Cisco Nexus 9000 シリーズ NX-OS システム管理設定ガイド、リリース 10.1(x)

最終更新：2025 年 11 月 11 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/

【注意】シスコ製品をご使用になる前に、安全上の注意（ **www.cisco.com/jp/go/safety_warning/** ）をご確認ください。本書は、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。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

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

このマニュアルは、Cisco Nexus スイッチの設置、設定、および維持に携わるネットワーク管理者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。

Cisco Nexus 3000 シリーズ スイッチの関連資料

Cisco Nexus 3000 シリーズ スイッチ全体のマニュアル セットは、次の URL にあります。

https://www.cisco.com/c/en/us/support/switches/nexus-3000-series-switches/tsd-products-support-series-home.html

マニュアルに関するフィードバック

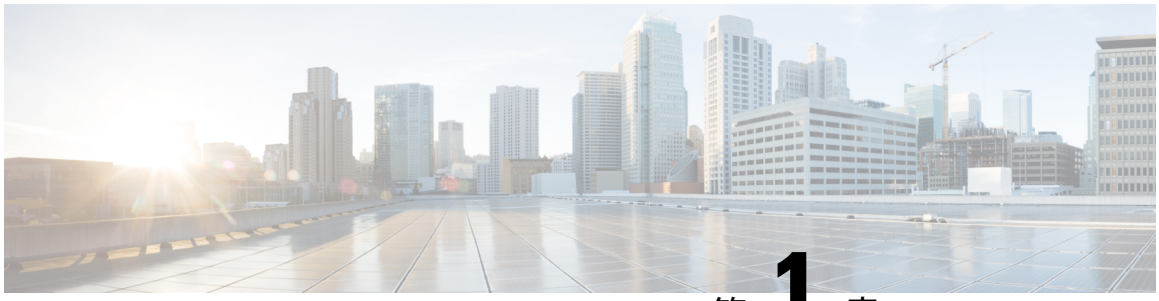
このマニュアルに関する技術的なフィードバック、または誤りや記載もれなどお気づきの点がございましたら、HTML ドキュメント内のフィードバック フォームよりご連絡ください。ご協力をよろしくお願いいたします。

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business results you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



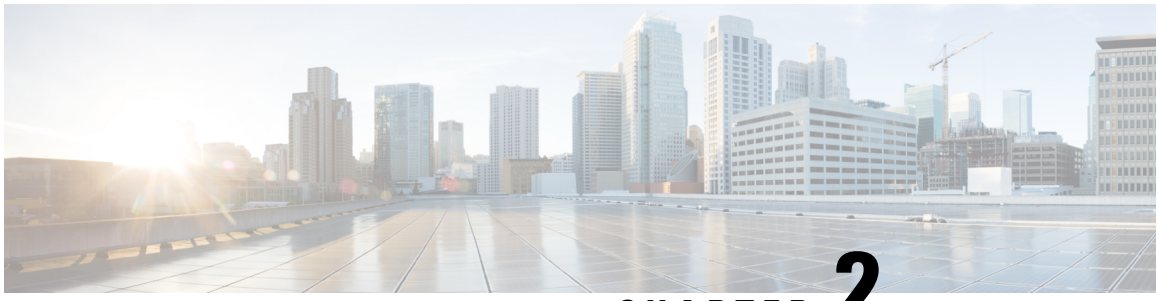
第 1 章

リリース 10.1（x）の新機能および変更された機能

新機能と更新情報

表は、Cisco NX-OS リリース 10.1（x）新機能および変更された機能をリストします：

特長	説明	追加/変更されたリリース	参照先
フラッシュ MIB SNMP ウォーク	Cisco Nexus 3000 シリーズ スイッチは、snmpwalk 要求に対して最大 10000 個のフラッシュファイルをサポートします。	リリース 10.1（1）	SNMP の注意事項および制約事項 （182 ページ）



CHAPTER 2

概要

この章は、次の内容で構成されています。

ライセンス要件

Cisco NX-OS ライセンス方式の推奨の詳細と、ライセンスの取得および適用の方法については、『*Cisco NX-OS* ライセンス ガイド』および『*Cisco NX-OS* ライセンス オプション ガイド』を参照してください。

システム管理機能

このマニュアルに記載されているシステム管理機能について説明します。

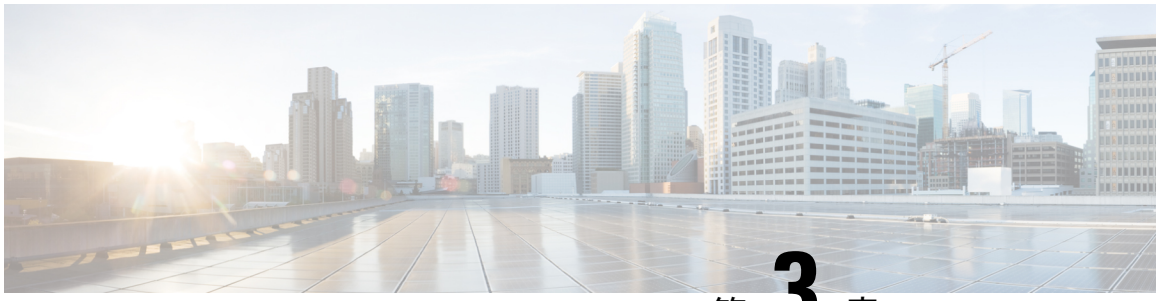
特長	説明
スイッチ プロファイル	設定の同期を使用すると、管理者は、設定変更を 1 台のスイッチで行い、ピア スイッチに自動的に設定を同期させることができます。この機能により、設定ミスがなくなり、管理上のオーバーヘッドが軽減されます。 設定同期モード（config-sync）を使用すると、ローカルおよびピア スイッチを同期するためにスイッチ プロファイルを作成できます。

特長	説明
Cisco Fabric Services	Cisco MDS NX-OS ソフトウェアは、データベースを効率的に分散し、デバイスの柔軟性を高めるため、Cisco Fabric Services（CFS）インフラストラクチャを使用します。CFS により、ファブリック内のすべてのスイッチに設定情報を自動的に配信できるため、SAN のプロビジョニングが簡単になります。
高精度時間プロトコル	高精度時間プロトコル（PTP）はネットワークに分散したノードの時刻同期プロトコルです。そのハードウェアのタイムスタンプ機能は、ネットワーク タイム プロトコル（NTP）などの他の時刻同期プロトコルより高い精度を実現します。
ユーザー アカウントおよび RBAC	ユーザーアカウントおよびロールベース アクセス コントロール（RBAC）では、割り当てられたロールのルールを定義できます。ロールは、ユーザーが管理操作にアクセスするための許可を制限します。各ユーザー ロールに複数のルールを含めることができ、各ユーザーが複数のロールを持つことができます。
Session Manager	Session Manager を使用すると、コンフィギュレーションを作成し、すべて正しく設定されていることを確認および検証したあとでバッチ モードで適用できます。
オンライン診断	Cisco Generic Online Diagnostics（GOLD）では、複数のシスコ プラットフォームにまたがる診断操作の共通フレームワークを定義しています。オンライン診断フレームワークでは、中央集中システムおよび分散システムに対応する、プラットフォームに依存しない障害検出アーキテクチャを規定しています。これには共通の診断 CLI とともに、起動時および実行時に診断するための、プラットフォームに依存しない障害検出手順が含まれます。 プラットフォーム固有の診断機能は、ハードウェア固有の障害検出テストを行い、診断テストの結果に応じて適切な対策を実行できます。

特長	説明
システム メッセージ ロギング	システム メッセージ ロギングを使用して宛先を制御し、システム プロセスが生成するメッセージのシビラティ（重大度）をフィルタリングできます。端末セッション、ログ ファイル、およびリモート システム上の syslog サーバーへのロギングを設定できます。 システム メッセージ ロギングは RFC 3164 に準拠しています。システム メッセージのフォーマットおよびデバイスが生成するメッセージの詳細については、『*Cisco NX-OS System Messages Reference*』を参照してください。
Smart Call Home	Call Home は重要なシステム ポリシーを電子メールで通知します。Cisco NX-OS では、ポケットベル サービス、標準的な電子メール、またはXMLベースの自動化された解析アプリケーションとの最適な互換性のために、広範なメッセージ形式が提供されています。この機能を使用して、ネットワーク サポート エンジニアやネットワーク オペレーション センターを呼び出せます。また、Cisco Smart Call Home サービスを使用して、TAC でケースを自動的に生成することもできます。
設定のロール バック	設定のロールバック機能を使用すると、Cisco NX-OS のコンフィギュレーションのスナップショットまたはユーザー チェックポイントを使用して、スイッチをリロードしなくても、いつでもそのコンフィギュレーションをスイッチに再適用できます。権限のある管理者であれば、チェックポイントで設定されている機能について専門的な知識がなくても、ロールバック機能を使用して、そのチェックポイント コンフィギュレーションを適用できます。
SNMP	簡易ネットワーク管理プロトコル（SNMP）は、SNMP マネージャとエージェント間の通信用メッセージ フォーマットを提供する、アプリケーション レイヤ プロトコルです。SNMP では、ネットワーク内のデバイスのモニタリングと管理に使用する標準フレームワークと共通言語が提供されます。

特長	説明
RMON	RMONは、各種のネットワーク エージェントおよびコンソール システムがネットワーク モニタリング データを交換できるようにするための、Internet Engineering Task Force（IETF）標準モニタリング仕様です。Cisco NX-OS では、Cisco NX-OS デバイスをモニターするための、RMON アラーム、イベント、およびログをサポートします。
SPAN	スイッチド ポート アナライザ（SPAN）機能（ポート ミラーリングまたはポート モニタリングとも呼ばれる）は、ネットワーク アナライザによる分析のためにネットワーク トラフィックを選択します。ネットワーク アナライザは、Cisco SwitchProbe、ファイバ チャネル アナライザ、またはその他のリモート モニタリング（RMON）プローブです。

特長	説明
ERSPAN	Encapsulated Remote Switched Port Analyzer（ERSPAN）は、IP ネットワークでミラーリングされたトラフィックを転送するために使用します。ERSPAN は異なるスイッチ上の送信元ポート、送信元 VLAN、および宛先をサポートし、ネットワーク上にある複数のスイッチのリモート モニタリングを可能にします。ERSPAN は、スイッチ間でトラフィックを伝送するために、Generic Routing Encapsulation（GRE）を使用します。 ERSPANは、ERSPAN送信元セッション、ルーティング可能なERSPAN GREカプセル化トラフィック、およびERSPAN宛先セッションで構成されています。異なるスイッチでERSPAN送信元セッションおよび宛先セッションを個別に設定します。 ERSPAN 送信元セッションを 1 台のスイッチ上で設定するには、送信元ポートまたはVLANのセットを、宛先IPアドレス、ERSPAN ID番号、および仮想ルーティングおよび転送（VRF）名に対応付けます。ERSPAN宛先セッションを別のスイッチ上で設定するには、宛先を送信元IPアドレス、ERSPAN ID番号、およびVRF名に対応付けます。 ERSPAN 送信元セッションは、送信元ポートまたは送信元 VLAN からのトラフィックをコピーし、このトラフィックを、ルーティング可能な GRE カプセル化パケットを使用して ERSPAN 宛先セッションに転送します。ERSPAN 宛先セッションはトラフィックを宛先にスイッチングします。



第 3 章

スイッチ プロファイルの設定

この章は、次の項で構成されています。

スイッチ プロファイルに関する情報

Cisco NX-OS リリース 6.0(2)U4(1) には、スイッチ プロファイルが導入されています。複数のアプリケーションは、ネットワーク内の Cisco Nexus シリーズ スイッチ間で整合性のある設定が必要です。コンフィギュレーションが一致しない場合、エラーやコンフィギュレーションエラーが生じる可能性があります。その結果、サービスが中断することがあります。

設定の同期（config-sync）機能では、1 つのスイッチ プロファイルを設定し、設定を自動的にピア スイッチに同期させることができます。スイッチプロファイルには次の利点があります。

- スイッチ間でコンフィギュレーションを同期化できます。
- 2 つのスイッチ間で接続が確立されると、コンフィギュレーションがマージされます。
- どのコンフィギュレーションを同期化するかを完全に制御できます。
- マージチェックおよび相互排除チェックを使用して、ピア全体でコンフィギュレーションの一貫性を確保します。
- verify 構文および commit 構文を提供します。

スイッチ プロファイル：コンフィギュレーション モード

スイッチ プロファイル機能には、次のコンフィギュレーション モードがあります。

- コンフィギュレーション同期化モード
- スイッチ プロファイル モード
- スイッチ プロファイル インポート モード

コンフィギュレーション同期モード

コンフィギュレーション同期モード（config-sync）では、プライマリとして使用するローカル スイッチ上で **config sync** コマンドを使用して、スイッチ プロファイルを作成できます。プロファイルの作成後、同期するピア スイッチで **config sync** コマンドを入力できます。

スイッチ プロファイル モード

スイッチ プロファイル モードでは、後でピア スイッチと同期化されるスイッチ プロファイルに、サポートされているコンフィギュレーション コマンドを追加できます。スイッチ プロファイル モードで入力したコマンドは、**commit** コマンドを入力するまでバッファに格納されます。

スイッチ プロファイル インポート モード

以前のリリースからアップグレードする場合、**import** コマンドを入力して、サポートされている実行コンフィギュレーション コマンドをスイッチ プロファイルにコピーすることができます。**import** コマンドを入力すると、スイッチ プロファイル モード（config-sync-sp）は、スイッチ プロファイル インポート モード（config-sync-sp-import）に変わります。スイッチ プロファイル インポート モードでは、既存のスイッチ設定を実行コンフィギュレーションからインポートし、どのコマンドをスイッチ プロファイルに含めるかを指定できます。

スイッチ プロファイルに含まれるコマンドはトポロジによって異なるため、**import** コマンド モードでは、インポートされたコマンド セットを特定のトポロジに合わせて変更できます。

インポート プロセスを完了し、スイッチ プロファイルにコンフィギュレーションを移動するには、**commit** コマンドを入力する必要があります。インポートプロセス中のコンフィギュレーション変更はサポートされていません。そのため、**commit** コマンドを入力する前に新しいコマンドを追加した場合、スイッチプロファイルは保存されていない状態であり、スイッチはスイッチ プロファイル インポート モードのままになります。追加したコマンドを削除するか、またはインポートを中断します。プロセスを中断すると、保存されていないコンフィギュレーションは失われます。インポートを完了したら、新しいコマンドをスイッチプロファイルに追加できます。

コンフィギュレーションの検証

次の 2 種類のコンフィギュレーション検証チェックを使用して、2 種類のスイッチ プロファイル エラーを識別できます。

- 相互排除チェック
- マージ チェック

相互排除チェック

スイッチプロファイルに含まれるコンフィギュレーションが上書きされる可能性を減らすためには、相互排除（mutex）でスイッチプロファイル コマンドをローカル スイッチに存在するコマンドとピア スイッチのコマンドに照合してチェックします。スイッチ プロファイルに含まれるコマンドは、そのスイッチプロファイルの外部またはピア スイッチでは設定できません。この要件により、既存のコマンドが意図せずに上書きされる可能性が減少します。

ピア スイッチに到達可能である場合、mutex チェックは、共通プロセスの一環として両方のスイッチで行われます。それ以外の場合は、mutex チェックはローカルで実行されます。設定端末から行われるコンフィギュレーション変更は、ローカル スイッチのみに反映されます。

mutex チェックがエラーを識別すると、mutex の障害として報告され、手動で修正する必要があります。

相互排除ポリシーには、次の例外が適用されます。

- インターフェイス設定：ポート チャネル インターフェイスは、スイッチ プロファイル モードまたはグローバル コンフィギュレーション モードで設定が済んでいる必要があります。

（注） 一部のポート チャネル サブコマンドは、スイッチ プロファイル モードで設定できません。ただしこれらのコマンドは、ポート チャネルがスイッチ プロファイル モードで作成、設定されている場合でも、グローバル コンフィギュレーション モードからであれば設定することができます。

たとえば、次のコマンドはグローバル コンフィギュレーション モードでのみ設定可能です。

switchport private-vlan association trunk *primary-vlan secondary-vlan*

- shutdown/no shutdown
- System QoS

マージ チェック

マージ チェックは、コンフィギュレーションを受信する側のピア スイッチで実行されます。マージチェックは、受信したコンフィギュレーションが、受信側のスイッチにすでに存在するスイッチ プロファイル コンフィギュレーションと競合しないようにします。マージ チェックは、マージ プロセスまたはコミット プロセス中に実行されます。エラーはマージ エラーとして報告され、手動で修正する必要があります。

1 つまたは両方のスイッチがリロードされ、コンフィギュレーションが初めて同期化される際には、マージ チェックによって、両方のスイッチのスイッチ プロファイル コンフィギュレーションが同じであることが検証されます。スイッチ プロファイルの相違はマージ エラーとして報告され、手動で修正する必要があります。

スイッチ プロファイルを使用したソフトウェアのアップグレードとダウングレード

以前のリリースにダウングレードすると、以前のリリースではサポートされていない既存のスイッチ プロファイルを削除するように要求されます。

以前のリリースからアップグレードする場合、スイッチ プロファイルに一部の実行コンフィギュレーション コマンドを移動することを選択できます。**import** コマンドでは、関連するスイッチ プロファイル コマンドをインポートできます。バッファされた（コミットされていない）コンフィギュレーションが存在する場合でもアップグレードを実行できますが、コミットされていないコンフィギュレーションは失われます。

スイッチ プロファイルに含まれるスイッチの 1 つで In Service Software Upgrade（ISSU）を実行しても、コンフィギュレーションを同期化することはできません。これは、ピアに到達できないためです。

スイッチ プロファイルの前提条件

スイッチ プロファイルには次の前提条件があります。

- **cfs ipv4 distribute** コマンドを入力して、両方のスイッチで mgmt0 上の Cisco Fabric Series over IP（CFSoIP）配信を有効にする必要があります。
- **config sync** および **switch-profile** コマンドを入力して、両方のピア スイッチで同じ名前のスイッチ プロファイルを設定する必要があります。
- **sync-peers destination** コマンドを入力して、各スイッチをピア スイッチとして設定します。

スイッチ プロファイルの注意事項および制約事項

スイッチプロファイルを設定する場合は、次の注意事項および制約事項を考慮してください。

- mgmt0 インターフェイスを使用してのみ設定同期化をイネーブルにできます。
- 設定の同期は、mgmt 0 インターフェイスを使用して実行され、管理 SVI を使用して実行できません。
- 同じスイッチ プロファイル名で同期されたピアを設定する必要があります。
- スイッチ プロファイル設定で使用可能なコマンドを、設定スイッチ プロファイル（config-sync-sp）モードで設定できます。
- 1 つのスイッチプロファイルセッションを一度に進行できます。別のセッションの開始を試みると失敗します。
- スイッチ プロファイル セッションの進行中は、コンフィギュレーション端末モードから実行されたサポートされているコマンドの変更はブロックされます。スイッチプロファイルセッションが進行しているときは、コンフィギュレーション端末モードからサポートされていないコマンドの変更を行わないでください。
- **commit** コマンドを入力し、ピア スイッチに到達可能である場合、設定は、両方のピア スイッチに適用されるか、いずれのスイッチにも適用されません。コミットの障害が発生した場合、コマンドは、スイッチ プロファイル バッファに残ります。その場合、必要な修正をし、コミットを再試行します。
- いったんスイッチ プロファイル モードで設定したポート チャネルを、グローバル コンフィギュレーション（config terminal）モードで設定することはできません。

（注） ポート チャネルに関する一部のサブコマンドは、スイッチ プロファイル モードでは設定できません。ただしこれらのコマンドは、ポート チャネルがスイッチ プロファイル モードで作成、設定されている場合でも、グローバル コンフィギュレーション モードからであれば設定することができます。

たとえば、次のコマンドはグローバル コンフィギュレーション モードでのみ設定可能です。

switchport private-vlan association trunk *primary-vlan secondary-vlan*

- shutdown および no shutdown は、グローバル コンフィギュレーション モードとスイッチ プロファイル モードのどちらでも設定できます。
- ポート チャネルをグローバル コンフィギュレーション モードで作成した場合は、メンバー インターフェイスを含むチャネル グループも、グローバル コンフィギュレーション モードを使用して作成する必要があります。
- スイッチ プロファイル モードで設定されたポート チャネルには、スイッチ プロファイルの内部と外部どちらからもメンバーにすることができます。
- メンバー インターフェイスをスイッチ プロファイルにインポートする場合は、メンバー インターフェイスを含むポート チャネルがスイッチ プロファイル内にも存在する必要があります。

接続の切断後の同期化の注意事項

- mgmt0 インターフェイスの接続が失われた後の設定の同期化：mgmt0 インターフェイスの接続が失われ、設定変更が必要な場合は、スイッチ プロファイルを使用して、両方のスイッチの設定変更を適用します。mgmt0 インターフェイスへの接続が復元されると、両方のスイッチが自動的に同期されます。

  設定変更を 1 台のスイッチだけで実行する場合、マージは、mgmt0 インターフェイスが起動し、設定が他のスイッチに適用されると実行されます。

スイッチ プロファイルの設定

スイッチ プロファイルは作成および設定できます。コンフィギュレーション同期モード（config-sync）で、**switch-profile** *name* コマンドを入力します。

始める前に

スイッチプロファイルは、各スイッチで同じ名前を使用して作成する必要があります。また、スイッチは互いにピアとして設定する必要があります。同じアクティブなスイッチプロファイルが設定されたスイッチ間で接続が確立されると、スイッチプロファイルが同期化されます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**cfs ipv4 distribute**
	例：
	`switch(config)# cfs ipv4 distribute`
	`switch(config)#`
		ピア スイッチ間の CFS 配信をイネーブルにします。

ステップ 3	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 4	**switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル同期コンフィギュレーション モードを開始します。

ステップ 5	**sync-peers destination** *IP-address*
	例：
	`switch(config-sync-sp)# sync-peers destination 10.1.1.1`
	`switch(config-sync-sp)#`
		ピア スイッチを設定します。

ステップ 6	（任意） **show switch-profile** *name* **status**
	例：
	`switch(config-sync-sp)# show switch-profile abc status`
	`switch(config-sync-sp)#`
		ローカル スイッチのスイッチ プロファイルおよびピア スイッチ情報を表示します。

ステップ 7	**exit**
	例：
	`switch(config-sync-sp)# exit`
	`switch#`
		スイッチ プロファイル コンフィギュレーション モードを終了し、EXEC モードに戻ります。

ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、スイッチプロファイルを設定し、スイッチプロファイルのステータスを表示する例を示します。

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# show switch-profile abc status
Start-time:  15801 usecs after Mon Aug 23 06:21:08 2010
End-time:   6480 usecs after Mon Aug 23 06:21:13 2010

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.1.1.1
Sync-status: In Sync.
Status: Commit Success
Error(s):
switch(config-sync-sp)# exit
switch#
```

スイッチ プロファイルへのスイッチの追加

スイッチ プロファイル コンフィギュレーション モードで **sync-peers destination** *destination IP* コマンドを入力し、スイッチ プロファイルにスイッチを追加します。

スイッチを追加する場合は、次の注意事項に従ってください。

- スイッチは IP アドレスで識別されます。
- 宛先 IP は同期するスイッチの IP アドレスです。
- コミットされたスイッチ プロファイルは、ピア スイッチでも設定の同期が設定されている場合に、新しく追加されたピアと（オンラインの場合）同期されます。

  メンバー インターフェイスをスイッチ プロファイルにインポートする場合は、メンバー インターフェイスを含むポート チャネルがスイッチ プロファイル内にも存在する必要があります。

始める前に

ローカル スイッチでスイッチ プロファイルを作成した後、同期に含まれる 2 番目のスイッチを追加する必要があります。

手順

	コマンドまたはアクション	目的

ステップ 1	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル同期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	**sync-peers destination** *destination IP*
	例：
	`switch(config-sync-sp)# sync-peers destination 10.1.1.1`
	`switch(config-sync-sp)#`
		スイッチ プロファイルにスイッチを追加します。

ステップ 4	**exit**
	例：
	`switch(config-sync-sp)# exit`
	`switch#`
		スイッチ プロファイル コンフィギュレーション モードを終了します。

ステップ 5	（任意） **show switch-profile peer** 例： `switch# show switch-profile peer`	スイッチ プロファイルのピアの設定を表示します。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

スイッチ プロファイルのコマンドの追加または変更

スイッチ プロファイルのコマンドを変更するには、変更されたコマンドをスイッチ プロファイルに追加し、**commit** コマンドを入力してコマンドを適用し、ピア スイッチが到達可能な場合にスイッチ プロファイルを同期します。

スイッチ プロファイル コマンドを追加または変更するときは、次の注意事項に従ってください。

- 追加または変更されたコマンドは、**commit** コマンドを入力するまでバッファに格納されます。
- コマンドは、バッファリングされた順序で実行されます。特定のコマンドに順序の依存関係がある場合（たとえば、QoS ポリシーは適用前に定義する必要がある）、その順序を維持する必要があります。そうしないとコミットに失敗する可能性があります。**show switch-profile name buffer** コマンド、**buffer-delete** コマンド、**buffer-move** コマンドなどのユーティリティ コマンドを使用して、バッファを変更し、入力済みのコマンドの順序を修正できます。

始める前に

ローカルおよびピア スイッチでスイッチ プロファイルを設定したら、スイッチ プロファイルにサポートされているコマンドを追加し、コミットする必要があります。コマンドは、**commit** コマンドを入力するまでスイッチ プロファイル バッファに追加されます。**commit** コマンドは次を行います。

- mutex チェックとマージ チェックを起動し、同期を確認します。
- ロールバック インフラストラクチャでチェックポイントを作成します。
- ローカル スイッチおよびピア スイッチのコンフィギュレーションを適用します。
- スイッチ プロファイル内の任意のスイッチでアプリケーション障害がある場合は、すべてのスイッチでロール バックを実行します。
- チェックポイントを削除します。

手順

	コマンドまたはアクション	目的

ステップ 1	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル同期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	*Command argument*
	例：
	`switch(config-sync-sp)# interface Port-channel100`
	`switch(config-sync-sp-if)# speed 1000`
	`switch(config-sync-sp-if)# interface`
		スイッチ プロファイルにコマンドを追加します。

	コマンドまたはアクション	目的
	Ethernet1/1 switch(config-sync-sp-if)# speed 1000 switch(config-sync-sp-if)# channel-group 100	
ステップ 4	（任意） **show switch-profile** *name* **buffer** **例：** switch(config-sync-sp)# show switch-profile abc buffer switch(config-sync-sp)#	スイッチ プロファイル バッファ内のコンフィギュレーション コマンドを表示します。
ステップ 5	**verify** **例：** switch(config-sync-sp)# verify	スイッチ プロファイル バッファ内のコマンドを確認します。
ステップ 6	**commit** **例：** switch(config-sync-sp)# commit	スイッチ プロファイルにコマンドを保存し、ピア スイッチと設定を同期します。
ステップ 7	（任意） **show switch-profile** *name* **status** **例：** switch(config-sync-sp)# show switch-profile abc status switch(config-sync-sp)#	ローカル スイッチのスイッチ プロファイルのステータスとピア スイッチのステータスを表示します。
ステップ 8	**exit** **例：** switch(config-sync-sp)# exit switch#	スイッチ プロファイル コンフィギュレーション モードを終了します。
ステップ 9	（任意） **copy running-config startup-config** **例：** switch# copy running-config startup-config	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

例

次に、スイッチ プロファイルを作成し、ピア スイッチを設定し、スイッチ プロファイルにコマンドを追加する例を示します。

```
switch# configuration terminal
switch(config)# cfs ipv4 distribute
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# sync-peers destination 10.1.1.1
switch(config-sync-sp)# interface port-channel100
switch(config-sync-sp-if)# speed 1000
switch(config-sync-sp-if)# interface Ethernet1/1
```

```
switch(config-sync-sp-if)# speed 1000
switch(config-sync-sp-if)# channel-group 100
switch(config-sync-sp)# verify
switch(config-sync-sp)# commit
switch(config-sync-sp)# exit
switch#
```

次に、定義されたスイッチプロファイルがある既存のコンフィギュレーションの例を示します。2 番目の例は、スイッチ プロファイルに変更されたコマンドを追加することによって、スイッチ プロファイル コマンドを変更する方法を示します。

```
switch# show running-config
switch-profile abc
  interface Ethernet1/1
    switchport mode trunk
    switchport trunk allowed vlan 1-10

switch# config sync
switch(config-sync)# switch-profile abc
switch(config-sync-sp)# interface Ethernet1/1
switch(config-sync-sp-if)# switchport trunk allowed vlan 5-10
switch(config-sync-sp-if)# commit

switch# show running-config
switch-profile abc
  interface Ethernet1/1
    switchport mode trunk
    switchport trunk allowed vlan 5-10
```

スイッチ プロファイルのインポート

インポートするコマンドのセットに基づいてスイッチプロファイルをインポートできます。コンフィギュレーション ターミナル モードを使用して、次のことを実行できます。

- 選択したコマンドをスイッチ プロファイルに追加する。
- インターフェイスに指定された、サポートされているコマンドを追加する。
- サポートされているシステムレベル コマンドを追加する。
- サポートされているシステムレベル コマンドを追加する（物理インターフェイス コマンドを除く）。

スイッチ プロファイルにコマンドをインポートする場合、スイッチプロファイル バッファが空である必要があります。

新しいコマンドがインポート中に追加されると、スイッチプロファイルが保存されていないままになり、スイッチはスイッチ プロファイル インポート モードのままになります。**abort** コマンドを入力してインポートを停止します。スイッチプロファイルのインポートの詳細については、「スイッチ プロファイル インポート モード」の項を参照してください。

手順

	コマンドまたはアクション	目的

ステップ 1	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル同期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	**import** {*interface port/slot* | *running-config* [**exclude interface ethernet**]}
	例：
	`switch(config-sync-sp)# import ethernet 1/2`
	`switch(config-sync-sp-import)#`
		インポートするコマンドを識別し、スイッチ プロファイル インポート モードを開始します。
		• <CR>：選択したコマンドを追加します。
		• **interface**：指定したインターフェイスのサポートされるコマンドを追加します。
		• **running-config**：サポートされるシステム レベル コマンドを追加します。
		• **running-config exclude interface ethernet**：サポートされるシステム レベル コマンドを追加します（物理インターフェイス コマンドを除く）。

ステップ 4	**commit** 例： `switch(config-sync-sp-import)# commit`	コマンドをインポートし、スイッチ プロファイルにコマンドを保存します。
ステップ 5	（任意） **abort** 例： `switch(config-sync-sp-import)# abort`	インポート プロセスを中止します。
ステップ 6	**exit** 例：	スイッチ プロファイル インポート モードを終了します。

	コマンドまたはアクション	目的
	`switch(config-sync-sp)# exit` `switch#`	
ステップ 7	（任意） **show switch-profile** 例： `switch# show switch-profile`	スイッチ プロファイル コンフィギュレーションを表示します。
ステップ 8	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

例

次に、sp というスイッチ プロファイルに、イーサネット インターフェイス コマンドを除く、サポートされるシステムレベル コマンドをインポートする例を示します。

```
switch(config-vlan)# conf sync
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# show switch-profile buffer

switch-profile  : sp
----------------------------------------------------------
Seq-no  Command
----------------------------------------------------------

switch(config-sync-sp)# import running-config exclude interface ethernet
switch(config-sync-sp-import)#
switch(config-sync-sp-import)# show switch-profile buffer

switch-profile  : sp
----------------------------------------------------------
Seq-no  Command
----------------------------------------------------------
3       vlan 100-299
4       vlan 300
4.1       state suspend
5       vlan 301-345
6       interface port-channel100
6.1       spanning-tree port type network
7       interface port-channel105

switch(config-sync-sp-import)#
```

スイッチ プロファイルのコマンドの確認

スイッチ プロファイル モードで **verify** コマンドを入力し、スイッチ プロファイルに含まれるコマンドを確認できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル同期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	**verify** 例： `switch(config-sync-sp)# verify`	スイッチ プロファイル バッファ内のコマンドを確認します。

ステップ 4	**exit**
	例：
	`switch(config-sync-sp)# exit`
	`switch#`
		スイッチ プロファイル コンフィギュレーション モードを終了します。

ステップ 5	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ピア スイッチの分離

スイッチ プロファイルを変更するためにピア スイッチを分離できます。このプロセスは、設定の同期をブロックする場合、または設定をデバッグするときに使用できます。

ピア スイッチを分離するには、スイッチ プロファイルからスイッチを削除し、スイッチ プロファイルにピア スイッチを追加する必要があります。

一時的にピア スイッチを分離するには、次の手順を実行します。

1. スイッチ プロファイルからピア スイッチを削除します。
2. スイッチ プロファイルを変更して、変更をコミットします。
3. debug コマンドを入力します。
4. 手順 2 でスイッチ プロファイルに対して行った変更を元に戻し、コミットします。

5. スイッチ プロファイルにピア スイッチを追加します。

スイッチ プロファイルの削除

all-config または **local-config** オプションを選択してスイッチ プロファイルを削除できます。

- **all-config**：両方のピア スイッチでスイッチ プロファイルを削除します（両方が到達可能な場合）。このオプションを選択し、ピアの1つが到達不能である場合、ローカル スイッチプロファイルだけが削除されます。**all-config** オプションは両方のピア スイッチでスイッチ プロファイルを完全に削除します。
- **local-config**：ローカル スイッチのみのスイッチ プロファイルを削除します。

手順

	コマンドまたはアクション	目的

ステップ 1	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**no switch-profile name** {**all-config** | **local-config**}
	例：
	`switch(config-sync)# no switch-profile abc local-config`
	`switch(config-sync-sp)#`
		次の手順に従って、スイッチ プロファイルを削除します。
		• **all-config**：ローカル スイッチおよびピア スイッチのスイッチ プロファイルを削除します。ピア スイッチが到達可能でない場合は、ローカル スイッチ プロファイルだけが削除されます。
		• **local-config**：スイッチ プロファイルおよびローカル コンフィギュレーションを削除します。

ステップ 3	**exit**
	例：
	`switch(config-sync-sp)# exit`
	`switch#`
		コンフィギュレーション同期モードを終了します。

ステップ 4	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

スイッチ プロファイルからのスイッチの削除

スイッチ プロファイルからスイッチを削除できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**config sync**
	例：
	`switch# config sync`
	`switch(config-sync)#`
		コンフィギュレーション同期モードを開始します。

ステップ 2	**switch-profile** *name*
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)#`
		スイッチプロファイルを設定し、スイッチプロファイルの名前を設定し、スイッチ プロファイル同期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	**no sync-peers destination** *destination IP*
	例：
	`switch(config-sync-sp)# no sync-peers destination 10.1.1.1`
	`switch(config-sync-sp)#`
		スイッチプロファイルから指定のスイッチを削除します。

ステップ 4	**exit**
	例：
	`switch(config-sync-sp)# exit`
	`switch#`
		スイッチ プロファイル コンフィギュレーション モードを終了します。

ステップ 5	（任意） **show switch-profile** 例： `switch# show switch-profile`	スイッチ プロファイル コンフィギュレーションを表示します。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

スイッチ プロファイル バッファの表示

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure sync**	コンフィギュレーション同期モードを開始します。
ステップ 2	switch(config-sync) # **switch-profile** *profile-name*	指定されたスイッチ プロファイルに対するスイッチプロファイル同期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	switch(config-sync-sp) # **show switch-profile***profile-name* **buffer**	指定されたインターフェイスに対するインターフェイス スイッチ プロファイル同期コンフィギュレーション モードを開始します。

例

次に、sp という名前のサービス プロファイルのスイッチ プロファイル バッファの表示例を示します。

```
switch# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       vlan 101
1.1       ip igmp snooping querier 10.101.1.1
2       mac address-table static 0000.0000.0001 vlan 101 drop
3       interface Ethernet1/2
3.1       switchport mode trunk
3.2       switchport trunk allowed vlan 101

switch(config-sync-sp)# buffer-move 3 1
switch(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       interface Ethernet1/2
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 101
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop
switch(config-sync-sp)#
```

スイッチのリブート後のコンフィギュレーションの同期化

スイッチ プロファイルを使用してピア スイッチで新しい設定をコミット中に Cisco Nexus シリーズ スイッチがリブートする場合、リロード後にピア スイッチを同期するには、次の手順を実行します。

手順

ステップ 1 リブート中にピア スイッチ上で変更された設定を再適用します。
ステップ 2 **commit** コマンドを入力します。
ステップ 3 設定が正しく適用されており、両方のピアが同期されていることを確認します。

例

スイッチ プロファイル設定の show コマンド

次の **show** コマンドは、スイッチ プロファイルに関する情報を表示します。

コマンド	目的
show switch-profile *name*	スイッチ プロファイル中のコマンドを表示します。
show switch-profile *name* **buffer**	スイッチ プロファイル中のコミットされていないコマンド、移動されたコマンド、削除されたコマンドを表示します。
show switch-profile *name* **peer** *IP-address*	ピア スイッチの同期ステータスが表示されます。
show switch-profile *name* **session-history**	最後の 20 のスイッチ プロファイル セッションのステータスを表示します。
show switch-profile *name* **status**	ピア スイッチのコンフィギュレーション同期ステータスを表示します。
show running-config exclude-provision	オフラインで事前プロビジョニングされた非表示のインターフェイスの設定を表示します。
show running-config switch-profile	ローカル スイッチのスイッチ プロファイルの実行コンフィギュレーションを表示します。

コマンド	目的
show startup-config switch-profile	ローカル スイッチのスイッチ プロファイルのスタートアップ コンフィギュレーションを表示します。

これらのコマンドの出力フィールドの詳細については、ご使用のプラットフォームの、システム管理コマンドのリファレンスを参照してください。

サポートされているスイッチ プロファイル コマンド

以下のスイッチ プロファイル コマンドがサポートされています。

- **logging event link-status default**
- [**no**] **vlan** *vlan-range*
- **ip access-list** *acl-name*
- **policy-map type network-qos jumbo-frames**
  - **class type network-qos class-default**
  - **mtu** *mtu value*
- **system qos**
  - **service-policy type network-qos jumbo-frames**
- **vlan configuration** *vlan id*
  - **ip igmp snooping querier** *ip*
- **spanning-tree port type edge default**
- **spanning-tree port type edge bpduguard default**
- **spanning-tree loopguard default**
- **no spanning-tree vlan** *vlan id*
- **port-channel load-balance ethernet source-dest-port**
- **interface port-channel** *number*
  - **description** *text*
  - **switchport mode trunk**
  - **switchport trunk allowed vlan** *vlan list*
  - **spanning-tree port type network**
  - **no negotiate auto**
  - **vpc peer-link**

- **interface port-channel** *number*
  - **switchport access vlan** *vlan id*
  - **spanning-tree port type edge**
  - **speed 10000**
  - **vpc** *number*
- **interface ethernet***x/y*
  - **switchport access vlan** *vlanid*
  - **spanning-tree port type edge**
  - **channel-group** *number* **mode active**
- **service dhcp**
- **ip dhcp relay**
- i**pv6 dhcp relay**
- **storm-control unicast level**

スイッチ プロファイルの設定例

ローカルおよびピア スイッチでのスイッチ プロファイルの作成例

次に、ローカルおよびピア スイッチで正常にスイッチ プロファイル設定を作成する例を示します。。

手順

	コマンドまたはアクション	目的

ステップ 1	ローカルおよびピア スイッチで CFSoIP 配信をイネーブルにします。
	例：
	`switch# configuration terminal`
	`switch(config)# cfs ipv4 distribute`	

ステップ 2	ローカルおよびピア スイッチでスイッチ プロファイルを作成します。
	例：
	`switch(config-sync)# switch-profile abc`
	`switch(config-sync-sp)# sync-peers destination 10.1.1.1`	

	コマンドまたはアクション	目的

ステップ 3	スイッチ プロファイルが、ローカルおよびピア スイッチで同じであることを確認します。
	例：
	`switch(config-sync-sp)# show switch-profile abc status`
	`Start-time: 15801 usecs after Mon Aug 23 06:21:08 2010`
	`End-time: 6480 usecs after Mon Aug 23 06:21:13 2010`
	`Profile-Revision: 1`
	`Session-type: Initial-Exchange`
	`Peer-triggered: Yes`
	`Profile-status: Sync Success`
	`Local information:`
	`----------`
	`Status: Commit Success`
	`Error(s):`
	`Peer information:`
	`----------`
	`IP-address: 10.1.1.1`
	`Sync-status: In Sync.`
	`Status: Commit Success`
	`Error(s):`	

ステップ 4	ローカル スイッチでスイッチ プロファイルにコンフィギュレーション コマンドを追加します。コマンドがコミットされたときに、コマンドがピア スイッチに適用されます。 **例**： `switch(config-sync-sp)# class-map type qos c1`	

ステップ 5	スイッチ プロファイルのコマンドを検証します。
	例：
	`switch(config-sync-sp-if)# verify`
	`Verification Successful`	

ステップ 6	スイッチ プロファイルにコマンドを適用し、ローカルとピア スイッチ間の設定を同期させます。 **例**：	

	コマンドまたはアクション	目的
	switch(config-sync-sp)# **commit** Commit Successful switch(config-sync)#	

同期ステータスの確認例

次に、ローカルとピア スイッチ間の同期ステータスを確認する例を示します。

```
switch(config-sync)# show switch-profile switch-profile status
Start-time: 804935 usecs after Mon Aug 23 06:41:10 2010
End-time: 956631 usecs after Mon Aug 23 06:41:20 2010

Profile-Revision: 2
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.1.1.1
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch(config-sync)#
```

実行コンフィギュレーションの表示

次に、ローカル スイッチでスイッチ プロファイルの実行コンフィギュレーションを表示する例を示します。

```
switch# configure sync
switch(config-sync)# show running-config switch-profile

switch(config-sync)#
```

ローカル スイッチとピア スイッチ間のスイッチ プロファイルの同期の表示

次に、2 台のピア スイッチの同期ステータスを表示する例を示します。

```
switch1# show switch-profile sp status

Start-time: 491815 usecs after Thu Aug 12 11:54:51 2010
End-time: 449475 usecs after Thu Aug 12 11:54:58 2010

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: No
```

```
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch1#


switch2# show switch-profile sp status

Start-time: 503194 usecs after Thu Aug 12 11:54:51 2010
End-time: 532989 usecs after Thu Aug 12 11:54:58 2010

Profile-Revision: 1
Session-type: Initial-Exchange
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch2#
```

ローカル スイッチとピア スイッチでの確認とコミットの表示

次に、ローカル スイッチおよびピア スイッチで正常に確認とコミットを設定する例を示します。

```
switch1# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# interface ethernet1/1
switch1(config-sync-sp-if)# description foo
switch1(config-sync-sp-if)# verify
Verification Successful
switch1(config-sync-sp)# commit
Commit Successful
switch1(config-sync)# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.52
  interface Ethernet1/1
    description foo
switch1(config-sync)# show switch-profile sp status
```

```
Start-time: 171513 usecs after Wed Aug 11 17:51:28 2010
End-time: 676451 usecs after Wed Aug 11 17:51:43 2010

Profile-Revision: 3
Session-type: Commit
Peer-triggered: No
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.52
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch1(config-sync)#


switch2# show running-config switch-profile
switch-profile sp
  sync-peers destination 10.193.194.51
  interface Ethernet1/1
    description foo
switch2# show switch-profile sp status

Start-time: 265716 usecs after Wed Aug 11 16:51:28 2010
End-time: 734702 usecs after Wed Aug 11 16:51:43 2010

Profile-Revision: 3
Session-type: Commit
Peer-triggered: Yes
Profile-status: Sync Success

Local information:
----------------
Status: Commit Success
Error(s):

Peer information:
----------------
IP-address: 10.193.194.51
Sync-status: In Sync.
Status: Commit Success
Error(s):

switch2#
```

同期の成功と失敗の例

次に、ピア スイッチにおけるスイッチ プロファイルの同期の成功例を示します。

```
switch# show switch-profile abc peer

switch# show switch-profile sp peer 10.193.194.52
Peer-sync-status          : In Sync.
Peer-status               : Commit Success
```

