2	[**no**] **remote path** *remote-path-name* 例： `apic1(config)# remote path myFiles`	リモート パスのコンフィギュレーション モードを開始します。
ステップ 3	**user** *username* 例： `apic1(config-remote)# user admin5`	リモート サーバにログインするユーザ名を設定します。パスワードを入力するように求められます。
ステップ 4	**path** {**ftp** | **scp** | **sftp**} *host*[*:port*] [**remote-directory**] 例： `apic1(config-remote)# path sftp filehost.example.com:21 remote-directory /reports/apic`	リモート サーバへのパスとプロトコルを設定します。パスワードを入力するように求められます。

例

次に、ファイルをエクスポートするためにリモート パスを設定する例を示します。

```
apic1# configure
apic1(config)# remote path myFiles
apic1(config-remote)# user admin5
You must reset the password when modifying the path:
Password:
Retype password:
apic1(config-remote)# path sftp filehost.example.com:21 remote-directory /reports/apic
You must reset the password when modifying the path:
Password:
Retype password:
```

オンデマンド テクニカルサポート ファイルの送信

GUI を使用したオンデマンド テクニカル サポート ファイルの送信

ステップ 1 メニュー バーで、[Admin] をクリックします。

ステップ 2 サブメニュー バーで、[Import/Export] をクリックします。

ステップ 3 [Navigation] ペインで、[Export Policies] を展開します。

ステップ 4 **[オンデマンド テクニカル サポート（On-demand Tech Support）]** を右クリックし、**[オンデマンド テクニカル サポートの作成（Create On-demand Tech Support）]** を選択します。

[オンデマンド テクニカル サポートの作成（Create On-demand Tech Support）] ダイアログ ボックスが表示されます。

ステップ 5 **[オンデマンド テクニカル サポートの作成（Create On-demand Tech Support）]** ダイアログ ボックスのフィールドに適切な値を入力します。

（注） フィールドの説明については、**[オンデマンド テクニカル サポートの作成（Create On-demand Tech Support）]** ダイアログ ボックスのヘルプアイコンをクリックします。ヘルプ ファイルが開いてプロパティの説明ページが表示されます。

ステップ 6 **[送信（Submit）]** をクリックし、テクニカル サポート ファイルを送信します。

（注） オンデマンドのテクニカルサポート ファイルは別の APIC に保存し、ストレージと CPU 条件のバランスを取ることができます。場所を確認するには、**[ナビゲーション（Navigation）]** ペインでオンデマンドのテクニカル サポート ポリシーをクリックし、**[作業（Work）]** ペインで **[操作（OPERATIONAL）]** タブをクリックします。コントローラが [EXPORT LOCATION] フィールドに表示されます。

ステップ 7 ポリシー名を右クリックし、[Collect Tech Support] を選択します。

ステップ 8 [Yes] を選択して、テクニカル サポート情報の収集を開始します。

REST API を使用したオンデマンド テクニカル サポート ファイルの送信

ステップ 1 REST API を使用して次の例のような XML を POST 送信し、テクニカル サポート ファイルのリモート宛先を設定します。

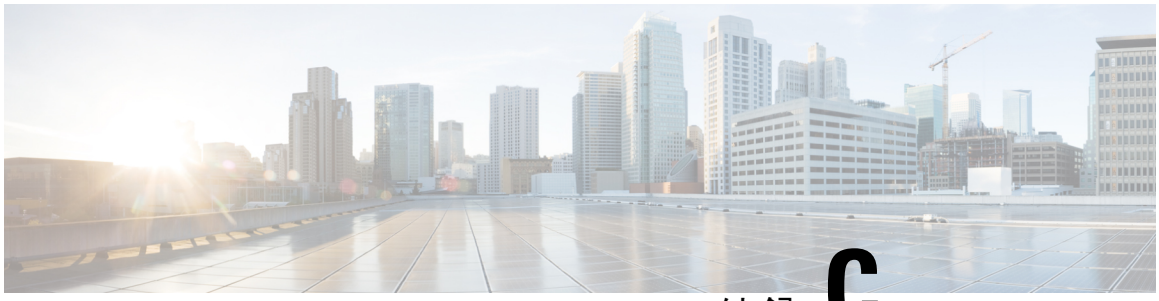
例：

```
<fileRemotePath userName="" remotePort="22" remotePath="" protocol="sftp" name="ToSupport"
host="192.168.200.2"
dn="uni/fabric/path-ToSupport" descr="">

<fileRsARemoteHostToEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default"/>

</fileRemotePath>
```