```
Peer-error(s)              :
switch1#
```

次に、到達不能ステータスのピアを使用した、ピア スイッチでのスイッチ プロファイルの同期の失敗例を示します。

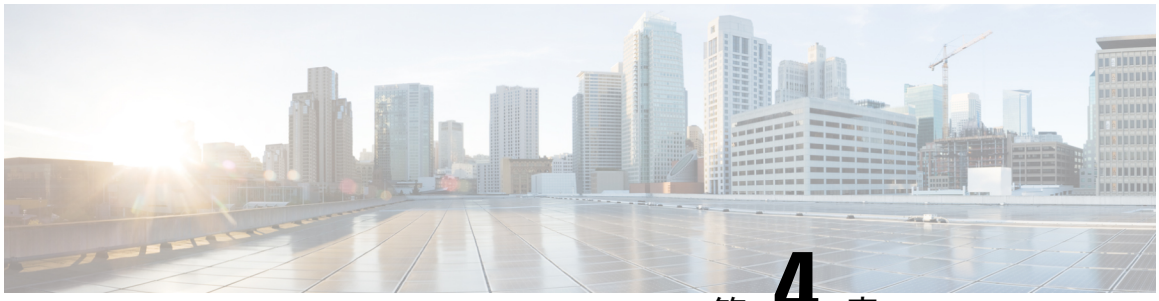
```
switch# show switch-profile sp peer 10.193.194.52
Peer-sync-status           : Not yet merged. pending-merge:1 received_merge:0
Peer-status                : Peer not reachable
Peer-error(s)              :
switch#
```

スイッチ プロファイル バッファの設定、バッファ移動、およびバッファの削除

次に、スイッチ プロファイル バッファの設定、バッファ移動、バッファ削除を設定する例を示します。

```
switch# configure sync
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# vlan 101
switch(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch(config-sync-sp-vlan)# exit
switch(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
switch(config-sync-sp)# interface ethernet1/2
switch(config-sync-sp-if)# switchport mode trunk
switch(config-sync-sp-if)# switchport trunk allowed vlan 101
switch(config-sync-sp-if)# exit
switch(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       vlan 101
1.1       ip igmp snooping querier 10.101.1.1
2       mac address-table static 0000.0000.0001 vlan 101 drop
3       interface Ethernet1/2
3.1       switchport mode trunk
3.2       switchport trunk allowed vlan 101

switch(config-sync-sp)# buffer-move 3 1
switch(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       interface Ethernet1/2
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 101
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch(config-sync-sp)# buffer-delete 1
switch(config-sync-sp)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
2       vlan 101
2.1       ip igmp snooping querier 10.101.1.1
```

```
3       mac address-table static 0000.0000.0001 vlan 101 drop

switch(config-sync-sp)# buffer-delete all
switch(config-sync-sp)# show switch-profile sp buffer
switch(config-sync-sp)#
```



第 4 章

CFS の使用

この章は、次の項で構成されています。

CFS について

Cisco Nexus シリーズ スイッチの一部の機能は、正常に動作するため、ネットワーク内の他のスイッチとの設定の同期化を必要とします。ネットワーク内のスイッチごとに手動設定によって同期化を行うことは、面倒で、エラーが発生しやすくなります。

CFS はネットワーク内の自動設定同期化に対して共通のインフラストラクチャを提供します。また、トランスポート機能、および機能に対する共通サービスのセットを提供します。CFS にはネットワーク内の CFS 対応スイッチを検出し、すべての CFS 対応スイッチの機能能力を検出する機能が備わっています。

Cisco Nexus シリーズ スイッチは、IPv4 または IPv6 ネットワークを介した CFS メッセージ配信をサポートします。

CFS には次の機能があります。

- CFS レイヤでクライアント/サーバー関係を持たないピアツーピア プロトコル。
- IPv4 ネットワークを介した CFS メッセージ配信。
- 3 つの配信モード。
  - 協調型配信：ネットワーク内で同時に 1 つの配信だけが許可されます。
  - 非協調型配信：協調型配信が進行中である場合を除いて、ネットワーク内で複数の同時配信を実行できます。

- 無制限の非協調型配信：既存の協調型配信がある場合でも、ネットワーク内で複数の同時配信が許可されます。無制限の非協調型配信は、他のすべての配信タイプの配信と同時に実行できます。

IP を介した CFS 配信では、次の機能がサポートされます。

- IP ネットワークを介した配信の 1 つの範囲：
  - 物理範囲：IP ネットワーク全体に配信されます。

CFS 配信

CFS 配信機能は、下位層の転送とは無関係です。Cisco Nexus シリーズ スイッチは IP を介した CFS 配信をサポートします。CFS を使用する機能は、下位層の転送を認識しません。

CFS の配信モード

CFS では異なる機能要件をサポートするために、3 つの配信モードをサポートします。

- 非協調型配信
- 協調型配信
- 無制限の非協調型配信

常に 1 つのモードだけを適用できます。

非協調型配信

非協調型配信は、ピアからの情報と競合させたくない情報を配信する場合に使用されます。1 つの機能に対して非協調的な並列配信を適用できます。

協調型配信

協調型配信は、いかなる時も 1 つの機能配信だけ適用できます。CFS は、ロックを使用してこの機能を強制します。ネットワーク内のいずれかの機能でロックが取得されていると、協調型配信は開始できません。協調型配信は、次の 3 段階で構成されています。

- ネットワーク ロックが取得されます。
- 設定が配信され、コミットされます。
- ネットワーク ロックが解除されます。

協調型配信には、次の 2 種類があります。

- CFS によるもの：機能が介在することなく、機能要求に応じて CFS が各段階を実行します。

• 機能によるもの：各段階は機能によって完全に管理されます。

協調型配信は、複数のスイッチから操作および配信が可能な情報を配信するのに使用されます。たとえば、ポート セキュリティの設定です。

無制限の非協調型配信

無制限の非協調型配信では、既存の協調型配信がある場合にネットワーク内で複数の同時配信が許可されます。無制限の非協調型配信は、他のすべての配信タイプの配信と同時に実行できます。

CFS 配信ステータスの確認

show cfs status コマンドを実行すると、スイッチの CFS 配信ステータスが表示されます。

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Enabled - mode IPv4
IPv4 multicast address : 239.255.70.83

Distribution over Ethernet : Enabled
```

アプリケーションの CFS サポート

CFS のアプリケーション要件

ネットワーク内のすべてのスイッチが CFS に対応している必要があります。CFS に対応していないスイッチは配信を受信できないため、ネットワークの一部が意図された配信を受信できなくなります。CFS には、次の要件があります。

- CFS の暗黙的な使用：CFS 対応アプリケーションの CFS 作業を初めて行う場合、設定変更プロセスが開始され、アプリケーションがネットワークをロックします。
- 保留データベース：保留データベースはコミットされていない情報を保持する一時的なバッファです。データベースが、ネットワーク内の他のスイッチのデータベースと確実に同期するために、コミットされていない変更はすぐには適用されません。変更をコミットすると、保留データベースはコンフィギュレーション データベース（別名、アクティブ データベースまたは有効データベース）を上書きします。
- アプリケーション単位でイネーブル化またはディセーブル化される CFS 配信：CFS 配信ステートのデフォルト（イネーブルまたはディセーブル）は、アプリケーション間で異なります。アプリケーションで CFS の配信がディセーブルにされている場合、そのアプリケーションは設定を配信せず、またネットワーク内のその他のスイッチからの配信も受け入れません。
- 明示的な CFS コミット：大半のアプリケーションでは、新しいデータベースをネットワークに配信したりネットワーク ロックを解除したりするために、一時的なバッファ内の変更

をアプリケーションデータベースにコピーする明示的なコミット操作が必要です。コミット操作を実行しないと、一時的バッファ内の変更は適用されません。

アプリケーションの CFS のイネーブル化

すべての CFS ベースのアプリケーションでは、配信機能をイネーブルまたはディセーブルにできます。

アプリケーションでは、配信はデフォルトでイネーブルにされています。

アプリケーションで配信が明示的にイネーブルにされていない場合は、CFS はそのアプリケーションの設定を配信しません。

アプリケーション登録スターテスの確認

show cfs application コマンドは、CFS に現在登録されているアプリケーションを表示します。最初のカラムには、アプリケーション名が表示されます。2 番めのカラムは、アプリケーションの配信がイネーブルであるかディセーブルであるかを示します（enabled または disabled）。最後のカラムは、アプリケーションの配信範囲を示します（論理、物理、またはその両方）。

Note **show cfs application** コマンドは、CFS に登録されているアプリケーションを表示するだけです。CFS を使用するコンディショナル サービスは、これらのサービスが稼働していなければ出力には示されません。

```
switch# show cfs application

----------------------------------------------
 Application    Enabled   Scope
----------------------------------------------
 ntp            No        Physical-all
 fscm           Yes       Physical-fc
 rscn           No        Logical
 fctimer        No        Physical-fc
 syslogd        No        Physical-all
 callhome       No        Physical-all
 fcdomain       Yes       Logical
 device-alias   Yes       Physical-fc
Total number of entries = 8
```

show cfs application name コマンドは、特定のアプリケーションの詳細を表示します。表示されるのは、イネーブル/ディセーブル ステート、CFS に登録されているタイムアウト、結合可能であるか（結合のサポートに対して CFS に登録されているか）、および配信範囲です。

```
switch# show cfs application name fscm
```

```
Enabled         : Yes
 Timeout        : 100s
 Merge Capable  : No
 Scope          : Physical-fc
```

ネットワークのロック

CFSインフラストラクチャを使用する機能（アプリケーション）を初めて設定する場合、この機能はCFS セッションを開始して、ネットワークをロックします。ネットワークがロックされた場合、スイッチソフトウェアでは、ロックを保持しているスイッチからのみこの機能への設定変更を行うことができます。別のスイッチから機能への設定変更を行う場合、ロックされているステータスを知らせるメッセージが、スイッチから発行されます。設定変更は、該当アプリケーションによって保留データベースに保持されます。

ネットワーク ロックを要求する CFS セッションを開始し、セッションを終了するのを忘れた場合は、管理者がそのセッションをクリアできます。いつでもネットワークをロックした場合、ユーザ名は再起動およびスイッチオーバーを行っても保持されます。（同じマシン上で）別のユーザーが設定タスクを実行しようとしても、拒否されます。

CFS ロック ステータスの確認

show cfs lock コマンドを実行すると、アプリケーションによって現在取得されているすべてのロックが表示されます。このコマンドにより、アプリケーションごとにアプリケーション名とロックの取得範囲が表示されます。

show cfs lock name コマンドは、指定したアプリケーションで使用されているロックの詳細情報を表示します。

変更のコミット

コミット操作により、すべてのアプリケーション ピアの保留データベースを保存し、すべてのスイッチのロックを解除します。

コミット機能はセッションを開始しません。セッションを開始するのは、ロック機能だけです。ただし、設定変更がこれまでに行われていなければ、空のコミットが可能です。この場合、コミット操作の結果として、ロックを取得し、現在のデータベースを配信するセッションが行われます。

CFSインフラストラクチャを使用して機能への設定変更をコミットすると、次のいずれかの応答に関する通知が届きます。

- 1 つまたは複数の外部スイッチが正常なステータスを報告する場合：アプリケーションは変更をローカルに適用し、ネットワーク ロックを解除します。

- どの外部スイッチも成功ステートを報告しない場合：アプリケーションはこのステートを失敗として認識し、ネットワーク内のどのスイッチにも変更を適用しません。ネットワーク ロックは解除されません。

commit コマンドを入力すると、指定した機能の変更をコミットできます。

変更の破棄

設定変更を廃棄すると、アプリケーションは保留中のデータベースを消去し、ネットワーク内のロックを解除します。中断およびコミット機能の両方を使用できるのは、ネットワーク ロックが取得されたスイッチだけです。

指定した機能に対して **abort** コマンドを使用すると、その機能の変更を廃棄できます。

設定の保存

まだ適用されていない変更内容（保留データベースにまだ存在する）は実行コンフィギュレーションには表示されません。変更をコミットすると、保留データベース内の設定変更が有効データベース内の設定を上書きします。

Caution 変更内容は、コミットしなければ、実行コンフィギュレーションに保存されません。

ロック済みセッションのクリア

ネットワーク内の任意のスイッチからアプリケーションが保持しているロックをクリアすると、ロックが取得されているにもかかわらず解除されていない状態から回復できます。この機能には、Admin 権限が必要になります。

Caution この機能を使用してネットワーク内のロックを解除する場合は、注意が必要です。ネットワーク内の任意のスイッチの保留中設定がフラッシュされ、内容が失われます。

CFS リージョン

CFS リージョンの概要

CFS リージョンは、物理配信範囲の所定の機能またはアプリケーションに対するスイッチのユーザー定義のサブセットです。ネットワークが広い範囲に及ぶ場合、場合によっては、物理的なプロキシミティに基づき、スイッチセット間での特定のプロファイルの配信を局所化または制限する必要があります。CFS リージョンを使用すると、ネットワーク内で特定の CFS 機

能またはアプリケーションに、配信の複数アイランドができます。CFS リージョンは、機能設定の配信をネットワーク内のスイッチの特定のセットまたはグループに制限するよう設計されています。

Note CFS リージョンの設定は、物理スイッチだけで行えます。CFS リージョンの設定は、VSAN では行えません。

シナリオ例

Smart Call Home アプリケーションは、困難な状況、あるいは異常が発生した時にネットワーク管理者にアラートを送信します。ネットワークが広い地域に及び、複数のネットワーク管理者がネットワーク内のスイッチの各サブセットを担当している場合は、Smart Call Home アプリケーションは、場所に関係なく、すべてのネットワーク管理者にアラートを送信します。Smart Call Home アプリケーションで、選択したネットワーク管理者にメッセージ アラートを送信するには、アプリケーションの物理範囲を微調整するか、絞り込む必要があります。CFS リージョンの実装によって、このシナリオを実現できます。

CFS リージョンは、0 ～ 200 の数字で識別されます。リージョン 0 はデフォルト リージョンとして予約されており、ネットワーク内のすべてのスイッチを含みます。1 ～ 200 のリージョンを設定できます。デフォルト リージョンでは下位互換性を維持しています。

機能が移動される、つまり、機能が新しいリージョンに割り当てられると、機能のスコープはそのリージョンに制限されます。他のすべてのリージョンは、配信やマージの対象から外されます。機能へのリージョンの割り当ては、配信において初期の物理スコープよりも優先されます。

複数の機能の設定を配信するように CFS リージョンを設定できます。ただし、特定のスイッチでは、一度に特定の機能設定を配信するように設定できる CFS リージョンは 1 つだけです。機能を CFS リージョンに割り当てた場合、この設定を別の CFS リージョン内に配信できません。

CFS リージョンの管理

CFS リージョンの作成

CFS リージョンを作成できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **cfs region** *region-id*	リージョンを作成します。

CFS リージョンへのアプリケーションの割り当て

スイッチでリージョンにアプリケーションを割り当てることができます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **cfs region** *region-id*	リージョンを作成します。
ステップ 3	switch(config-cfs-region)# *application*	リージョンにアプリケーションを追加します。 **Note** リージョンにスイッチ上の任意の数のアプリケーションを追加できます。同じリージョンにアプリケーションを複数回追加しようとすると、「Application already present in the same region」というエラー メッセージが表示されます。

Example

次に、リージョンにアプリケーションを割り当てる例を示します。

```
switch# configure terminal
switch(config)# cfs region 1
switch(config-cfs-region)# ntp
switch(config-cfs-region)# callhome
```

別の CFS リージョンへのアプリケーションの移動

あるリージョンから別のリージョンにアプリケーションを移動できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **cfs region** *region-id*	CFS リージョン サブモードを開始します。

	Command or Action	Purpose
ステップ 3	switch(config-cfs-region)# *application*	あるリージョンから別のリージョンに移動するアプリケーションを示します。 **Note** 同じリージョンにアプリケーションを複数回移動しようとすると、「Application already present in the same region」というエラー メッセージが表示されます。

Example

次に、リージョン 1 に割り当てられていたアプリケーションをリージョン 2 に移動する例を示します。

```
switch# configure terminal
switch(config)# cfs region 2
switch(config-cfs-region)# ntp
```

リージョンからのアプリケーションの削除

リージョンからのアプリケーションの削除は、アプリケーションをデフォルト リージョン（リージョン 0）に戻す場合と同じです。これによって、ネットワーク全体がアプリケーションの配信の範囲になります。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **cfs region** *region-id*	CFS リージョン サブモードを開始します。
ステップ 3	switch(config-cfs-region)# **no** *application*	リージョンに属しているアプリケーションを削除します。

CFS リージョンの削除

リージョンの削除とは、リージョン定義を無効にすることです。リージョンを削除すると、リージョンによってバインドされているすべてのアプリケーションがデフォルト リージョンに戻ります。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **no cfs region** *region-id*	リージョンを削除します。 **Note** 「All the applications in the region will be moved to the default region」という警告が表示されます。

IP を介した CFS の設定

IPv4 を介した CFS のイネーブル化

IPv4 を介した CFS をイネーブルまたはディセーブルにできます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **cfs ipv4 distribute**	スイッチのすべてのアプリケーションに対して IPv4 を介した CFS をグローバルでイネーブルにします。
ステップ 3	(Optional) switch(config)# **no cfs ipv4 distribute**	スイッチの IPv4 を介した CFS をディセーブルにします（デフォルト）。

IP を介した CFS 設定の確認

次に、IP を介した CFS 設定を確認する例を示します。

```
switch# show cfs status
Distribution : Enabled
Distribution over IP : Enabled - mode IPv4
IPv4 multicast address : 239.255.70.83
```

IP を介した CFS の IP マルチキャスト アドレスの設定

類似のマルチキャスト アドレスを持つ IP を介した CFS 対応スイッチのすべては、IP ネットワークを介した 1 つの CFS を形成します。ネットワーク トポロジ変更を検出するためのキープアライブ メカニズムのような CFS プロトコル特有の配信は、IP マルチキャスト アドレスを使用して情報を送受信します。

Note アプリケーション データの CFS 配信はダイレクト ユニキャストを使用します。

CFS の IPv4 マルチキャスト アドレスの設定

IP を介した CFS の IPv4 のマルチキャスト アドレス値を設定できます。デフォルトの IPv4 マルチキャスト アドレスは 239.255.70.83 です。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **cfs ipv4 mcast-address** *ipv4-address*	IPv4 を介した CFS 配信の IPv4 マルチキャスト アドレスを設定します。有効な IPv4 アドレスの範囲は 239.255.0.0 ～ 239.255.255.255 および 239.192/16 ～ 239.251/16 です。
ステップ 3	(Optional) switch(config)# **no cfs ipv4 mcast-address** *ipv4-address*	IPv4 を介した CFS 配信の デフォルトの IPv4 マルチキャスト アドレスに戻します。CFS のデフォルトの IPv4 マルチキャスト アドレスは 239.255.70.83 です。

IP を介した CFS の IP マルチキャスト アドレス設定の確認

次に、CFS over IP の IP マルチキャスト アドレス設定を確認する例を示します。

```
switch# show cfs status
Fabric distribution Enabled
IP distribution Enabled mode ipv4
IPv4 multicast address : 10.1.10.100
```

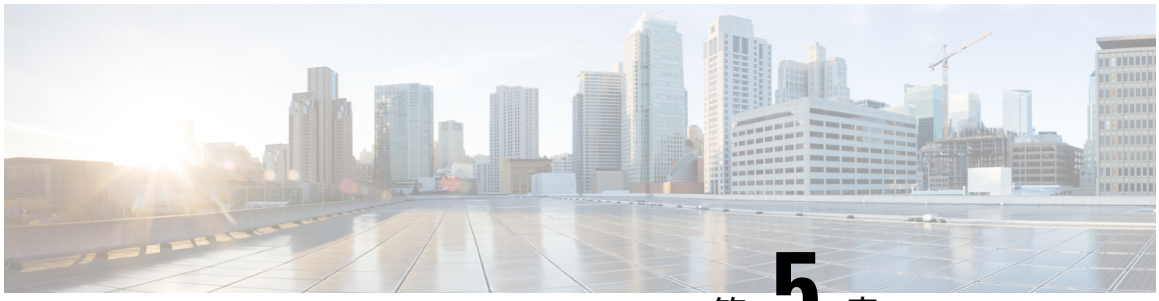
CFS のデフォルト設定

次の表に、CFS のデフォルト設定を示します。

Table 1: デフォルトの *CFS* パラメータ

パラメータ	デフォルト
スイッチでの CFS 配信	イネーブル
データベース変更	最初の設定変更によって暗黙的にイネーブル化
アプリケーションの配信	アプリケーションごとに異なる
コミット	明示的な設定が必要
IP を介した CFS	ディセーブル
IPv4 マルチキャストアドレス	239.255.70.83

CISCO-CFS-MIB には CFS 関連機能の SNMP 設定情報が含まれます。ご使用のプラットフォームの MIB リファレンスを参照してください。



第 5 章

PTP の設定

この章は、次の項で構成されています。

PTP に関する情報

PTPはネットワークに分散したノードの時刻同期プロトコルです。そのハードウェアのタイムスタンプ機能は、ネットワーク タイム プロトコル（NTP）などの他の時刻同期プロトコルよりも高い精度を実現します。

PTP システムは、PTP および非 PTP デバイスの組み合わせで構成できます。PTP デバイスには、オーディナリ クロック、境界クロック、およびトランスペアレント クロックが含まれます。非PTPデバイスには、通常のネットワーク スイッチやルータなどのインフラストラクチャ デバイスが含まれます。

PTPは、システムのリアルタイムPTPクロックが相互に同期する方法を指定する分散プロトコルです。これらのクロックは、グランドマスタークロック（階層の最上部にあるクロック）を持つマスター/スレーブ同期階層に編成され、システム全体の時間基準を決定します。同期は、タイミング情報を使用して階層のマスターの時刻にクロックを調整するメンバーと、PTP タイミング メッセージを交換することによって実現されます。PTPは、PTP ドメインと呼ばれる論理範囲内で動作します。

PTP は Cisco Nexus 3100 スイッチのリリース 6.0(2)U3(1) から 7.0(3)I2(4) でサポートされていません。ただし、PTP は Cisco Nexus 3100 スイッチのリリース 7.0(3)I4(1) 以上ではサポートされています。

PTP デバイス タイプ

次のクロックは、一般的な PTP デバイスです。

オーディナリ クロック

エンド ホストと同様に、単一の物理ポートに基づいてネットワークと通信します。オーディナリ クロックはグランドマスター クロックとして動作できます。

境界クロック

通常、複数の物理ポートがあり、各ポートはオーディナリ クロックのポートのように動作します。ただし、各ポートはローカル クロックを共有し、クロックのデータ セットはすべてのポートに共通です。各ポートは、境界クロックのその他すべてのポートから使用可能な最善のクロックに基づいて、個々の状態を、マスター（それに接続されている他のポートを同期する）またはスレーブ（ダウンストリーム ポートに同期する）に決定します。同期とマスター/スレーブ階層の確立に関するメッセージは、境界クロックのプロトコル エンジンで終了し、転送されません。

トランスペアレント クロック

通常のスイッチやルータなどのすべての PTP メッセージを転送しますが、スイッチでのパケットの滞留時間（パケットがトランスペアレント クロックを通過するために要した時間）と、場合によってはパケットの入力ポートのリンク遅延を測定します。トランスペアレント クロックはグランドマスター クロックに同期する必要がないため、ポートの状態はありません。

次の 2 種類のトランスペアレント クロックがあります。

エンドツーエンド トランスペアレント クロック

PTP メッセージの滞留時間を測定し、PTP メッセージまたは関連付けられたフォローアップ メッセージの修正フィールドの時間を収集します。

ピアツーピア トランスペアレント クロック

PTP メッセージの滞留時間を測定し、各ポートと、リンクを共有する他のノードの同じように装備されたポートとの間のリンク遅延を計算します。パケットの場合、この着信リンクの遅延は、PTP メッセージまたは関連付けられたフォローアップ メッセージの修正フィールドの滞留時間に追加されます。

（注） PTP は境界クロック モードのみで動作します。Grand Master Clock（10 MHz）アップストリームを導入することを推奨します。サーバーには、同期する必要があり、スイッチに接続されたクロックが含まれます。

エンドツーエンド トランスペアレント クロック モードとピアツーピア トランスペアレント クロック モードはサポートされません。

クロック モード

IEEE 1588 規格は、PTP をサポートするデバイスが 1 ステップと 2 ステップで動作するための 2 つのクロックモードを指定しています。

1 ステップ モード：

1 ステップ モードでは、クロック同期メッセージに、マスター ポートがメッセージを送信した時刻が含まれます。ASIC は、同期メッセージがポートを出るときにタイムスタンプを追加します。1 ステップ モードで動作するマスター ポートは、Cisco Nexus 9508-FM-R および 9504-FM-R ファブリック モジュールおよび Cisco Nexus 9636C-R、9636Q-R、および 9636C-RX ライン カードで使用できます。

スレーブ ポートは、同期メッセージの一部として送信されるタイムスタンプを使用します。

2 ステップ モード：

2 ステップ モードでは、同期メッセージがポートを出た時刻は後続のフォローアップ メッセージで送信されます。これは、デフォルトのモードです。

PTP プロセス

PTP プロセスは、マスター/スレーブ階層の確立とクロックの同期の 2 つのフェーズで構成されます。

PTP ドメイン内では、オーディナリ クロックまたは境界クロックの各ポートが、次のプロセスに従ってステートを決定します。

- 受信したすべての（マスター ステートのポートによって発行された）アナウンス メッセージの内容を検査します
- 外部マスターのデータ セット（アナウンス メッセージ内）とローカル クロックで、優先順位、クロック クラス、精度などを比較します
- 自身のステートがマスターまたはスレーブのいずれであるかを決定します

マスター/スレーブ階層が確立されると、クロックは次のように同期されます。

- マスターはスレーブに同期メッセージを送信し、送信された時刻を記録します。
- スレーブは同期メッセージを受信し、受信した時刻を記録します。すべての同期メッセージには、フォローアップ メッセージがあります。同期メッセージの数は、フォローアップ メッセージの数と同じである必要があります。
- スレーブはマスターに遅延要求メッセージを送信し、送信された時刻を記録します。
- マスターは遅延要求メッセージを受信し、受信した時刻を記録します。
- マスターはスレーブに遅延応答メッセージを送信します。遅延要求メッセージの数は、遅延応答メッセージの数と同じある必要があります。

- スレーブは、これらのタイムスタンプを使用して、クロックをマスターの時刻に調整します。

PTP のハイ アベイラビリティ

PTP のステートフル リスタートはサポートされません。

PTP の注意事項および制約事項

- Cisco Nexus 3000 および 3100 シリーズ スイッチでは、PTP クロック修正は 100 ～ 999 ナノ秒までの 3 桁の範囲に収まることが予想されます。
- PTP は境界クロック モードのみで動作します。エンドツーエンド トランスペアレント クロック モードとピアツーピア トランスペアレント クロック モードはサポートされません。
- PTP はユーザーデータグラムプロトコル（UDP）上の転送をサポートします。イーサネット上の転送はサポートされません。
- PTP はマルチキャスト通信だけをサポートします。ネゴシエートされたユニキャスト通信はサポートされません。
- PTP はネットワークごとに 1 つのドメインに制限されます。
- PTP 管理コマンドがサポートされています。
- PTP は、Cisco Nexus 36180YC-R スイッチおよび Cisco Nexus 3636C-R ラインカードでのみ、同期間隔 -2 でサポートされます。より高い同期間隔はサポートされません。
- PTP 対応ポートは、ポート上で PTP をイネーブルにしない場合、PTP パケットを識別せず、これらのパケットにタイムスタンプを適用したり、パケットをリダイレクトしたりしません。
- 1 packet per second（1 pps）入力はサポートされていません。
- IPv6 を介した PTP はサポートされていません。
- Cisco Nexus スイッチは、-2 ～ -5 の同期化ログ間隔を使用して、隣接マスターから同期する必要があります。
- ワンステップ PTP は、Cisco Nexus 3000 および 3500 シリーズ プラットフォーム スイッチではサポートされません。

PTP のデフォルト設定

次の表に、PTP パラメータのデフォルト設定を示します。

表 *2:* デフォルトの *PTP* パラメータ

パラメータ	デフォルト
PTP	ディセーブル
PTP バージョン	2
PTP ドメイン	0
クロックをアドバタイズする場合、PTP プライオリティ 1 値	255
クロックをアドバタイズする場合、PTP プライオリティ 2 値	255
PTP アナウンス間隔	1 ログ秒
PTP 同期間隔	– 2 ログ秒
PTP アナウンス タイムアウト	3 アナウンス間隔
PTP 最小遅延要求間隔	0 ログ秒
PTP VLAN	1

PTP の設定

PTP のグローバルな設定

デバイスでPTPをグローバルにイネーブルまたはディセーブルにできます。また、ネットワーク内のどのクロックがグランドマスターとして選択される優先順位が最も高いかを判別するために、さまざまな PTP クロック パラメータを構成できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # [**no**] **feature ptp**	デバイス上で PTP をイネーブルまたはディセーブルにします。 （注） スイッチの PTP をイネーブルにしても、各インターフェイスの PTP はイネーブルになりません。

	コマンドまたはアクション	目的
ステップ 3	switch(config) # [**no**] **ptp source** *ip-address* [**vrf** *vrf*]	すべての PTP パケットのソース IP アドレスを設定します。 *ip-address* には IPv4 形式を使用できます。
ステップ 4	（任意） switch(config) # [**no**] **ptp domain** *number*	このクロックで使用するドメイン番号を構成します。PTP ドメインを使用すると、1 つのネットワーク上で、複数の独立した PTP クロッキング サブドメインを使用できます。 *number* の範囲は 0 ～ 128 です。
ステップ 5	（任意） switch(config) # [**no**] **ptp priority1** *value*	このクロックをアドバタイズするときに使用する priority1 の値を構成します。この値はベスト マスター クロック選択のデフォルトの基準（クロック品質、クロック クラスなど）を上書きします。低い値が優先されます。 *value* の範囲は 0 ～ 255 です。
ステップ 6	（任意） switch(config) # [**no**] **ptp priority2** *value*	このクロックをアドバタイズするときに使用する priority2 の値を構成します。この値は、デフォルトの基準では同等に一致する 2 台のデバイスのうち、どちらを優先するかを決めるために使用されます。たとえば、priority2 値を使用して、特定のスイッチが他の同等のスイッチよりも優先されるようにすることができます。 *value* の範囲は 0 ～ 255 です。
ステップ 7	（任意） switch(config) # **show ptp brief**	PTP のステータスを表示します。
ステップ 8	（任意） switch(config) # **show ptp clock**	ローカル クロックのプロパティを表示します。
ステップ 9	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、デバイス上でPTPをグローバルに構成し、PTP通信用の送信元IPアドレスを指定し、クロックの優先レベルを構成する例を示します。

```
switch# configure terminal
switch(config)# feature ptp
switch(config)# ptp source 10.10.10.1
switch(config)# ptp priority1 1
switch(config)# ptp priority2 1
switch(config)# show ptp brief
PTP port status
-----------------------
Port State
------- --------------
switch(config)# show ptp clock
PTP Device Type: Boundary clock
Clock Identity : 0:22:55:ff:ff:79:a4:c1
Clock Domain: 0
Number of PTP ports: 0
Priority1 : 1
Priority2 : 1
Clock Quality:
Class : 248
Accuracy : 254
Offset (log variance) : 65535
Offset From Master : 0
Mean Path Delay : 0
Steps removed : 0
Local clock time:Sun Jul 3 14:13:24 2011
switch(config)#
```

インターフェイスでの PTP の設定

PTP をグローバルにイネーブルにしても、デフォルトで、サポートされているすべてのインターフェイス上でイネーブルになりません。PTPインターフェイスは個別にイネーブルに設定する必要があります。

始める前に

スイッチ上でグローバルに PTP をイネーブルにし、PTP 通信の送信元 IP アドレスを設定したことを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **interface ethernet** *slot/port*	PTP をイネーブルにするインターフェイスを指定し、インターフェイス構成モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	switch(config-if) # [**no**] **ptp**	インターフェイスで PTP をイネーブルまたはディセーブルにします。
ステップ 4	（任意） switch(config-if) # [**no**] **ptp announce** { **interval** *log seconds* | **timeout** *count*}	インターフェイス上の PTP アナウンスメッセージ間の間隔またはタイムアウトがインターフェイスで発生する前の PTP 間隔の数を構成します。 PTP アナウンス間隔の範囲は 0 ～ 4 秒で、間隔のタイムアウトの範囲は 2 ～ 10 です。
ステップ 5	（任意） switch(config-if) # [**no**] **ptp delay request minimum interval** *log seconds*	ポートがスレーブ ステートの場合に PTP 遅延要求メッセージ間で許可される最小間隔を構成します。 有効な範囲はログ -1 ～ 6 秒です。ログ（-2）は、1 秒あたり 4 フレームです。
ステップ 6	（任意） switch(config-if) # [**no**] **ptp sync interval** *log seconds*	インターフェイス上の PTP 同期メッセージの送信間隔を構成します。 Cisco Nexus 3000 シリーズ スイッチの PTP 同期間隔の範囲は -6 ログ秒～ 1 秒です。 Cisco Nexus 3548 シリーズ スイッチの PTP 同期間隔の範囲は -3 ログ秒～ 1 秒です。
ステップ 7	（任意） switch(config-if) # [**no**] **ptp vlan** *vlan-id*	PTP をイネーブルにするインターフェイスの VLAN を指定します。インターフェイスの 1 つの VLAN でイネーブルにできるのは、1 つの PTP のみです。 指定できる範囲は 1 ～ 4094 です。
ステップ 8	（任意） switch(config-if) # **show ptp brief**	PTP のステータスを表示します。
ステップ 9	（任意） switch(config-if) # **show ptp port interface** *interface slot/port*	PTP ポートのステータスを表示します。
ステップ 10	（任意） switch(config-if)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、インターフェイス上で PTP を構成し、アナウンス、遅延要求、および同期メッセージの間隔を構成する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 2/1
switch(config-if)# ptp
switch(config-if)# ptp announce interval 3
switch(config-if)# ptp announce timeout 2
switch(config-if)# ptp delay-request minimum interval 4
switch(config-if)# ptp sync interval -1
switch(config-if)# show ptp brief
PTP port status
-----------------------
Port State
------- --------------
Eth2/1 Master
switch(config-if)# show ptp port interface ethernet 2/1
PTP Port Dataset: Eth2/1
Port identity: clock identity: 0:22:55:ff:ff:79:a4:c1
Port identity: port number: 1028
PTP version: 2
Port state: Master
Delay request interval(log mean): 4
Announce receipt time out: 2
Peer mean path delay: 0
Announce interval(log mean): 3
Sync interval(log mean): -1
Delay Mechanism: End to End
Peer delay request interval(log mean): 0
switch(config-if)#
```

マスター ロールの割り当て

マスター ロールを割り当てるには、次の手順を実行します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効にするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
		（注）
		このコマンドを設定した後、Cisco NX-OS リリース 9.3(5) 以降の場合は、

	コマンドまたはアクション	目的
		ステップ 5 に進みます。Cisco NX-OS リリース 9.3(4) 以前の場合は、ステップ 3 に進みます。

ステップ 3	**[no] ptp transport ipv4 ucast master**
	例：
	`switch(config-if)# ptp transport ipv4 ucast master`
	`switch(config-if-ptp-master)#`
		特定のポート（レイヤ3インターフェイス）で PTP マスターをイネーブルにします。マスターサブモードでは、スレーブ IPv4 アドレスを入力できます。

ステップ 4	**slave ipv4** *<IP_address>*
	例：
	`switch-1(config)# interface ethernet 1/1`
	`switch-1(config-if)# ptp transport ipv4 ucast master`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.1`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.2`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.3`
	`switch-1(config-if-ptp-master)# slave ipv4 1.2.3.4`
	`switch-1(config-if-ptp-master)#`
		スレーブ IPv4 アドレスを入力します。マスターごとに最大 64 個の IP アドレスを使用できますが、実際の数は同期間隔の設定に応じて変わります。マスターは、これらのスレーブアドレスにのみ、アナウンス、同期、フォローアップ、および delay_resp を送信します。スレーブ IP が到達可能であることを確認する必要があります。
		（注）
		Cisco NX-OS リリース 9.3 (4) 以前の場合は、これで手順は終了です。

ステップ 5	**[no] ptp**
	例：
	`switch(config-if)# ptp`
	`switch(config-if)#`
		インターフェイスで PTP をイネーブルまたはディセーブルにします。
		（注）
		9.3(5) 以降では、このコマンドは、以下のユニキャスト コンフィギュレーション コマンドをインターフェイスに適用する前に必要です。

ステップ 6	**ptp transmission unicast**
	例：
	`switch(config-if)# ptp transmission unicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 7	**ptp role master**
	例：
	`switch(config-if)# ptp role master`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。
		master：マスタークロックは、インターフェイスの PTP ロールとして割り当てられます。

	コマンドまたはアクション	目的
		（注） このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 8	**ptp slave** *ipv4-address*
	例：
	`switch(config-if)# ptp slave 10.10.10.2`
	`switch(config-if)#`
		インターフェイスの PTP ロールが「master」に設定されている場合に、スレーブ クロックの IP アドレスを設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

スレーブ ロールの割り当て

スレーブ ロールを割り当てるには、次の手順を実行します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**interface ethernet** *slot/port*
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		PTP を有効にするインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
		（注）
		このコマンドを設定した後、Cisco NX-OS リリース 9.3(5) 以降の場合は、ステップ 5 に進みます。Cisco NX-OS リリース 9.3(4) 以前の場合は、ステップ 3 に進みます。

ステップ 3	[**no**] **ptp transport ipv4 ucast slave**
	例：
	`switch(config-if)# ptp transport ipv4 ucast slave`
	`switch(config-if-ptp-slave)#`
		特定のポート（レイヤ3インターフェイス）で PTP スレーブをイネーブルにします。スレーブ サブモードでは、ユーザーはマスター IPv4 アドレスを入力できます。

	コマンドまたはアクション	目的

ステップ 4	**master ipv4** *<IP_address>*
	例：
	`switch-1(config)# interface ethernet 1/1`
	`switch-1(config-if)# ptp transport ipv4 ucast slave`
	`switch-1(config-if-ptp-slave)# master ipv4 4.4.4.1`
	`switch-1(config-if-ptp-slave)# master ipv4 4.4.4.2`
	`switch-1(config-if-ptp-slave)# master ipv4 4.4.4.3`
		マスター IPv4 アドレスを入力します。
		（注）
		Cisco NX-OS リリース9.3 (4) 以前の場合は、これで手順は終了です。

ステップ 5	[**no**] **ptp**
	例：
	`switch(config-if)# ptp`
	`switch(config-if)#`
		インターフェイスで PTP をイネーブルまたはディセーブルにします。
		（注）
		このコマンドは、9.3(5) 以降で、以下のユニキャスト コンフィギュレーション コマンドをインターフェイスに適用する前に必要となるものです。

ステップ 6	**ptp transmission unicast**
	例：
	`switch(config-if)# ptp transmission unicast`
	`switch(config-if)#`
		インターフェイスで使用される PTP 伝送方式を設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 7	**ptp role slave**
	例：
	`switch(config-if)# ptp role slave`
	`switch(config-if)#`
		インターフェイスの PTP ロールを設定します。
		slave：スレーブ クロックがインターフェイスの PTP ロールとして割り当てられます。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

ステップ 8	**ptp master** *ipv4-address*
	例：
	`switch(config-if)# ptp master 10.10.10.1`
	`switch(config-if)#`
		インターフェイスの PTP ロールが「slave」に設定されている場合、マスター クロックの IP アドレスを設定します。
		（注）
		このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

PTP 混合モード

PTP は、接続されたクライアントから受信した **delay_req** メッセージのタイプに基づいて、Cisco Nexus デバイスによって自動的に検出される PTP メッセージを配信するための混合モードをサポートします。このモードでは、スレーブがユニキャスト メッセージで **delay_req** を送信すると、マスターもユニキャスト **delay_resp** メッセージで応答します。

PTP インターフェイスがマスター ステートを維持する設定

この手順では、エンドポイントによってポートがスレーブ ステートに移行するのを防ぐ方法について説明します。

始める前に

- スイッチ上でグローバルに PTP をイネーブルにし、PTP 通信の送信元 IP アドレスを設定したことを確認します。
- PTP をグローバルにイネーブルにしても、デフォルトで、サポートされているすべてのインターフェイス上でイネーブルになりません。PTP インターフェイスは個別にイネーブルに設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	switch # **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。
ステップ 2	switch(config) # **interface ethernet** *slot/port*	PTP をイネーブルにするインターフェイスを指定し、インターフェイス構成モードを開始します。
ステップ 3	switch(config-if) # **ptp**	インターフェイスで PTP をイネーブルまたはディセーブルにします。 （注） このコマンドを設定した後、Cisco NX-OS リリース 9.3(5) 以降の場合は、ステップ 5 に進みます。Cisco NX-OS リリース 9.3(4) 以前の場合は、ステップ 4 に進みます。
ステップ 4	switch(config-if) # **ptp multicast master-only**	マスター ステートを維持するようにポートを設定します。 （注）

	コマンドまたはアクション	目的
		このコマンドは、Cisco NX-OS リリース 9.3(4) 以前でサポートされています。Cisco NX-OS リリース 9.3(5) 以降では廃止されています。 Cisco NX-OS リリース 9.3 (4) 以前の場合は、これで手順は終了です。
ステップ 5	**ptp role master**	マスターステートを維持するようにポートを設定します。 （注） このコマンドは、Cisco NX-OS リリース 9.3(5) 以降でサポートされます。

例

この例では、インターフェイス上に PTP を設定し、インターフェイスがマスター ステートを維持するように設定する方法を示しています。

```
switch(config)# show ptp brief

PTP port status
-----------------------------------
Port                   State
----------------  ----------------
Eth1/1               Slave
switch(config)# interface ethernet 1/1
switch(config-if)# ptp multicast master-only
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_GM_CHANGE: Grandmaster clock has changed
 from 60:73:5c:ff:fe:62:a1:41 to 58:97:bd:ff:fe:0d:54:01 for the PTP protocol
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
 PTP_BMC_STATE_SLAVE to PTP_BMC_STATE_PRE_MASTER
2001 Jan  7 07:50:03 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_TIMESYNC_LOST: Lost sync with  master clock
2001 Jan  7 07:50:07 A3-MTC-CR-1 %$ VDC-1 %$ %PTP-2-PTP_STATE_CHANGE: Interface Eth1/1 change from
 PTP_BMC_STATE_PRE_MASTER to PTP_BMC_STATE_MASTER
```

平均パス遅延のしきい値の設定

平均パス遅延は、マスターおよびスレーブ間を移動するために PTP フレームが使用する最新の既知の良好な値です。超過すると Syslog メッセージをトリガーするしきい値を設定することができます。デフォルト値は、1 ナノ秒です。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	switch(config) # [**no**] **feature ptp**	デバイス上で PTP をイネーブルまたはディセーブルにします。 （注） スイッチの PTP をイネーブルにしても、各インターフェイスの PTP はイネーブルになりません。

ステップ 3	switch(config) # **ptp mean-path-delay** *threshold-value*
	例：
	`switch(config)# ptp mean-path-delay 20`
	`switch(config)# 2018 Jun 18 11:17:23 3548-XL-1`
	`%PTP-2-PTP_HIGH_MEAN_PATH_DELAY: PTP mean-path-delay 31 exceeds the threshold. Discarding the value.`
		Syslog メッセージをトリガーするしきい値の時間をナノ秒単位で指定します。
		平均パス遅延の *threshold-value* の範囲は 10 ～ 1000000000 です。
		デフォルト値は、1000000000 ナノ秒です。

例

次の例では、過去のいくつかの PTP 修正と、それらの平均パス遅延の情報を示します。

```
switch(config)# show ptp corrections
PTP past corrections
--------------------------------------------------------------------------------
Slave Port              SUP Time               Correction(ns)    MeanPath Delay(ns)
----------  -------------------------------  ------------------  ------------------
Eth1/2        Fri Dec 15 03:36:33 2017 226753    7                   36
Eth1/2        Fri Dec 15 03:36:32 2017 975282    -1                  36
Eth1/2        Fri Dec 15 03:36:32 2017 723901    0                   36
Eth1/2        Fri Dec 15 03:36:32 2017 472521    0                   36
Eth1/2        Fri Dec 15 03:36:32 2017 222255    -1                  38
Eth1/2        Fri Dec 15 03:36:31 2017 971076    -2                  38
Eth1/2        Fri Dec 15 03:36:31 2017 719685    -8                  38
Eth1/2        Fri Dec 15 03:36:31 2017 468215    15                  38
Eth1/2        Fri Dec 15 03:36:31 2017 217020    -2                  35
Eth1/2        Fri Dec 15 03:36:30 2017 965528    3                   35
Eth1/2        Fri Dec 15 03:36:30 2017 714151    -4                  35
Eth1/2        Fri Dec 15 03:36:30 2017 462905    0                   35
Eth1/2        Fri Dec 15 03:36:30 2017 212015    -1                  39
Eth1/2        Fri Dec 15 03:36:29 2017 960621    -2                  39
Eth1/2        Fri Dec 15 03:36:29 2017 709293    0                   39
Eth1/2        Fri Dec 15 03:36:29 2017 457782    5                   39
Eth1/2        Fri Dec 15 03:36:29 2017 206421    1                   36
Eth1/2        Fri Dec 15 03:36:28 2017 954986    1                   36
```

次の例では、設定されている平均パス遅延の値が表示されます。

```
switch(config)# show run all | grep mean-path-delay
ptp mean-path-delay 1000000000
```

タイムスタンプ タギング

タイムスタンプ タギング機能は、リモート デバイスでパケットが到達したときに正確な時間情報を提供し、実際の時間を追跡できるようにします。パケットは、PTP を使用してナノ秒の精度で切り捨てられ、タイムスタンプが付けられます。Cisco Nexus Data Broker とともにスイッチの TAP 集約機能を使用すると、SPAN を使用してネットワーク トラフィックをコピーし、トラフィックをフィルタリングしてタイムスタンプを付け、記録および分析のために送信できます。

インターフェイスで**ttag**を構成すると、すべての着信トラフィックがタグ付けされます。インターフェイスで **ttag-strip** を構成すると、ttag を持つすべての発信トラフィックが削除されます。

タイムスタンプ タギングの設定

（注） 9636C-R、9636C-RX、および 9636Q-R ライン カードを搭載した Cisco Nexus 9508 スイッチでは、タイムスタンプ タギングの設定はサポートされていません。

始める前に

PTP オフロードがグローバルに有効になっていることを確認します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		指定したインターフェイスに対してインターフェイス 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	[**no**] **ttag** 例： `switch(config-if)# ttag`	レイヤ 2 またはレイヤ 3 出力インターフェイスでタイムスタンプ タギングを設定します。これは、スイッチの出力時にタグ付けする必要があるトラフィックの入力ポートで必要です。これは、出力ポートでは必要ありません。

TTAG マーカー パケットと時間間隔の設定

ttag タイムスタンプ フィールドは、マーカー パケットに 48 ビットのタイムスタンプを付加します。この 48 ビットのタイムスタンプは、人間の読み取りやすい ASCII ベースのタイムスタンプではありません。この48ビットのタイムスタンプを人間が読み取れるようにするために、ttagマーカーパケットを使用して、48ビットのタイムスタンプ情報をデコードするための追加情報を提供できます。

フィールド	位置（バイト：ビット）	長さ	定義
Magic		16	デフォルトでは、このフィールドにはA6A6と表示されます。これにより、パケットストリーム上のttag-markerパケットを識別できます。
バージョン		8	バージョン番号。デフォルトのバージョンは 1 です。
精度		16	このフィールドは、48ビットのタイムスタンプサイズの粒度を表します。デフォルトの値は04で、これは 100 ピコ秒つまり 0.1 ナノ秒を表します。
UTc_offset		8	ASIC と UTC クロック間の utc_offset値です。デフォルト値は 0 です。
Timestamp_hi		32	48 ビットの ASIC ハードウェア タイムスタンプの上位 16 ビットです。
Timestamp_lo		32	48 ビットの ASIC ハードウェア タイムスタンプの下位 32 ビットです。
UTC sec		32	Cisco Nexus 9000 シリーズ スイッチの CPU クロックに基づく UTC タイムスタンプの秒の部分です。
UTC sec		32	Cisco Nexus 9000シリーズスイッチのCPUクロックに基づくUTCタイムスタンプのナノ秒の部分です。

予約済み		32	将来的な使用のために予約されています。
署名（Signature）		32	デフォルト値は 0xA5A5A5A5 です。これにより、マーカーパケットの前方検索が可能になり、UTC タイムスタンプへの参照が提供されるため、クライアント ソフトウェアはその参照 UTC を使用して、各パケットヘッダーの 32 ビットのハードウェアタイムスタンプを回復できます。
パッド		8	これは、ttag-marker の位置wo合わせを4バイト境界に変換するための位置合わせバイトです。

始める前に

PTP オフロードがグローバルにイネーブル化されていることを確認します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**ttag-marker-interval** *seconds* 例： `switch(config-if)# ttag-marker-interval 90`	スイッチが ttag-marker パケットを発信ポートに送信するまでの秒数を設定します。これはスイッチのグローバル設定です。デフォルトでは、ttag-marker パケットを 60 秒ごとに送信します。seconds の範囲は 1 ～ 25200 です。

ステップ 3	**interface** *type slot/port*
	例：
	`switch(config)# interface ethernet 2/2`
	`switch(config-if)#`
		指定したインターフェイスに対してインターフェイス コンフィギュレーション モードを開始します。

ステップ 4	[**no**] **ttag-marker enable** 例： `switch(config-if)# ttag-marker enable`	ttag-marker パケットを発信ポートに送信します。

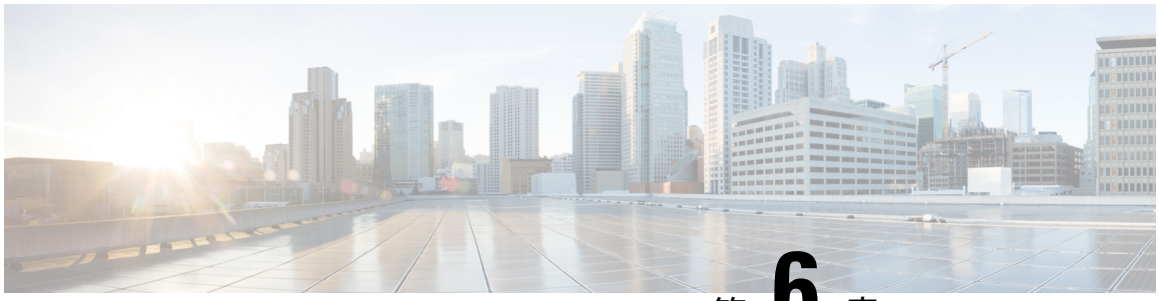
	コマンドまたはアクション	目的
ステップ 5	**ttag-strip** 例： `switch(config-if)# ttag-strip`	インターフェイスの出力パケットから TTAG を削除します。

PTP 設定の確認

次のいずれかのコマンドを使用して、設定を確認します。

表 *3 : PTP Show* コマンド

コマンド	目的
show ptp brief	PTP のステータスを表示します。
show ptp clock	ローカルクロックのプロパティ（クロック ID など）を表示します。
show ptp clock foreign-masters-record	PTP プロセスが認識している外部マスターの状態を表示します。外部マスターごとに、出力に、クロック ID、基本的なクロック プロパティ、およびクロックがグランドマスターとして使用されているかどうかが表示されます。
show ptp corrections	最後の数個の PTP 修正を表示します。
show ptp parent	PTP ペアレントのプロパティを表示します。
show ptp port interface ethernet *slot/port*	スイッチの PTP ポートのステータスを表示します。
show ptp counters [**all** | **interface ethernet** *slot/port*]	すべてのインターフェイスまたは指定したインターフェイスの PTP パケットカウンタを表示します。
show ptp time-property	PTP クロック プロパティを表示します。



第 6 章

NTP の設定

この章は、次の項で構成されています。

NTP の概要

ネットワーク タイム プロトコル（NTP）は、分散している一連のタイム サーバとクライアント間で 1 日の時間を同期させ、複数のネットワーク デバイスから受信するシステム ログや時間関連のイベントを相互に関連付けられるようにします。NTP ではトランスポート プロトコルとして、ユーザ データグラム プロトコル（UDP）を使用します。すべての NTP 通信は UTC を使用します。

NTP サーバは通常、タイム サーバに接続されたラジオ クロックやアトミック クロックなどの正規の時刻源から時刻を受信し、ネットワークを介してこの時刻を配信します。NTP はきわめて効率的で、毎分 1 パケット以下で 2 台のマシンを相互に 1 ミリ秒以内に同期します。

NTP ではストラタム（stratum）を使用して、ネットワーク デバイスと正規の時刻源の距離を表します。

- ストラタム 1 のタイム サーバは、信頼できる時刻源に直接接続されます（無線時計や原子時計または GPS 時刻源など）。

- ストラタム 2 の NTP サーバは、ストラタム 1 のタイム サーバから NTP を使用して時刻を受信します。

同期の前に、NTP は複数のネットワーク サービスが報告した時刻を比較し、1 つの時刻が著しく異なる場合は、それが Stratum 1 であっても、同期しません。Cisco NX-OS は、無線時計や原子時計に接続できず、ストラタム 1 サーバとして動作することはできないため、インターネット上で利用できるパブリック NTP サーバを使用することを推奨します。ネットワークがインターネットから切り離されている場合、Cisco NX-OS では、NTP によって時刻が同期されていなくても、NTP で同期されているものとして時刻を設定できます。

（注） NTP ピア関係を作成して、サーバで障害が発生した場合に、ネットワーク デバイスを同期させて、正確な時刻を維持するための時刻提供ホストを指定できます。

デバイス上の時刻は重要な情報であるため、NTP のセキュリティ機能を使用して、不正な時刻を誤って（または悪意を持って）設定できないように保護することを強く推奨します。その方法として、アクセス リストベースの制約方式と暗号化認証方式があります。

タイム サーバーとしての NTP

他のデバイスからタイム サーバとして設定できます。デバイスを正規の NTP サーバとして動作するよう設定し、外部の時刻源と同期していないときでも時刻を配信させることもできます。

CFS を使用した NTP の配信

Cisco Fabric Services（CFS）は、ローカル NTP コンフィギュレーションをネットワーク内のすべてのシスコ デバイスに配信します。

デバイス上で CFS をイネーブルにすると、NTP コンフィギュレーションが起動された場合には常に、ネットワーク全体のロックが NTP に適用されます。NTP コンフィギュレーションを変更した後で、これらの変更を破棄することもコミットすることもできます。

いずれの場合でも、CFS のロックはこのときに NTP アプリケーションから解放されます。

クロック マネージャ

クロックはさまざまなプロセス間で共有する必要のあるリソースです。

NTP や高精度時間プロトコル（PTP）といった複数の時刻同期プロトコルがシステムで稼働している可能性があります。

高可用性

NTP はステートレス リスタートをサポートします。リブート後またはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションが適用されます。

NTP ピアを設定すると、NTP サーバ障害の発生時に冗長性が得られます。

仮想化のサポート

NTP は Virtual Routing and Forwarding（VRF）インスタンスを認識します。NTP サーバおよび NTP ピアに対して特定の VRF を設定していない場合、NTP はデフォルトの VRF を使用します。

NTP の前提条件

NTP の前提条件は、次のとおりです。

- NTP を設定するには、NTP が動作している 1 つ以上のサーバに接続できなければなりません。

NTP の注意事項と制約事項

NTP に関する設定時の注意事項および制約事項は、次のとおりです。

- **show ntp session status** CLI コマンドには、最後のアクションのタイムスタンプ、最後のアクション、最後のアクションの結果、および最後のアクションの失敗理由は表示されません。
- NTP サーバー機能はサポートされます。
- 別のデバイスとの間にピア アソシエーションを設定できるのは、使用するクロックの信頼性が確実な場合（つまり、信頼できる NTP サーバーのクライアントである場合）に限られます。
- 単独で設定したピアは、サーバーの役割を担いますが、バックアップとして使用する必要があります。サーバが 2 台ある場合、いくつかのデバイスが一方のサーバに接続し、残りのデバイスが他方のサーバに接続するように設定できます。その後、2 台のサーバ間にピア アソシエーションを設定すると、信頼性の高い NTP 構成になります。
- サーバーが 1 台だけの場合は、すべてのデバイスをそのサーバーのクライアントとして設定する必要があります。
- 設定できる NTP エンティティ（サーバーおよびピア）は、最大 64 です。

- NTP に対して CFS がディセーブルになっていると、その NTP からコンフィギュレーションは配信されず、ネットワーク内の他のデバイスからの配信も受け入れません。
- NTP に対して CFS 配信をイネーブルにしても、**commit** コマンドを入力するまで、NTP コンフィギュレーション コマンドのエントリは NTP コンフィギュレーションに対してネットワークをロックします。ロック中は、ネットワーク内の（ロックを保持しているデバイス以外の）すべてのデバイスは NTP コンフィギュレーションを変更できません。
- CFS を使用して NTP をディセーブルにする場合、ネットワーク内のすべてのデバイスは、NTP に対して使用するよう設定したものと同じ VRF を持っている必要があります。
- VRF で NTP を設定する場合は、NTP サーバーおよびピアが、設定された VRF を介して相互にアクセスできることを確認します。
- ネットワーク全体の NTP サーバーおよび Cisco NX-OS デバイスに、NTP 認証キーを手動で配信する必要があります。
- スイッチをエッジ デバイスとして使用して NTP を利用したい場合は、**ntp access-group** コマンドを使用して必要なエッジ デバイスにのみ NTP をフィルタリングすることを推奨します。
- システムに **ntp passive**、**ntp broadcast client**、または **ntp multicast client** コマンドが設定されている場合、対称アクティブの着信パケット、ブロードキャスト パケット、マルチキャスト パケットを NTP が受信する際に、送信者と同期させるための一時的なピア アソシエーションを設定できます。

（注） 上記コマンドのいずれかを有効にする前に必ず **ntp authenticate** を指定してください。そうしないと、上記のパケット タイプのいずれかを送信する任意のデバイス（悪意のある攻撃者に制御されたデバイスを含む）とデバイスが同期される可能性があります。

- **ntp authenticate** コマンドが指定されている場合、対称アクティブ パケット、ブロードキャスト パケット、マルチキャスト パケットが受信されても、**ntp trusted-key** グローバル コンフィギュレーション コマンドで指定された認証キーの 1 つがパケットで運ばれていない限り、システムとピアの同期は行われません。
- **ntp access-group** コマンドなど他の方法で、デバイスの NTP サービスと非承認ホストとの通信防止の措置が取られている場合を除き、非承認のネットワーク ホストとの同期を避けるには、**ntp passive**、**ntp broadcast client**、**ntp multicast client** コマンドを指定した段階で随時 **ntp authenticate** コマンドを指定する必要があります。
- **ntp authenticate** コマンドは、**ntp server** および **ntp peer** コンフィギュレーション コマンドで設定されたピア アソシエーションを認証しません。**ntp server** および **ntp peer** アソシエーションを認証するには、**key** キーワードを指定します。
- 時刻の精度および信頼性要件が厳密ではない場合、NTP ブロードキャストまたはマルチキャスト アソシエーションを使用すると、ネットワークがローカル化され、ネットワーク

は 20 以上のクライアントを持ちます。帯域幅、システム メモリ、または CPU リソースが限られているネットワークでは NTP ブロードキャストまたはマルチキャスト アソシエーションの使用をお勧めします。

- 1 つの NTP アクセス グループに最大 4 つの ACL を設定できます。

（注） 情報の流れが一方向に限定されるため、NTP ブロードキャスト アソシエーションでは、時刻の精度がわずかに低下します。

デフォルト設定

次に、NTP パラメータのデフォルト設定を示します。

パラメータ	デフォルト
NTP	すべてのインターフェイスでイネーブル
NTP passive（アソシエーションを形成するために NTP をイネーブルにする）	イネーブル
NTP 認証	ディセーブル
NTP アクセス	イネーブル
NTP access group match all	ディセーブル
NTP ブロードキャスト サーバー	ディセーブル
NTP マルチキャスト サーバ	ディセーブル
NTP マルチキャスト クライアント	ディセーブル
NTP ロギング	無効化

NTP の設定

インターフェイスでの NTP のイネーブル化またはディセーブル化

特定のインターフェイスで NTP をイネーブルまたはディセーブルにできます。NTP は、すべてのインターフェイスでデフォルトでイネーブルに設定されています。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface** *type slot/port*	インターフェイス設定モードを開始します。
ステップ 3	switch(config-if)# **[no] ntp disable** {**ip** | **ipv6**}	指定のインターフェイスで NTP IPv4 または IPv6 をディセーブルにします。 インターフェイス上でNTPを再度イネーブルにするにはこのコマンドの **no** 形式を使用します。
ステップ 4	（任意） switch(config-if)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、インターフェイスで NTP をイネーブルまたはディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 6/1
switch(config-if)# ntp disable ip
switch(config-if)# copy running-config startup-config
```

正規の NTP サーバとしてのデバイスの設定

デバイスを正規の NTP サーバーとして動作するよう設定し、既存のタイム サーバーと同期していないときでも時刻を配信させることができます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **ntp master** [*stratum*]	正規の NTP サーバとしてデバイスを設定します。

	コマンドまたはアクション	目的
		NTP クライアントがこれらの時間を同期するのと別の階層レベルを指定できます。指定できる範囲は 1 ～ 15 です。
ステップ 3	（任意） **show running-config ntp**	NTP コンフィギュレーションを表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、正規の NTP サーバーとして Cisco NX-OS デバイスを別の階層レベルで設定する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp master 5
```

NTP サーバおよびピアの設定

NTP サーバーおよびピアを設定できます。

始める前に

NTP サーバーとそのピアの IP アドレスまたは DNS 名がわかっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# [**no**] **ntp server** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*]	1 つのサーバと 1 つのサーバアソシエーションを形成します。 NTP サーバとの通信で使用するキーを設定するには、**key** キーワードを使用します。 *key-id* 引数の範囲は 1 ～ 65535 です。 サーバをポーリングする最大および最小の間隔を設定するには、 **maxpoll** および **minpoll** キーワードを使用します。

	コマンドまたはアクション	目的
		max-poll および *min-poll* 引数の範囲は4ー16（2 の累乗として設定されます。つまり、実質的に16ー65536秒）で、デフォルト値はそれぞれ 6 と 4 です（*maxpoll* デフォルト = 64秒、*minpoll* デフォルト= 16秒）。 デバイスに対して対象の NTP サーバーを優先サーバーにするには、**prefer keyword** を使用します。 指定された VRF を介して通信するように NTP サーバを設定するには、**use-vrf** キーワードを使用します。 *vrf-name* 引数として、default、management、または大文字と小文字を区別した 32 文字までの任意の英数字の文字列を使用できます。 （注） NTP サーバーとの通信で使用するキーを設定する場合は、そのキーが、デバイス上の信頼できるキーとして存在していることを確認してください。
ステップ 3	switch(config)# [**no**] **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} [**key** *key-id*] [**maxpoll** *max-poll*] [**minpoll** *min-poll*] [**prefer**] [**use-vrf** *vrf-name*]	1 つのピアと 1 つのピア アソシエーションを形成します。複数のピアアソシエーションを指定できます。 NTP ピアとの通信で使用するキーを設定するには、**key** キーワードを使用します。*key-id* 引数の範囲は 1 ～ 65535 です。 サーバをポーリングする最大および最小の間隔を設定するには、 **maxpoll** および **minpoll** キーワードを使用します。*max-poll* および *min-poll* 引数の範囲は4ー16（2 の累乗として設定されます。つまり、実質的に 16ー131072 秒）で、デフォルト値はそれぞれ 6 と 4 です（*maxpoll* デフォルト = 64秒、*minpoll* デフォルト= 16秒）。 デバイスに対して対象の NTP ピアを優先にするには、**prefer** キーワードを使用します。

	コマンドまたはアクション	目的
		指定された VRF を介して通信するように NTP ピアを設定するには、**use-vrf** キーワードを使用します。*vrf-name* 引数には、**default**、**management**、または大文字と小文字が区別される最大 32 文字の任意の英数字文字列を指定できます。
ステップ 4	（任意） switch(config)# **show ntp peers**	設定されたサーバおよびピアを表示します。 （注） ドメイン名が解決されるのは、DNS サーバが設定されている場合だけです。
ステップ 5	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

NTP 認証の設定

ローカル ロックを同期させる時刻源を認証するようデバイスを設定できます。NTP 認証をイネーブルにすると、**ntp trusted-key** コマンドによって指定されたいずれかの認証キーを時刻源が保持している場合のみ、デバイスはその時刻源と同期します。デバイスは、認証チェックに失敗したすべてのパケットをドロップし、それらのパケットでローカルクロックがアップデートされないようにします。NTP 認証はデフォルトでディセーブルになっています。

始める前に

NTP サーバーと NTP ピアの認証は、**key** キーワードを各 **ntp server** および **ntp peer** コマンドで使用することにより、アソシエーションごとに設定されます。この手順で指定する予定の認証キーによって、すべての NTP サーバーとピア アソシエーションが設定されていることを確認します。**ntp server** または **ntp peer** コマンドで **key** キーワードを指定しない場合、認証なしでの動作が続けられます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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。

	コマンドまたはアクション	目的
ステップ 2	[**no**] **ntp authentication-key** *number* **md5** *md5-string* 例： `switch(config)# ntp authentication-key 42 md5 aNiceKey`	認証キーを定義します。デバイスが時刻源と同期するのは、時刻源がこれらの認証キーのいずれかを持ち、**ntp trusted-key** *number* コマンドによってキー番号が指定されている場合だけです。 認証キーの範囲は 1 ～ 65535 です。MD5 文字列の場合は、最大 8 文字の英数字を指定できます。
ステップ 3	**ntp server** *ip-address* **key** *key-id* 例： `switch(config)# ntp server 192.0.2.1 key 1001`	指定された NTP サーバーで認証を有効にし、サーバーとのアソシエーションを形成します。 NTP サーバとの通信で使用するキーを設定するには、**key** キーワードを使用します。*key-id* 引数の範囲は 1 ～ 65535 です。 認証を必須とする場合は、**key** キーワードを使用する必要があります。**ntp server** または **ntp peer** コマンドで **key** キーワードを指定しない場合、認証なしでの動作が続けられます。
ステップ 4	（任意） **show ntp authentication-keys** 例： `switch(config)# show ntp authentication-keys`	設定済みの NTP 認証キーを表示します。
ステップ 5	[**no**] **ntp trusted-key** *number* 例： `switch(config)# ntp trusted-key 42`	1 つ以上のキー（ステップ 2 で定義されているもの）を指定します。デバイスを時刻源と同期させるには、未設定のリモート シンメトリック、ブロードキャスト、およびマルチキャストの時刻源を NTP パケット内に入力する必要があります。trusted key の範囲は 1 ～ 65535 です。 このコマンドにより、デバイスが、信頼されていない時刻源と誤って同期する、ということが防止されます。
ステップ 6	（任意） **show ntp trusted-keys** 例：	設定済みの NTP の信頼されているキーを表示します。

	コマンドまたはアクション	目的
	`switch(config)# show ntp trusted-keys`	
ステップ 7	**[no] ntp authenticate** 例： `switch(config)# ntp authenticate`	ntp passive、ntp broadcast client、および ntp multicast で認証を有効または無効にします。NTP 認証はデフォルトでディセーブルになっています。
ステップ 8	（任意） **show ntp authentication-status** 例： `switch(config)# show ntp authentication-status`	NTP 認証の状況を表示します。
ステップ 9	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

NTP アクセス制限の設定

アクセス グループを使用して、NTP サービスへのアクセスを制御できます。具体的には、デバイスで許可する要求のタイプ、およびデバイスが応答を受け取るサーバを指定できます。

アクセス グループを設定しない場合は、すべてのデバイスにNTPアクセス権が付与されます。何らかのアクセス グループを設定した場合は、ソース IP アドレスがアクセス リストの基準をパスしたリモート デバイスに対してだけ、NTP アクセス権が付与されます。

Cisco NX-OS リリース 7.0(3)I7(3) 以降では、アクセス グループは次の方法で評価されます。

- **match-all** キーワードがない場合、パケットは permit が見つかるまでアクセス グループに対して（以下に示す順で）評価されます。permit が検出されない場合、パケットはドロップされます。
- **match-all** キーワードがある場合、パケットはすべてのアクセス グループに対して（以下に示す順で）評価され、最後に成功した評価（ACL が設定されている最後のアクセス グループ）に基づいてアクションが実行されます。

アクセス グループとパケットのタイプのマッピングは次のとおりです。

- peer：クライアント、対称アクティブ、対称パッシブ、サービス、コントロール、およびプライベート パケット（すべてのタイプ）を処理
- serve：クライアント、コントロール、およびプライベート パケットを処理
- serve-only：クライアント パケットだけを処理
- query-only：コントロールおよびプライベート パケットだけを処理

アクセス グループは、次の降順で評価されます。

1. peer（すべてのパケット タイプ）
2. serve（クライアント、コントロール、およびプライベート パケット）
3. query only（クライアント パケット）または query-only（コントロールおよびプライベート パケット）

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# [**no**] **ntp access-group match-all** | {{**peer** | **serve** | **serve-only** | **query-only** }*access-list-name*}	NTP のアクセスを制御し、基本の IP アクセス リストを適用するためのアクセス グループを作成または削除します。 アクセス グループのオプションは、次の順序で制限の緩いものから厳しいものへとスキャンされます。ただし、ピアに設定された拒否 ACL ルールに NTP が一致した場合、ACL 処理は停止し、次のアクセス グループ オプションへと継続しません。 • **peer** キーワードは、デバイスが時刻要求と NTP 制御クエリーを受信し、アクセス リストで指定されているサーバーと同期するようにします。 • **serve** キーワードは、アクセス リストに指定されているサーバーからの時刻要求と NTP 制御クエリーをデバイスが受信できるようにしますが、指定されたサーバーとは同期しないようにします。 • **serve-only** キーワードは、デバイスがアクセス リストで指定されたサーバーからの時刻要求だけを受信するようにします。 • **query-only** キーワードは、デバイスがアクセス リストで指定されたサーバーからの NTP 制御クエリーのみを受信するようにします。

	コマンドまたはアクション	目的
		• **match-all** キーワードを使用すると、アクセス グループ オプションが、制限の最も緩いものから最も厳しいもの、peer、serve、serve-only、query-only の順序でスキャンされるようにできます。着信パケットが peer アクセス グループの ACL に一致しない場合、パケットは serve アクセス グループに送信され、処理されます。パケットが serve アクセス グループの ACL に一致しない場合、serve-only アクセス グループに送られ、これが継続されます。 （注） **match-all** キーワードは、Cisco NX-OS リリース 7.0(3)I6(1) 以降で使用可能です。
ステップ 3	switch(config)# **show ntp access-groups**	（任意）NTP アクセス グループのコンフィギュレーションを表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、アクセス グループ「accesslist1」からピアと同期できるようデバイスを設定する例を示します。

```
switch# configure terminal
switch(config)# ntp access-group peer accesslist1
switch(config)# show ntp access-groups
Access List Type
----------------------------
accesslist1 Peer
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

NTP ソース IP アドレスの設定

NTP は、NTP パケットが送信されたインターフェイスのアドレスに基づいて、すべての NTP パケットにソース IP アドレスを設定します。特定のソース IP アドレスを使用するよう NTP を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **ntp source** *ip-address*	すべての NTP パケットにソース IP アドレスを設定します。*ip-address* には IPv4 または IPv6 形式を使用できます。

例

次に、NTP ソース IP アドレスに 192.0.2.2 を設定する例を示します。

```
switch# configure terminal
switch(config)# ntp source 192.0.2.2
```

NTP ソース インターフェイスの設定

特定のインターフェイスを使用するよう NTP を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **ntp source-interface** *interface*	すべての NTP パケットに対してソース インターフェイスを設定します。次のリストに、*interface* として有効な値を示します。 • ethernet • loopback • mgmt • port-channel • vlan

例

次に、NTP 送信元インターフェイスを設定する例を示します。

```
switch# configure terminal
switch(config)# ntp source-interface ethernet
```

NTP ブロードキャスト サーバの設定

インターフェイス上で NTP IPv4 ブロードキャスト サーバを設定できます。デバイスは、そのインターフェイスを介してブロードキャストパケットを定期的に送信します。クライアントは応答を送信する必要はありません。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface** *type slot/port*	インターフェイス設定モードを開始します。
ステップ 3	switch(config-if)# **[no] ntp broadcast** [**destination** *ip-address*] [**key** *key-id*] [*version number*]	指定されたインターフェイスの IPv4 NTP ブロードキャスト サーバをイネーブルにします。 • **destination** *ip-address*：ブロードキャスト宛先 IP アドレスを設定します。 • **key** *key-id*：ブロードキャスト認証キー番号を設定します。有効な範囲は 1 ～ 65535 です。 • *version number*：NTP バージョンを設定します。範囲は 2 ～ 4 です。
ステップ 4	switch(config-if)# **exit**	インターフェイス コンフィギュレーション モードを終了します。
ステップ 5	（任意） switch(config)# **[no] ntp broadcastdelay** *delay*	推定のブロードキャストラウンドトリップ遅延をマイクロ秒単位で設定します。範囲は 1 ～ 999999 です。
ステップ 6	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、NTP ブロードキャスト サーバーを設定する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 6/1
switch(config-if)# ntp broadcast destination 192.0.2.10
switch(config-if)# exit
switch(config)# ntp broadcastdelay 100
switch(config)# copy running-config startup-config
```

NTP マルチキャスト サーバの設定

インターフェイスに対して NTP IPv4 または IPv6 マルチキャスト サーバを設定できます。デバイスは、そのインターフェイスを介してマルチキャスト パケットを定期的に送信します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface** *type slot/port*	インターフェイス設定モードを開始します。
ステップ 3	switch(config-if)# **[no] ntp multicast** [*ipv4-address* | *ipv6-address*] [**key** *key-id*] [*ttl value*] [*version number*]	指定したインターフェイスの NTP IPv4 または IPv6 マルチキャスト サーバーをイネーブルにします。 • *ipv4-address* または *ipv6-address*：マルチキャスト IPv4 または IPv6 アドレス。 • **key** *key-id*：ブロードキャスト認証キー番号を設定します。有効な範囲は 1 ～ 65535 です。 • *ttl value*：マルチキャスト パケットの存続可能時間値。範囲は 1 ～ 255 です。 • *version number*：NTP バージョン。範囲は 2 ～ 4 です。
ステップ 4	（任意） switch(config-if)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、NTPマルチキャストパケットを送信するようにイーサネットインターフェイスを設定する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 2/2
switch(config-if)# ntp multicast FF02::1:FF0E:8C6C
switch(config-if)# copy running-config startup-config
```

NTP マルチキャスト クライアントの設定

インターフェイス上でNTP マルチキャスト クライアントを設定できます。デバイスはNTP マルチキャストメッセージをリッスンし、マルチキャストが設定されていないインターフェイスからのメッセージを廃棄します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface** *type slot/port*	インターフェイス設定モードを開始します。
ステップ 3	switch(config-if)# **[no] ntp multicast client** [*ipv4-address* | *ipv6-address*]	指定されたインターフェイスが NTP マルチキャスト パケットを受信できるようにします。
ステップ 4	（任意） switch(config-if)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、NTPマルチキャストパケットを受信するようにイーサネットインターフェイスを設定する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 2/3
switch(config-if)# ntp multicast client FF02::1:FF0E:8C6C
switch(config-if)# copy running-config startup-config
```

NTP ロギングの設定

重要な NTP イベントでシステム ログを生成するよう、NTP ロギングを設定できます。NTP ロギングはデフォルトでディセーブルになっています。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# [**no**] **ntp logging**	重要な NTP イベントでシステム ログを生成することをイネーブルまたはディセーブルにします。NTP ロギングはデフォルトでディセーブルになっています。
ステップ 3	（任意） switch(config)# **show ntp logging-status**	NTP ロギングのコンフィギュレーション状況を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、重要な NTP イベントによってシステム ログを生成するよう、NTP ロギングをイネーブルにする例を示します。

```
switch# configure terminal
switch(config)# ntp logging
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

NTP 用の CFS 配信のイネーブル化

NTP コンフィギュレーションを他の CFS 対応デバイスに配信するために、NTP 用の CFS 配信をイネーブルにできます。

始める前に

デバイスの CFS 配信をイネーブルにしていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# [**no**] **ntp distribute**	CFS を介して配信される NTP コンフィギュレーションのアップデートをデバイスが受信することを、イネーブルまたはディセーブルにします。
ステップ 3	（任意） switch(config)# **show ntp status**	NTP CFS の配信状況を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、デバイスが CFS を介して NTP 設定の更新を受信できるようにする例を示します。

```
switch# configure terminal
switch(config)# ntp distribute
switch(config)# copy running-config startup-config
```

NTP 設定変更のコミット

NTP コンフィギュレーションの変更をコミットすると、保留データベースのコンフィギュレーション変更によって有効なデータベースが上書きされ、ネットワーク内のすべてのデバイスが同じコンフィギュレーションを受け取ります。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **ntp commit**	ネットワーク内のすべての Cisco NX-OS デバイスに NTP コンフィギュレーションの変更を配信し、CFS ロックを解放します。このコマンドは、保留データベースに対して行われた変更によって、有効なデータベースを上書きします。

NTP 設定変更の廃棄

コンフィギュレーション変更の後で、これらの変更をコミットせずに、破棄するよう選択することもできます。変更を破棄すると、Cisco NX-OS によって保留データベースの変更が削除され、CFS ロックが解放されます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **ntp abort**	保留データベースで NTP コンフィギュレーションの変更を破棄して、CFS ロックを解放します。このコマンドは、NTP コンフィギュレーションを起動したデバイスで使用します。

CFS セッション ロックの解放

NTP コンフィギュレーションを実行したが、変更をコミットまたは破棄してロックを解放し忘れた場合は、自分で、または他の管理者がネットワーク内の任意のデバイスからロックを解放できます。また、この操作では、保留データベースの変更が破棄されます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **clear ntp session**	保留データベースで NTP コンフィギュレーションの変更を破棄して、CFS ロックを解放します。

NTP の設定確認

コマンド	目的
show ntp access-groups	NTP アクセス グループのコンフィギュレーションを表示します。
show ntp authentication-keys	設定済みの NTP 認証キーを表示します。
show ntp authentication-status	NTP 認証の状況を表示します。

コマンド	目的
show ntp logging-status	NTP のロギング状況を表示します。
show ntp peer-status	すべての NTP サーバおよびピアのステータスを表示します。
show ntp peer	すべての NTP ピアを表示します。
show ntp pending	NTP 用の一時 CFS データベースを表示します。
show ntp pending-diff	保留 CFS データベースと現行の NTP コンフィギュレーションの差異を表示します。
show ntp rts-update	RTS アップデートの状況を表示します。
show ntp session status	NTP CFS 配信セッションの情報を表示します。
show ntp source	設定済みの NTP ソース IP アドレスを表示します。
show ntp source-interface	設定済みの NTP ソースインターフェイスを表示します。
show ntp statistics {**io** | **local** | **memory** | **peer** {**ipaddr** {*ipv4-addr*} | **name** *peer-name*}}	NTP 統計情報を表示します。
show ntp status	NTP CFS の配信状況を表示します。
show ntp trusted-keys	設定済みの NTP の信頼されているキーを表示します。
show running-config ntp	NTP 情報を表示します。

NTP の設定例

NTP の設定例

次に、NTP サーバーおよびピアを設定し、NTP 認証をイネーブルにして、NTP ロギングをイネーブルにした後で、そのスタートアップの設定を保存し、リブートとリスタートを通して保存されるようにする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp peer 192.0.2.105
switch(config)# show ntp peers
-------------------------------------------------
Peer IP Address Serv/Peer
```

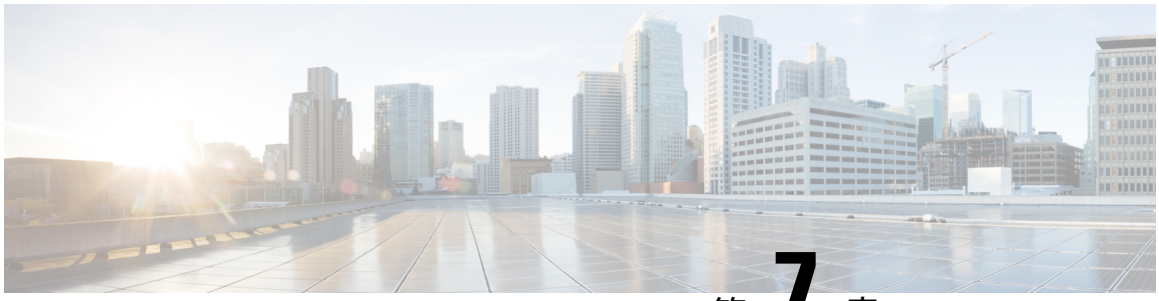
```
-----------------------------------------------
192.0.2.100 Peer (configured)
192.0.2.105 Server (configured)
switch(config)# ntp authentication-key 42 md5 aNiceKey
switch(config)# show ntp authentication-keys
----------------------------
Auth key MD5 String
----------------------------
42 aNicekey
switch(config)# ntp trusted-key 42
switch(config)# show ntp trusted-keys
Trusted Keys:
42
switch(config)# ntp authenticate
switch(config)# show ntp authentication-status
Authentication enabled.
switch(config)# ntp logging
switch(config)# show ntp logging
NTP logging enabled.
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

次に、以下の制約事項のある NTP アクセス グループの設定の例を示します。

- peer の制約事項は、「peer-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。
- serve の制約事項は、「serve-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。
- serve-only の制約事項は、「serve-only-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。
- query-only の制約事項は、「query-only-acl」というアクセス リストの条件を満たす IP アドレスに適用されます。

```
switch# configure terminal
switch(config)# ntp peer 10.1.1.1
switch(config)# ntp peer 10.2.2.2
switch(config)# ntp peer 10.3.3.3
switch(config)# ntp peer 10.4.4.4
switch(config)# ntp peer 10.5.5.5
switch(config)# ntp peer 10.6.6.6
switch(config)# ntp peer 10.7.7.7
switch(config)# ntp peer 10.8.8.8
switch(config)# ntp access-group peer peer-acl
switch(config)# ntp access-group serve serve-acl
switch(config)# ntp access-group serve-only serve-only-acl
switch(config)# ntp access-group query-only query-only-acl
switch(config)# ip access-list peer-acl
switch(config-acl)# 10 permit ip host 10.1.1.1 any
switch(config-acl)# 20 permit ip host 10.8.8.8 any
switch(config)# ip access-list serve-acl
switch(config-acl)# 10 permit ip host 10.4.4.4 any
switch(config-acl)# 20 permit ip host 10.5.5.5 any
switch(config)# ip access-list serve-only-acl
switch(config-acl)# 10 permit ip host 10.6.6.6 any
switch(config-acl)# 20 permit ip host 10.7.7.7 any
switch(config)# ip access-list query-only-acl
```

```
switch(config-acl)# 10 permit ip host 10.2.2.2 any
switch(config-acl)# 20 permit ip host 10.3.3.3 any
```



第 7 章

ユーザ アカウントおよび RBAC の設定

この章は、次の項で構成されています。

ユーザー アカウントおよび RBAC の概要

Cisco Nexus シリーズ スイッチは、ロールベース アクセス コントロール（RBAC）を使用して、ユーザーがスイッチにログインするときに各ユーザーが持つアクセス権の量を定義します。

RBAC では、1 つまたは複数のユーザー ロールを定義し、各ユーザー ロールがどの管理操作を実行できるかを指定します。スイッチのユーザー アカウントを作成するとき、そのアカウントにユーザー ロールを関連付けます。これにより個々のユーザーがスイッチで行うことができる操作が決まります。

ユーザ ロール

ユーザー ロールには、そのロールを割り当てられたユーザーが実行できる操作を定義するルールが含まれています。各ユーザー ロールに複数のルールを含めることができ、各ユーザーが複数のロールを持つことができます。たとえば、role1 では設定操作へのアクセスだけが許可されており、role2 ではデバッグ操作へのアクセスだけが許可されている場合、role1 と role2 の両方に属するユーザーは、設定操作とデバッグ操作にアクセスできます。特定の 、VLAN、およびインターフェイスへのアクセスを制限することもできます。

スイッチには、次のデフォルト ユーザー ロールが用意されています。

network-admin（スーパーユーザー）

スイッチ全体に対する完全な読み取りと書き込みのアクセス権。

network-operator

スイッチに対する完全な読み取りアクセス権。

Note 複数のロールに属するユーザは、そのロールで許可されるすべてのコマンドの組み合わせを実行できます。コマンドへのアクセス権は、コマンドへのアクセス拒否よりも優先されます。たとえば、ユーザが、コンフィギュレーション コマンドへのアクセスが拒否されたロール A を持っていたとします。しかし、同じユーザが ロール B も持ち、このロールではコンフィギュレーション コマンドにアクセスできるとします。この場合、このユーザはコンフィギュレーション コマンドにアクセスできます。

ルール

ルールは、ロールの基本要素です。ルールは、そのロールがユーザにどの操作の実行を許可するかを定義します。ルールは次のパラメータで適用できます。

コマンド

正規表現で定義されたコマンドまたはコマンド グループ

機能

Cisco Nexus デバイスにより提供される機能に適用されるコマンド。**show role feature** コマンドを入力すると、このパラメータに指定できる機能名が表示されます。

機能グループ

機能のデフォルト グループまたはユーザ定義グループ**show role feature-group** コマンドを入力すると、このパラメータに指定できるデフォルトの機能グループが表示されます。

OID

SNMP オブジェクト ID（OID）。

これらのパラメータは、階層状の関係を作成します。最も基本的な制御パラメータはコマンドです。次の制御パラメータは機能です。これは、その機能にアソシエートされているすべてのコマンドを表します。最後の制御パラメータが、機能グループです。機能グループは、関連する機能を組み合わせたものです。機能グループによりルールを簡単に管理できます。

SNMP OID は RBAC でサポートされています。SNMP OID に読み取り専用ルールまたは読み取り/書き込みルールを設定できます。

ロールごとに最大 256 のルールを設定できます。ルールが適用される順序は、ユーザ指定のルール番号で決まります。ルールは降順で適用されます。たとえば、1 つのロールが 3 つのルールを持っている場合、ルール 3 がルール 2 よりも前に適用され、ルール 2 はルール 1 よりも前に適用されます。

ユーザー ロール ポリシー

ユーザーがアクセスできるスイッチ リソースを制限するために、またはインターフェイスと VLAN へのアクセスを制限するために、ユーザー ロール ポリシーを定義できます。

ユーザ ロール ポリシーは、ロールに定義されているルールで制約されます。たとえば、特定のインターフェイスへのアクセスを許可するインターフェイス ポリシーを定義した場合、**interface** コマンドを許可するコマンドルールをロールに設定しないと、ユーザはインターフェイスにアクセスできません。

コマンドルールが特定のリソース（インターフェイス、VLAN、）へのアクセスを許可した場合、ユーザーがそのユーザーに関連付けられたユーザー ロール ポリシーに表示されていなくても、ユーザーはこれらのリソースへのアクセスを許可されます。

ユーザー アカウントの設定の制限事項

次の語は予約済みであり、ユーザー設定に使用できません。

- adm
- bin
- daemon
- ftp
- ftpuser
- games
- gdm
- gopher
- halt
- lp
- mail
- mailnull
- man
- mtsuser
- news
- nobody
- san-admin
- shutdown
- sync
- sys

- uucp
- xfs

注意 Cisco Nexus シリーズ スイッチでは、すべて数字のユーザー名が TACACS+ または RADIUS で作成されている場合でも、すべて数字のユーザー名はサポートされません。AAA サーバに数字だけのユーザ名が登録されていて、ログイン時に入力しても、スイッチはログイン要求を拒否します。

ユーザ パスワードの要件

Cisco Nexus デバイス パスワードには大文字小文字の区別があり、英数字を含むことができます。

（注） Cisco Nexus デバイスのパスワードには、ドル記号（$）やパーセント記号（%）などの特殊文字を使用できます。

パスワードが脆弱な場合（短い、解読されやすいなど）、Cisco Nexus デバイスはパスワードを拒否します。各ユーザーアカウントには強力なパスワードを設定するようにしてください。強力なパスワードは、次の特性を持ちます。