ステップ 2 REST API を使用して次のような XML を POST 送信し、オンデマンドのテクニカル サポート ファイルを生成します。

例：

```
<dbgexpTechSupOnD upgradeLogs="no" startTime="unspecified" name="Tech_Support_9-20-16"
  exportToController="no" endTime="unspecified" dn="uni/fabric/tsod-Tech_Support_9-20-16" descr=""
  compression="gzip" category="forwarding" adminSt="untriggered">
    <dbgexpRsExportDest tDn="uni/fabric/path-ToSupport"/>
    <dbgexpRsTsSrc tDn="topology/pod-1/node-102/sys"/>
    <dbgexpRsTsSrc tDn="topology/pod-1/node-103/sys"/>
    <dbgexpRsTsSrc tDn="topology/pod-1/node-101/sys"/>
    <dbgexpRsData tDn="uni/fabric/tscont"/>
</dbgexpTechSupOnD>
<fabricFuncP>
    <fabricCtrlrPGrp name="default">
        <fabricRsApplTechSupOnDemand tnDbgexpTechSupOnDName=" Tech_Support_9-20-16"/>
    </fabricCtrlrPGrp>
</fabricFuncP>
```



付録 C

スイッチ インベントリの検索

スイッチのモデルとシリアル番号を知っていると、TAC サポートがファブリックのトラブルシューティングを行うのに役立ちます。このセクションでは、Cisco APIC GUI、CLI、および REST API を使用してスイッチのモデルとシリアル番号を見つける方法について説明します。

GUI を使用してスイッチ インベントリを検索する

このセクションでは、Cisco APIC GUI を使用してスイッチのモデルとシリアル番号を検索する方法について説明します。

始める前に

Cisco APIC GUI にアクセスできる必要があります。

ステップ 1 メニュー バーで **[ファブリック（Fabric）] > [インベントリ（Inventory）]** を選択します。

ステップ 2 ナビゲーション ペインで **[ポッド（Pod）]** アイコンをクリックします。
ナビゲーション ペインにスイッチ アイコンが表示されます。

ステップ 3 ナビゲーション ペインでスイッチ アイコンをクリックします。
作業ウィンドウの上部にタブのリストが表示されます。

ステップ 4 [General] タブをクリックします。
作業ペインにスイッチ情報が表示されます。

NX-OS CLI を使用したスイッチ インベントリの検索

このセクションでは、NX-OS CLI を使用してスイッチのモデルとシリアル番号を見つける方法について説明します。

スイッチ インベントリの検索

次のようにスイッチ インベントリを見つけます。

例：

```
switch# show hardware
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Documents: http://www.cisco.com/en/US/products/ps9372/tsd_products_support_series_home.html
Copyright (c) 2002-2014, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php

Software
  BIOS:      version 07.56
  kickstart: version 12.1(1h) [build 12.1(1h)]
  system:    version 12.1(1h) [build 12.1(1h)]
  PE:        version 2.1(1h)
  BIOS compile time:       06/08/2016
  kickstart image file is: /bootflash/aci-n9000-dk9.12.1.1h.bin
  kickstart compile time:  10/01/2016 20:10:40 [10/01/2016 20:10:40]
  system image file is:    /bootflash/auto-s
  system compile time:     10/01/2016 20:10:40 [10/01/2016 20:10:40]


Hardware
  cisco N9K-C93180YC-EX ("supervisor")
   Intel(R) Xeon(R) CPU  @ 1.80GHz with 16400384 kB of memory.
  Processor Board ID FDO20101H1W

  Device name: ifav41-leaf204
  bootflash:    62522368 kB

Kernel uptime is 02 day(s), 21 hour(s), 42 minute(s), 31 second(s)

Last reset at 241000 usecs after Sun Oct 02 01:27:25 2016
  Reason: reset-by-installer
  System version: 12.1(1e)
  Service: Upgrade

plugin
  Core Plugin, Ethernet Plugin
--------------------------------
Switch hardware ID information
--------------------------------

Switch is booted up
Switch type is : Nexus C93180YC-EX Chassis
Model number is N9K-C93180YC-EX
H/W version is 0.2010
Part Number is 73-15298-01
Part Revision is 1
Manufacture Date is Year 20 Week 10
Serial number is FDO20101H1W
CLEI code is 73-15298-01

--------------------------------
Chassis has one slot
```