- 長さが 8 文字以上である
- 複数の連続する文字（「abcd」など）を含んでいない
- 複数の同じ文字の繰り返し（「aaabbb」など）を含んでいない
- 辞書に載っている単語を含んでいない
- 正しい名前を含んでいない
- 大文字および小文字の両方が含まれている
- 数字が含まれている

強力なパスワードの例を次に示します。

- If2CoM18
- 2009AsdfLkj30
- Cb1955S21

（注） セキュリティ上の理由から、ユーザ パスワードはコンフィギュレーション ファイルに表示されません。

ユーザー アカウントの注意事項および制約事項

ユーザー アカウントおよび RBAC を設定する場合、ユーザー アカウントには次の注意事項および制約事項があります。

- ユーザ ロールに設定された読み取り/書き込みルールに関係なく、一部のコマンドは、あらかじめ定義された network-admin ロールでのみ実行できます。
- 最大 256 個のルールをユーザー ロールに追加できます。
- 最大 64 個のユーザー ロールをユーザー アカウントに割り当てることができます。
- 1 つのユーザー ロールを複数のユーザー アカウントに割り当てることができます。
- network-admin、network-operator、san-admin などの事前定義されたロールは編集不可です。
- ルールの追加、削除、編集は、SAN 管理者ユーザー ロールではサポートされません。
- インターフェイス、VLAN、または VSAN 範囲は SAN 管理者ユーザー ロールでは変更できません。

（注） ユーザー アカウントは、少なくとも 1 つのユーザー ロールを持たなければなりません。

ユーザ アカウントの設定

Note ユーザーアカウントの属性に加えられた変更は、そのユーザーがログインして新しいセッションを作成するまで有効になりません。

ユーザー名の最初の文字として、任意の英数字または _（アンダースコア）を使用できます。最初の文字にその他の特殊文字を使用することはできません。ユーザー名に許可されていない文字が含まれている場合、指定したユーザーはログインできません。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	(Optional) switch(config)# **show role**	使用可能なユーザ ロールを表示します。必要に応じて、他のユーザ ロールを設定できます。

	Command or Action	Purpose
ステップ 3	switch(config) # **username** *user-id* [**password** *password*] [**expire** *date*] [**role** *role-name*]	ユーザー アカウントを設定します。 *user-id* は、最大 28 文字の英数字の文字列で、大文字と小文字が区別されます。 デフォルトの *password* は定義されていません。 **Note** パスワードを指定しなかった場合、ユーザーはスイッチにログインできない場合があります。 **Note** リリース 7.0(3)I2(1) 以降では、パスワード強度をチェックするための新しい内部関数が実装されています。リリース 7.0(3)I2(1) の Cisco Nexus 3000 シリーズ プラットフォームでパスワード強度チェックを有効にすると、以前のリリースとは異なる基準が適用されます。 **expire** *date* オプションのフォーマットは YYYY-MM-DD です。デフォルトでは、失効日はありません。
ステップ 4	switch(config) # **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。
ステップ 5	(Optional) switch# **show user-account**	ロール設定を表示します。
ステップ 6	(Optional) switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Example

次に、ユーザ アカウントを設定する例を示します。

```
switch# configure terminal
switch(config)# username NewUser password 4Ty18Rnt
switch(config)# exit
switch# show user-account
```

次に、リリース 7.0(3)I2(1) 以降でパスワード強度チェックを有効にする基準の例を示します。

```
switch(config)# username xyz password nbv12345
password is weak
Password should contain characters from at least three of the following classes: lower
```

```
case letters, upper case letters, digits and special characters.
switch(config)# username xyz password Nbv12345
password is weak
it is too simplistic/systematic
switch(config)#
```

SAN 管理者ユーザの設定

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **username** *user-id* **role san-admin password** *password*	指定したユーザに対する SAN 管理者ユーザ ロールのアクセス権を設定します。
ステップ 3	（任意） switch(config) # **show user-account**	ロール設定を表示します。
ステップ 4	（任意） switch(config) # **show snmp-user**	SNMP ユーザの設定を表示します。
ステップ 5	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、SAN 管理者ユーザを設定し、ユーザ アカウントおよび SNMP ユーザ設定を表示する例を示します。

```
switch# configure terminal
switch(config)# username user1 role san-admin password xyz123
switch(config)# show user-account
user:admin
     this user account has no expiry date
     roles:network-admin
user:user1
     this user account has no expiry date
     roles:san-admin
switch(config) # show snmp user
_____________________________________________________________________
     SNMP USERS
_____________________________________________________________________

User     Auth  Priv(enforce)  Groups
____     ____  _____________  ______
admin    md5   des(no)        network-admin
user1    md5   des(no)        san-admin

_____________________________________________________________________
```

```
NOTIFICATION TARGET USES (configured for sending V3 Inform)
_____________________________________________________________________

User     Auth   Priv
____     ____   ____

switch(config) #
```

RBAC の設定

ユーザ ロールおよびルールの作成

指定したルール番号は、ルールが適用される順番を決定します。ルールは降順で適用されます。たとえば、1 つのロールが 3 つのルールを持っている場合、ルール 3 がルール 2 よりも前に適用され、ルール 2 はルール 1 よりも前に適用されます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **role name** *role-name*	ユーザー ロールを指定し、ロール コンフィギュレーション モードを開始します。 *role-name* 引数は、最大 16 文字の英数字の文字列で、大文字と小文字が区別されます。
ステップ 3	switch(config-role) # **rule** *number* {**deny** | **permit**} **command** *command-string*	コマンド ルールを設定します。 *command-string* には、スペースおよび正規表現を含めることができます。たとえば、「interface ethernet *」は、すべてのイーサネット インターフェイスが含まれます。 必要な規則の数だけこのコマンドを繰り返します。
ステップ 4	switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**}	すべての操作の読み取り専用ルールまたは読み取り/書き込みルールを設定します。
ステップ 5	switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**} **feature** *feature-name*	機能に対して、読み取り専用規則か読み取りと書き込みの規則かを設定します。

	Command or Action	Purpose
		機能リストを表示するには、**show role feature** コマンドを使用します。 必要な規則の数だけこのコマンドを繰り返します。
ステップ 6	switch(config-role)# **rule** *number* {**deny** | **permit**} {**read** | **read-write**} **feature-group** *group-name*	機能グループに対して、読み取り専用規則か読み取りと書き込みの規則かを設定します。 機能グループのリストを表示するには、**show role feature-group** コマンドを使用します。 必要な規則の数だけこのコマンドを繰り返します。
ステップ 7	(Optional) switch(config-role)# **description** *text*	ロールの説明を設定します。説明にはスペースも含めることができます。
ステップ 8	switch(config-role)# **end**	ロール コンフィギュレーション モードを終了します。
ステップ 9	(Optional) switch# **show role**	ユーザ ロールの設定を表示します。
ステップ 10	(Optional) switch# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、ユーザ ロールを作成してルールを指定する例を示します。

```
switch# configure terminal
switch(config)# role name UserA
switch(config-role)# rule deny command clear users
switch(config-role)# rule deny read-write
switch(config-role)# description This role does not allow users to use clear commands
switch(config-role)# end
switch(config)# show role
```

機能グループの作成

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **role feature-group** *group-name*	ユーザー ロール機能グループを指定して、ロール機能グループ コンフィギュレーション モードを開始します。 *group-name* は、最大 32 文字の英数字の文字列で、大文字と小文字が区別されます。
ステップ 3	switch(config) # **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。
ステップ 4	(Optional) switch# **show role feature-group**	ロール機能グループ設定を表示します。
ステップ 5	(Optional) switch# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、機能グループを作成する例を示します。

```
switch# configure terminal
switch(config) # role feature-group group1
switch(config) # exit
switch# show role feature-group
switch# copy running-config startup-config
switch#
```

ユーザ ロール インターフェイス ポリシーの変更

ユーザー ロール インターフェイス ポリシーを変更することで、ユーザーがアクセスできるインターフェイスを制限できます。ロールがアクセスできるインターフェイスのリストを指定します。これを必要なインターフェイスの数だけ指定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

	Command or Action	Purpose
ステップ 2	switch(config) # **role name** *role-name*	ユーザー ロールを指定し、ロール コンフィギュレーション モードを開始します。
ステップ 3	switch(config-role) # **interface policy deny**	ロール インターフェイス ポリシー コンフィギュレーション モードを開始します。
ステップ 4	switch(config-role-interface) # **permit interface** *interface-list*	ロールがアクセスできるインターフェイスのリストを指定します。 必要なインターフェイスの数だけこのコマンドを繰り返します。 このコマンドの場合、イーサネット インターフェイスを指定できます。
ステップ 5	switch(config-role-interface) # **exit**	ロール インターフェイス ポリシー コンフィギュレーション モードを終了します。
ステップ 6	(Optional) switch(config-role) # **show role**	ロール設定を表示します。
ステップ 7	(Optional) switch(config-role) # **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Example

次に、ユーザーがアクセスできるインターフェイスを制限するために、ユーザー ロール インターフェイス ポリシーを変更する例を示します。

```
switch# configure terminal
switch(config)# role name UserB
switch(config-role)# interface policy deny
switch(config-role-interface)# permit interface ethernet 2/1
switch(config-role-interface)# permit interface fc 3/1
switch(config-role-interface)# permit interface vfc 30/1
```

ユーザ ロール VLAN ポリシーの変更

ユーザー ロール VLAN ポリシーを変更することで、ユーザーがアクセスできる VLAN を制限できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **role name** *role-name*	ユーザー ロールを指定し、ロール コンフィギュレーション モードを開始します。
ステップ 3	switch(config-role)# **vlan policy deny**	ロール VLAN ポリシー コンフィギュレーション モードを開始します。
ステップ 4	switch(config-role-vlan # **permit vlan** *vlan-list*	ロールがアクセスできる VLAN の範囲を指定します。 必要な VLAN の数だけこのコマンドを繰り返します。
ステップ 5	switch(config-role-vlan) # **exit**	ロール VLAN ポリシー コンフィギュレーション モードを終了します。
ステップ 6	(Optional) switch# **show role**	ロール設定を表示します。
ステップ 7	(Optional) switch# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

ユーザ ロール VSAN ポリシーの変更

ユーザー ロール VSAN ポリシーを変更して、ユーザーがアクセスできる VSAN を制限できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config-role) # **role name** *role-name*	ユーザー ロールを指定し、ロール コンフィギュレーション モードを開始します。
ステップ 3	switch(config-role) # **vsan policy deny**	ロール VSAN ポリシー コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 4	switch(config-role-vsan) # **permit vsan** *vsan-list*	ロールがアクセスできる VSAN 範囲を指定します。 必要な VSAN の数だけ、このコマンドを繰り返します。
ステップ 5	switch(config-role-vsan) # **exit**	ロール VSAN ポリシー コンフィギュレーション モードを終了します。
ステップ 6	(Optional) switch# **show role**	ロール設定を表示します。
ステップ 7	(Optional) switch# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

ユーザー アカウントと RBAC の設定の確認

次のいずれかのコマンドを使用して、設定を確認します。

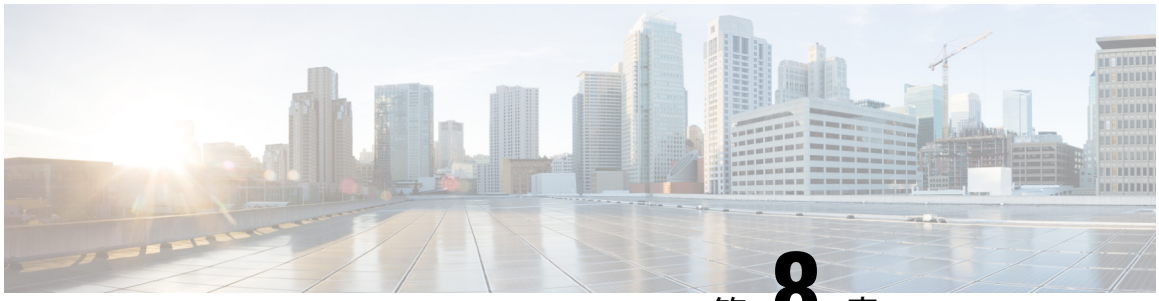
コマンド	目的
show role [*role-name*]	ユーザー ロールの設定を表示します。
show role feature	機能リストを表示します。
show role feature-group	機能グループの設定を表示します。
show startup-config security	スタートアップ コンフィギュレーションのユーザ アカウント設定を表示します。
show running-config security [**all**]	実行コンフィギュレーションのユーザ アカウント設定を表示します。**all** キーワードを指定すると、ユーザ アカウントのデフォルト値が表示されます。
show user-account	ユーザ アカウント情報を表示します。

ユーザー アカウントおよび RBAC のユーザー アカウント デフォルト設定

次の表に、ユーザー アカウントおよび RBAC パラメータのデフォルト設定を示します。

Table 4: デフォルトのユーザー アカウントおよび *RBAC* パラメータ

パラメータ	デフォルト
ユーザ アカウント パスワード	未定義。
ユーザー アカウントの有効期限	なし。
インターフェイス ポリシー	すべてのインターフェイスにアクセス可能。
VLAN ポリシー	すべての VLAN にアクセス可能。
VFC ポリシー	すべての VFC にアクセス可能。
VETH ポリシー	すべての VETH にアクセス可能。



第 8 章

システムメッセージロギングの設定

この章は、次の項で構成されています。

システム メッセージ ロギングの概要

システムメッセージロギングを使用して宛先を制御し、システムプロセスが生成するメッセージの重大度をフィルタリングできます。端末セッション、ログ ファイル、およびリモート システム上の Syslog サーバへのロギングを設定できます。

システムメッセージのフォーマットおよびデバイスが生成するメッセージの詳細については、『*Cisco NX-OS System Messages Reference*』を参照してください。

デフォルトでは、Cisco Nexus デバイスはメッセージをターミナル セッションへ出力します。

デフォルトでは、スイッチはシステム メッセージをログ ファイルに記録します。

次の表に、システムメッセージで使用されている重大度を示します。重大度を設定する場合、システムはそのレベル以下のメッセージを出力します。

Table 5: **システム メッセージの重大度**

レベル	説明
0：緊急	システムが使用不可
1：アラート	即時処理が必要
2：クリティカル	クリティカル状態
3：エラー	エラー状態

レベル	説明
4：警告	警告状態
5：通知	正常だが注意を要する状態
6：情報	単なる情報メッセージ
7：デバッグ	デバッグ実行時にのみ表示

重大度 0、1、または 2 の最新のメッセージを 100 個まで不揮発性 RAM（NVRAM）ログに記録します。NVRAM へのロギングは設定できません。

メッセージを生成したファシリティと重大度に基づいて記録するシステム メッセージを設定できます。

Syslogサーバ

syslog サーバーは、syslog プロトコルに基づいてシステム メッセージを記録するよう設定されたリモート システムで稼働します。最大 8 台の syslog サーバーにログを送信するように Cisco Nexus シリーズ スイッチを設定できます。CFS が有効の場合は、最大 3 台の syslog サーバーを構成できます。

ファブリック内のすべてのスイッチで syslog サーバーの同じ構成をサポートするために、Cisco Fabric Services（CFS）を使用して syslog サーバー構成を配布できます。

Note スイッチを最初に初期化する場合、ネットワークが初期化されてからメッセージが Syslog サーバーに送信されます。

セキュアな Syslog サーバ

Cisco NX-OS リリース 9.2(1) 以降では、リモート ロギング サーバへのセキュアな TLS トランスポート接続をサポートするように Syslog サーバを設定できます。さらに、相互認証の設定によって NX-OS スイッチ（クライアント）のアイデンティティを強化することができます。NX-OS スイッチの場合、この機能は TLSv1.1 および TLSv1.2 をサポートします。

セキュアな Syslog サーバの機能では、デバイス認証および暗号化を提供するために TCP/TLS トランスポートおよびセキュリティ プロトコルを使用します。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）Cisco NX-OS デバイスが、ロギングにセキュアな接続をサポートする（サーバとして機能している）リモート Syslog サーバ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。

システムメッセージロギングの注意事項および制約事項

システム メッセージ ロギングには、次の注意事項と制約事項があります。

- システム メッセージは、デフォルトでコンソールおよびログ ファイルに記録されます。
- Cisco Nexus 3000 シリーズのプラットフォームの Syslog は、MAC の衝突イベントを示します。syslog メッセージには、送信元 MAC アドレス、VLAN、内部ポートの番号情報などの詳細が含まれています。さまざまなセットアップで観察されるように、テーブルの使用率が約 75 % になると、MAC の衝突は普通に発生し、予想されるものです。次の syslog の例を参照してください。 `2015 Mar 26 06:20:37 switch%-SLOT1-5-BCM_L2_HASH_COLLISION: L2 ENTRY unit=0 mac=00:11:11:f7:46:40 vlan=1998 port=0x0800082e.`
- Cisco NX-OS リリース 9.2(1) 以降では、リモート ロギング サーバへのセキュアな TLS トランスポート接続をサポートするように Syslog サーバを設定できます。この機能は、TLSv1.1 および TLSv1.2 をサポートします。

システム メッセージ ロギングのデフォルト設定

次の表に、システム メッセージ ロギング パラメータのデフォルト設定を示します。

Table 6: **デフォルトのシステム メッセージ ロギング パラメータ**

パラメータ	デフォルト
コンソール ロギング	重大度 2 でイネーブル
モニタ ロギング	重大度 2 でイネーブル
ログ ファイル ロギング	重大度 5 のメッセージ ロギングがイネーブル
モジュール ロギング	重大度 5 でイネーブル
ファシリティ ロギング	イネーブル
タイムスタンプ単位	秒
Syslog サーバ ロギング	ディセーブル
Syslog サーバ設定の配布	無効化

システム メッセージ ロギングの設定

ターミナル セッションへのシステム メッセージ ロギングの設定

コンソール、Telnet、およびセキュア シェル セッションに対するシビラティ（重大度）によって、メッセージを記録するようスイッチを設定できます。

デフォルトでは、ターミナル セッションでロギングはイネーブルです。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **terminal monitor**	コンソールから現在の端末セッションに syslog メッセージをコピーします。
ステップ 2	switch# **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。
ステップ 3	switch(config)# **logging console** [*severity-level*]	指定されたシビラティ（重大度）（またはそれ以上）に基づくコンソール セッションへのメッセージの記録をイネーブルにします（数字が小さいほうがシビラティ（重大度）が高いことを示します）。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 重大度が指定されていない場合、デフォルトの 2 が使用されます。
ステップ 4	(Optional) switch(config)# **no logging console** [*severity-level*]	コンソールへのロギング メッセージをディセーブルにします。

	Command or Action	Purpose
ステップ 5	switch(config)# **logging monitor** [*severity-level*]	指定されたシビラティ（重大度）（またはそれ以上）に基づくモニターへのメッセージの記録をイネーブルにします（数字が小さいほうがシビラティ（重大度）が高いことを示します）。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 重大度が指定されていない場合、デフォルトの 2 が使用されます。 設定は Telnet および SSH セッションに適用されます。
ステップ 6	(Optional) switch(config)# **no logging monitor** [*severity-level*]	Telnet および SSH セッションへのメッセージ ロギングをディセーブルにします。
ステップ 7	(Optional) switch# **show logging console**	コンソール ロギング設定を表示します。
ステップ 8	(Optional) switch# **show logging monitor**	モニタ ロギング設定を表示します。
ステップ 9	(Optional) switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Example

次に、コンソールのロギング レベルを 3 に設定する例を示します。

```
switch# configure terminal
switch(config)# logging console 3
```

次に、コンソールのロギングの設定を表示する例を示します。

```
switch# show logging console
Logging console:                enabled (Severity: error)
```

次に、コンソールのロギングをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# no logging console
```

次に、ターミナル セッションのロギング レベルを 4 に設定する例を示します。

```
switch# terminal monitor
switch# configure terminal
switch(config)# logging monitor 4
```

次に、ターミナル セッションのロギングの設定を表示する例を示します。

```
switch# show logging monitor
Logging monitor:                enabled (Severity: warning)
```

次に、ターミナル セッションのロギングをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# no logging monitor
```

ファイルへのシステム メッセージ ロギングの設定

システム メッセージをファイルに記録するようスイッチを設定できます。デフォルトでは、システム メッセージはファイル log:messages に記録されます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **logging logfile** *logfile-name severity-level* [**size** *bytes*]	システム メッセージを保存するのに使用するログ ファイルの名前と、記録する最小シビラティ（重大度）を設定します。任意で最大ファイル サイズを指定できます。デフォルトの重大度は 5 です。ファイル サイズは 4194304 です。 重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート

	Command or Action	Purpose
		• 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ ファイル サイズは 4096 ～ 10485760 バイトです。
ステップ 3	(Optional) switch(config)# **no logging logfile** [*logfile-name severity-level* [**size** *bytes*]]	ログ ファイルへのロギングをディセーブルにします。任意で最大ファイル サイズを指定できます。デフォルトの重大度は 5 です。ファイル サイズは 4194304 です。
ステップ 4	(Optional) switch# **show logging info**	ロギング設定を表示します。任意で最大ファイル サイズを指定できます。デフォルトの重大度は 5 です。ファイル サイズは 4194304 です。
ステップ 5	(Optional) switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Example

次に、システム メッセージをファイルに記録するようスイッチを設定する例を示します。

```
switch# configure terminal
switch(config)# logging logfile my_log 6 size 4194304
```

次の例は、ロギング設定の表示方法を示しています（簡潔にするため、一部の出力が削除されています）。

```
switch# show logging info
Logging console:                enabled (Severity: debugging)
Logging monitor:                enabled (Severity: debugging)

Logging timestamp:              Seconds
Logging server:                 disabled
Logging logfile:                enabled
        Name - my_log: Severity - informational Size - 4194304
Facility        Default Severity        Current Session Severity
--------        ----------------        ------------------------
```

```
aaa                     3                       3
aclmgr                          3                               3
afm                     3                       3
altos                   3                       3
auth                    0                       0
authpriv                3                       3
bootvar                 5                       5
callhome                2                       2
capability              2                       2
cdp                     2                       2
cert_enroll             2                       2
...
```

モジュールおよびファシリティ メッセージのロギングの設定

モジュールおよびファシリティに基づいて記録するメッセージの重大度およびタイムスタンプの単位を設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **logging module** [*severity-level*]	指定された重大度またはそれ以上の重大度であるモジュール ログ メッセージをイネーブルにします。重大度は 0 ～ 7 の範囲です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 重大度が指定されていない場合、デフォルトの 5 が使用されます。
ステップ 3	switch(config)# **logging level** *facility severity-level*	指定された重大度またはそれ以上の重大度である指定のファシリティからのロギ

	Command or Action	Purpose
		ングメッセージをイネーブルにします。重大度は 0 ～ 7 です。 • 0：緊急 • 1：アラート • 2：クリティカル • 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 同じ重大度をすべてのファシリティに適用するには、all ファシリティを使用します。デフォルト値については、**show logging level** コマンドを参照してください。 **Note** リリース 7.0(3)I2(1) 以降、BCM_USD、ETHPC、FWM、およびNOHMS プロセスのログ レベルは設定できません。BCM_USD プロセスの場合、**attach module 1** コマンドを使用して、ログ レベルを設定します。 **Note** コンポーネントの現行セッションのシビラティ（重大度）がデフォルトのシビラティ（重大度）と同じ場合には、実行中のコンフィギュレーションでそのコンポーネントのログ レベルが表示されないことが予想されます。デフォルトのログ レベルは、実行中のコンフィギュレーションでは表示されませんが、**show logging level** コマンドで表示されます。
ステップ 4	(Optional) switch(config)# **no logging module** [*severity-level*]	モジュール ログ メッセージをディセーブルにします。

	Command or Action	Purpose
ステップ 5	(Optional) switch(config)# **no logging level** [*facility severity-level*]	指定されたファシリティのロギングシビラティ（重大度）をデフォルト レベルにリセットします。ファシリティおよびシビラティ（重大度）を指定しないと、スイッチはすべてのファシリティをデフォルト レベルにリセットします。
ステップ 6	(Optional) switch# **show logging module**	モジュールロギング設定を表示します。
ステップ 7	(Optional) switch# **show logging level** [*facility*]	ファシリティごとに、ロギング レベル設定およびシステムのデフォルト レベルを表示します。ファシリティを指定しないと、スイッチはすべてのファシリティのレベルを表示します。
ステップ 8	(Optional) switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Example

次に、モジュールおよび特定のファシリティ メッセージのシビラティ（重大度）を設定する例を示します。

```
switch# configure terminal
switch(config)# logging module 3
switch(config)# logging level aaa 2
```

ロギング タイムスタンプの設定

Cisco Nexus シリーズ スイッチによって記録されるメッセージのタイムスタンプの単位を設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}	ロギング タイムスタンプ単位を設定します。デフォルトでは、単位は秒です。

	Command or Action	Purpose
ステップ 3	(Optional) switch(config)# **no logging timestamp** {**microseconds** | **milliseconds** | **seconds**}	ロギング タイムスタンプ単位をデフォルトの秒にリセットします。
ステップ 4	(Optional) switch# **show logging timestamp**	設定されたロギング タイムスタンプ単位を表示します。
ステップ 5	(Optional) switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

Example

次に、メッセージのタイムスタンプ単位を設定する例を示します。

```
switch# configure terminal
switch(config)# logging timestamp milliseconds
switch(config)# exit
switch# show logging timestamp
Logging timestamp:              Milliseconds
```

RFC 5424 に準拠したロギング syslog の構成

コマンドは、次の方法で変更できます：

- [*no*] **logging rfc-strict 5424**
- **show logging rfc-strict 5424**

手順

	コマンドまたはアクション	目的
ステップ 1	`switch(config)#`[*no*] logging rfc-strict 5424	（オプション）コマンドを無効にするか、またはそのデフォルトに設定します
ステップ 2	`switch(config)#` **logging rfc-strict 5424**	メッセージ ロギング ファシリティを変更し、メッセージが準拠する必要のある RFC を設定します。
ステップ 3	`switch(config)#`show logging rfc-strict 5424	RFC 5424 に準拠する syslog を表示します

ACL ロギング キャッシュの設定

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **logging ip access-list cache entries** *num_entries*	ソフトウェア内にキャッシュする最大ログ エントリ数を設定します。範囲は 0 ～ 1000000 エントリです。デフォルト値は 8000 エントリです。
ステップ 3	switch(config)# **logging ip access-list cache interval** *seconds*	ログの更新の間隔を秒数で設定します。この時間中エントリが非アクティブの場合、キャッシュから削除されます。指定できる範囲は 5 ～ 86400 秒です。デフォルト値は 300 秒です。
ステップ 4	switch(config)# **logging ip access-list cache threshold** *num_packets*	エントリがログに記録されるまでに一致するパケット数を設定します。範囲は 0 ～ 1000000 パケットです。デフォルト値は 0 パケットです。つまり、パケットの一致数によってロギングがトリガーされることはありません。
ステップ 5	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、ログ エントリの最大数を 5000、間隔を 120 秒、しきい値を 500000 に設定する例を示します。

```
switch# configure terminal
switch(config)# logging ip access-list cache entries 5000
switch(config)# logging ip access-list cache interval 120
switch(config)# logging ip access-list cache threshold 500000
switch(config)# copy running-config startup-config
```

インターフェイスへの ACL ロギングの適用

始める前に

- ロギング用に設定された少なくとも 1 つのアクセス コントロール エントリ（ACE）で IP アクセス リストを作成します。
- ACL ロギング キャッシュを設定します。
- ACL ログの一致レベルを設定します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface mgmt0**	mgmt0 インターフェイスを指定します。
ステップ 3	switch(config-if)# **ip access-group** *name* **in**	指定したインターフェイスの入力トラフィックで ACL ロギングをイネーブルにします。
ステップ 4	（任意） switch(config-if)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、すべての入力トラフィックに対して acl1 で指定されたロギングに mgmt0 インターフェイスを適用する例を示します。

```
switch# configure terminal
switch(config)# interface mgmt0
switch(config-if)# ip access-group acl1 in
switch(config-if)# copy running-config startup-config
```

Source-Interface ロギングの設定

syslog メッセージがどのインターフェイスを使用してルータを出るかにかかわらず、syslog サーバーに送信されるすべてのシステム ロギング（syslog）メッセージに、送信元アドレスと同じ IP アドレスを含めるように設定できます。送信元インターフェイスで指定されている syslog パケットにユーザー設定の送信元 IP を設定できます。

（注） 有効な IP アドレスが割り当てられていない場合、syslog が作成され、メッセージが出口インターフェイス IP アドレスとともに送信されます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# [**no**] **logging source-interface** [**ethernet** *slot/port* | **loopback** *interface-number* | **mgmt** *interface-number* | **port-channel** *port channel-number* | **vlan** *interface-number* | **tunnel** *interface-number*]	• **ethernet**：イーサネット オプションの送信元インターフェイスの範囲は 1 ～ 253 です。 • **loopback**：ループバック オプションの送信元インターフェイスの範囲は 1 ～ 1023 です。 • **mgmt**：管理オプションの送信元インターフェイスのインターフェイス番号は 0 です。 • **port-channel**：ポート チャネル オプションの送信元インターフェイスの範囲は 1 ～ 4096 です。
ステップ 3	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、送信元インターフェイスをイーサネット インターフェイスとして設定する例を示します。

```
switch# configure terminal
switch(config)# logging source-interface ethernet 2/1
switch(config)# copy running-config startup-config
```

ACL ログの一致レベルの設定

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **acllog match-log-level** *number*	ACL ログ（acllog）で記録されるエントリと一致するようにログ レベルを指定します。*number* は 0 ～ 7 までの値です。デフォルト値は 6 です。 （注） ログに入力するログ メッセージでは、ACL ログ ファシリティ（acllog）のログ レベルとログ ファイルのロギングシビラティ（重大度）は、ACL ログの一致ログ レベル設定よりも大きいか、同じです。詳細については、「モジュールおよびファシリティ メッセージのロギングの設定 （114 ページ）」および「ファイルへのシステム メッセージ ロギングの設定 （112ページ）」を参照してください。
ステップ 3	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

syslog サーバの設定

システム メッセージを記録する、リモート システムを参照する syslog サーバーを最大で 8 台設定できます。

Note シスコは、管理仮想ルーティングおよび転送（VRF）インスタンスを使用するサーバとして、syslog サーバを設定することを推奨します。VRF の詳細情報については、『Cisco Nexus 3000 Series NX-OS Unicast Routing Configuration Guide』を参照してください。

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		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	**logging server** *host* [*severity-level* [**use-vrf** *vrf-name* [**facility** *facility*]]] **Example:** `switch(config)# logging server 172.28.254.254 5 use-vrf default facility local3`	ホストが syslog メッセージを受信するように設定します。 • *host* 引数は、syslog サーバー ホストのホスト名または IPv4 または IPv6 アドレスを示します。 • *severity-level* 引数は、指定したレベルに syslog サーバーへのメッセージのロギングを制限します。シビラティ（重大度）は 0 ～ 7 の範囲です。Table 5: システム メッセージの重大度 , on page 107を参照してください。 • **use vrf** *vrf-name* キーワードは、VRF 名のデフォルトまたは管理値を示します。特定の VRF が指定されない場合は、management がデフォルトです。 **show running** コマンドの出力には、次の構成シナリオに基づいて VRF が表示される場合と表示されない場合があります： • VRF が構成されていない場合、システムは管理 VRF をデフォルトとして使用します。この VRF は出力に表示されません。 • 管理 VRF を構成していたとします。この場合、この VRF はデフォルトとして識別されるため、出力には表示されません。 • 他の VRF を構成していたとします。それから、この VRF が出力に表示されます。

	Command or Action	Purpose
		Note 現在の Cisco Fabric Services（CFS）配信では VRF をサポートしていません。CFS 配信がイネーブルの場合、デフォルト VRF で構成されているロギング サーバーは管理 VRF として配布されます。 • facility 引数は syslog ファシリティ タイプを指定します。デフォルトの発信ファシリティは local7 です。 ファシリティは、使用している Cisco Nexus シリーズ ソフトウェアのコマンド リファレンスに記載されています。 **Note** デバッグは CLI ファシリティですが、デバッグの syslog はサーバーに送信されません。
ステップ 3	(Optional) **no logging server** *host* **Example:** `switch(config)# no logging server 172.28.254.254 5`	指定されたホストのロギング サーバーを削除します。
ステップ 4	(Optional) **show logging server** **Example:** `switch# show logging server`	Syslog サーバー構成を表示します。
ステップ 5	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、syslog サーバーを設定する例を示します。

```
switch# configure terminal
switch(config)# logging server 172.28.254.254 5
use-vrf default facility local3

switch# configure terminal
```

```
switch(config)# logging server 172.28.254.254 5 use-vrf management facility local3
```

UNIX または Linux システムでの syslog の設定

/etc/syslog.conf ファイルに次の行を追加して、UNIX または Linux システム上に syslog サーバーを設定できます。

facility.level <five tab characters> *action*

次の表に、設定可能な syslog フィールドを示します。

Table 7: syslog.conf の syslog フィールド

フィールド	説明
Facility	メッセージの作成者。auth、authpriv、cron、daemon、kern、lpr、mail、mark、news、syslog、user、local0 ～ local7 です。アスタリスク（*）を使用するとすべてを指定します。これらのファシリティ指定により、発信元に基づいてメッセージの宛先を制御できます。 **Note** ローカル ファシリティを使用する前に設定をチェックします。
Level	メッセージを記録する最小重大度。debug、info、notice、warning、err、crit、alert、emerg です。アスタリスク（*）を使用するとすべてを指定します。none を使用するとファシリティをディセーブルにできます。
Action	メッセージの宛先。ファイル名、前にアットマーク（@）が付いたホスト名、カンマで区切られたユーザー リストです。アスタリスク（*）を使用するとすべてのログイン ユーザーを指定します。

Procedure

ステップ 1 /etc/syslog.conf ファイルに次の行を追加して、ファイル /var/log/myfile.log に local7 ファシリティのデバッグ メッセージを記録します。

```
debug.local7            /var/log/myfile.log
```

ステップ 2 シェル プロンプトで次のコマンドを入力して、ログ ファイルを作成します。

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

ステップ 3 次のコマンドを入力して、システム メッセージ ロギング デーモンが myfile.log をチェックして、新しい変更を取得するようにします。

```
$ kill -HUP ~cat /etc/syslog.pid~
```

セキュアな Syslog サーバの設定

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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**] **logging server** *host* [*severity-level* [**port** *port-number*][**secure**[**trustpoint client-identity** *trustpoint-name*]][**use-vrf** *vrf-name*]]
	例：
	`switch(config)# logging server 192.0.2.253 secure`
	例：
	`switch(config)# logging server 2001::3 5 secure trustpoint client-identity myCA use-vrf red`
		指定されたホスト名、あるいは IPv4 または IPv6 アドレスで Syslog サーバを設定します。必要に応じて、CA によって署名されるクライアント アイデンティティ証明書をインストールし、trustpoint client-identity オプションを使用することで相互認証を適用できます。
		セキュアな TLS 接続のデフォルト宛先ポートは 6514 です。

ステップ 3	（任意） **logging source-interface** *interface name* 例： `switch(config)# logging source-interface lo0`	リモート Syslog サーバの送信元インターフェイスをイネーブルにします。
ステップ 4	（任意） **show logging server** 例： `switch(config)# show logging server`	Syslog サーバ設定を表示します。secure オプションを設定する場合、出力のエントリにトランスポート情報が含まれるようになります。デフォルトでは、secure オプションが設定されていない場合、トランスポートは UDP です。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

CA 証明書の設定

セキュアな Syslog 機能のサポートには、トラストポイントの設定によってリモート サーバを認証する必要があります。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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**] **crypto ca trustpoint** *trustpoint-name*
	例：
	`switch(config)# crypto ca trustpoint winca`
	`switch(config-trustpoint)#`
		トラストポイントを設定します。
		（注）
		トラストポイントの設定の前に ip domain-name を設定する必要があります。

ステップ 3	必須: **crypto ca authenticate** *trustpoint-name* 例： `switch(config-trustpoint)# crypto ca authenticate winca`	トラストポイントの CA 証明書を設定します。
ステップ 4	（任意） **show crypto ca certificate** 例： `switch(config)# show crypto ca certificates`	設定されている証明書/チェーンと、関連付けられているトラストポイントを表示します。
ステップ 5	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	デバイスのリロード後にトラストポイントが持続されるように、実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします。

CA 証明書の登録

NX-OS スイッチ（クライアント）が識別するようリモートサーバによって要求される相互認証では、ピア認証が必須であるため、これは証明書をスイッチに登録するための追加設定です。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	必須: **crypto key generate rsa label** *key name* **exportable modules 2048** 例： `switch(config-trustpoint)# crypto key generate rsa label myKey exportable modulus 2048`	RSA キー ペアを設定します。デフォルトでは、Cisco NX-OS ソフトウェアは 1024 ビットの RSA キーを作成します。

ステップ 3	[**no**] **crypto ca trustpoint** *trustpoint-name*
	例：
	`switch(config)# crypto ca trustpoint myCA`
	`switch(config-trustpoint)#`
		トラストポイントを設定します。
		（注）
		トラストポイントの設定の前に ip domain-name を設定する必要があります。

ステップ 4	必須: **rsakeypair** *key-name* 例： `switch(config-trustpoint)# rsakeypair myKey`	トラストポイント CA に生成されたキーペアを関連付けます。
ステップ 5	**crypto ca trustpoint** *trustpoint-name* 例： `switch(config)# crypto ca authenticate myCA`	トラストポイントの CA 証明書を設定します。
ステップ 6	[**no**] **crypto ca enroll** *trustpoint-name* 例： `switch(config)# crypto ca enroll myCA`	CA に登録するスイッチのアイデンティティ証明書を生成します。
ステップ 7	**crypto ca import** *trustpoint-name* **certificate** 例： `switch(config-trustpoint)# crypto ca import myCA certificate`	CA によって署名されたアイデンティティ証明書をスイッチにインポートします。
ステップ 8	（任意） **show crypto ca certificates** 例： `switch# show crypto ca certificates`	設定されている証明書またはチェーンと、関連付けられているトラストポイントを表示します。

	コマンドまたはアクション	目的
ステップ 9	必須: **copy running-config startup-config** 例： `switch# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

syslog サーバー設定の配布の設定

Cisco Fabric Services（CFS）インフラストラクチャを使用して、ネットワーク内の他のスイッチへ Syslog サーバー設定を配布できます。

Syslog サーバー設定の配布をイネーブルにすると、配布設定をコミットする前に Syslog サーバー設定を変更し、保留中の変更を表示できます。配布がイネーブルである限り、スイッチは Syslog サーバー設定に対する保留中の変更を維持します。

Note スイッチを再起動すると、揮発性メモリに保存されている syslog サーバー設定の変更は失われることがあります。

Before you begin

1 つまたは複数の syslog サーバーを設定しておく必要があります。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **logging distribute**	CFS インフラストラクチャを使用して、ネットワーク スイッチへの syslog サーバー設定の配布をイネーブルにします。デフォルトでは、配布はディセーブルです。
ステップ 3	switch(config)# **logging commit**	ファブリック内のスイッチへ配布するための Syslog サーバー設定に対する保留中の変更をコミットします。
ステップ 4	switch(config)# **logging abort**	Syslog サーバー設定に対する保留中の変更をキャンセルします。
ステップ 5	(Optional) switch(config)# **no logging distribute**	CFS インフラストラクチャを使用して、ネットワーク スイッチへの syslog サーバー設定の配布をディセーブルにしま

	Command or Action	Purpose
		す。設定変更が保留中の場合は、配布をディセーブルにできません。**logging commit** および **logging abort** コマンドを参照してください。デフォルトでは、配布はディセーブルです。
ステップ 6	(Optional) switch# **show logging pending**	Syslog サーバー設定に対する保留中の変更を表示します。
ステップ 7	(Optional) switch# **show logging pending-diff**	syslog サーバー設定の保留中の変更に対して、現在の syslog サーバー設定との違いを表示します。
ステップ 8	(Optional) switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ログ ファイルの表示およびクリア

ログ ファイルおよび NVRAM のメッセージを表示したり消去したりできます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **show logging last** *number-lines*	ロギング ファイルの最終行番号を表示します。最終行番号には 1 ～ 9999 を指定できます。
ステップ 2	switch# **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	入力されたスパン内にタイム スタンプがあるログ ファイルのメッセージを表示します。終了時間を入力しないと、現在の時間が使用されます。月の時間フィールドには3文字を、年と日の時間フィールドには数値を入力します。
ステップ 3	switch# **show logging nvram** [**last** *number-lines*]	NVRAM のメッセージを表示します。表示される行数を制限するには、表示する最終行番号を入力できます。最終行番号には 1 ～ 100 を指定できます。
ステップ 4	switch# **clear logging logfile**	ログ ファイルの内容をクリアします。
ステップ 5	switch# **clear logging nvram**	NVRAM の記録されたメッセージをクリアします。

Example

次に、ログ ファイルのメッセージを表示する例を示します。

```
switch# show logging last 40
switch# show logging logfile start-time 2007 nov 1 15:10:0
switch# show logging nvram last 10
```

次に、ログ ファイルのメッセージをクリアする例を示します。

```
switch# clear logging logfile
switch# clear logging nvram
```

システム メッセージ ロギングの設定確認

システム メッセージのロギング設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show logging console	コンソール ロギング設定を表示します。
show logging info	ロギング設定を表示します。
show logging ip access-list cache	IP アクセス リスト キャッシュを表示します。
show logging ip access-list cache detail	IP アクセス リスト キャッシュに関する詳細情報を表示します。
show logging ip access-list status	IP アクセス リスト キャッシュのステータスを表示します。
show logging last *number-lines*	ログ ファイルの末尾から指定行数を表示します。
show logging level [*facility*]	ファシリティ ロギングシビラティ（重大度）設定を表示します。
show logging logfile [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	ログ ファイルのメッセージを表示します。
show logging module	モジュール ロギング設定を表示します。
show logging monitor	モニタ ロギング設定を表示します。
show logging nvram [**last** *number-lines*]	NVRAM ログのメッセージを表示します。
show logging pending	Syslog サーバーの保留中の配布設定を表示します。

コマンド	目的
show logging pending-diff	Syslog サーバーの保留中の配布設定の違いを表示します。
show logging server	Syslog サーバー設定を表示します。
show logging session	ロギング セッションのステータスを表示します。
show logging status	ロギング ステータスを表示します。
show logging timestamp	ロギング タイムスタンプ単位設定を表示します。
show running-config acllog	ACL ログ ファイルの実行コンフィギュレーションを表示します。

繰り返されるシステム ロギング メッセージ

システム プロセスはロギング メッセージを生成します。生成される重大度レベルを制御するために使用されるフィルタによっては、多数のメッセージが生成され、その多くが繰り返されます。

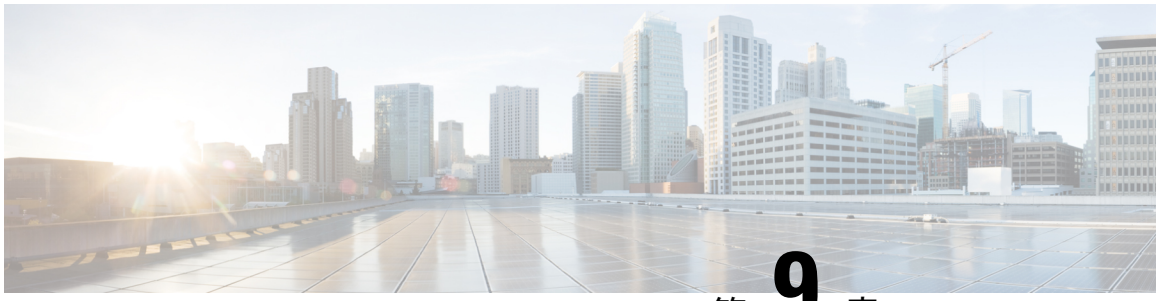
ロギング メッセージの量を管理するスクリプトの開発を容易にし、**show logging log** コマンドの出力の「フラッディング」から繰り返されるメッセージを排除するために、繰り返されるメッセージをロギングする次の方法が使用されます。

以前の方法では、同じメッセージが繰り返された場合、デフォルトでは、メッセージ内でメッセージが再発生した回数が示されていました。

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting
Port
Identity is 00:1c:73:ff:ff:ee:f6:e5
2019 Mar 11 13:43:15 Cisco-customer last message repeated 242 times
```

新しいメソッドは、繰り返しメッセージの最後に繰り返し回数を追加するだけです。

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting
Port
Identity is 00:1c:73:ff:ff:ee:f6:e5

2019 Mar 11 13:43:15 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting
Port
Identity is 00:1c:73:ff:ff:ee:f6:e5 (message repeated 242 times)
```



第 9 章

Smart Call Home の設定

この章は、次の項で構成されています。

Smart Call Home に関する情報

Smart Call Home は、重要なシステム イベントを E メールで通知します。Cisco Nexus シリーズ スイッチは、幅広いメッセージ フォーマットを提供し、ポケットベル サービス、標準 E メール、または XML ベースの自動解析アプリケーションと最適な互換性を保てます。この機能を使用して、ネットワーク サポート エンジニアやネットワーク オペレーション センターを呼び出せます。また、Cisco Smart Call Home サービスを使用して、TAC でケースを自動的に生成することもできます。

シスコと直接サービス契約を結んでいる場合は、Smart Call Home サービス用のデバイスを登録できます。Smart Call Home は、ご使用のデバイスから送信された Smart Call Home メッセージを分析し、背景情報および推奨事項を提供して、システムの問題を迅速に解決します。既知と特定できる問題、特に GOLD 診断エラーについては、シスコ TAC によって自動サービス リクエストが生成されます。

Smart Call Home には、次の機能があります。

- 継続的なデバイス ヘルス モニタリングとリアルタイムの診断アラート。
- ご使用のデバイスからの Smart Call Home メッセージの分析と、必要に応じた自動サービス リクエストの生成は、問題を迅速に解決するための詳細な診断情報とともに、適切な TAC チームにルーティングされます。

- セキュアなメッセージ転送が、ご使用のデバイスから直接、またはダウンロード可能な Transport Gateway（TG）集約ポイントを経由して行われます。複数のデバイスでサポートを必要としている場合、またはセキュリティ要件の関係でご使用のデバイスをインターネットに直接接続できない場合は、TG 集約ポイントを使用できます。
- Smart Call Home メッセージと推奨事項、すべての Smart Call Home デバイスのインベントリおよび設定情報、および Field Notice、セキュリティ勧告、およびサポート終了日情報への Web ベースのアクセス。

Smart Call Home の概要

Smart Call Home を使用すると、重要なイベントがデバイスで発生した場合に外部エンティティに通知できます。Smart Call Home では、ユーザーが宛先プロファイルに設定する複数の受信者にアラートが配信されます。

Smart Call Home には、スイッチで事前に定義された一連のアラートが含まれます。これらのアラートはアラート グループにグループ化され、アラート グループのアラートが発生したときに実行する CLI コマンドが割り当てられています。スイッチには、転送された Smart Call Home メッセージのコマンド出力が含まれます。

Smart Call Home 機能には、次のものがあります。

- 関連する CLI コマンド出力の実行および添付が自動化されます。
- 次のような、複数のメッセージ フォーマット オプションがあります。
  - ショート テキスト：ポケットベルまたは印刷されたレポートに適している文字。
  - フルテキスト：人間が判読しやすいように完全にフォーマットされたメッセージ情報です。
  - XML：Extensible Markup Language（XML）および Adaptive Messaging Language（AML）XML スキーマ定義（XSD）を使用した、判読可能なフォーマットです。XML 形式では、シスコ TAC と通信できます。
- 複数のメッセージ宛先への同時配信が可能。各宛先プロファイルには最大 50 件の電子メール宛先アドレスを設定できます。

Smart Call Home 宛先プロファイル

Smart Call Home 宛先プロファイルには、次の情報が含まれています。

- 1 つ以上のアラート グループ：アラートの発生時に、特定の Smart Call Home メッセージを送信するアラートのグループ。
- 1 つ以上の電子メール宛先：この宛先プロファイルに割り当てられたアラート グループによって生成された Smart Call Home メッセージの受信者リスト。

- メッセージ フォーマット：Smart Call Home メッセージのフォーマット（ショート テキスト、フル テキスト、または XML）。
- メッセージシビラティ（重大度）：スイッチが宛先プロファイル内のすべての電子メール アドレスに対して Smart Call Home メッセージを生成するまで、アラートが満たす必要がある Smart Call Home シビラティ（重大度）。アラートの Smart Call Home シビラティ（重大度）が、宛先プロファイルに設定されたメッセージシビラティ（重大度）よりも低い場合、スイッチはアラートを生成しません。

定期メッセージを日別、週別、月別で送信するコンポーネントアラートグループを使用して、定期的なコンポーネント アップデート メッセージを許可するよう宛先プロファイルを設定することもできます。

Cisco Nexus スイッチは、次の定義済み宛先プロファイルをサポートします。

- CiscoTAC-1：XML メッセージ フォーマットの Cisco-TAC アラート グループをサポートします。
- full-text-destination：フル テキスト メッセージ フォーマットをサポートします。
- short-text-destination：ショート テキスト メッセージ フォーマットをサポートします。

Smart Call Home アラート グループ

アラート グループは、すべての Cisco Nexus デバイスでサポートされる Smart Call Home アラートの定義済みサブセットです。アラート グループを使用すると、定義済みまたはカスタム宛先プロファイルに送信する一連の Smart Call Home アラートを選択できます。Smart Call Home アラートが宛先プロファイルにアソシエートされたいずれかのアラート グループに属する場合、およびアラートで、Smart Call Home メッセージシビラティ（重大度）が宛先プロファイルに設定されているメッセージシビラティ（重大度）と同じか、それ以上である場合のみ、スイッチは Smart Call Home アラートを宛先プロファイルの電子メールの宛先に送信します。

次の表に、サポートされるアラート グループと、アラート グループ用に生成された Smart Call Home メッセージに含まれるデフォルトの CLI コマンド出力を示します。

Table 8: アラート グループおよび実行されるコマンド

アラート グループ	説明	実行されるコマンド
Cisco-TAC	Smart Call Home 宛ての、他のアラート グループからのすべてのクリティカル アラート。	アラートを発信するアラート グループに基づいてコマンドを実行します。
診断	診断によって生成されたイベント。	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**

アラートグループ	説明	実行されるコマンド
スーパーバイザ ハードウェア	スーパーバイザ モジュールに関連するイベント。	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**
ラインカード ハードウェア	標準またはインテリジェント スイッチング モジュールに関連するイベント。	**show diagnostic result module all detail** **show moduleshow version** **show tech-support platform callhome**
設定	設定に関連した定期的なイベント。	**show version** **show module** **show running-config all** **show startup-config**
システム	装置の動作に重要なソフトウェア システムの障害によって生成されるイベント	**show system redundancy status** **show tech-support**
環境	電源、ファン、および温度アラームなどの環境検知要素に関連するイベント。	**show environment** **show logging last 1000** **show module show version** **show tech-support platform callhome**
インベントリ	装置がコールド ブートした場合、またはFRU の取り付けまたは取り外しを行った場合に示されるコンポーネント ステータス。このアラートは重要でないイベントであり、情報はステータスおよび使用権に使用されます。	**show module** **show version** **show license usage** **show inventory** **show sprom all** **show system uptime**

Smart Call Home は、syslog のシビラティ（重大度）を、syslog ポート グループ メッセージの対応する Smart Call Home のシビラティ（重大度）に対応させます。

特定のイベントが発生し、Smart Call Home メッセージを含む **show** 出力を送信した場合に、追加の **show** コマンドを実行するために、定義済みのアラート グループをカスタマイズできます。

show コマンドは、フル テキストおよび XML 宛先プロファイルにのみ追加できます。ショート テキスト宛先プロファイルは、128 バイトのテキストに制限されているため、追加の **show** コマンドをサポートしていません。

Smart Call Home のメッセージ レベル

Smart Call Home を使用すると、緊急度に基づいてメッセージをフィルタリングできます。各宛先プロファイル（定義済みおよびユーザー定義）を、Smart Call Home メッセージ レベルしきい値にアソシエートすることができます。宛先プロファイルのこのしきい値よりも小さい値を持つ Smart Call Home メッセージは、スイッチによって生成されません。Smart Call Home メッセージ レベルの範囲は 0（緊急度が最小）～9（緊急度が最大）です。デフォルトは 0 です（スイッチはすべてのメッセージを送信します）。

syslog アラート グループに送信される Smart Call Home メッセージでは、syslog のシビラティ（重大度）が Smart Call Home のメッセージ レベルにマッピングされます。

Note Smart Call Home は、メッセージ テキストで syslog メッセージ レベルを変更しません。

次の表に、各 Smart Call Home メッセージ レベルのキーワードと、syslog ポート アラート グループの対応する syslog レベルを示します。

Table 9: **重大度と *syslog* レベルのマッピング**

Smart Call Home レベル	キーワード	Syslog レベル	説明
9	Catastrophic	該当なし	ネットワーク全体に壊滅的な障害が発生しています。
8	Disaster	該当なし	ネットワークに重大な影響が及びます。
7	Fatal	緊急（0）	システムが使用不可能な状態。
6	Critical	アラート（1）	クリティカルな状況で、すぐに対応する必要があります。
5	Major	重要（2）	重大な状態。
4	Minor	エラー（3）	軽微な状態。
3	警告	警告（4）	警告状態。
2	通知	通知（5）	基本的な通知および情報メッセージです。
1	標準	情報（6）	標準状態に戻ることを示す標準イベントです。
0	Debugging	デバッグ（7）	デバッグ メッセージ。

Call Home のメッセージ形式

Call Home では、次のメッセージ フォーマットがサポートされます。

- ショート テキスト メッセージ フォーマット
- すべてのフル テキストと XML メッセージに共通のフィールド
- 対処的または予防的イベント メッセージに挿入されるフィールド
- コンポーネント イベント メッセージの挿入フィールド
- ユーザーが作成したテスト メッセージの挿入フィールド

次の表に、すべてのメッセージ タイプのショート テキスト書式設定オプションを示します。

Table 10: ショート テキスト メッセージ フォーマット

データ項目	説明
デバイス ID	設定されたデバイス名
日時スタンプ	起動イベントのタイム スタンプ
エラー判別メッセージ	起動イベントの簡単な説明（英語）
アラームの緊急度	システム メッセージに適用されるようなエラー レベル

次の表に、フル テキストまたは XML の共通するイベント メッセージ形式について説明します。

Table 11: すべてのフル テキストと ***XML*** メッセージに共通のフィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
タイム スタンプ	ISO 時刻通知でのイベントの日付/タイム スタンプ *YYYY-MM-DD HH:MM:SS GMT+HH:MM*	/aml/header/time
メッセージ名	メッセージの名前。特定のイベント名は上記の表に記載	/aml/header/name
メッセージ タイプ	リアクティブまたはプロアクティブなどのメッセージ タイプの名前。	/aml/header/type
メッセージ グループ	Syslog などのアラート グループの名前。	/aml/header/group

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
重大度	メッセージの重大度	/aml/header/level
送信元 ID	ルーティングのための製品タイプ	/aml/header/source
デバイス ID	メッセージを生成したエンドデバイスの固有デバイス識別情報（UDI）。メッセージがデバイスに対して固有でない場合は、このフィールドを空にする必要があります。形式は、*type@Sid@serial*。 • *type* は、バックプレーン IDPROM からの製品の型番。 • @ は区切り文字です。 • *Sid* は C で、シリアル ID をシャーシ シリアル番号として特定します。 • *serial* は、Sid フィールドによって識別される番号です。 例：WS-C6509@C@12345678	/aml/ header/deviceID
カスタマー ID	サポート サービスによって契約情報やその他の ID に使用されるオプションのユーザ設定可能なフィールド	/aml/ header/customerID
連絡先 ID	サポート サービスによって契約情報やその他の ID に使用されるオプションのユーザ設定可能なフィールド	/aml/ header /contractID
サイト ID	シスコが提供したサイト ID または別のサポート サービスにとって意味のあるその他のデータに使用されるオプションのユーザ設定可能なフィールド	/aml/ header/siteID

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
サーバー ID	デバイスからメッセージが生成された場合、これはデバイスの Unique Device Identifier（UDI）フォーマットです。 形式は、*type@Sid@serial*。 • *type* は、バックプレーン IDPROM からの製品の型番。 • @ は区切り文字です。 • *Sid* は C で、シリアル ID をシャーシ シリアル番号として特定します。 • *serial* は、Sid フィールドによって識別される番号です。 例：WS-C6509@C@12345678	/aml/header/serverID
メッセージの説明	エラーを説明するショート テキスト。	/aml/body/msgDesc
デバイス名	イベントが発生したノード（デバイスのホスト名）。	/aml/body/sysName
担当者名	イベントが発生したノード関連の問題について問い合わせる担当者名。	/aml/body/sysContact
連絡先電子メール	この装置の担当者の E メール アドレス。	/aml/body/sysContactEmail
連絡先電話番号	このユニットの連絡先である人物の電話番号	/aml/body/sysContactPhoneNumber
住所	この装置関連の返品許可（RMA）部品の送付先住所を保存するオプション フィールド。	/aml/body/sysStreetAddress
モデル名	デバイスのモデル名（製品ファミリ名に含まれる具体的なモデル）。	/aml/body/chassis/name

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
シリアル番号	ユニットのシャーシのシリアル番号	/aml/body/chassis/serialNo
シャーシの部品番号	シャーシの最上アセンブリ番号	/aml/body/chassis/partNo
特定のアラート グループ メッセージの固有のフィールドは、ここに挿入されます。		
このアラート グループに対して複数の CLI コマンドが実行されると、次のフィールドが繰り返される場合があります。		
Command output name	実行された CLI コマンドの正確な名前。	/aml/attachments/attachment/name
添付ファイルの種類	特定のコマンド出力。	/aml/attachments/attachment/type
MIME タイプ	プレーン テキストまたは符号化タイプ。	/aml/attachments/attachment/mime
コマンド出力テキスト	自動的に実行されるコマンドの出力	/aml/attachments/attachment/atdata

次の表に、フル テキストまたは XML のリアクティブ イベント メッセージ形式について説明します。

Table 12: 対処的または予防的イベント メッセージに挿入されるフィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
シャーシのハードウェア バージョン	シャーシのハードウェア バージョン。	/aml/body/chassis/hwVersion
スーパーバイザ モジュールのソフトウェア バージョン	最上レベルのソフトウェア バージョン	/aml/body/chassis/swVersion
影響のある FRU 名	イベント メッセージを生成する関連 FRU の名前。	/aml/body/fru/name
影響のある FRU のシリアル番号	関連 FRU のシリアル番号。	/aml/body/fru/serialNo
影響のある FRU の製品番号	関連 FRU の部品番号。	/aml/body/fru/partNo
FRU スロット	イベント メッセージを生成する FRU のスロット番号。	/aml/body/fru/slot

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
FRU ハードウェア バージョン	関連 FRU のハードウェア バージョン。	/aml/body/fru/hwVersion
FRU ソフトウェアのバージョン	関連 FRU で稼働しているソフトウェア バージョン。	/aml/body/fru/swVersion

次の表に、フル テキストまたは XML のコンポーネント イベント メッセージ形式について説明します。

Table 13: コンポーネント イベント メッセージの挿入フィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
シャーシのハードウェア バージョン	シャーシのハードウェア バージョン。	/aml/body/chassis/hwVersion
スーパーバイザ モジュールのソフトウェア バージョン	最上レベルのソフトウェア バージョン	/aml/body/chassis/swVersion
FRU 名	イベント メッセージを生成する関連 FRU の名前。	/aml/body/fru/name
FRU s/n	FRU のシリアル番号。	/aml/body/fru/serialNo
FRU 製品番号	FRU の部品番号。	/aml/body/fru/partNo
FRU スロット	FRU のスロット番号。	/aml/body/fru/slot
FRU ハードウェア バージョン	FRU のハードウェア バージョン。	/aml/body/fru/hwVersion
FRU ソフトウェアのバージョン	FRU で稼働しているソフトウェア バージョン。	/aml/body/fru/swVersion

次の表に、フルテキストまたは XML のユーザーが作成したテスト メッセージ形式について説明します。

Table 14: ユーザーが作成したテスト メッセージの挿入フィールド

データ項目（プレーンテキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
プロセス ID	固有のプロセス ID	/aml/body/process/id
プロセス状態	プロセスの状態（実行中、中止など）	/aml/body/process/processState

データ項目（プレーンテキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
プロセス例外	原因コードの例外	/aml/body/process/exception

Smart Call Home の注意事項および制約事項

- IP 接続がない場合、またはプロファイル宛先への仮想ルーティングおよびフォワーディング（VRF）インスタンス内のインターフェイスがダウンしている場合、スイッチは Smart Call Home メッセージを送信できません。
- 任意の SMTP 電子メール サーバーで動作します。

（注） SNMP sysContact は、デフォルトでは設定されていません。明示的に **snmp-server contact** <*sys-contact*> コマンドを使用して、SNMP sysContact を設定する必要があります。このコマンドを設定すると、callhome 機能が有効になります。

Smart Call Home の前提条件

- 電子メール サーバーに接続できる必要があります。
- コンタクト名（SNMP サーバーのコンタクト）、電話番号、および住所情報へアクセスできる必要があります。
- スイッチと電子メール サーバー間に IP 接続が必要です。
- 設定するデバイスに対して有効なサービス契約が必要です。

Call Home のデフォルト設定

Table 15: デフォルトの ***Call Home*** パラメータ

パラメータ	デフォルト
フル テキスト フォーマットで送信するメッセージの宛先メッセージ サイズ	4000000
XML フォーマットで送信するメッセージの宛先メッセージ サイズ	4000000

パラメータ	デフォルト
ショートテキストフォーマットで送信するメッセージの宛先メッセージ サイズ	4000
ポートを指定しなかった場合の SMTP サーバ ポート	25
プロファイルとアラート グループのアソシエート	フルテキスト宛先プロファイルおよびショートテキスト宛先プロファイルの場合はすべて。CiscoTAC-1 宛先プロファイルの場合は cisco-tac アラート グループ
フォーマット タイプ	XML
Call Home のメッセージ レベル	0（ゼロ）

Smart Call Home の設定

Smart Call Home の登録

始める前に

- ご使用のスイッチの sMARTnet 契約番号を確認してください
- 電子メール アドレスを確認してください
- Cisco.com ID を確認してください

手順

ステップ 1 ブラウザで、次の Smart Call Home Web ページに移動します。

http://www.cisco.com/go/smartcall/

ステップ 2 [**Getting Started**] で、Smart Call Home の登録指示に従ってください。

次のタスク

連絡先情報を設定します。

連絡先情報の設定

Smart Call Home には、電子メール、電話番号、住所の各情報を指定する必要があります。契約 ID、カスタマー ID、サイト ID、およびスイッチ プライオリティ情報を任意で指定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **snmp-server contact** *sys-contact*	SNMP sysContact を設定します。
ステップ 3	switch(config)# **callhome**	Smart Call Home コンフィギュレーション モードを開始します。
ステップ 4	switch(config-callhome)# **email-contact** *email-address*	スイッチの担当者の電子メールアドレスを設定します。 *email-address* には、電子メール アドレスの形式で、最大 255 の英数字を使用できます。 **Note** 任意の有効な E メール アドレスを使用できます。アドレスには、空白を含めることはできません。
ステップ 5	switch(config-callhome)# **phone-contact** *international-phone-number*	デバイスの担当者の電話番号を国際電話フォーマットで設定します。 *international-phone-number* は、最大 17 文字の英数字で、国際電話フォーマットにする必要があります。 **Note** 電話番号には、空白を含めることはできません。番号の前にプラス（+）プレフィックスを使用します。
ステップ 6	switch(config-callhome)# **streetaddress** *address*	スイッチの主担当者の住所を設定します。 *address* には、最大 255 の英数字を使用できます。スペースを使用できます。
ステップ 7	(Optional) switch(config-callhome)# **contract-id** *contract-number*	サービス契約からこのスイッチの契約番号を設定します。

	Command or Action	Purpose
		contract-number には最大 255 の英数字を使用できます。
ステップ 8	(Optional) switch(config-callhome)# **customer-id** *customer-number*	サービス契約からこのスイッチのカスタマー番号を設定します。 *customer-number* には最大 255 の英数字を使用できます。
ステップ 9	(Optional) switch(config-callhome)# **site-id** *site-number*	このスイッチのサイト番号を設定します。 *site-number* は、最大 255 文字の英数字を自由なフォーマットで指定できます。
ステップ 10	(Optional) switch(config-callhome)# **switch-priority** *number*	このスイッチのスイッチ プライオリティを設定します。 指定できる範囲は 0 ～ 7 です。0 は最高のプライオリティを、7 は最低のプライオリティを示します。デフォルト値は 7 です。
ステップ 11	(Optional) switch# **show callhome**	Smart Call Home コンフィギュレーションの概要を表示します。
ステップ 12	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、Call Home に関する担当者情報を設定する例を示します。

```
switch# configuration terminal
switch(config)# snmp-server contact personname@companyname.com
switch(config)# callhome
switch(config-callhome)# email-contact personname@companyname.com
switch(config-callhome)# phone-contact +1-800-123-4567
switch(config-callhome)# street-address 123 Anystreet St., Anycity, Anywhere
```

What to do next

宛先プロファイルを作成します。

宛先プロファイルの作成

ユーザー定義の宛先プロファイルを作成し、新しい宛先プロファイルにメッセージフォーマットを設定する必要があります。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome)# **destination-profile** {**ciscoTAC-1** { **alert-group** *group* | **email-addr** *address* | **http** *URL* | **transport-method** {**email** | **http**}} | *profilename* { **alert-group** *group* | **email-addr** *address* | **format** {**XML** | **full-txt** | **short-txt**} | **http** *URL* | **message-level** *level* | **message-size** *size* | **transport-method** {**email** | **http**}} | **full-txt-destination** { **alert-group** *group* | **email-addr** *address* | **http** *URL* | **message-level** *level* | **message-size** *size* | **transport-method** {**email** | **http**}} | **short-txt-destination** { **alert-group** *group* | **email-addr** *address* | **http** *URL* | **message-level** *level* | **message-size** *size* | **transport-method** {**email** | **http**}}}	新しい宛先プロファイルを作成し、そのプロファイルのメッセージ フォーマットを設定します。プロファイル名は、最大 31 文字の英数字で指定できます。 このコマンドについての詳細は、プラットフォームのコマンド リファレンスを参照してください。
ステップ 4	(Optional) switch# **show callhome destination-profile** [**profile** *name*]	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 5	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、Smart Call Home の宛先プロファイルを作成する例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile Noc101 format full-text
```

宛先プロファイルの変更

定義済みまたはユーザー定義の宛先プロファイルの次の属性を変更できます。

- 宛先アドレス：アラートの送信先となる実際のアドレス（トランスポート メカニズムに関係します）。
- メッセージ フォーマット：アラート送信に使用されるメッセージ フォーマット（フル テキスト、ショート テキスト、または XML）。
- メッセージ レベル：この宛先プロファイルの Call Home メッセージのシビラティ（重大度）。
- メッセージ サイズ：この宛先プロファイルの E メール アドレスに送信された Call Home メッセージの長さ。

Note CiscoTAC-1 宛先プロファイルは変更または削除できません。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome)# **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **email-addr** *address*	ユーザー定義または定義済みの宛先プロファイルに E メール アドレスを設定します。宛先プロファイルには、最大 50 個の E メール アドレスを設定できます。
ステップ 4	**destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **message-level** *number*	この宛先プロファイルの Smart Call Home メッセージのシビラティ（重大度）を設定します。Smart Call Home シビラティ（重大度）が一致する、またはそれ以上であるアラートのみが、このプロファイルの宛先に送信されます。*number* に指定できる範囲は 0 ～ 9 です。9 は最大のシビラティ（重大度）を示します。
ステップ 5	switch(config-callhome)# **destination-profile** {*name* | **full-txt-destination** | **short-txt-destination**} **message-size** *number*	この宛先プロファイルの最大メッセージ サイズを設定します。full-txt-destination の値の範囲は 0 ～ 5000000 で、デフォルトは 2500000 です。short-txt-destination の値の範囲は 0 ～ 100000 で、デフォル

	Command or Action	Purpose
		トは 4000 です。CiscoTAC-1 では、値は 5000000 で、これは変更不可能です。
ステップ 6	(Optional) switch# **show callhome destination-profile** [**profile** *name*]	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 7	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、Smart Call Home の宛先プロファイルを変更する例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile full-text-destination email-addr
person@example.com
switch(config-callhome)# destination-profile full-text-destination message-level 5
switch(config-callhome)# destination-profile full-text-destination message-size 10000
switch(config-callhome)#
```

What to do next

アラート グループと宛先プロファイルをアソシエートします。

アラート グループと宛先プロファイルのアソシエート

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome)# **destination-profile** *name* **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **Environmental** | **Inventory** | **License** | **Linecard-Hardware** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**}	アラート グループをこの宛先プロファイルにアソシエートします。キーワード **All** を使用して、すべてのアラート グループをこの宛先プロファイルにアソシエートします。
ステップ 4	(Optional) switch# **show callhome destination-profile** [**profile** *name*]	1 つまたは複数の宛先プロファイルに関する情報を表示します。

	Command or Action	Purpose
ステップ 5	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、すべてのアラート グループを宛先プロファイル Noc101 にアソシエートする例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# destination-profile Noc101 alert-group All
switch(config-callhome)#
```

What to do next

オプションで **show** コマンドをアラート グループに追加し、SMTP 電子メール サーバーを設定することができます。

アラート グループへの show コマンドの追加

1 つのアラート グループには、最大 5 個のユーザー定義 **show** コマンドを割り当てることができます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome)# **alert-group** {**Configuration** | **Diagnostic** | **Environmental** | **Inventory** | **License** | **Linecard-Hardware** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **user-def-cmd** *show-cmd*	**show** コマンド出力を、このアラート グループに送信された Call Home メッセージに追加します。有効な **show** コマンドだけが受け入れられます。 **Note** CiscoTAC-1 宛先プロファイルには、ユーザー定義の **show** コマンドを追加できません。

	Command or Action	Purpose
ステップ 4	(Optional) switch# **show callhome user-def-cmds**	アラート グループに追加されたすべてのユーザー定義 **show** コマンドに関する情報を表示します。
ステップ 5	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、**show ip routing** コマンドを Cisco-TAC アラート グループに追加する例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# alert-group Configuration user-def-cmd show ip routing
switch(config-callhome)#
```

What to do next

SMTP 電子メール サーバーに接続するように Smart Call Home を設定します。

電子メール サーバーの詳細の設定

Smart Call Home 機能が動作するよう SMTP サーバー アドレスを設定します。送信元および返信先 E メール アドレスも設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome)# **transport email smtp-server** *ip-address* [**port** *number*] [**use-vrf** *vrf-name*]	SMTP サーバーを、ドメイン ネーム サーバー（DNS）名、IPv4 アドレス、または IPv6 アドレスのいずれかとして設定します。 番号の範囲は 1 ～ 65535 です。デフォルトのポート番号は 25 です。

	Command or Action	Purpose
		この SMTP サーバーと通信する際に使用するよう任意で VRF インスタンスを設定できます。
ステップ 4	(Optional) switch(config-callhome)# **transport email from** *email-address*	Smart Call Home メッセージの送信元電子メール フィールドを設定します。
ステップ 5	(Optional) switch(config-callhome)# **transport email reply-to** *email-address*	Smart Call Home メッセージの返信先電子メール フィールドを設定します。
ステップ 6	(Optional) switch# **show callhome transport-email**	Smart Call Home の電子メール設定に関する情報を表示します。
ステップ 7	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、Smart Call Home メッセージの電子メール オプションを設定する例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# transport email smtp-server 192.0.2.10 use-vrf Red
switch(config-callhome)# transport email from person@example.com
switch(config-callhome)# transport email reply-to person@example.com
switch(config-callhome)#
```

What to do next

定期的なインベントリ通知を設定します。

定期的なインベントリ通知の設定

ハードウェアのインベントリ情報に加えて、デバイス上で現在イネーブルになっているすべてのソフトウェア サービスおよび実行中のすべてのソフトウェア サービスのインベントリに関するメッセージを定期的に送信するようにスイッチを設定できます。スイッチは 2 つの Smart Call Home 通知（定期的な設定メッセージと定期的なインベントリ メッセージ）を生成します。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

	Command or Action	Purpose
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome)# **periodic-inventory notification** [**interval** *days*] [**timeofday** *time*]	定期的なインベントリ メッセージを設定します。 **interval** *days* の範囲は 1 ～ 30 日です。 デフォルトは 7 日です。 **timeofday** *time* は HH:MM の形式です。
ステップ 4	(Optional) switch# **show callhome**	Smart Call Home に関する情報を表示します。
ステップ 5	(Optional) switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、定期的なインベントリ メッセージを 20 日ごとに生成するよう設定する例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# periodic-inventory notification interval 20
switch(config-callhome)#
```

What to do next

重複メッセージ抑制をディセーブルにします。

重複メッセージ抑制のディセーブル化

同じイベントについて受信する重複メッセージの数を制限できます。デフォルトでは、スイッチは同じイベントについて受信する重複メッセージの数を制限します。2 時間の時間枠内で送信された重複メッセージの数が 30 メッセージを超えると、スイッチは同じアラート タイプの以降のメッセージを廃棄します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome) # **no duplicate-message throttle**	Smart Call Home の重複メッセージ抑制をディセーブルにします。 重複メッセージ抑制はデフォルトでイネーブルです。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、重複メッセージ抑制をディセーブルにする例を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# no duplicate-message throttle
switch(config-callhome)#
```

次のタスク

Smart Call Home をイネーブルにします。

Smart Call Home のイネーブル化またはディセーブル化

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome) # [**no**] **enable**	Smart Call Home をイネーブルまたはディセーブルにします。 Smart Call Home は、デフォルトでディセーブルです。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ

	コマンドまたはアクション	目的
		コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次の例は、Smart Call Home をイネーブルにする方法を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# enable
switch(config-callhome)#
```

次のタスク

任意でテスト メッセージを生成します。

Smart Call Home 設定のテスト

始める前に

宛先プロファイルのメッセージ レベルが 2 以下に設定されていることを確認します。

重要 Smart Call Home のテストは、宛先プロファイルのメッセージ レベルが 3 以上に設定されている場合は失敗します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **callhome**	Smart Call Home 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	switch(config-callhome) # **callhome send diagnostic**	設定されたすべての宛先に指定の Smart Call Home テスト メッセージを送信します。
ステップ 4	switch(config-callhome) # **callhome test**	設定されたすべての宛先にテスト メッセージを送信します。
ステップ 5	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ

	コマンドまたはアクション	目的
		コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次の例は、Smart Call Home をイネーブルにする方法を示します。

```
switch# configuration terminal
switch(config)# callhome
switch(config-callhome)# callhome send diagnostic
switch(config-callhome)# callhome test
switch(config-callhome)#
```

Smart Call Home 設定の確認

次のいずれかのコマンドを使用して、設定を確認します。

コマンド	目的
show callhome	Smart Call Home のステータスを表示します。
show callhome destination-profile *name*	1 つまたは複数の Smart Call Home 宛先プロファイルを表示します。
show callhome pending-diff	保留中の Smart Call Home 設定と実行中の Smart Call Home 設定の違いを表示します。
show callhome status	Smart Call Home ステータスを表示します。
show callhome transport-email	Smart Call Home の電子メール設定を表示します。
show callhome user-def-cmds	任意のアラート グループに追加された CLI コマンドを表示します。
show running-config [**callhome** | **callhome-all**]	Smart Call Home の実行コンフィギュレーションを表示します。
show startup-config callhome	Smart Call Home のスタートアップ コンフィギュレーションを表示します。
show tech-support callhome	Smart Call Home のテクニカル サポート出力を表示します。

フル テキスト形式での syslog アラート通知の例

次の例では、Syslog ポート アラート グループ通知のフル テキスト形式を示します。

```
source:MDS9000
Switch Priority:7
Device Id:WS-C6509@C@FG@07120011
Customer Id:Example.com
Contract Id:123
Site Id:San Jose
Server Id:WS-C6509@C@FG@07120011
Time of Event:2004-10-08T11:10:44
Message Name:SYSLOG_ALERT
Message Type:Syslog
Severity Level:2
System Name:10.76.100.177
Contact Name:User Name
Contact Email:person@example.com
Contact Phone:+1-408-555-1212
Street Address:#1234 Any Street, Any City, Any State, 12345
Event Description:2006 Oct  8 11:10:44 10.76.100.177 %PORT-5-IF_TRUNK_UP:
%$VLAN 1%$ Interface e2/5, vlan 1 is up
syslog_facility:PORT
start chassis information:
Affected Chassis:WS-C6509
Affected Chassis Serial Number:FG@07120011
Affected Chassis Hardware Version:0.104
Affected Chassis Software Version:3.1(1)
Affected Chassis Part No:73-8607-01
end chassis information:
```

XML 形式での syslog アラート通知の例

次の例では、Syslog ポート アラート グループ通知の XML を示します。

```
From: example
Sent: Wednesday, April 25, 2007 7:20 AM
To: User (user)
Subject: System Notification From Router - syslog - 2007-04-25 14:19:55
GMT+00:00
<?xml version="1.0" encoding="UTF-8"?>
<soap-env:Envelope xmlns:soap-env="http://www.w3.org/2003/05/soap-envelope">
<soap-env:Header>
<aml-session:Session xmlns:aml-session="http://www.example.com/2004/01/aml-session"
soap-env:mustUnderstand="true" soap-env:role=
"http://www.w3.org/2003/05/soap-envelope/role/next">
<aml-session:To>http://tools.example.com/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.example.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.example.com/appliance/uri</aml-session:From>
<aml-session:MessageId>M2:69000101:C9D9E20B</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.example.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.example.com/2005/05/callhome/syslog</aml-block:Type>
```

```
<aml-block:CreationDate>2007-04-25 14:19:55 GMT+00:00</aml-block:CreationDate>
<aml-block:Builder>
<aml-block:Name>Cat6500</aml-block:Name>
<aml-block:Version>2.0</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>G3:69000101:C9F9E20C</aml-block:GroupId>
<aml-block:Number>0</aml-block:Number>
<aml-block:IsLast>true</aml-block:IsLast>
<aml-block:IsPrimary>true</aml-block:IsPrimary>
<aml-block:WaitForPrimary>false</aml-block:WaitForPrimary>
</aml-block:BlockGroup>
<aml-block:Severity>2</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:Call Home xmlns:ch="http://www.example.com/2005/05/callhome" version="1.0">
<ch:EventTime>2007-04-25 14:19:55 GMT+00:00</ch:EventTime>
<ch:MessageDescription>03:29:29: %CLEAR-5-COUNTERS: Clear counter on all
interfaces by console</ch:MessageDescription>
<ch:Event>
<ch:Type>syslog</ch:Type>
<ch:SubType>
</ch:SubType>
<ch:Brand>Cisco Systems</ch:Brand>
<ch:Series>Catalyst 6500 Series Switches</ch:Series>
</ch:Event>
<ch:CustomerData>
<ch:UserData>
<ch:Email>person@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:CustomerId>12345</ch:CustomerId>
<ch:SiteId>building 1</ch:SiteId>
<ch:ContractId>abcdefg12345</ch:ContractId>
<ch:DeviceId>WS-C6509@C@69000101</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>Router</ch:Name>
<ch:Contact>
</ch:Contact>
<ch:ContactEmail>user@example.com</ch:ContactEmail>
<ch:ContactPhoneNumber>+1-408-555-1212</ch:ContactPhoneNumber>
<ch:StreetAddress>#1234 Any Street, Any City, Any State, 12345
</ch:StreetAddress>
</ch:SystemInfo>
</ch:CustomerData>
<ch:Device>
<rme:Chassis xmlns:rme="http://www.example.com/rme/4.0">
<rme:Model>WS-C6509</rme:Model>
<rme:HardwareVersion>1.0</rme:HardwareVersion>
<rme:SerialNumber>69000101</rme:SerialNumber>
<rme:AdditionalInformation>
<rme:AD name="PartNumber" value="73-3438-03 01" />
<rme:AD name="SoftwareVersion" value="4.0(20080421:012711)" />
</rme:AdditionalInformation>
</rme:Chassis>
</ch:Device>
</ch:Call Home>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging</aml-block:Name>
<aml-block:Data encoding="plain">
<![CDATA[Syslog logging: enabled (0 messages dropped, 0 messages
```

```
rate-limited, 0 flushes, 0 overruns, xml disabled, filtering disabled)
    Console logging: level debugging, 53 messages logged, xml disabled,
filtering disabled    Monitor logging: level debugging, 0 messages logged,
xml disabled,filtering disabled    Buffer logging: level debugging,
53 messages logged, xml disabled,      filtering disabled    Exception
Logging: size (4096 bytes)    Count and timestamp logging messages: disabled
    Trap logging: level informational, 72 message lines logged
Log Buffer (8192 bytes):
00:00:54: curr is 0x20000
00:00:54: RP: Currently running ROMMON from F2 region
00:01:05: %SYS-5-CONFIG_I: Configured from memory by console
00:01:09: %SYS-5-RESTART: System restarted --Cisco IOS Software,
s72033_rp Software (s72033_rp-ADVENTERPRISEK9_DBG-VM), Experimental
Version 12.2(20070421:012711) Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 15:54 by xxx
Firmware compiled 11-Apr-07 03:34 by integ Build [100]00:01:01: %PFREDUN-6-ACTIVE:
 Initializing as ACTIVE processor for this switch00:01:01: %SYS-3-LOGGER_FLUSHED:
System was paused for 00:00:00 to ensure console debugging output.00:03:00: SP: SP:
 Currently running ROMMON from F1 region00:03:07: %C6K_PLATFORM-SP-4-CONFREG_BREAK
_ENABLED: The default factory setting for config register is 0x2102.It is advisable
 to retain 1 in 0x2102 as it prevents returning to ROMMON when break is issued.00:03:18:

 %SYS-SP-5-RESTART: System restarted --Cisco IOS Software, s72033_sp Software
 (s72033_sp-ADVENTERPRISEK9_DBG-VM), Experimental Version 12.2(20070421:012711)Copyright

 (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-07 18:00 by xxx
00:03:18: %SYS-SP-6-BOOTTIME: Time taken to reboot after reload =  339 seconds
00:03:18: %OIR-SP-6-INSPS: Power supply inserted in slot 1
00:03:18: %C6KPWR-SP-4-PSOK: power supply 1 turned on.
00:03:18: %OIR-SP-6-INSPS: Power supply inserted in slot00:01:09: %SSH-5-ENABLED:
 SSH 1.99 has been enabled
00:03:18: %C6KPWR-SP-4-PSOK: power supply 2 turned on.
00:03:18: %C6KPWR-SP-4-PSREDUNDANTMISMATCH: power supplies rated outputs do not match.
00:03:18: %C6KPWR-SP-4-PSREDUNDANTBOTHSUPPLY: in power-redundancy mode, system is
 operating on both power supplies.
00:01:10: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
00:01:10: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
00:03:20: %C6KENV-SP-4-FANHIOUTPUT: Version 2 high-output fan-tray is in effect
00:03:22: %C6KPWR-SP-4-PSNOREDUNDANCY: Power supplies are not in full redundancy,
 power usage exceeds lower capacity supply
00:03:26: %FABRIC-SP-5-FABRIC_MODULE_ACTIVE: The Switch Fabric Module in slot 6
 became active.
00:03:28: %DIAG-SP-6-RUN_MINIMUM: Module 6: Running Minimal Diagnostics...
00:03:50: %DIAG-SP-6-DIAG_OK: Module 6: Passed Online Diagnostics
00:03:50: %OIR-SP-6-INSCARD: Card inserted in slot 6, interfaces are now online
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 3: Running Minimal Diagnostics...
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 7: Running Minimal Diagnostics...
00:03:51: %DIAG-SP-6-RUN_MINIMUM: Module 9: Running Minimal Diagnostics...
00:01:51: %MFIB_CONST_RP-6-REPLICATION_MODE_CHANGE: Replication Mode Change Detected.
 Current system replication mode is Ingress
00:04:01: %DIAG-SP-6-DIAG_OK: Module 3: Passed Online Diagnostics
00:04:01: %OIR-SP-6-DOWNGRADE: Fabric capable module 3 not at an appropriate hardware
 revision level, and can only run in flowthrough mode
00:04:02: %OIR-SP-6-INSCARD: Card inserted in slot 3, interfaces are now online
00:04:11: %DIAG-SP-6-DIAG_OK: Module 7: Passed Online Diagnostics
00:04:14: %OIR-SP-6-INSCARD: Card inserted in slot 7, interfaces are now online
00:04:35: %DIAG-SP-6-DIAG_OK: Module 9: Passed Online Diagnostics
00:04:37: %OIR-SP-6-INSCARD: Card inserted in slot 9, interfaces are now online
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-07 03:34 by integ Build [100]
00:00:22: %SYS-DFC4-5-RESTART: System restarted --
Cisco DCOS Software, c6lc2 Software (c6lc2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
```

```
Compiled Thu 26-Apr-08 17:20 by xxx
00:00:23: DFC4: Currently running ROMMON from F2 region
00:00:25: %SYS-DFC2-5-RESTART: System restarted --
Cisco IOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 12.2
(20070421:012711)Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:26: DFC2: Currently running ROMMON from F2 region
00:04:56: %DIAG-SP-6-RUN_MINIMUM: Module 4: Running Minimal Diagnostics...
00:00:09: DaughterBoard (Distributed Forwarding Card 3)
Firmware compiled 11-Apr-08 03:34 by integ Build [100]
slot_id is 8
00:00:31: %FLASHFS_HES-DFC8-3-BADCARD: /bootflash:: The flash card seems to
 be corrupted
00:00:31: %SYS-DFC8-5-RESTART: System restarted --
Cisco DCOS Software, c6lc2 Software (c6lc2-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 17:20 by username1
00:00:31: DFC8: Currently running ROMMON from S (Gold) region
00:04:59: %DIAG-SP-6-RUN_MINIMUM: Module 2: Running Minimal Diagnostics...
00:05:12: %DIAG-SP-6-RUN_MINIMUM: Module 8: Running Minimal Diagnostics...
00:05:13: %DIAG-SP-6-RUN_MINIMUM: Module 1: Running Minimal Diagnostics...
00:00:24: %SYS-DFC1-5-RESTART: System restarted --
Cisco DCOS Software, c6slc Software (c6slc-SPDBG-VM), Experimental Version 4.0
(20080421:012711)Copyright (c) 1986-2008 by Cisco Systems, Inc.
Compiled Thu 26-Apr-08 16:40 by username1
00:00:25: DFC1: Currently running ROMMON from F2 region
00:05:30: %DIAG-SP-6-DIAG_OK: Module 4: Passed Online Diagnostics
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
 session 0 is Centralized
00:05:31: %SPAN-SP-6-SPAN_EGRESS_REPLICATION_MODE_CHANGE: Span Egress HW
Replication Mode Change Detected. Current replication mode for unused asic
 session 1 is Centralized
00:05:31: %OIR-SP-6-INSCARD: Card inserted in slot 4, interfaces are now online
00:06:02: %DIAG-SP-6-DIAG_OK: Module 1: Passed Online Diagnostics
00:06:03: %OIR-SP-6-INSCARD: Card inserted in slot 1, interfaces are now online
00:06:31: %DIAG-SP-6-DIAG_OK: Module 2: Passed Online Diagnostics
00:06:33: %OIR-SP-6-INSCARD: Card inserted in slot 2, interfaces are now online
00:04:30: %XDR-6-XDRIPCNOTIFY: Message not sent to slot 4/0 (4) because of IPC
 error timeout. Disabling linecard. (Expected during linecard OIR)
00:06:59: %DIAG-SP-6-DIAG_OK: Module 8: Passed Online Diagnostics
00:06:59: %OIR-SP-6-DOWNGRADE_EARL: Module 8 DFC installed is not identical to
 system PFC and will perform at current system operating mode.
00:07:06: %OIR-SP-6-INSCARD: Card inserted in slot 8, interfaces are now online
Router#]]>
</aml-block:Data>
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```

第 10 章

Session Manager の設定

この章は、次の項で構成されています。

Session Manager の概要

Session Manager を使用すると、設定変更をバッチ モードで実行できます。Session Manager は次のフェーズで機能します。

- コンフィギュレーション セッション：Session Manager モードで実行するコマンドのリストを作成します。
- 検証：設定の基本的なセマンティック チェックを行います。Cisco NX-OS は、設定の一部でセマンティクス検査が失敗した場合にエラーを返します。
- 検証：既存のハードウェア設定、ソフトウェア設定、およびリソースに基づいて、設定全体を確認します。Cisco NX-OS は、設定がこの確認フェーズで合格しなかった場合にエラーを返します。
- コミット：Cisco NX-OS は設定全体を確認して、デバイスに対する変更をアトミックに実行します。エラーが発生すると、Cisco NX-OS は元の設定に戻ります。
- 打ち切り：設定変更を実行しないで廃棄します。

任意で、変更をコミットしないでコンフィギュレーションセッションを終了できます。また、コンフィギュレーション セッションを保存することもできます。

Session Manager の注意事項および制約事項

Session Manager には、次の注意事項および制限事項があります。

- Session Manager は、アクセス コントロール リスト（ACL）機能のみサポートします。
- 作成できるコンフィギュレーション セッションの最大数は 32 です。
- すべてのセッションで設定できるコマンドの最大数は 20,000 です。

Session Manager の設定

セッションの作成

作成できるコンフィギュレーション セッションの最大数は 32 です。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure session** *name*	コンフィギュレーション セッションを作成し、セッション コンフィギュレーション モードを開始します。名前は任意の英数字ストリングです。 セッションの内容を表示します。
ステップ 2	(Optional) switch(config-s)# **show configuration session** [*name*]	セッションの内容を表示します。
ステップ 3	(Optional) switch(config-s)# **save** *location*	セッションをファイルに保存します。保存場所には、bootflash または volatile を指定できます。

セッションでの ACL の設定

コンフィギュレーション セッションで ACL を設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure session** *name*	コンフィギュレーション セッションを作成し、セッション コンフィギュレーション モードを開始します。名前は任意の英数字ストリングです。
ステップ 2	switch(config-s)# **ip access-list** *name*	ACL を作成します。

	Command or Action	Purpose
ステップ 3	(Optional) switch(config-s-acl)# **permit** *protocol source destination*	ACL に許可文を追加します。
ステップ 4	switch(config-s-acl)# **interface** *interface-type number*	インターフェイス コンフィギュレーション モードを開始します。
ステップ 5	switch(config-s-if)# **ip port access-group** *name* **in**	インターフェイスにポート アクセス グループを追加します。
ステップ 6	(Optional) switch# **show configuration session** [*name*]	セッションの内容を表示します。

セッションの確認

セッションを確認するには、セッション モードで次のコマンドを使用します。

コマンド	目的
switch(config-s)# **verify** [**verbose**]	コンフィギュレーション セッションのコマンドを確認します。

セッションのコミット

セッションをコミットするには、セッション モードで次のコマンドを使用します。

コマンド	目的
switch(config-s)# **commit** [**verbose**]	コンフィギュレーションセッションのコマンドをコミットします。

セッションの保存

セッションを保存するには、セッション モードで次のコマンドを使用します。

コマンド	目的
switch(config-s)# **save** *location*	（任意）セッションをファイルに保存します。保存場所には、bootflash または volatile を指定できます。

セッションの廃棄

セッションを廃棄するには、セッション モードで次のコマンドを使用します。

コマンド	目的
switch(config-s)# **abort**	コマンドを適用しないで、コンフィギュレーション セッションを廃棄します。

Session Manager のコンフィギュレーション例

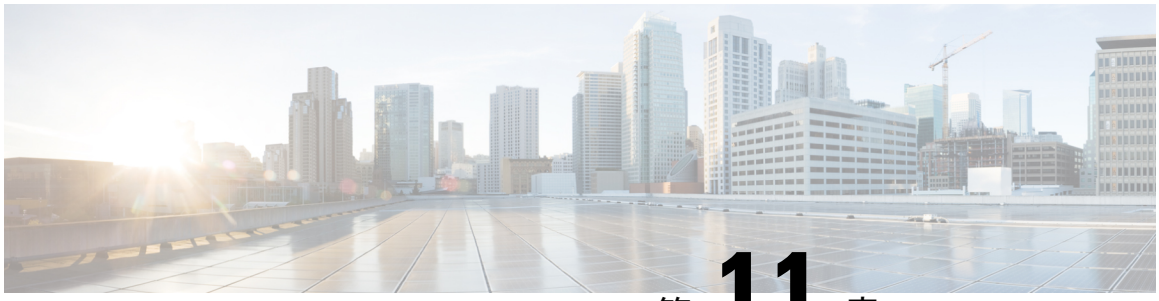
次に、ACL 用のコンフィギュレーション セッションを作成する例を示します。