スイッチ インベントリの検索

```
--------------------------------

Module1 ok
  Module type is : 48x10/25G
  1 submodules are present
  Model number is N9K-C93180YC-EX
  H/W version is 0.2110
  Part Number is 73-17776-02
  Part Revision is 11
  Manufacture Date is Year 20 Week 10
  Serial number is FDO20101H1W
  CLEI code is 73-17776-02

GEM ok
  Module type is : 6x40/100G Switch
  1 submodules are present
  Model number is N9K-C93180YC-EX
  H/W version is 0.2110
  Part Number is 73-17776-02
  Part Revision is 11
  Manufacture Date is Year 20 Week 10
  Serial number is FDO20101H1W
  CLEI code is 73-17776-02

---------------------------------------
Chassis has  2 PowerSupply Slots
---------------------------------------

PS1 shut
  Power supply type is : 54.000000W 220v AC
  Model number is NXA-PAC-650W-PE
  H/W version is 0.0
  Part Number is 341-0729-01
  Part Revision is A0
  Manufacture Date is Year 19 Week 50
  Serial number is LIT19500ZEK
  CLEI code is 341-0729-01

PS2 ok
  Power supply type is : 54.000000W 220v AC
  Model number is NXA-PAC-650W-PE
  H/W version is 0.0
  Part Number is 341-0729-01
  Part Revision is A0
  Manufacture Date is Year 19 Week 50
  Serial number is LIT19500ZEA
  CLEI code is 341-0729-01

---------------------------------------
Chassis has  4 Fans
---------------------------------------

FT1 ok

 Fan1(sys_fan1)(fan_model:NXA-FAN-30CFM-F)                                  is inserted but info
 is not available

FT2 ok

 Fan2(sys_fan2)(fan_model:NXA-FAN-30CFM-F)                                  is inserted but info
 is not available

FT3 ok
```

```
 Fan3(sys_fan3)(fan_model:NXA-FAN-30CFM-F)                                    is inserted but info
 is not available

FT4 ok

 Fan4(sys_fan4)(fan_model:NXA-FAN-30CFM-F)                                    is inserted but info
 is not available


==============================================================================
```

REST API を使用したスイッチ インベントリの検索

このセクションでは、REST API を使用してスイッチのモデルとシリアル番号を見つける方法について説明します

次のようにスイッチ インベントリを見つけます。

例：

```
GET
https://192.0.20.123/api/node/mo/topology/pod-1.json?query-target=children&target-subtree-class=fabricNode
```