```
switch# configure session name test2
switch(config-s)# ip access-list acl2
switch(config-s-acl)# permit tcp any any
switch(config-s-acl)# exit
switch(config-s)# interface Ethernet 1/4
switch(config-s-ip)# ip port access-group acl2 in
switch(config-s-ip)# exit
switch(config-s)# verify
switch(config-s)# exit
switch# show configuration session test2
```

Session Manager 設定の確認

Session Manager の設定情報を確認するには、次の作業のいずれかを行います。

コマンド	目的
show configuration session [*name*]	コンフィギュレーション ファイルの内容を表示します。
show configuration session status [*name*]	コンフィギュレーション セッションのステータスを表示します。
show configuration session summary	すべてのコンフィギュレーション セッションのサマリーを表示します。



第 11 章

スケジューラの設定

この章は、次の項で構成されています。

スケジューラの概要

スケジューラを使用すると、次のようなメンテナンス作業のタイムテーブルを定義し、設定することができます。

- QoS（Quality of Service）ポリシーの変更
- データのバックアップ
- 設定の保存

ジョブは、定期的な作業を定義する単一または複数のコマンドで構成されています。ジョブは、1 回だけ、または定期的な間隔でスケジューリングすることができます。

スケジューラでは、ジョブと、そのタイムテーブルを次のように定義できます。

ジョブ

コマンドリストとして定義され、指定されたスケジュールに従って実行される定期的なタスク。

スケジュール

ジョブを実行するためのタイムテーブル。1 つのスケジュールに複数のジョブを割り当てることができます。

1 つのスケジュールは、定期的、または 1 回だけ実行するように定義されます。

- 定期モード：ジョブを削除するまで続行される繰り返しの間隔。次のタイプの定期的な間隔を設定できます。
  - Daily：ジョブは 1 日 1 回実行されます。
  - Weekly：ジョブは毎週 1 回実行されます。
  - Monthly：ジョブは毎月 1 回実行されます。
  - Delta：ジョブは、指定した時間に開始され、以後、指定した間隔（days:hours:minutes）で実行されます。
- 1 回限定モード：ジョブは、指定した時間に 1 回だけ実行されます。

リモート ユーザ認証

ジョブの開始前に、スケジューラはジョブを作成したユーザーを認証します。リモート認証からのユーザー クレデンシャルは、スケジュールされたジョブをサポートできるだけの十分に長い時間保持されないため、ジョブを作成するユーザーの認証パスワードをローカルで設定する必要があります。これらのパスワードは、スケジューラのコンフィギュレーションに含まれ、ローカル設定のユーザとは見なされません。

ジョブを開始する前に、スケジューラはローカル パスワードとリモート認証サーバに保存されたパスワードを照合します。

スケジューラ ログ ファイル

スケジューラは、ジョブ出力を含むログ ファイルを管理します。ジョブ出力のサイズがログ ファイルのサイズより大きい場合、出力内容は切り捨てられます。

スケジューラの注意事項および制約事項

- ジョブの実行中に次のいずれかの状況が発生した場合、スケジューラは失敗する可能性があります。
  - 機能ライセンスが、その機能のジョブがスケジュールされている時間に期限切れになった場合。
  - 機能が、その機能を使用するジョブがスケジューリングされている時間にディセーブルになっている場合。
- 時刻が設定されていることを確認します。スケジューラはデフォルトのタイムテーブルを適用しません。スケジュールを作成し、ジョブを割り当てても、時刻を設定しなければ、ジョブは開始されません。
- ジョブは開始されると非インタラクティブ方式で実行されるため、ジョブの定義中、インタラクティブなコマンドや中断を伴うコマンド（例：**copy bootflash:** *file* **ftp:***URI*、**write**

erase、**reload**、およびその他類似のコマンド）が指定されていないことを確認してください。特定の時間にリロードジョブがスケジュールされ、実行されると、スイッチはブートループに入ります。したがって、スケジューラ構成では使用しないでください。

スケジューラのデフォルト設定

表 *16:* コマンド スケジューラのパラメータのデフォルト

パラメータ	デフォルト
スケジューラの状態	ディセーブル
ログ ファイル サイズ	16 KB

スケジューラの設定

スケジューラのイネーブル化

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **feature scheduler**	スケジューラをイネーブルにします。
ステップ 3	（任意） switch(config) # **show scheduler config**	スケジューラ設定を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、スケジューラをイネーブルにする例を示します。

```
switch# configure terminal
switch(config)# feature scheduler
switch(config)# show scheduler config
config terminal
   feature scheduler
```

```
   scheduler logfile size 16
end
switch(config)#
```

スケジューラ ログ ファイル サイズの定義

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **scheduler logfile size** *value*	スケジューラ ログ ファイル サイズをキロバイト（KB）で定義します。 範囲は 16 ～ 1024 です。デフォルトのログ ファイル サイズは 16 です。 （注） ジョブ出力のサイズがログ ファイルのサイズより大きい場合、出力内容は切り捨てられます。
ステップ 3	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、スケジューラ ログ ファイルのサイズを定義する例を示します。

```
switch# configure terminal
switch(config)# scheduler logfile size 1024
switch(config)#
```

リモート ユーザ認証の設定

リモート ユーザーは、ジョブを作成および設定する前に、クリア テキスト パスワードを使用して認証する必要があります。

show running-config コマンドの出力では、リモート ユーザー パスワードは常に暗号化された状態で表示されます。コマンドの暗号化オプション（**7**）は、ASCII デバイス設定をサポートします。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **scheduler aaa-authentication password** [**0** | **7**] *password*	現在ログインしているユーザーのパスワードを設定します。 クリア テキスト パスワードを設定するには、**0** を入力します。 暗号化されたパスワードを設定するには、**7** を入力します。
ステップ 3	switch(config) # **scheduler aaa-authentication username** *name* **password** [**0** | **7**] *password*	リモート ユーザーのクリア テキスト パスワードを設定します。
ステップ 4	（任意） switch(config) # **show running-config | include "scheduler aaa-authentication"**	スケジューラのパスワード情報を表示します。
ステップ 5	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、NewUser という名前のリモート ユーザーのクリア テキスト パスワードを設定する例を示します。

```
switch# configure terminal
switch(config) # scheduler aaa-authentication
username NewUser password z98y76x54b
switch(config) # copy running-config startup-config
switch(config) #
```

ジョブの定義

一旦ジョブを定義すると、コマンドの変更、削除はできません。ジョブを変更するには、そのジョブを削除して新しいジョブを作成する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **scheduler job name** *name*	ジョブを指定された名前で作成し、ジョブ構成モードを開始します。 *name* は 31 文字までに制限されています。
ステップ 3	switch(config-job) # *command1* **;** [*command2* **;***command3* **;** ...	特定のジョブに対応するコマンド シーケンスを定義します。複数のコマンドは、スペースとセミコロンで（;）で区切る必要があります。 ファイル名は現在のタイムスタンプとスイッチ名を使用して作成します。
ステップ 4	（任意） switch(config-job) # **show scheduler job** [*name*]	ジョブ情報を表示します。 *name* は 31 文字までに制限されています。
ステップ 5	（任意） switch(config-job) # **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次の例は、次の方法を示します。

- 「backup-cfg」という名前のスケジューラ ジョブを作成示します。
- 実行中の構成をブートフラッシュ上のファイルに保存します。
- ファイルをブートフラッシュから TFTP サーバーにコピーします。
- 変更がスタートアップ構成に保存されます。

```
switch# configure terminal
switch(config) # scheduler job name backup-cfg
switch(config-job) # copy running-config
tftp://1.2.3.4/$(SWITCHNAME)-cfg.$(TIMESTAMP) vrf management
switch(config-job) # copy running-config startup-config
```

ジョブの削除

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **no scheduler job name** *name*	特定のジョブおよびそこで定義されたすべてのコマンドを削除します。 *name* は 31 文字までに制限されています。
ステップ 3	（任意） switch(config-job) # **show scheduler job** [*name*]	ジョブ情報を表示します。
ステップ 4	（任意） switch(config-job) # **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、configsave という名前のジョブを削除する例を示します。

```
switch# configure terminal
switch(config)# no scheduler job name configsave
switch(config-job)# copy running-config startup-config
switch(config-job)#
```

タイムテーブルの定義

タイムテーブルを設定する必要があります。設定しないと、ジョブがスケジューリングされません。

time コマンドで時刻を設定しない場合は、スケジューラは現在の時刻を使用します。たとえば、現在の時刻が 2008 年 3 月 24 日の 22 時 00 分である場合、ジョブは次のように開始されます。

- スケジューラは、**time start 23:00 repeat 4:00:00** コマンドの開始時刻が、2008 年 3 月 24 日 23 時 00 分であると見なします。
- スケジューラは、**time daily 55** コマンドの開始時刻が、毎日 22 時 55 分であると見なします。
- スケジューラは、**time weekly 23:00** コマンドの開始時刻が、毎週金曜日の 23 時 00 分であると見なします。

- スケジューラは、**time monthly 23:00** コマンドの開始時刻が、毎月 24 日の 23 時 00 分であると見なします。

（注） スケジューラは、1 つ前のジョブが完了しない限り、次のジョブを開始しません。たとえば、1 分間隔で実行するジョブを 22 時 00 分に開始するようジョブをスケジューリングしたが、ジョブを完了するには 2 分間必要である場合、ジョブは次のように実行されます。スケジューラは 22 時 00 分に最初のジョブを開始し、22 時 02 分に完了します。次に 1 分間待機し、22 時 03 分に次のジョブを開始します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **scheduler schedule name** *name*	新しいスケジューラを作成し、そのスケジュールのスケジュール コンフィギュレーション モードを開始します。 *name* は 31 文字までに制限されています。
ステップ 3	switch(config-schedule) # **job name** *name*	このスケジュールにジョブを関連付けます。1 つのスケジュールに複数のジョブを追加できます。 *name* は 31 文字までに制限されています。
ステップ 4	switch(config-schedule) # **time daily** *time*	ジョブが毎日 HH:MM の形式で指定された時刻に開始することを意味します。
ステップ 5	switch(config-schedule) # **time weekly** [[*day-of-week*:] *HH*:] *MM*	ジョブが週の指定された曜日に開始することを意味します。 曜日は整数（たとえば、日曜日は **1**、月曜日は **2**）または略語（たとえば、**sun**、**mon**）で表します。 引数全体の最大長は 10 文字です。
ステップ 6	switch(config-schedule) # **time monthly** [[*day-of-month*:] *HH*:] *MM*	ジョブが月の特定の日に開始することを意味します。

	コマンドまたはアクション	目的
		29、30 または 31 のいずれかを指定した場合、そのジョブは各月の最終日に開始されます。
ステップ 7	switch(config-schedule) # **time start** { **now** **repeat** *repeat-interval* | *delta-time* [**repeat** *repeat-interval*]}	ジョブが定期的に開始することを意味します。 start-time の形式は [[[[yyyy:]mmm:]dd:]HH]:MM です。 • *delta-time*：スケジュールの設定後、ジョブの開始までの待機時間を指定します。 • **now**：ジョブが今から 2 分後に開始することを指定します。 • **repeat** *repeat-interval*：ジョブを反復する回数を指定します。
ステップ 8	（任意） switch(config-schedule) # **show scheduler config**	スケジューラの情報を表示します。
ステップ 9	（任意） switch(config-schedule) # **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、ジョブが毎月 28 日の 23 時 00 分に開始するタイムテーブルを定義する例を示します。

```
switch# configure terminal
switch(config)# scheduler schedule name weekendbackupqos
switch(config-scheduler)# job name offpeakzoning
switch(config-scheduler)# time monthly 28:23:00
switch(config-scheduler)# copy running-config startup-config
switch(config-scheduler)#
```

スケジューラ ログ ファイルの消去

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **clear scheduler logfile**	スケジューラ ログ ファイルを消去します。

例

次に、スケジューラ ログ ファイルを消去する例を示します。

```
switch# configure terminal
switch(config)# clear scheduler logfile
```

スケジューラのディセーブル化

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config) # **no feature scheduler**	スケジューラをディセーブルにします。
ステップ 3	（任意） switch(config) # **show scheduler config**	スケジューラ設定を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、スケジューラをディセーブルにする例を示します。

```
switch# configure terminal
switch(config) # no feature scheduler
switch(config) # copy running-config startup-config
switch(config) #
```

スケジューラの設定確認

次のいずれかのコマンドを使用して、設定を確認します。

表 *17 :* スケジューラの *show* コマンド

コマンド	目的
show scheduler config	スケジューラ設定を表示します。
show scheduler job [**name** *name*]	設定されているジョブを表示します。
show scheduler logfile	スケジューラ ログ ファイルの内容を表示します。
show scheduler schedule [**name** *name*]	設定されているスケジュールを表示します。

スケジューラの設定例

スケジューラ ジョブの作成

この例では、実行コンフィギュレーションをブートフラッシュ内のファイルに保存するスケジュールジョブを作成する方法を示します。このジョブは、その後で、ブートフラッシュからTFTP サーバにファイルをコピーします（現在のタイムスタンプとスイッチ名を使用してファイル名を作成します）。

```
switch# configure terminal
switch(config)# scheduler job name backup-cfg
switch(config-job)# copy running-config
tftp://1.2.3.4/$(SWITCHNAME)-cfg.$(TIMESTAMP) vrf management
switch(config-job)# end
switch(config)#
```

スケジューラ ジョブのスケジューリング

次に、backup-cfgという名前のスケジューラ ジョブを、毎日午前1時に実行するようスケジューリングする例を示します。

```
switch# configure terminal
switch(config)# scheduler schedule name daily
switch(config-schedule)# job name backup-cfg
switch(config-schedule)# time daily 1:00
switch(config-schedule)# end
switch(config)#
```

ジョブ スケジュールの表示

次に、ジョブ スケジュールを表示する例を示します。

```
switch# show scheduler schedule
Schedule Name       : daily
---------------------------
User Name           : admin
Schedule Type       : Run every day at 1 Hrs 00 Mins
Last Execution Time : Fri Jan 2 1:00:00 2009
Last Completion Time: Fri Jan 2 1:00:01 2009
Execution count     : 2
-----------------------------------------------
     Job Name            Last Execution Status
-----------------------------------------------
back-cfg                            Success (0)
switch(config)#
```

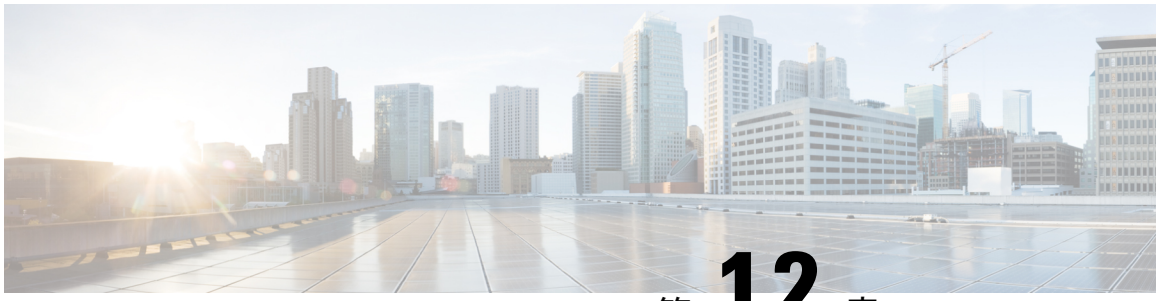
スケジューラ ジョブの実行結果の表示

次に、スケジューラによって実行されたスケジューラ ジョブの結果を表示する例を示します。

```
switch# show scheduler logfile
Job Name       : back-cfg                           Job Status: Failed (1)
Schedule Name  : daily                              User Name : admin
Completion time: Fri Jan 1  1:00:01 2009
-------------------------------- Job Output ---------------------------------
`cli var name timestamp 2009-01-01-01.00.00`
`copy running-config bootflash:/$(HOSTNAME)-cfg.$(timestamp)`
`copy bootflash:/switch-cfg.2009-01-01-01.00.00 tftp://1.2.3.4/ vrf management `
copy: cannot access file '/bootflash/switch-cfg.2009-01-01-01.00.00'
==============================================================================
Job Name       : back-cfg                           Job Status: Success (0)
Schedule Name  : daily                              User Name : admin
Completion time: Fri Jan 2  1:00:01 2009
-------------------------------- Job Output ---------------------------------
`cli var name timestamp 2009-01-02-01.00.00`
`copy running-config bootflash:/switch-cfg.2009-01-02-01.00.00`
`copy bootflash:/switch-cfg.2009--01-02-01.00.00 tftp://1.2.3.4/ vrf management `
Connection to Server Established.
[                       ]         0.50KBTrying to connect to tftp server......
[######                 ]        24.50KB
TFTP put operation was successful
==============================================================================
switch#
```

スケジューラの標準

この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。



第 12 章

SNMP の設定

この章は、次の項で構成されています。

SNMP に関する情報

簡易ネットワーク管理プロトコル（SNMP）は、SNMP マネージャとエージェント間の通信用メッセージフォーマットを提供する、アプリケーションレイヤプロトコルです。SNMP では、ネットワーク内のデバイスのモニタリングと管理に使用する標準フレームワークと共通言語が提供されます。

SNMP 機能の概要

SNMP フレームワークは 3 つの部分で構成されます。

- SNMPマネージャ：SNMPを使用してネットワークデバイスのアクティビティを制御し、モニタリングするシステム
- SNMPエージェント：デバイスのデータを維持し、必要に応じてこれらのデータを管理システムに報告する、管理対象デバイス内のソフトウェア コンポーネント。Cisco Nexus デバイスはエージェントおよび MIB をサポートします。SNMP エージェントをイネーブルにするには、マネージャとエージェントの関係を定義する必要があります。
- MIB（Management Information Base; 管理情報ベース）：SNMP エージェントの管理対象オブジェクトの集まり

Note Cisco NX-OS は、イーサネット MIB の SNMP セットをサポートしません。

Cisco Nexus デバイスは、SNMPv1、SNMPv2c、および SNMPv3 をサポートします。SNMPv1 および SNMPv2c はどちらも、コミュニティベース形式のセキュリティを使用します。

SNMP は、RFC 3410（http://tools.ietf.org/html/rfc3410）、RFC 3411（http://tools.ietf.org/html/rfc3411）、RFC 3412（http://tools.ietf.org/html/rfc3412）、RFC 3413（http://tools.ietf.org/html/rfc3413）、RFC 3414（http://tools.ietf.org/html/rfc3414）、RFC 3415（http://tools.ietf.org/html/rfc3415）、RFC 3416（http://tools.ietf.org/html/rfc3416）、RFC 3417（http://tools.ietf.org/html/rfc3417）、RFC 3418（http://tools.ietf.org/html/rfc3418）、および RFC 3584（http://tools.ietf.org/html/rfc3584）で定義されています。

SNMP 通知

SNMP の重要な機能の 1 つは、SNMP エージェントから通知を生成できることです。これらの通知では、要求を SNMP マネージャから送信する必要はありません。通知は、不正なユーザ認証、再起動、接続の切断、隣接ルータとの接続の切断、その他の重要なイベントを表示します。

Cisco NX-OS は、トラップまたはインフォームとして SNMP 通知を生成します。トラップは、エージェントからホスト レシーバ テーブルで指定された SNMP マネージャに送信される、非同期の非確認応答メッセージです。インフォームは、SNMP エージェントから SNMP マネージャに送信される非同期メッセージで、マネージャは受信したという確認応答が必要です。

トラップの信頼性はインフォームより低くなります。SNMP マネージャはトラップを受信しても確認応答（ACK）を送信しないからです。このため、トラップが受信されたかどうかをスイッチが判断できません。インフォーム要求を受信する SNMP マネージャは、SNMP 応答プロトコル データ ユニット（PDU）でメッセージの受信を確認応答します。Cisco Nexus デバイスが応答を受信しない場合、インフォーム要求を再び送信できます。

複数のホスト レシーバーに通知を送信するよう Cisco NX-OS を設定できます。

SNMPv3

SNMPv3 は、ネットワーク経由のフレームの認証と暗号化を組み合わせることによって、デバイスへのセキュア アクセスを実現します。SNMPv3 が提供するセキュリティ機能は次のとおりです。

- メッセージの完全性：パケットが伝送中に改ざんされていないことを保証します。
- 認証：メッセージのソースが有効かどうかを判別します。
- 暗号化：許可されていないソースにより判読されないように、パケットの内容のスクランブルを行います。

SNMPv3 では、セキュリティ モデルとセキュリティ レベルの両方が提供されています。セキュリティ モデルは、ユーザおよびユーザが属するロールを設定する認証方式です。セキュリティ レベルとは、セキュリティ モデル内で許可されるセキュリティのレベルです。セキュリティ モデルとセキュリティ レベルの組み合わせにより、SNMP パケット処理中に採用されるセキュリティ メカニズムが決まります。

SNMPv1、SNMPv2、SNMPv3 のセキュリティ モデルおよびセキュリティ レベル

セキュリティ レベルは、SNMP メッセージを開示から保護する必要があるかどうか、およびメッセージを認証するかどうか判断します。セキュリティ モデル内のさまざまなセキュリティ レベルは、次のとおりです。

- noAuthNoPriv：認証または暗号化を実行しないセキュリティ レベル。このレベルは、SNMPv3 ではサポートされていません。
- authNoPriv：認証は実行するが、暗号化を実行しないセキュリティ レベル。
- authPriv：認証と暗号化両方を実行するセキュリティ レベル。

SNMPv1、SNMPv2c、および SNMPv3 の 3 つのセキュリティ モデルを使用できます。セキュリティ モデルとセキュリティ レベルの組み合わせにより、SNMP メッセージの処理中に適用されるセキュリティ メカニズムが決まります。

***Table 18: SNMP* セキュリティ モデルおよびセキュリティ レベル**

モデル	レベル	認証	暗号化	結果
v1	noAuthNoPriv	コミュニティ ストリング	なし	コミュニティ ストリングの照合を使用して認証します。
v2c	noAuthNoPriv	コミュニティ ストリング	なし	コミュニティ ストリングの照合を使用して認証します。

モデル	レベル	認証	暗号化	結果
v3	authNoPriv	HMAC-MD5 または HMAC-SHA	未対応	Hash-Based Message Authentication Code（HMAC）メッセージ ダイジェスト 5（MD5）アルゴリズムまたは HMAC Secure Hash Algorithm（SHA）アルゴリズムに基づいて認証します。
v3	authPriv	HMAC-MD5 または HMAC-SHA	DES	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。データ暗号規格（DES）の 56 ビット暗号化、および暗号ブロック連鎖（CBC）DES（DES-56）標準に基づいて認証します。

ユーザベースのセキュリティ モデル

SNMPv3 ユーザーベース セキュリティ モデル（USM）は SNMP メッセージレベル セキュリティを参照し、次のサービスを提供します。

- メッセージの完全性：メッセージが不正な方法で変更または破壊されず、データ シーケンスが悪意なく起こり得る範囲を超えて変更されていないことを保証します。
- メッセージの発信元の認証：データを受信したユーザーが提示した ID の発信元を確認します。
- メッセージの機密性：情報が使用不可であること、または不正なユーザ、エンティティ、またはプロセスに開示されないことを保証します。

SNMPv3 は、設定済みユーザによる管理動作のみを許可し、SNMP メッセージを暗号化します。

Cisco NX-OSは、次の 2 つの SNMPv3 認証プロトコルを使用します。

- HMAC-MD5-96 認証プロトコル
- HMAC-SHA-96 認証プロトコル

Cisco NX-OS は、SNMPv3 メッセージ暗号化用プライバシープロトコルの 1 つとして、Advanced Encryption Standard（AES）を使用し、RFC 3826 に準拠します。

priv オプションで、SNMP セキュリティ暗号化方式として、DES または 128 ビット AES 暗号化を選択できます。**priv** オプションと **aes-128** トークンを併用すると、このプライバシー パスワードは 128 ビットの AES キー番号を生成するためのパスワードになります。AES priv パスワードは、8 文字以上の長さにできます。パスフレーズをクリアテキストで指定する場合、最大 64 文字を指定できます。ローカライズド キーを使用する場合は、最大 130 文字を指定できます。

Note 外部の AAA サーバーを使用して SNMPv3 を使う場合、外部 AAA サーバーのユーザー設定でプライバシー プロトコルに AES を指定する必要があります。

CLI および SNMP ユーザの同期

SNMPv3 ユーザ管理は、Access Authentication and Accounting（AAA）サーバ レベルで集中化できます。この中央集中型ユーザ管理により、Cisco NX-OSの SNMP エージェントは AAA サーバのユーザ認証サービスを利用できます。ユーザ認証が検証されると、SNMP PDU の処理が進行します。AAA サーバはユーザ グループ名の格納にも使用されます。SNMP はグループ名を使用して、スイッチでローカルに使用できるアクセス ポリシーまたはロール ポリシーを適用します。

ユーザ グループ、ロール、またはパスワードの設定が変更されると、SNMP と AAA の両方のデータベースが同期化されます。

Cisco NX-OS は、次のようにユーザー設定を同期化します。

- **snmp-server user** コマンドで指定された **auth** パスフレーズは、CLI ユーザーのパスワードになります。
- **username** コマンドで指定されたパスワードは、SNMP ユーザーの **auth** および **priv** パスフレーズになります。
- SNMP または CLI を使用してユーザを作成または削除すると、SNMP と CLI の両方でユーザが作成または削除されます。
- ユーザとロールの対応関係の変更は、SNMP と CLI で同期化されます。
- ロール変更（CLI からの削除または変更）は、SNMP と同期化されます。

Note パスフレーズまたはパスワードをローカライズしたキーおよび暗号形式で設定した場合、Cisco NX-OS はユーザー情報（パスワード、ルールなど）を同期させません。

グループベースの SNMP アクセス

Note グループは業界全体で使用されている標準的なSNMP用語なので、SNMPに関する説明では、「ロール」ではなく「グループ」を使用します。

SNMP アクセス権は、グループ別に編成されます。SNMP 内の各グループは、CLI を使用する場合のロールに似ています。各グループは3つのアクセス権により定義されます。つまり、読み取りアクセス、書き込みアクセス、および通知アクセスです。それぞれのアクセスを、各グループでイネーブルまたはディセーブルに設定できます。

ユーザ名が作成され、ユーザのロールが管理者によって設定され、ユーザがそのロールに追加されていれば、そのユーザはエージェントとの通信を開始できます。

SNMP の注意事項および制約事項

SNMP には、次の注意事項および制限事項があります。

- アクセス コントロール リスト（ACL）は、スイッチに設定されたローカル SNMPv3 ユーザのみに適用できます。ACL は、認証、許可、アカウンティング（AAA）サーバに保存されるリモート SNMPv3 ユーザに適用できません。
- Cisco NX-OS は、イーサネット MIB への読み取り専用アクセスをサポートします。詳細については次の URL http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml にアクセスして、Cisco NX-OS の MIB サポート リストを参照してください。
- Cisco NX-OS は、SNMPv3 noAuthNoPriv セキュリティ レベルをサポートしていません。
- Cisco NX-OS Release 7.0(3)I6(1) から以前のリリースへの無停止ダウングレード パスを行う場合、ローカル エンジン ID を設定していたなら、ローカル エンジン ID の設定を戻してから、SNMP ユーザとコミュニティ文字列を再設定する必要があります。
- Cisco Nexus 3000 シリーズ スイッチは、 要求に対して最大 10000 個のフラッシュ ファイルをサポートします。

SNMP のデフォルト設定

Table 19: デフォルトの ***SNMP*** パラメータ

パラメータ	デフォルト
ライセンス通知	イネーブル
linkUp/Down 通知タイプ	ietf-extended

SNMP の設定

SNMP 送信元インターフェイスの設定

特定のインターフェイスを使用するように SNMP を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **snmp-server source-interface** {**inform** | **trap**} *type slot/port*	すべての SNMP パケットの送信元インターフェイスを設定します。次のリストに、*interface* として有効な値を示します。 • ethernet • loopback • mgmt • port-channel • vlan
ステップ 3	switch(config)# **show snmp source-interface**	設定済みの SNMP 送信元インターフェイスを表示します。

例

次に、SNMP 送信元インターフェイスを設定する例を示します。

```
switch(config)# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server source-interface inform ethernet 1/10
switch(config)# snmp-server source-interface trap ethernet 1/10
switch(config)# show snmp source-interface
-----------------------------------------------------------------
Notification                    source-interface
-----------------------------------------------------------------
trap                              Ethernet1/10
inform                              Ethernet1/10
-----------------------------------------------------------------
```

SNMP ユーザの設定

Note Cisco NX-OS で SNMP ユーザーを構成するために使用するコマンドは、Cisco IOS でユーザーを構成するために使用されるものとは異なります。

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		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	switch(config)# **snmp-server user** *name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID** *id*] [**localizedkey**]] **Example:** `switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh`	認証およびプライバシー パラメータのある SNMP ユーザを設定します。 パスフレーズには最大 64 文字の英数字を使用できます。大文字と小文字が区別されます。 **localizedkey** キーワードを使用する場合は、パスフレーズに大文字と小文字を区別した英数字を 130 文字まで使用できます。 engineID の形式は、12 桁のコロンで区切った 10 進数字です。
ステップ 3	(Optional) switch# **show snmp user** **Example:** `switch(config) # show snmp user`	1 人または複数の SNMP ユーザーに関する情報を表示します。
ステップ 4	(Optional) **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

Example

次に、SNMP ユーザーを構成する例を示します。

```
switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh
```

SNMP メッセージ暗号化の適用

着信要求に認証または暗号化が必要となるよう SNMP を設定できます。デフォルトでは、SNMP エージェントは認証および暗号化を行わないでも SNMPv3 メッセージを受け付けます。プライバシーを適用する場合、Cisco NX-OS は、**noAuthNoPriv** または **authNoPriv** のいずれかのセキュリティ レベル パラメータを使用するすべての SNMPv3 PDU 要求に対して、許可エラーで応答します。

SNMP メッセージの暗号化を特定のユーザーに強制するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
switch(config)# **snmp-server user** *name* **enforcePriv**	このユーザーに対して SNMP メッセージ暗号化を適用します。

SNMP メッセージの暗号化をすべてのユーザーに強制するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
switch(config)# **snmp-server globalEnforcePriv**	すべてのユーザーに対して SNMP メッセージ暗号化を適用します。

SNMPv3 ユーザに対する複数のロールの割り当て

SNMP ユーザーを作成した後で、そのユーザーに複数のロールを割り当てることができます。

Note 他のユーザーにロールを割り当てることができるのは、network-admin ロールに属するユーザーだけです。

コマンド	目的
switch(config)# **snmp-server user** *name group*	この SNMP ユーザーと設定されたユーザー ロールをアソシエートします。

SNMP コミュニティの作成

SNMPv1 または SNMPv2c の SNMP コミュニティを作成できます。

コマンド	目的
switch(config)# **snmp-server community** *name group* {**ro** | **rw**}	SNMP コミュニティ ストリングを作成します。

SNMP 要求のフィルタリング

アクセス コントロール リスト（ACL）をコミュニティに割り当てて、着信 SNMP 要求にフィルタを適用できます。割り当てた ACL により着信要求パケットが許可される場合、SNMP はその要求を処理します。ACL により要求が拒否される場合、SNMP はその要求を廃棄して、システム メッセージを送信します。

ACL は次のパラメータで作成します。

- 送信元 IP アドレス
- 宛先 IP アドレス
- 送信元ポート
- 宛先ポート
- プロトコル（UDP または TCP）

ACL は、UDP および TCP を介する IPv4 および IPv6 の両方に適用されます。ACL を作成したら、ACL を SNMP コミュニティに割り当てます。

ヒント ACL の作成の詳細については、使用している Cisco Nexus シリーズ ソフトウェアの NX-OS セキュリティ コンフィギュレーション ガイドを参照してください。

ACL をコミュニティに割り当てて SNMP 要求をフィルタするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
switch(config)# **snmp-server community** *community name* **use-acl** *acl-name* **Example:** `switch(config)# snmp-server community public use-acl my_acl_for_public`	SNMP コミュニティに IPv4 ACL または IPv6 ACL を割り当てて SNMP 要求をフィルタします。

SNMP 通知レシーバの設定

複数のホスト レシーバーに対して SNMP 通知を生成するよう Cisco NX-OSを設定できます。

グローバル コンフィギュレーション モードで SNMPv1 トラップのホスト レシーバを設定できます。

コマンド	目的
switch(config)# **snmp-server host** *ip-address* **traps version 1** *community* [**udp_port** *number*]	SNMPv1 トラップのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。コミュニティは、最大 255 文字の英数字で指定できます。UDP ポート番号の範囲は 0 ～ 65535 です。

グローバル コンフィギュレーション モードで SNMPv2c トラップまたはインフォームのホスト レシーバを設定できます。

コマンド	目的
switch(config)# **snmp-server host** *ip-address* {**traps** | **informs**} **version 2c** *community* [**udp_port** *number*]	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。コミュニティは、最大 255 文字の英数字で指定できます。UDP ポート番号の範囲は 0 ～ 65535 です。

グローバル コンフィギュレーション モードで SNMPv3 トラップまたはインフォームのホスト レシーバを設定できます。

コマンド	目的
switch(config)# **snmp-server host** *ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port** *number*]	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。ユーザー名は、最大 255 文字の英数字で指定できます。UDP ポート番号の範囲は 0 ～ 65535 です。

Note SNMP マネージャは、SNMPv3 メッセージを認証し暗号解除するため、Cisco Nexus デバイスの SNMP engineID に基づくユーザー クレデンシャル（authKey/PrivKey）を認識していなければなりません。

次に、SNMPv1 トラップのホスト レシーバを設定する例を示します。

```
switch(config)# snmp-server host 192.0.2.1 traps version 1 public
```

次に、SNMPv2 インフォームのホスト レシーバを設定する例を示します。

```
switch(config)# snmp-server host 192.0.2.1 informs version 2c public
```

次に、SNMPv3 インフォームのホスト レシーバを設定する例を示します。

```
switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS
```

VRF を使用する SNMP 通知レシーバの設定

設定された VRF をホスト レシーバに接続するように Cisco NX-OS を設定できます。SNMP 通知レシーバの VRF 到達可能性およびフィルタリング オプションを設定すると、SNMP によって CISCO-SNMP-TARGET-EXT-MIB の cExtSnmpTargetVrfTable にエントリが追加されます。

（注） VRF 到達可能性またはフィルタリング オプションを設定する前に、ホストを設定する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch# **snmp-server host** *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*]	特定の VRF を使用してホスト レシーバと通信するように SNMP を設定します。IP アドレスは、IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。
ステップ 3	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、IP アドレス 192.0.2.1 の SNMP サーバー ホストを「Blue」という名前の VRF を使用するように設定する例を示します。

```
switch# configuration terminal
switch(config)# snmp-server host 192.0.2.1 use-vrf Blue
switch(config)# copy running-config startup-config
```

VRF に基づく SNMP 通知のフィルタリング

通知が発生した VRF に基づいて、Cisco NX-OS 通知をフィルタリングするように設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **snmp-server host** *ip-address* **filter-vrf** *vrf_name* [**udp_port** *number*]	設定された VRF に基づいて、通知ホスト レシーバへの通知をフィルタリングします。IP アドレスは、IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。 このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。
ステップ 3	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、VRF に基づいて SNMP 通知のフィルタリングを設定する例を示します。

```
switch# configuration terminal
switch(config)# snmp-server host 192.0.2.1 filter-vrf Red
switch(config)# copy running-config startup-config
```

インバンド アクセスのための SNMP の設定

次のものを使用して、インバンド アクセス用に SNMP を設定できます。

- コンテキストのない SNMP v2 の使用：コンテキストにマッピングされたコミュニティを使用できます。この場合、SNMP クライアントはコンテキストについて認識する必要はありません。
- コンテキストのある SNMP v2 の使用：SNMP クライアントはコミュニティ、たとえば、<community>@<context> を指定して、コンテキストを指定する必要があります。
- SNMP v3 の使用：コンテキストを指定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configuration 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。
ステップ 2	switch(config)# **snmp-server context** *context-name* **vrf** *vrf-name*	管理 VRF またはデフォルト VRF に SNMP コンテキストをマッピングします。カスタム VRF はサポートされません。 名前には最大 32 の英数字を使用できます。
ステップ 3	switch(config)# **snmp-server community** *community-name* **group** *group-name*	SNMPv2c コミュニティと SNMP コンテキストにマッピングし、コミュニティが属するグループを識別します。名前には最大 32 の英数字を使用できます。
ステップ 4	switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*	SNMPv2c コミュニティを SNMP コンテキストにマッピングします。名前には最大 32 の英数字を使用できます。

例

次の SNMPv2 の例は、コンテキストに snmpdefault という名前のコミュニティをマッピングする方法を示しています。

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community snmpdefault group network-admin
switch(config)# snmp-server mib community-map snmpdefault context def
switch(config)#
```

次の SNMPv2 の例は、マッピングされていないコミュニティ comm を設定し、インバンド アクセスする方法を示しています。

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community comm group network-admin
switch(config)#
```

次の SNMPv3 の例は、v3 ユーザー名とパスワードを使用する方法を示しています。

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)#
```

SNMP 通知のイネーブル化

通知をイネーブルまたはディセーブルにできます。通知名を指定しないと、Cisco NX-OSは通知をすべてイネーブルにします。

Note **snmp-server enable traps** CLI コマンドを使用すると、設定通知ホスト レシーバによっては、トラップとインフォームの両方をイネーブルにできます。

次の表に、Cisco NX-OS MIB の通知をイネーブルにする CLI コマンドを示します。

***Table 20: SNMP* 通知のイネーブル化**

MIB	関連コマンド
すべての通知	**snmp-server enable traps**
CISCO-ERR-DISABLE-MIB	**snmp-server enable traps show interface status**
Q-BRIDGE-MIB	**snmp-server enable traps show mac address-table**
CISCO-SWITCH-QOS-MIB	**snmp-server enable traps show hardware internal buffer info pkt-stats**
BRIDGE-MIB	**snmp-server enable traps bridge newroot** **snmp-server enable traps bridge topologychange**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa**
ENITY-MIB、 CISCO-ENTITY-FRU-CONTROL-MIB、 CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity fru**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license**
IF-MIB	**snmp-server enable traps link**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-FCC-MIB	**snmp-server enable traps fcc**
CISCO-DM-MIB	**snmp-server enable traps fcdomain**
CISCO-NS-MIB	**snmp-server enable traps fcns**
CISCO-FCS-MIB	**snmp-server enable traps fcs discovery-complete** **snmp-server enable traps fcs request-reject**

MIB	関連コマンド
CISCO-FDMI-MIB	**snmp-server enable traps fdmi**
CISCO-FSPF-MIB	**snmp-server enable traps fspf**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
CISCO-RSCN-MIB	**snmp-server enable traps rscn** **snmp-server enable traps rscn els** **snmp-server enable traps rscn ils**
CISCO-ZS-MIB	**snmp-server enable traps zone** **snmp-server enable traps zone default-zone-behavior-change** **snmp-server enable traps zone enhanced-zone-db-change** **snmp-server enable traps zone merge-failure** **snmp-server enable traps zone merge-success** **snmp-server enable traps zone request-reject** **snmp-server enable traps zone unsupp-mem**
CISCO-CONFIG-MAN-MIB **Note** ccmCLIRunningConfigChanged 通知を除き、MIB オブジェクトをサポートしていません。	**snmp-server enable traps config**

Note ライセンス通知は、デフォルトではイネーブルです。

グローバル コンフィギュレーション モードで指定の通知をイネーブルにするには、次の作業を行います。

コマンド	目的
switch(config)# **snmp-server enable traps**	すべての SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps aaa** [**server-state-change**]	AAA SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps entity** [**fru**]	ENTITY-MIB SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps license**	ライセンス SNMP 通知をイネーブルにします。

コマンド	目的
switch(config)# **snmp-server enable traps port-security**	ポート セキュリティ SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps snmp** [**authentication**]	SNMP エージェント通知をイネーブルにします。

リンクの通知の設定

デバイスに対して、イネーブルにする linkUp/linkDown 通知を設定できます。次のタイプの linkUp/linkDown 通知をイネーブルにできます。

- cieLinkDown：シスコ拡張リンク ステート ダウン通知をイネーブルにします。
- cieLinkUp：シスコ拡張リンク ステート アップ通知をイネーブルにします。
- cisco-xcvr-mon-status-chg：シスコ インターフェイス トランシーバ モニター ステータス変更通知をイネーブルにします。
- delayed-link-state-change：遅延リンク ステート変更をイネーブルにします。
- extended-linkUp：IETF 拡張リンク ステート アップ通知をイネーブルにします。
- extended-linkDown：IETF 拡張リンク ステート ダウン通知をイネーブルにします。
- linkDown：IETF リンク ステート ダウン通知をイネーブルにします。
- linkUp：IETF リンク ステート アップ通知をイネーブルにします。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**snmp-server enable traps link** [**cieLinkDown** | **cieLinkUp** | **cisco-xcvr-mon-status-chg** | **delayed-link-state-change**] | **extended-linkUp** | **extended-linkDown** | **linkDown** | **linkUp**] 例： `switch(config)# snmp-server enable traps link cieLinkDown`	リンク SNMP 通知をイネーブルにします。

インターフェイスでのリンク通知のディセーブル化

個別のインターフェイスで linkUp および linkDown 通知をディセーブルにできます。これにより、フラッピングインターフェイス（アップとダウン間の移行を繰り返しているインターフェイス）に関する通知を制限できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **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。
ステップ 2	switch(config)# **interface** *type slot/port*	変更するインターフェイスを指定します。
ステップ 3	switch(config -if)# **no snmp trap link-status**	インターフェイスの SNMP リンクステート トラップをディセーブルにします。この機能は、デフォルトでイネーブルにされています。

TCP での SNMP に対するワンタイム認証のイネーブル化

TCP セッション上で SNMP に対するワンタイム認証をイネーブルにできます。

コマンド	目的
switch(config)# **snmp-server tcp-session** [**auth**]	TCP セッション上で SNMP に対するワンタイム認証をイネーブルにします。この機能はデフォルトで無効に設定されています。

SNMP スイッチの連絡先および場所の情報の割り当て

スイッチの連絡先情報（スペースを含めず、最大 32 文字まで）およびスイッチの場所を割り当てることができます。

Procedure

	Command or Action	**Purpose**
ステップ 1	switch# **configuration 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。
ステップ 2	switch(config)# **snmp-server contact** *name*	sysContact（SNMP 担当者名）を設定します。

	Command or Action	Purpose
ステップ 3	switch(config)# **snmp-server location** *name*	sysLocation（SNMP ロケーション）を設定します。
ステップ 4	(Optional) switch# **show snmp**	1 つまたは複数の宛先プロファイルに関する情報を表示します。
ステップ 5	(Optional) switch# **copy running-config startup-config**	この設定変更を保存します。

コンテキストとネットワーク エンティティ間のマッピング設定

プロトコル インスタンス、VRF などの論理ネットワーク エンティティに対する SNMP コンテキストのマッピングを設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configuration 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。
ステップ 2	switch(config)# **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	SNMP コンテキストをプロトコル インスタンス、VRF、またはトポロジにマッピングします。名前には最大 32 の英数字を使用できます。
ステップ 3	switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*	SNMPv2c コミュニティを SNMP コンテキストにマッピングします。名前には最大 32 の英数字を使用できます。
ステップ 4	(Optional) switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	SNMP コンテキストとプロトコル インスタンス、VRF、またはトポロジ間のマッピングを削除します。名前には最大 32 の英数字を使用できます。 **Note** コンテキスト マッピングを削除する目的で、インスタンス、VRF、またはトポロジを入力しないでください。**instance**、**vrf**、または **topology** キーワードを使用すると、コンテキストとゼロ長ストリング間のマッピングが設定されます。

SNMP ローカル エンジン ID の設定

Cisco NX-OS リリース 7.0(3)I6(1) 以降では、ローカル デバイスにエンジン ID を設定できます。

Note SNMP ローカル エンジン ID を設定すると、すべての SNMP ユーザ、V3 ユーザに設定されたホスト、およびコミュニティ ストリングを再設定する必要があります。Cisco NX-OS リリース 7.0(3)I7(1) 以降では、SNMP ユーザとコミュニティ ストリングのみを再設定する必要があります。

Procedure

	Command or Action	Purpose

ステップ 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		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	**snmp-server engineID local** *engineid-string* **Example:** `switch(config)# snmp-server engineID local AA:BB:CC:1A:2C:10`	ローカル デバイスの SNMP engineID を変更します。 ローカル エンジン ID は、コロンで指定された 16 進数オクテットのリストとして設定する必要があります。ここでは 10 ～ 64 の範囲の偶数 16 進数文字が使用され、2 つの 16 進数文字ごとにコロンで区切られます。たとえば、i80:00:02:b8:04:61:62:63 です。
ステップ 3	**show snmp engineID** **Example:** `switch(config)# show snmp engineID`	設定されている SNMP エンジンの ID を表示します。
ステップ 4	**[no] snmp-server engineID local** *engineid-string* **Example:** `switch(config)# no snmp-server engineID local AA:BB:CC:1A:2C:10`	ローカル エンジン ID を無効にし、自動生成されたデフォルトのエンジン ID を設定します。
ステップ 5	Required: **copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SNMP のディセーブル化

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	switch(config) # **no snmp-server protocol enable** 例： `no snmp-server protocol enable`	SNMP をディセーブルにします。 SNMPは、デフォルトでディセーブルになっています。

SNMP 設定の確認

SNMP 設定情報を表示するには、次の作業を行います。

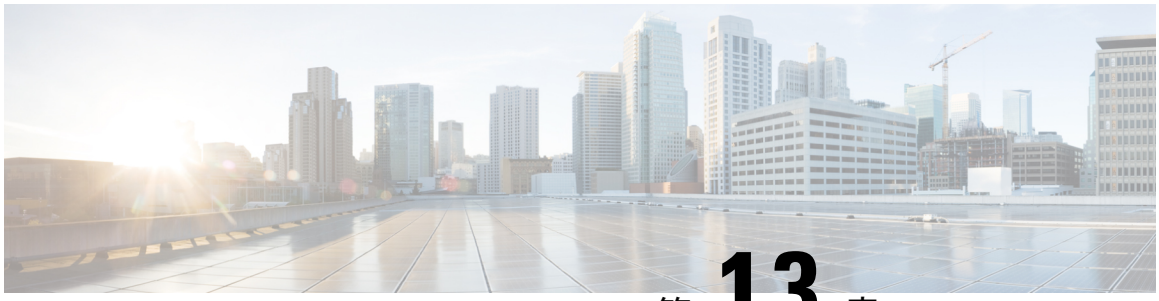
コマンド	目的
show snmp	SNMP ステータスを表示します。
show snmp community	SNMP コミュニティ ストリングを表示します。
show interface snmp-ifindex	すべてのインターフェイスについて（IF-MIB から）SNMP の ifIndex 値を表示します。
show running-config snmp [all]	SNMP の実行コンフィギュレーションを表示します。
show snmp engineID	SNMP engineID を表示します。
show snmp group	SNMP ロールを表示します。
show snmp sessions	SNMP セッションを表示します。
show snmp context	SNMP コンテキスト マッピングを表示します。
show snmp host	設定した SNMP ホストの情報を表示します。
show snmp source-interface	設定した発信元インターフェイスの情報を表示します。
show snmp trap	イネーブルまたはディセーブルである SNMP 通知を表示します。

コマンド	目的
show snmp user	SNMPv3 ユーザを表示します。

その他の参考資料

MIB

MIB	MIB のリンク
SNMP に関連する MIB	サポートされている MIB を検索およびダウンロード 次の URL にアクセスしてください。 https://cisco.github.io/cisco-mibs/supportlists/nexus3000/Nexus3000MIBSupportList.html



第 13 章

PCAP SNMP パーサーの使用

この章は、次の項で構成されています。

PCAP SNMP パーサーの使用

PCAP SNMP パーサーは、.pcap 形式でキャプチャされた SNMP パケットを分析するツールです。スイッチ上で動作し、スイッチに送信されるすべての SNMP get、getnext、getbulk、set、trap、および response 要求の統計情報レポートを生成します。

PCAP SNMP パーサーを使用するには、次のいずれかのコマンドを使用します。

- **debug packet-analysis snmp** [**mgmt0** | **inband**] **duration** *seconds* [*output-file*] [**keep-pcap**]：Tshark を使用して指定の秒数間のパケットをキャプチャし、一時 .pcap ファイルに保存します。次に、その .pcap ファイルに基づいてパケットを分析します。

  結果は出力ファイルに保存されます。出力ファイルが指定されていない場合は、コンソールに出力されます。**keep-pcap** オプションを使用する場合を除き、一時 .pcap ファイルはデフォルトで削除されます。パケット キャプチャは、デフォルトの管理インターフェイス（mgmt0）、または帯域内インターフェイスで実行できます。

  例：

```
switch# debug packet-analysis snmp duration 100

switch# debug packet-analysis snmp duration 100 bootflash:snmp_stats.log

switch# debug packet-analysis snmp duration 100 bootflash:snmp_stats.log keep-pcap

switch# debug packet-analysis snmp inband duration 100

switch# debug packet-analysis snmp inband duration 100 bootflash:snmp_stats.log

switch# debug packet-analysis snmp inband duration 100 bootflash:snmp_stats.log
keep-pcap
```

- **debug packet-analysis snmp** *input-pcap-file* [*output-file*]：既存の .pcap ファイルにあるキャプチャしたパケットを分析します。

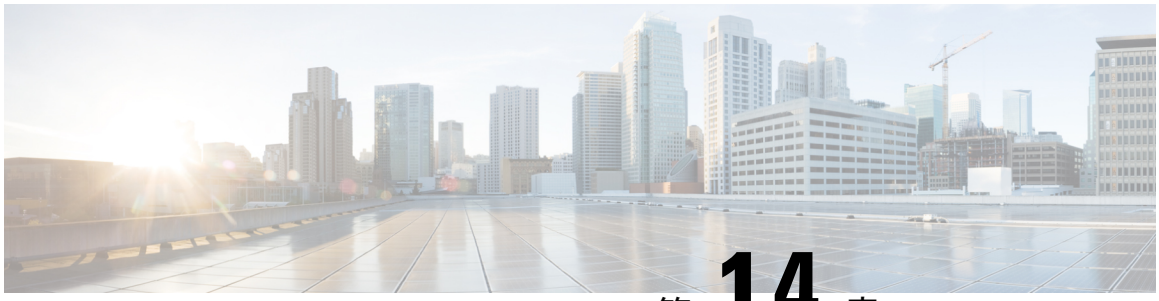
PCAP SNMP パーサーの使用

例：

```
switch# debug packet-analysis snmp bootflash:snmp.pcap

switch# debug packet-analysis snmp bootflash:snmp.pcap bootflash:snmp_stats.log
```

次に、**debug packet-analysis snmp** [**mgmt0** | **inband**] **duration** コマンドの統計情報レポートの例を示します。

```
switch# debug packet-analysis snmp duration 10
Capturing on eth0
36
wireshark-cisco-mtc-dissector: ethertype=0xde09, devicetype=0x0
wireshark-broadcom-rcpu-dissector: ethertype=0xde08, devicetype=0x0

Started analyzing. It may take several minutes, please wait!

Statistics Report
----------------------------------------
SNMP Packet Capture Duration: 0 seconds
Total Hosts: 1
Total Requests: 18
Total Responses: 18
Total GET: 0
Total GETNEXT: 0
Total WALK: 1 (NEXT: 18)
Total GETBULK: 0
Total BULKWALK: 0 (BULK: 0)
Total SET: 0
Total TRAP: 0
Total INFORM: 0

Hosts        GET  GETNEXT  WALK(NEXT) GETBULK  BULKWALK(BULK) SET  TRAP  INFORM  RESPONSE
-------------------------------------------------------------------------------------------
10.22.27.244   0     0        1(18)       0         0(0)          0    0      0       18

Sessions
--------
1

MIB Objects GET  GETNEXT  WALK(NEXT) GETBULK(Non_rep/Max_rep) BULKWALK(BULK,
Non_rep/Max_rep)
-------------------------------------------------------------------------------------------
ifName      0       0        1(18)       0                          0


SET     Hosts
--------------------
0       10.22.27.244
```



第 14 章

RMON の設定

この章は、次の項で構成されています。

RMON について

RMON は、各種のネットワーク エージェントおよびコンソール システムがネットワーク モニタリング データを交換できるようにするための、Internet Engineering Task Force（IETF）標準モニタリング仕様です。Cisco NX-OS は、Cisco Nexus デバイスをモニタリングするための RMON アラーム、イベント、およびログをサポートします。

RMONアラームは、指定された期間、特定の管理情報ベース（MIB）オブジェクトをモニタリングし、指定されたしきい値でアラームを発生させ、別のしきい値でアラームをリセットします。アラームと RMON イベントを組み合わせて使用し、RMON アラームが発生したときにログ エントリまたは SNMP 通知を生成できます。

Cisco Nexus デバイスでは RMON はデフォルトでディセーブルに設定されており、イベントまたはアラームは設定されていません。RMONアラームおよびイベントを設定するには、CLI または SNMP 互換ネットワーク管理ステーションを使用します。

RMON アラーム

SNMP INTEGER タイプの解決を行う任意の MIB オブジェクトにアラームを設定できます。指定されたオブジェクトは、標準のドット付き表記（たとえば、1.3.6.1.2.1.2.2.1.17 は ifOutOctets.17 を表します）の既存の SNMP MIB オブジェクトでなければなりません。

アラームを作成する場合、次のパラメータを指定します。

- モニタリングする MIB オブジェクト

- サンプリング間隔：MIB オブジェクトのサンプル値を収集するのに Cisco Nexus デバイスが使用する間隔
- サンプル タイプ：絶対サンプルでは、MIB オブジェクト値の現在のスナップショットを使用します。デルタ サンプルは連続した2つのサンプルを使用し、これらの差を計算します。
- 上限しきい値：Cisco Nexus デバイスが上限アラームを発生させる、または下限アラームをリセットするときの値
- 下限しきい値：Cisco Nexus デバイスが下限アラームを発生させる、または上限アラームをリセットするときの値
- イベント：アラーム（上限または下限）の発生時に Cisco Nexus デバイスが実行するアクション

Note hcalarms オプションを使用して、アラームを 64 ビットの整数の MIB オブジェクトに設定します。

たとえば、エラー カウンタ MIB オブジェクトにデルタ タイプ上限アラームを設定できます。エラー カウンタ デルタがこの値を超えた場合、SNMP 通知を送信し、上限アラーム イベントを記録するイベントを発生させることができます。この上限アラームは、エラーカウンタのデルタ サンプルが下限しきい値を下回るまで再度発生しません。

Note 下限しきい値には、上限しきい値よりも小さな値を指定してください。

RMON イベント

特定のイベントを各 RMON アラームにアソシエートさせることができます。RMON は次のイベント タイプをサポートします。

- SNMP 通知：関連したアラームが発生したときに、SNMP risingAlarm または fallingAlarm 通知を送信します。
- ログ：関連したアラームが発生した場合、RMON ログテーブルにエントリを追加します。
- 両方：関連したアラームが発生した場合、SNMP 通知を送信し、RMON ログ テーブルにエントリを追加します。

下限アラームおよび上限アラームに異なるイベントを指定できます。

RMON の設定時の注意事項および制約事項

RMON には、次の注意事項および制限事項があります。

- SNMP通知イベントタイプを使用するには、SNMPユーザおよび通知レシーバを設定する必要があります。
- 整数になる MIB オブジェクトに、RMON アラームのみを設定できます。

RMON 設定の確認

RMON の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show rmon alarms	RMON アラームに関する情報を表示します。
show rmon events	RMON イベントに関する情報を表示します。
show rmon hcalarms	RMON高容量アラームに関する情報を表示します。
show rmon logs	RMON ログに関する情報を表示します。

デフォルトの RMON 設定

次の表に、RMON パラメータのデフォルト設定を示します。

Table 21: デフォルトの ***RMON*** パラメータ

パラメータ	デフォルト
アラーム	未設定
イベント	未設定

RMON アラームの設定

任意の整数の SNMP MIB オブジェクトに RMON アラームを設定できます。

次のパラメータを任意で指定することもできます。

- 上限および下限しきい値が指定値を超えた場合に発生させるイベント番号

- アラームのオーナー

SNMP ユーザが設定され、SNMP 通知がイネーブルであることを確認します。

Before you begin

SNMP ユーザーが設定され、SNMP 通知がイネーブルであることを確認します。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **rmon alarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-index*] **falling-threshold** *value* [*event-index*] [**owner** *name*]	RMON アラームを作成します。値の範囲は -2147483647 ～ 2147483647 です。オーナー名は任意の英数字ストリングです。
ステップ 3	switch(config)# **rmon hcalarm** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold-high** *value* **rising-threshold-low** *value* [*event-index*] **falling-threshold-high** *value* **falling-threshold-low** *value* [*event-index*] [**owner** *name*] [**storagetype** *type*]	RMON 高容量アラームを作成します。値の範囲は -2147483647 ～ 2147483647 です。オーナー名は任意の英数字ストリングです。 ストレージタイプの範囲は 1 ～ 5 です。
ステップ 4	(Optional) switch# **show rmon** {**alarms** | **hcalarms**}	RMON アラームまたは高容量アラームに関する情報を表示します。
ステップ 5	(Optional) switch# **copy running-config startup-config**	この設定変更を保存します。

Example

次に、RMON アラームを設定する例を示します。

```
switch# configure terminal
switch(config)# rmon alarm 1 1.3.6.1.2.1.2.2.1.17.83886080 5 delta rising-threshold 5 1
 falling-threshold 0 owner test
switch(config)# exit
switch# show rmon alarms
Alarm 1 is active, owned by test
 Monitors 1.3.6.1.2.1.2.2.1.17.83886080 every 5 second(s)
 Taking delta samples, last value was 0
 Rising threshold is 5, assigned to event 1
 Falling threshold is 0, assigned to event 0
```

```
On startup enable rising or falling alarm
```

RMON イベントの設定

RMON アラームとアソシエートするよう RMON イベントを設定できます。複数の RMON アラームで同じイベントを再利用できます。

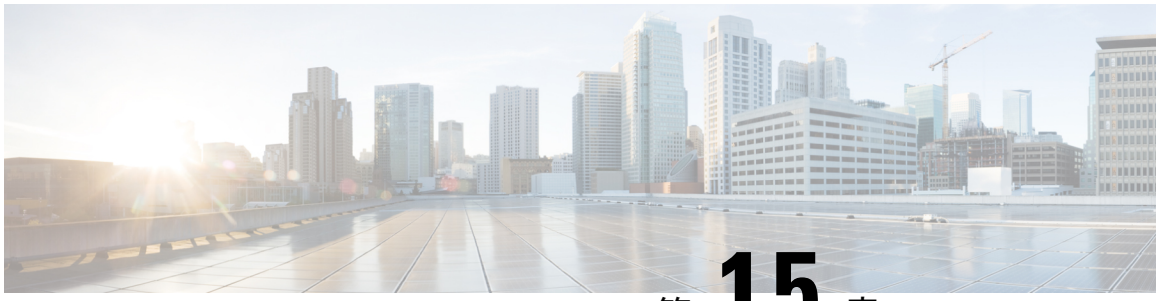
SNMP ユーザが設定され、SNMP 通知がイネーブルであることを確認します。

Before you begin

SNMP ユーザーが設定され、SNMP 通知がイネーブルであることを確認します。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **rmon event** *index* [**description** *string*] [**log**] [**trap**] [**owner** *name*]	RMON イベントを設定します。説明のストリングおよびオーナー名は、任意の英数字ストリングです。
ステップ 3	(Optional) switch(config)# **show rmon** {**alarms** | **hcalarms**}	RMON アラームまたは高容量アラームに関する情報を表示します。
ステップ 4	(Optional) switch# **copy running-config startup-config**	この設定変更を保存します。



第 15 章

オンライン診断の設定

この章は、次の項で構成されています。

オンライン診断について

オンライン診断では、スイッチの起動時またはリセット時にハードウェア コンポーネントを確認し、通常の動作時にはハードウェアの状態を監視します。

Cisco Nexus シリーズ スイッチは、起動時診断および実行時診断をサポートします。起動時診断には、システム起動時とリセット時に実行する、中断を伴うテストおよび非中断テストが含まれます。

実行時診断（ヘルス モニタリング診断）には、スイッチの通常の動作時にバックグラウンドで実行する非中断テストが含まれます。

ブートアップ診断

起動時診断は、スイッチをオンラインにする前にハードウェアの障害を検出します。起動診断では、スーパーバイザと ASIC の間のデータ パスと制御パスの接続も確認します。次の表に、スイッチの起動時またはリセット時にだけ実行される診断を示します。

Table 22: ブートアップ診断

診断	説明
PCIe	PCI express（PCIe）アクセスをテストします。
NVRAM	NVRAM（不揮発性 RAM）の整合性を確認します。

診断	説明
インバンドポート	インバンドポートとスーパーバイザの接続をテストします。
管理ポート	管理ポートをテストします。
メモリ	DRAM の整合性を確認します。

起動時診断には、ヘルス モニタリング診断と共通するテスト セットも含まれます。

起動時診断では、オンボード障害ロギング（OBFL）システムに障害を記録します。また、障害により LED が表示され、診断テストのステート（on、off、pass、または fail）を示します。

起動診断テストをバイパスするように Cisco Nexus デバイス を設定することも、またはすべての起動診断テストを実行するように設定することもできます。

ヘルス モニタリング診断

ヘルス モニタリング診断では、スイッチの状態に関する情報を提供します。実行時のハードウェア エラー、メモリ エラー、ソフトウェア障害、およびリソースの不足を検出します。

ヘルス モニタリング診断は中断されずにバックグラウンドで実行され、ライブ ネットワーク トラフィックを処理するスイッチの状態を確認します。

次の表に、スイッチのヘルス モニタリング診断を示します。

Table 23: **ヘルス モニタリング診断テスト**

診断	説明
LED	ポートおよびシステムのステータス LED を監視します。
電源モジュール	電源装置のヘルス ステータスを監視します。
温度センサー	温度センサーの読み取り値を監視します。
テスト ファン	ファンの速度およびファンの制御をモニターします。

Note スイッチが吸気温度のしきい値に達し、120 秒の制限内には温度が低下しない場合、スイッチを復旧するには、スイッチの電源をオフにして、電源装置を再装着する必要があります。

次の表に、システム起動時とリセット時にも実行されるヘルス モニタリング診断を示します。

Table 24: ヘルス モニタリングおよび起動時診断テスト

診断	説明
SPROM	バックプレーンとスーパーバイザ SPROM の整合性を確認します。
ファブリック エンジン	スイッチ ファブリック ASIC をテストします。
ファブリック ポート	スイッチ ファブリック ASIC 上のポートをテストします。
転送エンジン	転送エンジン ASIC をテストします。
転送エンジン ポート	転送エンジン ASIC 上のポートをテストします。
前面ポート	前面ポート上のコンポーネント（PHY および MAC など）をテストします。

Note スイッチが 70 度（摂氏）の内部温度しきい値を超え、120 秒以内にしきい値の制限以下に温度が低下しない場合、スイッチを復旧するには、スイッチの電源をオフにして、スイッチの電源を再投入する必要があります。

拡張モジュール診断

スイッチの起動時またはリセット時の起動時診断には、スイッチのインサービス拡張モジュールのテストが含まれます。

稼働中のスイッチに拡張モジュールを挿入すると、診断テストセットが実行されます。次の表に、拡張モジュールの起動時診断を示します。これらのテストは、起動時診断と共通です。起動時診断が失敗した場合、拡張モジュールはサービス状態になりません。

Table 25: 拡張モジュールの起動時診断およびヘルス モニタリング診断

診断	説明
SPROM	バックプレーンとスーパーバイザ SPROM の整合性を確認します。
ファブリック エンジン	スイッチ ファブリック ASIC をテストします。
ファブリック ポート	スイッチ ファブリック ASIC 上のポートをテストします。
転送エンジン	転送エンジン ASIC をテストします。
転送エンジン ポート	転送エンジン ASIC 上のポートをテストします。
前面ポート	前面ポート上のコンポーネント（PHY および MAC など）をテストします。

ヘルス モニタリング診断は、IS 拡張モジュールで実行されます。次の表で、拡張モジュールのヘルス モニタリング診断に固有の追加のテストについて説明します。

Table 26: 拡張モジュールのヘルス モニタリング診断

診断	説明
LED	ポートおよびシステムのステータス LED を監視します。
温度センサー	温度センサーの読み取り値を監視します。

オンライン診断の注意事項と制約事項

オンライン診断には、次の注意事項と制限事項があります。

- 中断を伴うオンライン診断テストをオンデマンド方式で実行することはできません。
- BootupPortLoopback テストはサポートされていません。
- インターフェイス Rx および Tx パケット カウンタは、シャットダウン状態のポートで増えます（およそ 15 分ごとに 4 パケット）。
- 管理ダウン ポートでは、ユニキャスト パケット Rx および Tx のカウンタが、GOLD ループバック パケットに対して追加されます。PortLoopback テストがオン デマンドなのは Cisco NX-OS 7.0(3)I1(2) より前のリリースであるため、パケット カウンタが追加されるのは、テストを管理ダウン ポートで実行する場合だけです。Cisco NX-OS リリース 7.0(3)I1(2) 以降では PortLoopback テストは定期的に行われるため、パケット カウンタは管理ダウン ポートで30分ごとに追加されます。テストは管理ダウン ポートでのみ実行されます。ポートが閉じられている場合は、カウンタは影響を受けません。

オンライン診断の設定

完全なテストセットを実行するよう起動時診断を設定できます。もしくは、高速モジュール起動時のすべての起動時診断テストをバイパスできます。

Note 起動時オンライン診断レベルを complete に設定することを推奨します。起動時オンライン診断をバイパスすることは推奨しません。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **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。
ステップ 2	switch(config)# **diagnostic bootup level** [**complete** | **bypass**]	デバイスの起動時に診断を実行するよう起動時診断レベルを次のように設定します。 • **complete**：すべての起動時診断を実行します。これはデフォルト値です。 • **bypass**：起動時診断を実行しません。
ステップ 3	(Optional) switch# **show diagnostic bootup level**	現在、スイッチで実行されている起動時診断レベル（bypass または complete）を表示します。

Example

次に、完全な診断を実行するよう起動時診断レベルを設定する例を示します。

```
switch# configure terminal
switch(config)# diagnostic bootup level complete
```

オンライン診断設定の確認

オンライン診断の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show diagnostic bootup level	起動時診断レベルを表示します。
show diagnostic result module *slot*	診断テストの結果を表示します。

オンライン診断のデフォルト設定

次の表に、オンライン診断パラメータのデフォルト設定を示します。

Table 27: デフォルトのオンライン診断パラメータ

パラメータ	デフォルト
起動時診断レベル	complete

パリティ エラーの診断

パリティ エラーのクリア

hardware profile parity-error {l2-table | l3-table} clear コマンドを使用して、パリティ エラーが検出された場合、対応するレイヤ 2 またはレイヤ 3 テーブル エントリ（0 付き）をクリアできます。このコマンドは、実行コンフィギュレーションでのシステムの起動時に有効です。また、このコマンドは有効にする必要があるため、設定を保存後、システムを再起動してコマンドを有効にします。

重要 このコマンドは、Cisco NX-OS リリース 6.0(2)U2(1) 以降のバージョンではサポートされていません。

次のガイドラインが適用されます。

- l2_entry テーブルにこのコマンドが使用されている場合、トラフィック パターンのためにクリアされたエントリを再学習する必要があります。
- l3_entry_only（ホスト）テーブルにこのコマンドが使用されている場合、クリアされたエントリは再学習されません。

このコマンドは、次のお客様の設定で役立ちます。

- L2_Entry テーブル（スタティック L2_entry テーブル エントリなし）

  L2_Entry テーブル エントリがクリアされている場合、エントリはトラフィック パターンから動的に学習する必要があります。IGMP やマルチキャストから学習することはできません。

- L3_Entry_only（ホスト）テーブル

  お客様はホスト テーブルを使用できません。**hardware profile unicast enable-host-ecmp** コマンドを有効にする必要があります。この場合、カスタマー ノードの L3_Entry_only テーブルには有効なエントリが存在しないため、L3_Entry_only エントリ テーブルをクリアしても何の影響も生じません。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **hardware profile parity-error l2-table clear**	レイヤ 2 テーブルのパリティ エラー エントリをクリアします。
ステップ 3	switch(config)# **hardware profile parity-error l3-table clear**	レイヤ 3 テーブルのパリティ エラー エントリをクリアします。

例

次に、レイヤ 2 テーブルのパリティ エラーをクリアする例を示します。

```
switch# configure terminal
switch(config)# hardware profile parity-error l2-table clear
switch(config)# copy running-config startup-config
switch(config)# reload
```

次に、レイヤ 3 テーブルのパリティ エラーをクリアする例を示します。

```
switch# configure terminal
switch(config)# hardware profile parity-error l3-table clear
switch(config)# copy running-config startup-config
switch(config)# reload
```

ソフト エラー リカバリ

Cisco NX-OS リリース 6.0(2)U2(1) には、フォワーディング エンジンの内蔵メモリ テーブルにおけるソフト エラーに対するソフトウェア エラー リカバリ（SER）が導入されています。この機能は、デフォルトでイネーブルにされています。

フォワーディングエンジンの内蔵コントロールテーブルとパケット メモリは、エラー訂正コード（ECC）、パリティ保護、またはテーブルのパリティ チェックに基づいたソフトウェア スキャンなど、さまざまなメカニズムによって保護されます。ソフトウェアのキャッシュは、大部分のハードウェア テーブルで保持されます。パリティ エラーおよび ECC エラーは、トラフィックが影響を受けているエントリにヒットすると検出されます。Ternary Content Addressable Memory（TCAM）の場合、CPU によってソフトウェア シャドウ エントリとハードウェア エントリが比較されるときにエラーが検出されます。これらのいずれかのタイプのエラーが検出されると、そのメモリのエラーを報告するための割り込みが発生します。

修正メカニズムは、ハードウェア テーブルごとに異なります。ソフトウェア シャドウがあるハードウェア テーブルの場合は、影響を受けているエントリがソフトウェア キャッシュからコピーされて、割り込みがクリアされます。レイヤ 3 ホスト ルックアップ テーブルや ACL TCAM テーブルなどのハードウェア テーブルは、この方法で検出されて修正されます。ソフトウェア シャドウがないハードウェア テーブルの場合は、影響を受けているエントリがクリ

アされるか、またはゼロ設定されます。ハードウェア学習されたレイヤ2エントリ テーブルなどのハードウェアテーブルおよびカウンタのメモリは、この方法で検出されて修正されます。

パケットのフォワーディングルックアップ時にハードウェアでパリティエラーが発生すると、パリティエラーが発生したテーブルによってはパケットがドロップされます。パリティエラーの検出から修正までのリカバリ時間は、この場合、1 エントリで 600 マイクロ秒以上かかります。トラフィックがこのエントリにヒットしている場合、この期間のトラフィックは失われます。

パリティ保護されていない TCAM テーブルの場合、パリティ エラーを検出するために、テーブル エントリに対する定期的なソフトウェア スキャンが実行されます。パリティ エラーが検出された場合、影響を受けているメモリ位置がソフトウェア シャドウからコピーされて、エラーが修正されます。ソフトウェア起動のスキャンは 10 秒ごとに行われ、1 回のスキャンで 4,000 エントリがスキャンされます。フォワーディング エンジンには、スキャン対象の TCAM エントリが約 36,000 あります。最悪の場合、これらのテーブルのパリティ エラーを検出して修正するのに 90 秒以上かかります。リカバリ時間は、システムの負荷に基づき算出されます。

回復不能なパリティ エラーの場合、次の例のような、syslog イベント通知が生成されます。

```
2013 Nov 14 12:37:32 switch %USER-3-SYSTEM_MSG: bcm_usd_isr_switch_event_cb_log:658:
slot_num 0, event 2, memory error type: Detection(0x1), table name: Ingress ACL result
table(0x830004b5), index: 1790  - bcm_usd
```

メモリ テーブルの状態の確認

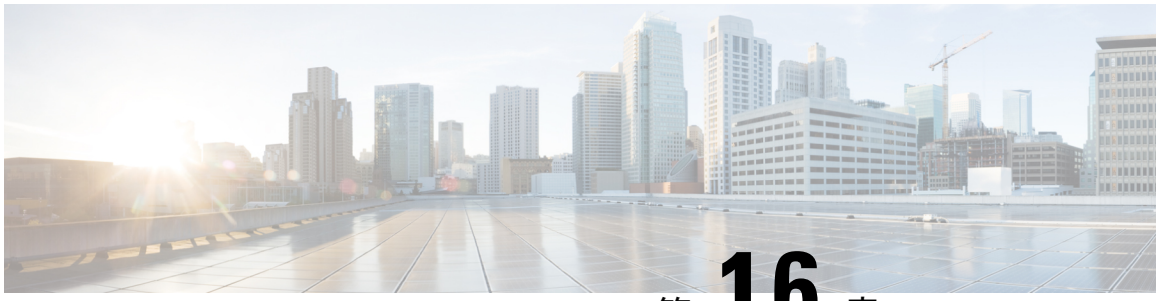
ASIC メモリ テーブルで発生したパリティ エラー数の概要を表示するには、次のコマンドを実行します。

コマンド	目的
show hardware forwarding memory health summary	ASIC メモリ テーブルのパリティ エラー数の概要を表示します。

例

次に、ASIC メモリ テーブルのパリティ エラー数の概要を表示する例を示します。

```
switch# show hardware forwarding memory health summary
Parity error counters:
Total parity error detections: 7
Total parity error corrections: 7
Total TCAM table parity error detections: 1
Total TCAM table parity error corrections: 1
Total SRAM table parity error detections: 6
Total SRAM table parity error corrections: 6
Parity error summary:
Table ID: L2 table      Detections: 1   Corrections: 1
Table ID: L3 Host table Detections: 1   Corrections: 1
Table ID: L3 LPM table  Detections: 1   Corrections: 1
Table ID: L3 LPM result table   Detections: 1   Corrections: 1
Table ID: Ingress pre-lookup ACL result table   Detections: 1   Corrections: 1
Table ID: Ingress ACL result table      Detections: 1   Corrections: 1
Table ID: Egress ACL result table       Detections: 1   Corrections: 1
```



第 16 章

Embedded Event Manager の設定

この章は、次の項で構成されています。

Embedded Event Manager について

Cisco NX-OS システム内のクリティカル イベントを検出して処理する機能は、ハイ アベイラビリティにとって重要です。Embedded Event Manager（EEM）は、デバイス上で発生するイベントをモニターし、設定に基づいてこれらのイベントを回復またはトラブルシューティングするためのアクションを実行することによってシステム内のイベントを検出して処理する、中央のポリシー駆動型のフレームワークを提供します。

EEM は次の 3 種類の主要コンポーネントからなります。

イベント文

何らかのアクション、回避策、または通知が必要になる可能性のある、別の Cisco NX-OS コンポーネントからモニターするイベント。

アクション文

電子メールの送信やインターフェイスのディセーブル化などの、イベントから回復するために EEM が実行できるアクション。

ポリシー

イベントのトラブルシューティングまたはイベントからの回復を目的とした 1 つまたは複数のアクションとペアになったイベント。

EEM を使用しない場合は、個々のコンポーネントが独自のイベントの検出および処理を行います。たとえば、ポートでフラップが頻繁に発生する場合は、「errDisable ステートにする」のポリシーが ETHPM に組み込まれます。

Embedded Event Manager ポリシー

EEM ポリシーは、イベント文および 1 つまたは複数のアクション文からなります。イベント文では、探すイベントとともに、イベントのフィルタリング特性を定義します。アクション文では、イベントの発生時に EEM が実行するアクションを定義します。

たとえば、いつカードがデバイスから取り外されたかを識別し、カードの取り外しに関する詳細を記録する EEM ポリシーを設定できます。カードの取り外しのインスタンスすべてを探すようにシステムに指示するイベント文および詳細を記録するようにシステムに指示するアクション文を設定します。

コマンドライン インターフェイス（CLI）または VSH スクリプトを使用して EEM ポリシーを設定できます。

EEM からデバイス全体のポリシー管理ビューが得られます。EEM ポリシーが設定されると、対応するアクションがトリガーされます。トリガーされたイベントのすべてのアクション（システムまたはユーザー設定）がシステムによって追跡され、管理されます。

設定済みのシステム ポリシー

Cisco NX-OS には、設定済みのさまざまなシステム ポリシーがあります。これらのシステム ポリシーでは、デバイスに関連する多数の一般的なイベントおよびアクションが定義されています。システム ポリシー名は、2 個の下線記号（__）から始まります。

一部のシステム ポリシーは上書きできます。このような場合、イベントまたはアクションに対する上書きを設定できます。設定した上書き変更がシステム ポリシーの代わりになります。

（注） 上書きポリシーにはイベント文を含める必要があります。イベント文が含まれていない上書きポリシーは、システム ポリシーで想定されるすべてのイベントを上書きします。

設定済みのシステム ポリシーを表示し、上書きできるポリシーを決定するには、**show event manager system-policy** コマンドを使用します。

ユーザー作成ポリシー

ユーザー作成ポリシーを使用すると、ネットワークの EEM ポリシーをカスタマイズできます。ユーザー ポリシーがイベントに対して作成されると、ポリシーのアクションは、EEM が同じイベントに関連するシステム ポリシー アクションをトリガーした後にのみトリガーされます。

ログ ファイル

EEM ポリシーの一致に関連するデータが格納されたログ ファイルは、/log/event_archive_1 ディレクトリにある event_archive_1 ログ ファイルで維持されます。

イベント文

対応策、通知など、一部のアクションが実行されるデバイス アクティビティは、EEM によってイベントと見なされます。イベントは通常、インターフェイスやファンの誤動作といったデバイスの障害に関連します。

イベント文は、どのイベントがポリシー実行のトリガーになるかを指定します。

ヒント　ポリシー内に複数の EEM イベントを作成し、区別してから、カスタム アクションをトリガーするためのイベントの組み合わせを定義することで、イベントの組み合わせに基づいた EEM ポリシーをトリガーするように EEM を設定できます。

EEM ではイベント フィルタを定義して、クリティカル イベントまたは指定された時間内で繰り返し発生したイベントだけが関連付けられたアクションのトリガーになるようにします。

一部のコマンドまたは内部イベントが他のコマンドを内部的にトリガーします。これらのコマンドは表示されませんが、引き続きアクションをトリガーするイベント指定と一致します。これらのコマンドがアクションをトリガーするのを防ぐことはできませんが、どのイベントがアクションを引き起こしたかを確認できます。

サポートされるイベント

EEM はイベント文で次のイベントをサポートします。

- カウンタ イベント
- ファン欠損イベント
- ファン不良イベント
- メモリしきい値イベント
- 上書きされたシステム ポリシーで使用されるイベント
- SNMP 通知イベント
- syslog イベント
- システム マネージャ イベント
- 温度イベント
- 追跡イベント

アクション文

アクション文は、イベントが発生したときに、ポリシーによってトリガーされるアクションを説明します。各ポリシーに複数のアクション文を設定できます。ポリシーにアクションを関連付けなかった場合、EEM はイベント観察を続けますが、アクションは実行されません。

トリガーされたイベントがデフォルト アクションを処理するために、デフォルト アクションを許可する EEM ポリシーを設定する必要があります。たとえば、一致文で CLI コマンドを照合する場合、EEM ポリシーに event-default アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。

（注） ユーザー ポリシーまたは上書きポリシー内のアクション文を設定する場合、アクション文が、相互に否定したり、関連付けられたシステム ポリシーに悪影響を与えるようなことがないように確認することが重要です。

サポートされるアクション

EEM がアクション文でサポートするアクションは、次のとおりです。

- CLI コマンドの実行
- カウンタのアップデート
- デバイスのリロード
- syslog メッセージの生成
- SNMP 通知の生成
- システム ポリシー用デフォルト アクションの使用

VSH スクリプト ポリシー

テキスト エディタを使用して、VSH スクリプトでポリシーを作成できます。VSH スクリプトを使用して作成されたポリシーには、他のポリシーと同様にイベント文とアクション文が含まれます。また、これらのポリシーはシステム ポリシーを拡張するか、または無効にすることができます。

VSH スクリプト ポリシーを定義したら、それをデバイスにコピーしてアクティブにします。

Embedded Event Manager のライセンス要件

この機能には、ライセンスは必要ありません。ライセンス パッケージに含まれていない機能はすべて Cisco NX-OS システム イメージにバンドルされており、追加費用は一切発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

Embedded Event Manager の前提条件

EEM を設定するには、network-admin の権限が必要です。

Embedded Event Manager の注意事項および制約事項

EEM の設定を計画するときは、次の点を考慮します。

- 設定可能な EEM ポリシーの最大数は 500 です。
- ユーザ ポリシーまたは上書きポリシー内のアクション文が、相互に否定したり、関連付けられたシステム ポリシーに悪影響を与えたりするようなことがないようにする必要があります。
- 発生したイベントでデフォルトのアクションを処理できるようにするには、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。たとえば、一致文でコマンドを照合する場合、EEM ポリシーに event-default アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。
- イベント ログの自動収集とバックアップには、次の注意事項があります。
  - デフォルトでは、スイッチのログ収集を有効にすると、サイズ、規模、コンポーネントのアクティビティに応じて、15 分から数時間のイベントログが利用できるようになります。
  - 長期間にわたる関連ログを収集できるようにするには、必要な特定のサービス/機能に対してのみイベントログの保持を有効にします。「単一サービスの拡張ログ ファイル保持の有効化」を参照してください。内部イベント ログをエクスポートすることもできます。「外部ログ ファイル ストレージ」を参照してください。
  - トラブルシューティングを行うときは、内部イベント ログのスナップショットを手動によりリアルタイムで収集することをお勧めします。「最近のログ ファイルのローカル コピーの生成」を参照してください。
- イベント文が指定されていて、アクション文が指定されていない上書きポリシーを設定した場合、アクションは開始されません。また、障害も通知されません。
- 上書きポリシーにイベント文が含まれていないと、システム ポリシーで可能性のあるイベントがすべて上書きされます。
- 通常コマンドの表現の場合：すべてのキーワードを拡張する必要があり、アスタリスク（*）記号のみが引数の置換に使用できます。
- EEM イベント相関は 1 つのポリシーに最大 4 つのイベント文をサポートします。イベント タイプは同じでも別でもかまいませんが、サポートされるイベント タイプは、cli、カウンタ、snmp、syslog、追跡だけです。
- 複数のイベント文が EEM ポリシーに存在する場合は、各イベント文に **tag** キーワードと一意な tag 引数が必要です。
- EEM イベント相関はシステムのデフォルト ポリシーを上書きしません。
- デフォルト アクション実行は、タグ付きのイベントで設定されているポリシーではサポートされません。

- イベント指定が CLI のパターンと一致する場合、SSH 形式のワイルド カード文字を使用できます。

  たとえば、すべての show コマンドを照合する場合は、**show *** コマンドを入力します。**show . *** コマンドを入力すると、機能しません。

- イベント指定が一致する syslog メッセージの正規表現の場合、適切な正規表現を使用できます。

  たとえば、syslog が生成されているポート上で ADMIN_DOWN イベントを検出するには、**.ADMIN_DOWN.** を使用します。**ADMIN_DOWN** コマンドを入力すると、機能しません。

- syslog のイベント指定では、regex は、EEM ポリシーのアクションとして生成される syslog メッセージと一致しません。

- EEM イベントが CLI の **show** コマンドと一致し、画面に表示するために（および EEM ポリシーによってブロックされないために）**show** コマンドの出力が必要な場合は、EEM ポリシーの最初のアクションに対して、**event-default** コマンドを指定する必要があります。