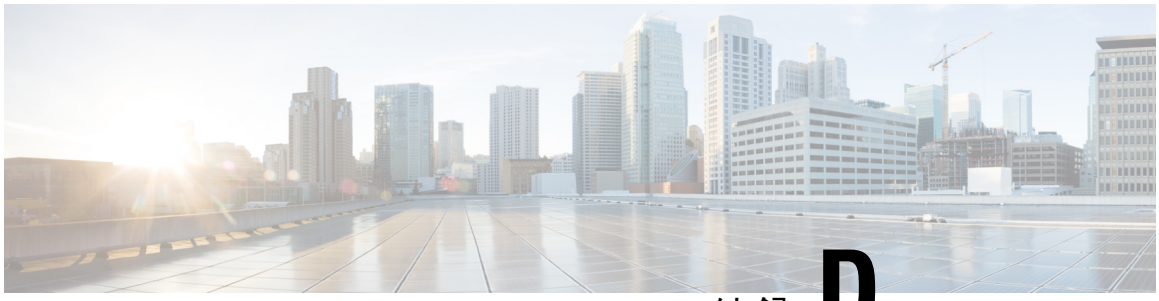
次の応答が返されます：

```
response:
  {
     "totalCount":"8",
     "imdata":
     [{
         "fabricNode":{
           "attributes":{
              "adSt":"on",
              "childAction":"",
              "delayedHeartbeat":"no",
              "dn":"topology/pod-1/node-103",
              "fabricSt":"active",
              "id":"103",
              "lcOwn":"local",
              "modTs":"2016-10-08T14:49:35.665+00:00",
              "model":"N9K-C9396PX",
              "monPolDn":"uni/fabric/monfab-default",
              "name":"leaf3",
              "nameAlias":"",
              "role":"leaf",
              "serial":"TEP-1-103",
              "status":"","uid":"0",
              "vendor":"Cisco Systems, Inc",
              "version":""}
            }
     },{
         "fabricNode":{
           "attributes":{
             "adSt":"on",
             "childAction":"",
             "delayedHeartbeat":"no",
```

スイッチ インベントリの検索


```
        "dn":"topology/pod-1/node-105",
        "fabricSt":"active",
        "id":"105",
        "lcOwn":"local",
        "modTs":"2016-10-08T14:47:52.011+00:00",
        "model":"N9K-C9508",
        "monPolDn":"uni/fabric/monfab-default",
        "name":"spine2",
        "nameAlias":"",
        "role":"spine",
        "serial":"TEP-1-105","status":"",
        "uid":"0",
        "vendor":"Cisco Systems, Inc",
        "version":""
     ...
  [TRUNCATED]
     ...
}
```

スイッチ インベントリの検索



付録 D

Cisco APIC SSD の交換

この手順を使用して、Cisco APIC のソリッド ステート ドライブ（SSD）を交換します。

（注） この手順は、クラスタに正常な SSD を備えた APIC が少なくとも 1 つあり、完全に適合している場合にのみ実行する必要があります。クラスタ内のすべての APIC コントローラに障害が発生した SSD がある場合は、Cisco Technical Assistance Center（TAC）でケースをオープンしてください。

• Cisco APIC のソリッド ステート ドライブ（SSD）の交換 （211 ページ）

Cisco APIC のソリッド ステート ドライブ（SSD）の交換

始める前に

- Cisco IMC リリースが 2.0(9c) より前の場合は、ソリッド ステート ドライブ (SSD) を交換する前に Cisco IMC ソフトウェアをアップグレードする必要があります。対象の Cisco IMC リリースのリリース ノートを参照して、現在のリリースから対象のリリースへの推奨されるアップグレードパスを確認してください。このリンクにある『*Cisco Host Upgrade Utility（HUU）User Guide*』の現在のバージョンの指示に従って、アップグレードを実行します。
- Cisco IMC BIOS で、トラステッドプラットフォーム モジュール（TPM）の状態が「有効」に設定されていることを確認します。KVM コンソールを使用して BIOS 設定にアクセスすると、**[高度（Advanced）] > [トラステッド コンピューティング（Trusted Computing）] > [TPM ステート（TPM State）]** で TPM の状態を表示および構成できます。

（注） TPM ステートが「無効」の場合、APIC は起動に失敗します。

- シスコ ソフトウェア ダウンロード サイトから APIC `.iso` イメージを取得します。

Cisco APIC SSD の交換

（注） APIC .isoイメージのリリースバージョンは、クラスタ内の他のAPIC コントローラと同じバージョンである必要があります。

ステップ 1 クラスタ内の別の APIC から、SSD を交換する APIC を廃止します。

a) メニューバーで、 **System** > **Controllers** を選択します。

b) **Navigation** ウィンドウで、 **Controllers** > **apic_controller_name** > **Cluster as Seen by Node** を展開します。**apic_controller_name** には、廃止されていない APIC コントローラを指定します。

c) 継続する前に、**Work** ウィンドウで、クラスタの **Health State** (**Active Controllers** サマリ テーブルに示されているもの) が **Fully Fit** になっていることを確認します。

d) 同じ **[作業（Work）]** ペインで、廃止するコントローラを選択し、**[アクション（Actions）]** > **[廃止（Decommission）]** をクリックします。

e) **Yes** をクリックします。
解放されたコントローラは [Operational State] 列に [Unregistered] と表示されます。コントローラは稼動対象外になり、**[作業（Work）]** ウィンドウには表示されなくなります。

ステップ 2 古い SSD があればそれを物理的に取り外し、新しい SSD を追加します。

ステップ 3 Cisco IMC で、新しく取り付けた SSD を使用して RAID ボリュームを作成します。

Cisco IMC については、『*Cisco UCS C* シリーズ統合管理コントローラ *GUI* 構成ガイド』を参照してください。「ストレージアダプタの管理」の章の「未使用の物理ドライブからの仮想ドライブの作成」の手順に従って、RAID 0 仮想ドライブを作成および初期化します。

ステップ 4 Cisco IMC で、仮想メディアを使用して APIC イメージをインストールします。この手順では、SSD がパーティション分割され、APIC ソフトウェアが HDD にインストールされます。

（注） Cisco APIC リリース 4.x 以降の新規インストールについては、『*Cisco APIC* のインストール、アップグレード、およびダウングレード ガイド』を参照してください。

a) Cisco IMC vMedia 機能を使用して、APIC .iso イメージをマウントします。

b) コントローラを起動し電源を再投入します。

c) 起動プロセス中を押して **F6** を選択、 **Cisco vKVM マッピング vDVD** ワンタイム ブート デバイスとして。BIOS パスワードを入力する必要があります。デフォルトのパスワードは「password」です。

d) 最初の起動時に、構成スクリプトが実行されます。画面の指示に従って、APIC ソフトウェアの初期設定を構成します。

e) インストールが完了したら、仮想メディア マウントのマッピングを解除します。

ステップ 5 クラスタ内の APIC から、廃止された APIC を起動します。

a) クラスタの一部である他の APIC を選択します。メニュー バーで、 **[システム (System)]** > **[コントローラ (Controllers)]** を選択します。

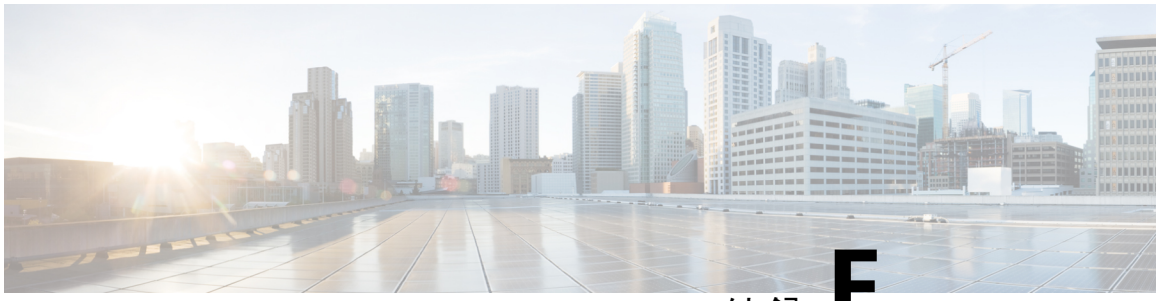
b) **Navigation** ウィンドウで、 **Controllers** > **apic_controller_name** > **Cluster as Seen by Node** を展開します。**apic_controller_name** には、クラスタの一部であるアクティブなコントローラーを指定します。

c) **[作業（Work）]** ウィンドウで、**未登録（Unregistered）** と **稼働状態（Operational State）** 列に表示されている廃止されているコントローラをクリックします。

d) **Work** ウィンドウで、**Actions** > **Commission** をクリックします。
e) **Confirmation** ダイアログボックスで **Yes** をクリックします。

稼働済みコントローラには、正常性状態が**完全適合**と表示され、動作状態が**使用可**能と表示されます。これで、コントローラが **[作業（Work）]** ペインに表示されます。

Cisco APIC SSD の交換



付録 E

予想される出力エラー

予想される出力エラー

Cisco Nexus ハードウェア -EX、-FX1-3、および N93xxC は、内部インターフェイス カウンタに出力エラーを表示し、ACI 環境で障害（F119936）を発生させる可能性があります。**show interface** の出力エラー カウンタが変更されていない限り、これは予期される動作です。

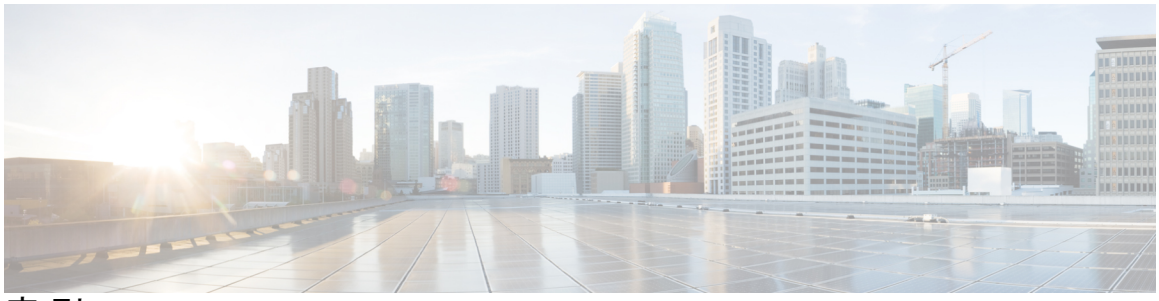
また、**show platform internal counters** ポート出力エラーが増加することに注意してください。ただし、**show interface** で同じポートをチェックすると、出力エラー率は増加しません。

このセクションでは、予想される出力エラーの例を示します。

```
module-1# show platform internal counters port 51
Stats for port 51
(note: forward drops includes sup redirected packets too)
IF          LPort                    Input                      Output
                                Packets       Bytes          Packets     Bytes
eth-1/51    51  Total           669974   110547179       692398    194500094
                Unicast         112138    30292113       439809    161274739
                Multicast            0           0       251315     33075023
                Flood           261736    32880023         1274       150332
            Total Drops         296100                   261736
                Buffer               0                        0
                Error                0                   261736
                <...>

leaf-101# show interface ethernet 1/51
Ethernet1/51 is up
admin state is up, Dedicated Interface
  Hardware: 1000/10000/100000/40000 Ethernet, address: 0000.0000.0000 (bia a023.9f56.48f3)

  MTU 9366 bytes, BW 40000000 Kbit, DLY 1 usec
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, medium is broadcast
  Port mode is routed
  full-duplex, 40 Gb/s, media type is 40G
  FEC (forward-error-correction) : disable-fec
  Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned off
```

予想される出力エラー

```
Rate mode is dedicated
Switchport monitor is off
EtherType is 0x8100
EEE (efficient-ethernet) : n/a
Last link flapped 1d14h
Last clearing of "show interface" counters never
1 interface resets
30 seconds input rate 4912 bits/sec, 3 packets/sec
30 seconds output rate 1944 bits/sec, 2 packets/sec
Load-Interval #2: 5 minute (300 seconds)
  input rate 3360 bps, 2 pps; output rate 10504 bps, 4 pps
L3 in Switched:
  ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
L3 out Switched:
  ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
RX
  352942 unicast packets  317417 multicast packets  0 broadcast packets
  670359 input packets  110608007 bytes
  8643 jumbo packets  0 storm suppression bytes
  0 runts  0 giants  0 CRC  0 no buffer
  0 input error  0 short frame  0 overrun   0 underrun  0 ignored
  0 watchdog  0 bad etype drop  0 bad proto drop  0 if down drop
  0 input with dribble  0 input discard
  0 Rx pause
TX
  417109 unicast packets  275682 multicast packets  0 broadcast packets
  692791 output packets  194559643 bytes
  7173 jumbo packets
  0 output error  0 collision  0 deferred  0 late collision
  0 lost carrier  0 no carrier  0 babble  0 output discard
  0 Tx pause
```



索 引

て

と

ふ

翻訳について
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。