Embedded Event Manager のデフォルト設定

表 *28 :* デフォルトの *EEM* パラメータ

パラメータ	デフォルト
システム ポリシー	アクティブ

Embedded Event Manager の設定

環境変数の定義

環境変数の定義はオプションの手順ですが、複数のポリシーで繰り返し使用する共通の値を設定する場合に役立ちます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**event manager environment** *variable-name variable-value*	EEM 用の環境変数を作成します。

	コマンドまたはアクション	目的
	例： `switch(config) # event manager environment emailto "admin@anyplace.com"`	*variable-name* は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。 *variable-value* は大文字と小文字が区別され、引用符で囲んだ最大 39 文字の英数字を使用できます。
ステップ 3	（任意） **show event manager environment** {*variable-name* | **all**} 例： `switch(config) # show event manager environment all`	設定した環境変数に関する情報を表示します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

ユーザー ポリシーを設定します。

CLI によるユーザ ポリシーの定義

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `switch# configure terminal switch(config)#`	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	**event manager applet** *applet-name* 例： `switch(config)# event manager applet monitorShutdown switch(config-applet)#`	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。 applet-name は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。
ステップ 3	（任意） **description** *policy-description* 例：	ポリシーの説明になるストリングを設定します。

	コマンドまたはアクション	目的
	`switch(config-applet)# description "Monitors interface shutdown."`	string には最大 80 文字の英数字を使用できます。ストリングは引用符で囲みます。
ステップ 4	**event** *event-statement* 例： `switch(config-applet)# event cli match "shutdown"`	ポリシーのイベント文を設定します。
ステップ 5	（任意） **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] { **happens** *occurs* **in** *seconds*} 例： `switch(config-applet)# tag one or two happens 1 in 10000`	ポリシー内の複数のイベントを相互に関連付けます。 *occurs* 引数の範囲は 1 ～ 4294967295 です。 *seconds* 引数の範囲は 0 ～ 4294967295 秒です。
ステップ 6	**action** *number*[*.number2*] *action-statement* 例： `switch(config-applet)# action 1.0 cli show interface e 3/1`	ポリシーのアクション文を設定します。アクション文が複数ある場合、このステップを繰り返します。
ステップ 7	（任意） **show event manager policy-state** *name* [**module** *module-id*] 例： `switch(config-applet)# show event manager policy-state monitorShutdown`	設定したポリシーの状態に関する情報を表示します。
ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

イベント文およびアクション文を設定します。

イベント文の設定

イベント文を設定するには、EEM コンフィギュレーション モード（config-applet）で次のいずれかのコマンドを使用します。

（注） 多くの機能が展開されている場合、ベースラインのメモリでは、マイナー、重大、およびクリティカルのしきい値を定義する必要があります。デフォルトのしきい値は DRAM サイズに応じて起動時に計算されるため、その値はプラットフォームで使用されている DRAM サイズによって異なります。しきい値は、system memory-thresholds minor percentage severe percentage critical percentage コマンドを使用して設定できます。メモリの少ないプラットフォーム、たとえば 4GB DRAM を搭載したデバイスでは、誤ったアラームが発生しないようにメモリのしきい値を高い値に設定します。

始める前に

ユーザー ポリシーを定義します。

手順

	コマンドまたはアクション	目的
ステップ 1	**event cli** [**tag** *tag*] **match** *expression* [**count** *repeats* | **time** *seconds* 例： `switch(config-applet) # event cli match "shutdown"`	正規表現と一致するコマンドが入力された場合に、イベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 *repeats* の範囲は 1 ～ 65000 です。 *time* の範囲は 0 ～ 4294967295 です。0 は無制限を示します。
ステップ 2	**event counter** [**tag** *tag*] **name** *counter* **entry-val** *entry* **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} { **exit-val** *exit* **exit-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} 例： `switch(config-applet) # event counter name mycounter entry-val 20 gt`	カウンタが、開始演算子に基づいて開始のしきい値を超えた場合にイベントを発生させます。イベントはただちにリセットされます。任意で、カウンタが終了のしきい値を超えたあとでリセットされるように、イベントを設定できます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 *counter* name は大文字と小文字を区別し、最大 28 の英数字を使用できます。

	コマンドまたはアクション	目的
		entry および *exit* の値の範囲は 0 ～ 2147483647 です。
ステップ 3	**event fanabsent** [**fan** *number*] **time** *seconds* 例： `switch(config-applet) # event fanabsent time 300`	秒数で設定された時間を超えて、ファンがデバイスから取り外されている場合に、イベントを発生させます。 *number* の範囲はモジュールに依存します。 *seconds* の範囲は 10 ～ 64000 です。
ステップ 4	**event fanbad** [**fan** *number*] **time** *seconds* 例： `switch(config-applet) # event fanbad time 3000`	秒数で設定された時間を超えて、ファンが故障状態の場合に、イベントを発生させます。 *number* の範囲はモジュールに依存します。 *seconds* の範囲は 10 ～ 64000 です。
ステップ 5	**event memory** {**critical** | **minor** | **severe**} 例： `switch(config-applet) # event memory critical`	メモリのしきい値を超えた場合にイベントを発生させます。
ステップ 6	**event policy-default count** *repeats* [**time** *seconds*] 例： `switch(config-applet) # event policy-default count 3`	システムポリシーで設定されているイベントを使用します。このオプションは、ポリシーを上書きする場合に使用します。 *repeats* の範囲は 1 ～ 65000 です。 *seconds* の範囲は 0 ～ 4294967295 秒です。0 は無制限を示します。
ステップ 7	**event snmp** [**tag** *tag*] **oid** *oid* **get-type** {**exact** | **next**} **entry-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **entry-val** *entry* [**exit-comb** {**and** | **or**}]**exit-op** {**eq** | **ge** | **gt** | **le** | **lt** | **ne**} **exit-val** *exit* **exit-time** *time* **polling-interval** *interval* 例： `switch(config-applet) # event snmp oid 1.3.6.1.2.1.31.1.1.1.6 get-type next entry-op lt 300 entry-val 0 exit-op eq 400 exit-time 30 polling-interval 300`	SNMP OID が、開始演算子に基づいて開始のしきい値を超えた場合にイベントを発生させます。イベントはただちにリセットされます。または任意で、カウンタが終了のしきい値を超えたあとでリセットされるように、イベントを設定できます。OID はドット付き10進表記です。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。

	コマンドまたはアクション	目的
		entry および *exit* の値の範囲は 0 ～ 18446744073709551615 です。 *time* の範囲は 0 ～ 2147483647 秒です。 *interval* の範囲は 0 ～ 2147483647 秒です。
ステップ 8	**event sysmgr memory** [**module** *module-num*] **major** *major-percent* **minor** *minor-percent* **clear** *clear-percent* 例： `switch(config-applet) # event sysmgr memory minor 80`	指定したシステムマネージャのメモリのしきい値を超えた場合にイベントを発生させます。 *percent* の範囲は 1 ～ 99 です。
ステップ 9	**event temperature** [**module** *slot*] [**sensor** *number*] **threshold** {**any** | **down** | **up**} 例： `switch(config-applet) # event temperature module 2 threshold any`	温度センサーが設定されたしきい値を超えた場合に、イベントを発生させます。 *sensor* の範囲は 1 ～ 18 です。
ステップ 10	**event track** [**tag** *tag*] *object-number* **state** {**any** | **down** | **up** 例： `switch(config-applet) # event track 1 state down`	トラッキング対象オブジェクトが設定された状態になった場合に、イベントを発生させます。 **tag** *tag* キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 指定できる *object-number* の範囲は 1 ～ 500 です。

次のタスク

アクション文を設定します。

すでにアクション文を設定した場合、または設定しないことを選択した場合は、次のオプション作業のいずれかを実行します。

- VSH スクリプトを使用してポリシーを定義します。その後、VSH スクリプト ポリシーを登録し、アクティブにします。
- メモリのしきい値を設定します。
- EEM パブリッシャとして syslog を設定します。
- EEM 設定を確認します。

アクション文の設定

EEM のコンフィギュレーション モード（config-applet）で次のいずれかのコマンドを使用して、アクションを設定できます。

（注） 発生したイベントでデフォルトのアクションを処理できるようにする場合は、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。

たとえば、一致文でコマンドを照合する場合、EEM ポリシーに event-default アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。**terminal event-manager bypass** コマンドを使用すると、一致するすべての EEM ポリシーでコマンドを実行できます。

始める前に

ユーザー ポリシーを定義します。

手順

	コマンドまたはアクション	目的
ステップ 1	**action** *number*[*.number2*] **cli** *command1*[*command2.*] [**local**] 例： `switch(config-applet) # action 1.0 cli "show interface e 3/1"`	設定済みコマンドを実行します。任意で、イベントが発生したモジュール上でコマンドを実行できます。 アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。
ステップ 2	**action** *number*[*.number2*] **counter name** *counter* **value** *val* **op** {**dec** | **inc** | **nop** | **set**} 例： `switch(config-applet) # action 2.0 counter name mycounter value 20 op inc`	設定された値および操作でカウンタを変更します。 アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。 *counter* は大文字と小文字を区別し、最大 28 文字の英数字を使用できます。 *val* には 0 ～ 2147483647 の整数または置換パラメータを指定できます。

	コマンドまたはアクション	目的
ステップ 3	**action** *number*[.*number2*] **event-default** 例： `switch(config-applet) # action 1.0 event-default`	関連付けられたイベントのデフォルトアクションを実行します。 アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。
ステップ 4	**action** *number*[.*number2*] **policy-default** 例： `switch(config-applet) # action 1.0 policy-default`	上書きしているポリシーのデフォルトアクションを実行します。 アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。
ステップ 5	**action** *number*[.*number2*] **reload** [**module** *slot* [**-** *slot*]] 例： `switch(config-applet) # action 1.0 reload module 3-5`	システム全体に 1 つ以上のモジュールをリロードします。 アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。
ステップ 6	**action** *number*[.*number2*] **snmp-trap** [**intdata1** *integer-data1*] [**intdata2** *integer-data2*] [**strdata** *string-data*] 例： `switch(config-applet) # action 1.0 snmp-trap strdata "temperature problem"`	設定されたデータを使用して SNMP トラップを送信します。アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。 *data* 要素には 80 桁までの任意の数を指定できます。 *string* には最大 80 文字の英数字を使用できます。
ステップ 7	**action** *number*[.*number2*] **syslog** [**priority** *prio-val*] **msg** *error-message* 例：	設定されたプライオリティで、カスタマイズした syslog メッセージを送信します。

	コマンドまたはアクション	目的
	`switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"`	アクション ラベルのフォーマットは number1.number2 です。 *number* には 1 ～ 16 桁の任意の番号を指定できます。 *number2* の範囲は 0 ～ 9 です。 *error-message* には最大 80 文字の英数字を引用符で囲んで使用できます。

次のタスク

イベント文を設定します。

すでにイベント文を設定した場合、または設定しないことを選択した場合は、次のオプション作業のいずれかを実行します。

- VSH スクリプトを使用してポリシーを定義します。その後、VSH スクリプト ポリシーを登録し、アクティブにします。
- メモリのしきい値を設定します。
- EEM パブリッシャとして syslog を設定します。
- EEM 設定を確認します。

VSH スクリプトによるポリシーの定義

これはオプションのタスクです。VSH スクリプトを使用して EEM ポリシーを記述する場合は、次の手順を実行します。

手順

ステップ 1 テキスト エディタで、ポリシーを定義するコマンド リストを指定します。
ステップ 2 テキスト ファイルに名前をつけて保存します。
ステップ 3 次のシステム ディレクトリにファイルをコピーします。bootflash://eem/user_script_policies

次のタスク

VSH スクリプト ポリシーを登録してアクティブにします。

VSH スクリプト ポリシーの登録およびアクティブ化

これはオプションのタスクです。VSH スクリプトを使用して EEM ポリシーを記述する場合は、次の手順を実行します。

始める前に

ポリシーを VSH スクリプトを使用して定義し、システム ディレクトリにファイルをコピーします。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**event manager policy** *policy-script* 例： `switch(config)# event manager policy moduleScript`	EEM スクリプト ポリシーを登録してアクティブにします。 *policy-script* は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。
ステップ 3	（任意） **event manager policy internal** *name* 例： `switch(config)# event manager policy internal moduleScript`	EEM スクリプト ポリシーを登録してアクティブにします。 *policy-script* は大文字と小文字を区別し、最大 29 の英数字を使用できます。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

システム要件に応じて、次のいずれかを実行します。

- メモリのしきい値を設定します。
- EEM パブリッシャとして syslog を設定します。
- EEM 設定を確認します。

システム ポリシーの上書き

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	（任意） **show event manager policy-state** *system-policy*
	例：
	`switch(config-applet)# show event manager policy-state __ethpm_link_flap`
	`Policy __ethpm_link_flap`
	`Cfg count : 5`
	`Cfg time interval : 10.000000 (seconds)`
	`Hash default, Count 0`
		上書きするシステム ポリシーの情報をしきい値を含めて表示します。**show event manager system-policy** コマンドを使用して、システム ポリシーの名前を探します。

ステップ 3	**event manager applet** *applet-name* **override** *system-policy*
	例：
	`switch(config-applet)# event manager applet`
	`ethport override __ethpm_link_flap`
	`switch(config-applet)#`
		システムポリシーを上書きし、アプレット コンフィギュレーション モードを開始します。
		applet-name は大文字と小文字を区別し、最大 80 文字の英数字を使用できます。
		system-policy は、システム ポリシーの 1 つにする必要があります。

ステップ 4	**description** *policy-description* 例： `switch(config-applet)# description "Overrides link flap policy"`	ポリシーの説明になるストリングを設定します。 *policy-description* は大文字と小文字を区別し、最大 80 文字の英数字を使用できますが、引用符で囲む必要があります。
ステップ 5	**event** *event-statement* 例： `switch(config-applet)# event policy-default count 2 time 1000`	ポリシーのイベント文を設定します。

ステップ 6	**section** *number action-statement*
	例：
	`switch(config-applet)# action 1.0 syslog`
	`priority warnings msg "Link is flapping."`
		ポリシーのアクション文を設定します。複数のアクション文では、この手順を繰り返します。

	コマンドまたはアクション	目的
ステップ 7	（任意） **show event manager policy-state** *name* 例： `switch(config-applet)# show event manager policy-state ethport`	設定したポリシーに関する情報を表示します。
ステップ 8	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

EEM パブリッシャとしての syslog の設定

EEM パブリッシャとして syslog を設定すると、スイッチから syslog メッセージをモニターできます。

（注） syslog メッセージをモニターする検索文字列の最大数は 10 です。

始める前に

- EEM が syslog による登録で利用できることを確認します。
- syslog デーモンが設定され、実行されていることを確認します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**event manager applet** *applet-name*
	例：
	`switch(config)# event manager applet abc`
	`switch (config-appliet)#`
		EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**event syslog** [**tag** *tag*] { **occurs** *number* | **period** *seconds* | **pattern** *msg-text* | **priority** *priority*} 例： `switch(config-applet)# event syslog occurs 10`	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

EEM 設定を確認します。

Embedded Event Manager の設定確認

次のいずれかのコマンドを使用して、設定を確認します。

コマンド	目的
show event manager environment [*variable-name* | **all**]	イベント マネージャの環境変数に関する情報を表示します。
show event manager event-types [*event* | **all** | **module** *slot*]	イベント マネージャのイベント タイプに関する情報を表示します。
show event manager history events [**detail**] [**maximum** *num-events*] [**severity** {**catastrophic** | **minor** | **moderate** | **severe**}]	すべてのポリシーについて、イベント履歴を表示します。
show event manager policy-state *policy-name*	しきい値を含め、ポリシーの状態に関する情報を表示します。
show event manager script system [*policy-name* | **all**]	スクリプト ポリシーに関する情報を表示します。
show event manager system-policy [**all**]	定義済みシステム ポリシーに関する情報を表示します。
show running-config eem	EEM の実行コンフィギュレーションに関する情報を表示します。

コマンド	目的
show startup-config eem	EEM のスタートアップ コンフィギュレーションに関する情報を表示します。

Embedded Event Manager の設定例

次に、モジュール3の中断のないアップグレードの障害のしきい値だけを変更することによって、__lcm_module_failure システム ポリシーを上書きする例を示します。また、syslog メッセージも送信します。その他のすべての場合、システム ポリシー __lcm_module_failure の設定値が適用されます。

```
event manager applet example2 override __lcm_module_failure
event module-failure type hitless-upgrade-failure module 3 count 2
   action 1 syslog priority errors msg module 3 "upgrade is not a hitless upgrade!"
   action 2 policy-default
```

次に、__ethpm_link_flap システム ポリシーを上書きし、インターフェイスをシャットダウンする例を示します。

```
event manager applet ethport override __ethpm_link_flap
   event policy-default count 2 time 1000
   action 1 cli conf t
   action 2 cli int et1/1
   action 3 cli no shut
```

次に、ユーザーがデバイスでコンフィギュレーション モードを開始すると、コマンドを実行できるが、SNMP 通知をトリガーする EEM ポリシーを作成する例を示します。

```
event manager applet TEST
   event cli match "conf t"
   action 1.0 snmp-trap strdata "Configuration change"
   action 2.0 event-default
```

（注） EEM ポリシーに **event-default** アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。

次に、EEM ポリシーの複数イベントを関連付け、イベント トリガーの組み合わせに基づいてポリシーを実行する例を示します。この例では、EEM ポリシーは、指定された syslog パターンのいずれかが 120 秒以内に発生したときにトリガーされます。

```
event manager applet eem-correlate
   event syslog tag one pattern "copy bootflash:.* running-config.*"
   event syslog tag two pattern "copy run start"
   event syslog tag three pattern "hello"
   tag one or two or three happens 1 in 120
   action 1.0 reload module 1
```

イベント ログの自動収集とバックアップ

自動的に収集されたイベントログは、スイッチのメモリにローカルに保存されます。イベントログ ファイル ストレージは、一定期間ファイルを保存する一時バッファです。時間が経過すると、バッファのロールオーバーによって次のファイルのためのスペースが確保されます。ロールオーバーでは、先入れ先出し方式が使用されます。

Cisco NX-OS リリース 9.3(3) 以降、EEM は以下の収集およびバックアップ方法を使用します。

- 拡張ログ ファイルの保持
- トリガーベースのイベント ログの自動収集

拡張ログ ファイルの保持

Cisco NX-OS リリース 9.3 (3) 以降、すべての Cisco Nexus プラットフォーム スイッチは、少なくとも 8 GB のシステムメモリを備え、イベント ロギング ファイルの拡張保持をサポートします。ログ ファイルをスイッチにローカルに保存するか、外部コンテナを介してリモートに保存すると、ロールオーバーによるイベント ログの損失を削減できます。

すべてのサービスの拡張ログ ファイル保持のイネーブル化

拡張ログ ファイル保持は、スイッチで実行されているすべてのサービスに対してデフォルトで有効になっています。スイッチでログ ファイル保持機能がイネーブルになっていない場合（**no bloggerd log-dump** が設定されている場合）、次の手順を使用してイネーブルにします。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**bloggerd log-dump all**
	例：
	`switch(config)# bloggerd log-dump all`
	`switch(config)#`
		すべてのサービスのログ ファイル保持機能をイネーブルにします。

例

```
switch# configure terminal
switch(config)# bloggerd log-dump all
Sending Enable Request to Bloggerd
```

```
Bloggerd Log Dump Successfully enabled
switch(config)#
```

すべてのサービスの拡張ログ ファイル保持の無効化

拡張ログファイル保持は、スイッチ上のすべてのサービスに対してデフォルトで無効になっています。スイッチのログファイル保持機能がすべてのサービスに対して有効になっている場合は、次の手順を実行します。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 bloggerd log-dump all**
	例：
	`switch(config)# no bloggerd log-dump all`
	`switch(config)#`
		スイッチ上のすべてのサービスのログ ファイル保持機能を無効にします。

例

```
switch# configure terminal
switch(config)# no bloggerd log-dump all
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)#
```

単一サービスの拡張ログファイル保持の有効化

拡張ログファイル保持は、スイッチで実行されているすべてのサービスに対してデフォルトで有効になっています。スイッチで（**no bloggerd log-dump**が設定されていて）ログ ファイル保持機能が有効になっていない場合、次の手順を使用して単一のサービスに対して有効にします。

手順

	コマンドまたはアクション	目的
ステップ 1	**show system internal sysmgr service name** *service-type* 例：	サービス SA P番号を含む ACL Manager に関する情報を表示します。

	コマンドまたはアクション	目的
	`switch# show system internal sysmgr service name aclmgr`	

ステップ 2	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 3	**bloggerd log-dump sap** *number* 例： `switch(config)# bloggerd log-dump sap 351`	ACL Manager サービスのログ ファイル保持機能をイネーブルにします。
ステップ 4	**show system internal bloggerd info log-dump-info** 例： `switch(config)# show system internal bloggerd info log-dump-info`	スイッチ上のログ ファイル保持機能に関する情報を表示します。

例

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
        UUID = 0x182, PID = 653, SAP = 351
        State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
        Restart count: 1
        Time of last restart: Mon Nov  4 11:10:39 2019.
        The service never crashed since the last reboot.
        Tag = N/A
        Plugin ID: 0
switch(config)# configure terminal
switch(config)# bloggerd log-dump sap 351
Sending Enable Request to Bloggerd
Bloggerd Log Dump Successfully enabled
switch(config)# show system internal bloggerd info log-dump-info
------------------------------------------------------------
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
------------------------------------------------------------
Module     | VDC        | SAP                       | Enabled?
------------------------------------------------------------
1          | 1          | 351 (MTS_SAP_ACLMGR     ) | Enabled
------------------------------------------------------------
------------------------------------------------------------
Log Dump Throttle Switch-Wide Config:
-------------------------------------
Log Dump Throttle                                    : ENABLED
Minimum buffer rollover count (before throttling)    : 5
Maximum allowed rollover count per minute            : 1
------------------------------------------------------------

switch(config)#
```

拡張ログ ファイルの表示

スイッチに現在保存されているイベント ログ ファイルを表示するには、次の作業を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**dir debug:log-dump/** 例： `switch# dir debug:log-dump/`	スイッチに現在保存されているイベント ログ ファイルを表示します。

例

```
switch# dir debug:log-dump/

3676160 Dec 05 02:43:01 2019 20191205023755_evtlog_archive.tar
3553280 Dec 05 06:05:06 2019 20191205060005_evtlog_archive.tar

Usage for debug://sup-local
913408 bytes used
4329472 bytes free
5242880 bytes total
```

単一サービスに対する拡張ログファイル保持の無効化

拡張ログファイル保持は、スイッチ上のすべてのサービスに対してデフォルトで有効になっています。スイッチで単一またはすべてのサービス（Cisco NX-OSリリース9.3(5) ではデフォルト）に対してログファイル保持機能が有効になっている場合に、特定のサービスを無効にするには、次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**show system internal sysmgr service name** *service-type* 例： `switch# show system internal sysmgr service name aclmgr`	サービス SA P番号を含む ACL Manager に関する情報を表示します。
ステップ 2	**configure terminal** 例： `switch# configure terminal switch(config)#`	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。

	コマンドまたはアクション	目的
ステップ 3	**no bloggerd log-dump sap** *number* 例： `switch(config)# no bloggerd log-dump sap 351`	ACL Manager サービスのログ ファイル保持機能を無効にします。
ステップ 4	**show system internal bloggerd info log-dump-info** 例： `switch(config)# show system internal bloggerd info log-dump-info`	スイッチ上のログ ファイル保持機能に関する情報を表示します。

例

次に、「aclmgr」という名前のサービスの拡張ログ ファイル保持を無効にする例を示します。

```
switch# show system internal sysmgr service name aclmgr
Service "aclmgr" ("aclmgr", 80):
        UUID = 0x182, PID = 653, SAP = 351
        State: SRV_STATE_HANDSHAKED (entered at time Mon Nov  4 11:10:41 2019).
        Restart count: 1
        Time of last restart: Mon Nov  4 11:10:39 2019.
        The service never crashed since the last reboot.
        Tag = N/A
        Plugin ID: 0
switch(config)# configure terminal
switch(config)# no bloggerd log-dump sap 351
Sending Disable Request to Bloggerd
Bloggerd Log Dump Successfully disabled
switch(config)# show system internal bloggerd info log-dump-info
------------------------------------------------------------
Log Dump config is READY
Log Dump is DISABLED for ALL application services in the switch
Exceptions to the above rule (if any) are as follows:
------------------------------------------------------------
Module     | VDC        | SAP                        | Enabled?
------------------------------------------------------------
1          | 1          | 351 (MTS_SAP_ACLMGR     ) | Disabled
------------------------------------------------------------
------------------------------------------------------------
Log Dump Throttle Switch-Wide Config:
-------------------------------------
Log Dump Throttle                                    : ENABLED
Minimum buffer rollover count (before throttling)    : 5
Maximum allowed rollover count per minute            : 1
------------------------------------------------------------

switch(config)#
```

トリガーベースのイベント ログの自動収集

トリガーベースのログ収集機能：

- 問題発生時に関連データを自動的に収集します。
- コントロール プレーンへの影響なし
- カスタマイズ可能な設定ですか：
  - シスコが入力するデフォルト
  - 収集対象は、ネットワーク管理者または Cisco TACによって、選択的に上書きされます。
  - イメージのアップグレード時は新しいトリガーを自動的に更新します。
- ログをスイッチにローカルに保存するか、外部サーバにリモートで保存します。
- 重大度 0、1、および 2 の syslog をサポートします：
- アドホック イベントのカスタム syslog（syslog と接続する自動収集コマンド）

トリガーベースのログ ファイルの自動収集の有効化

ログ ファイルのトリガーベースの自動作成を有効にするには、__syslog_trigger_default システム ポリシーのオーバーライド ポリシーをカスタム YAML ファイルで作成し、情報を収集する特定のログを定義する必要があります。

ログ ファイルの自動収集を有効にするカスタム YAML ファイルの作成の詳細については、自動収集 YAML ファイルの設定 （240 ページ） を参照してください。

自動収集 YAML ファイル

EEM 機能の **action** コマンドで指定される自動収集 YAML ファイルは、さまざまなシステムまたは機能コンポーネントのアクションを定義します。このファイルは、スイッチ ディレクトリ: /bootflash/scripts にあります。デフォルトの YAML ファイルに加えて、コンポーネント固有の YAML ファイルを作成し、同じディレクトリに配置できます。コンポーネント固有の YAML ファイルの命名規則は **component-name.yaml** です。コンポーネント固有のファイルが同じディレクトリに存在する場合は、**action** コマンドで指定されたファイルよりも優先されます。たとえば、アクションファイル **bootflash/scripts/platform.yaml** がデフォルトのアクションファイル **/bootflash/scripts** とともに **bootflash/scripts/test.yaml**ディレクトリにある場合、**platform.yaml** ファイルで定義された命令がデフォルトの **test.yaml** ファイルに存在するプラットフォーム コンポーネントの手順よりも優先します。

コンポーネントの例としては、ARP、BGP、IS-IS などがあります。すべてのコンポーネント名に精通していない場合は、シスコ カスタマー サポートに連絡して、コンポーネント固有のアクション（およびデフォルトの **test.yaml** ファイル）の YAML ファイルを定義してください。

例：

```
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

自動収集 YAML ファイルの設定

YAMLファイルの内容によって、トリガーベースの自動収集時に収集されるデータが決まります。スイッチには YAML ファイルが 1 つだけ存在しますが、任意の数のスイッチ コンポーネントとメッセージの自動収集メタデータを含めることができます。

スイッチの次のディレクトリで YAML ファイルを見つけます。

```
/bootflash/scripts
```

次の例を使用して、トリガーベース収集のYAMLファイルを呼び出します。この例は、ユーザ定義の YAML ファイルを使用してトリガーベース収集を実行するために最低限必要な設定を示しています。

```
switch# show running-config eem
!Command: show running-config eem
!Running configuration last done at: Mon Sep 30 19:34:54 2019
!Time: Mon Sep 30 22:24:55 2019
version 9.3(3) Bios:version 07.59
event manager applet test_1 override __syslog_trigger_default
  action 1.0 collect test.yaml $_syslog_msg
```

上記の例では、「`test_1`」がアプレットの名前で、「`test.yaml`」が /bootflash/scripts ディレクトリにあるユーザ設定の YAML ファイルの名前です。

YAML ファイルの例

次に、トリガーベースのイベント ログ自動収集機能をサポートする基本的な YAML ファイルの例を示します。ファイル内のキー/値の定義を次の表に示します。

（注） YMAL ファイルに適切なインデントがあることを確認します。ベスト プラクティスとして、スイッチで使用する前に任意の「オンライン YAML 検証」を実行します。

```
bash-4.3$ cat /bootflash/scripts/test.yaml
version: 1
components:
   securityd:
        default:
            tech-sup: port
            commands: show module
   platform:
        default:
            tech-sup: port
            commands: show module
```

キー：値	説明
バージョン：1	1 に設定します。他の番号を使用すると、自動収集スクリプトに互換性がなくなります。
コンポーネント：	以下がスイッチ コンポーネントであることを指定するキーワード。
securityd：	syslog コンポーネントの名前（`securityd` は syslog のファシリティ名）。

キー：値	説明
デフォルト：	コンポーネントに属するすべてのメッセージを識別します。
tech-sup：port	securityd syslog コンポーネントのポート モジュールのテクニカル サポートを収集します。
コマンド：show module	securityd syslog コンポーネントの show module コマンド出力を収集します。
プラットフォーム：	syslog コンポーネントの名前（platform は syslog のファシリティ名）。
tech-sup：port	platform syslog コンポーネントのポート モジュールのテクニカル サポートを収集します。
コマンド：show module	platform syslog コンポーネントの show module コマンド出力を収集します。

特定のログにのみ自動収集メタデータを関連付けるには、次の例を使用します。たとえば、SECURITYD-2-FEATURE_ENABLE_DISABLE

```
securityd:
                feature_enable_disable:
                        tech-sup: security
                        commands: show module
```

キー：値	説明
securityd：	syslog コンポーネントの名前（securityd は syslog のファシリティ名）。
feature_enable_disable：	syslog メッセージのメッセージ ID。
tech-sup：security	securityd syslog コンポーネントのセキュリティモ ジュールのテクニカル サポートを収集します。
コマンド：show module	セキュリティ syslog コンポーネントの show module コマンド出力を収集します。

上記の YAML エントリの syslog 出力の例：

```
2019 Dec 4 12:41:01 n9k-c93108tc-fx %SECURITYD-2-FEATURE_ENABLE_DISABLE: User
has enabled the feature bash-shell
```

複数の値を指定するには、次の例を使用します。

```
version: 1
components:
   securityd:
        default:
            commands: show module;show version;show module
            tech-sup: port;lldp
```

（注） 複数の show コマンドとテクニカル サポート キーの値を区切るには、セミコロンを使用します（前の例を参照）。

コンポーネントあたりの自動収集の量の制限

自動収集の場合、コンポーネント イベントあたりのバンドル数の制限はデフォルトで 3 に設定されています。1 つのコンポーネントで 3 つ以上のイベントが発生すると、イベントはドロップされ、ステータス メッセージ **EVENTLOGLIMITREACHED** が表示されます。イベント ログがロールオーバーすると、コンポーネント イベントの自動収集が再開されます。

例：

```
switch# show system internal event-logs auto-collect history
DateTime              Snapshot ID  Syslog                   Status/Secs/Logsize(Bytes)
2020-Jun-27 07:20:03  1140276903   ACLMGR-0-TEST_SYSLOG     EVENTLOGLIMITREACHED
2020-Jun-27 07:15:14  1026359228   ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:15:09  384952880    ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:13:55  1679333688   ACLMGR-0-TEST_SYSLOG     PROCESSED:2:9332278
2020-Jun-27 07:13:52  1679333688   ACLMGR-0-TEST_SYSLOG     PROCESSING
2020-Jun-27 07:12:55  502545693    ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:12:25  1718497217   ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:08:25  1432687513   ACLMGR-0-TEST_SYSLOG     PROCESSED:2:10453823
2020-Jun-27 07:08:22  1432687513   ACLMGR-0-TEST_SYSLOG     PROCESSING
2020-Jun-27 07:06:16  90042807     ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:03:26  1737578642   ACLMGR-0-TEST_SYSLOG     RATELIMITED
2020-Jun-27 07:02:56  40101277     ACLMGR-0-TEST_SYSLOG     PROCESSED:3:10542045
2020-Jun-27 07:02:52  40101277     ACLMGR-0-TEST_SYSLOG     PROCESSING
```

自動収集ログ ファイル

自動収集ログ ファイルについて

YAML ファイルの設定によって、自動収集ログ ファイルの内容が決まります。収集ログ ファイルで使用されるメモリの量は設定できません。保存後のファイルが消去される頻度は設定できます。

自動収集ログ ファイルは、次のディレクトリに保存されます。

```
switch# dir bootflash:eem_snapshots
   44205843    Sep 25 11:08:04 2019
1480625546_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz
  Usage for bootflash://sup-local
 6940545024 bytes used
44829761536 bytes free
51770306560 bytes total
```

ログ ファイルへのアクセス

コマンド キーワード「debug」を使用してログを検索します。

```
switch# dir debug:///
...
       26    Oct 22 10:46:31 2019  log-dump
```

```
24    Oct 22 10:46:31 2019  log-snapshot-auto
26    Oct 22 10:46:31 2019  log-snapshot-user
```

次の表に、ログの場所と保存されるログの種類を示します。

場所	説明
log-dump	このフォルダには、ログ ロールオーバー時にイベント ログが保存されます。
log-snapshot-auto	このフォルダには、syslogイベント0、1、2の自動収集ログが含まれます。
log-snapshot-user	このフォルダには、`bloggerd log-snapshot` の実行時に収集されたログが保存されます。

ログ ロールオーバーで生成されたログ ファイルを表示するには、次の例を参考にしてください。

```
switch# dir debug:log-dump/
debug:log-dump/20191022104656_evtlog_archive.tar
debug:log-dump/20191022111241_evtlog_archive.tar
debug:log-dump/20191022111841_evtlog_archive.tar
debug:log-dump/20191022112431_evtlog_archive.tar
debug:log-dump/20191022113042_evtlog_archive.tar
debug:log-dump/20191022113603_evtlog_archive.tar
```

ログ tar ファイルの解析

tar ファイル内のログを解析するには、次の例を参考にしてください。

```
switch# show system internal event-logs parse
debug:log-dump/20191022104656_evtlog_archive.tar
--------LOGS:/tmp/BLOGGERD0.991453012199/tmp/1-191022104658-191022110741-device_test-M27-V1-I1:0-P884.gz--------
2019 Oct 22 11:07:41.597864 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Data Space
Limits(bytes): Soft: -1  Ha rd: -1
2019 Oct 22 11:07:41.597857 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Stack Space
Limits(bytes): Soft: 500000  Hard: 500000
2019 Oct 22 11:07:41.597850 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):AS: 1005952076
 -1
2019 Oct 22 11:07:41.597406 E_DEBUG Oct 22 11:07:41 2019(device_test_process_events):Sdwrap
 msg unknown
2019 Oct 22 11:07:41.597398 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Going back to
 select
2019 Oct 22 11:07:41.597395 E_DEBUG Oct 22 11:07:41 2019(nvram_test):TestNvram examine
27 blocks
2019 Oct 22 11:07:41.597371 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Parent: Thread
 created test index:4 thread_id:-707265728
2019 Oct 22 11:07:41.597333 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):Node inserted
2019 Oct 22 11:07:41.597328 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):The test index
 in diag is 4
2019 Oct 22 11:07:41.597322 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):result severity
 level
2019 Oct 22 11:07:41.597316 E_DEBUG Oct 22 11:07:41 2019(diag_test_start):callhome alert
 level
```

次の表に、特定の tar ファイルの解析に使用できる追加のキーワードを示します。

キーワード	説明
component	プロセス名で識別されるコンポーネントに属するログをデコードします。

キーワード	説明
from-datetime	yy [mm [dd [HH [MM [SS]]]]] 形式で指定した、特定の日時のログをデコードします。
instance	デコードする SDWRAP バッファ インスタンスのリスト（カンマ区切り）。
module	SUP や LC などのモジュールからのログをデコードします（モジュール ID を使用）。
to-datetime	yy [mm [dd [HH [MM [SS]]]]] 形式で指定した、特定の日時までのログをデコードします。

別の場所へログをコピーする

リモート サーバなどの別の場所にログをコピーするには、次の例を参考にしてください。

```
switch# copy debug:log-dump/20191022104656_evtlog_archive.tar
scp://<ip-adress>/nobackup/<user> vrf management  use-kstack
Enter username: user@<ip-address>'s password:
20191022104656_evtlog_archive.tar                                        100%  130KB
 130.0KB/s   00:00
Copy complete, now saving to disk (please wait)...
Copy complete.
```

自動収集ログファイルの消去

生成されるトリガー ベースの自動収集ログには、EventHistory と EventBundle の 2 種類があります。

EventHistory ログの消去ロジック

イベント履歴の場合は、/var/sysmgr/srv_logs/xport フォルダで消去が行われます。250 MB のパーティション RAM が、/var/sysmgr/srv_logs ディレクトリにマウントされます。

/var/sysmgr/srv_logs のメモリ使用率が、割り当てられた 250 MB の 65% 未満の場合、ファイルは消去されません。メモリ使用率が 65% の制限レベルに達すると、新しいログの保存を続行するのに十分なメモリが使用可能になるまで、最も古いファイルから消去されます。

EventBundle ログの消去ロジック

イベント バンドルの場合、消去ロジックは /bootflash/eem_snapshots フォルダでの状態に基づいて実行されます。自動収集されたスナップショットを保存するために、EEM 自動収集スクリプトは、ブートフラッシュ ストレージの 5% を割り当てます。ブート フラッシュ容量の 5% が使用されると、ログは消去されます。

新しい自動収集ログが利用可能になっているものの、ブートフラッシュに保存するスペースがない場合（すでに 5% の容量に達している）、システムは次のことを確認します。

1. 12 時間以上経過した既存の自動収集ファイルがある場合、システムはファイルを削除し、新しいログをコピーします。
2. 既存の自動収集ファイルが 12 時間未満の場合、新しく収集されたログは保存されずに廃棄されます。

デフォルトパージ時間である 12 時間は、次のコマンドを使用して変更できます。コマンドで指定する時間は分単位です。

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml purge-time 300 $_syslog_msg
```

event manager command: *test* は、ポリシー例の名前です。**__syslog_trigger_default** は、オーバーライドする必要のあるシステム ポリシーの名前です。この名前は、二重アンダースコア（__）で始まる必要があります。

action command: **1.0** は、アクションの実行順番を示している例となっています。**collect** は、データが YAMU ファイルを使用して収集されることを示しています。*test.yaml* は、YAML ファイルの名前の例です。**$_syslog_msg** は、コンポーネントの名前です。

（注） どの時点でも、進行中のトリガーベースの自動収集イベントは1つだけです。自動収集がすでに発生しているときに別の新しいログ イベントを保存しようとすると、新しいログ イベントは破棄されます。

デフォルトでは、トリガーベースのバンドルは 5 分（300 秒）ごとに 1 つだけ収集されます。このレート制限は、次のコマンドでも設定できます。コマンドで指定する時間は秒単位です。

```
switch(config)# event manager applet test override __syslog_trigger_default
switch(config-applet)# action 1.0 collect test.yaml rate-limit 600 $_syslog_msg
```

event manager command: *test* はポリシーの名前の例です。**__syslog_trigger_default** は、オーバーライドするシステム ポリシーの名前の例です。この名前は、二重アンダースコア（__）で始まる必要があります。

action command: **1.0** は、アクションの実行順番を示している例となっています。**collect** は、データが YAMU ファイルを使用して収集されることを示しています。*test.yaml* は、YAML ファイルの名前の例です。**$_syslog_msg** は、コンポーネントの名前です。

自動収集の統計情報と履歴

トリガーベースの収集統計情報の例を次に示します。

```
switch# show system internal event-logs auto-collect statistics
---------------------EEM Auto Collection Statistics--------------------
Syslog Parse Successful :88 Syslog Parse Failure :0
Syslog Ratelimited :0 Rate Limit Check Failed :0
Syslog Dropped(Last Action In Prog) :53 Storage Limit Reached :0
User Yaml Action File Unavailable :0 User Yaml Parse Successful :35
User Yaml Parse Error :0 Sys Yaml Action File Unavailable :11
Sys Yaml Parse Successful :3 Sys Yaml Parse Error :0
Yaml Action Not Defined :0 Syslog Processing Initiated :24
Log Collection Failed :0 Tar Creation Error :0
Signal Interrupt :0 Script Exception :0
Syslog Processed Successfully :24 Logfiles Purged :0
```

次の例は、CLI コマンドを使用して取得されたトリガーベースの収集履歴（処理された syslog 数、処理時間、収集されたデータのサイズ）を示しています。

```
switch# show system internal event-logs auto-collect history
DateTime Snapshot ID Syslog Status/Secs/Logsize(Bytes)
2019-Dec-04 05:30:32 1310232084 VPC-0-TEST_SYSLOG PROCESSED:9:22312929
```

```
2019-Dec-04 05:30:22 1310232084 VPC-0-TEST_SYSLOG PROCESSING
2019-Dec-04 04:30:13 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSED:173:33194665
2019-Dec-04 04:28:47 897805674 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:28:47 947981421 SYSLOG-1-SYSTEM_MSG DROPPED-LASTACTIONINPROG
2019-Dec-04 04:27:19 1618762270 ACLMGR-0-TEST_SYSLOG PROCESSING
2019-Dec-04 02:17:16 1957148102 CARDCLIENT-2-FPGA_BOOT_GOLDEN NOYAMLFILEFOUND
```

トリガーベースのログ収集の確認

次の例のように**show event manager system-policy | i trigger** コマンドを入力して、トリガーベースのログ収集機能が有効になっていることを確認します。

```
switch# show event manager system-policy | i trigger n 2
          Name : __syslog_trigger_default
    Description : Default policy for trigger based logging
    Overridable : Yes
     Event type : 0x2101
```

トリガーベースのログ ファイル生成の確認

トリガーベースの自動収集機能によってイベント ログ ファイルが生成されたかどうかを確認できます。次の例のいずれかのコマンドを入力します。

```
switch# dir bootflash:eem_snapshots
9162547 Nov 12 22:33:15 2019
1006309316_SECURITYD_2_FEATURE_ENABLE_DISABLE_eem_snapshot.tar.gz

Usage for bootflash://sup-local
8911929344 bytes used
3555950592 bytes free
12467879936 bytes total
```

```
switch# dir debug:log-snapshot-auto/
63435992 Dec 03 06:28:52 2019
20191203062841_1394408030_PLATFORM_2_MOD_PWRDN_eem_snapshot.tar.gz

Usage for debug://sup-local
544768 bytes used
4698112 bytes free
5242880 bytes total
```

ローカル ログ ファイルのストレージ

ローカル ログ ファイルのストレージ機能：

- ローカルデータ ストレージ時間の量は、導入の規模とタイプによって異なります。モジュラ スイッチと非モジュラ スイッチの両方で、ストレージ時間は 15 分から数時間のデータです。長期間にわたる関連ログを収集するには、次の手順を実行します。
  - 必要な特定のサービス/機能に対してのみイベント ログの保持を有効にします。「単一サービスの拡張ログファイル保持の有効化 （235 ページ）」を参照してください。
  - スイッチから内部イベント ログをエクスポートします。「外部ログ ファイルのストレージ （249 ページ）」を参照してください。
- 圧縮されたログは RAM に保存されます。

- 250MB のメモリは、ログ ファイル ストレージ用に予約されています。
- ログ ファイルは tar 形式で最適化されます（5 分ごとに 1 ファイルまたは 10 MB のいずれか早い方）。
- スナップ ショット収集を許可します。

最近のログ ファイルのローカル コピーの生成

拡張ログファイル保持は、スイッチで実行されているすべてのサービスに対してデフォルトで有効になっています。ローカル ストレージの場合、ログ ファイルは、フラッシュ メモリに保存されます。次の手順を使用して、最新のイベント ログ ファイルのうち最大 10 個のイベント ログ ファイルを生成します。

手順

	コマンドまたはアクション	目的
ステップ 1	**bloggerd log-snapshot** [*file-name*] [**bootflash:** *file-path* | **logflash:** *file-path* | **usb1:**] [**size** *file-size*] [**time** *minutes*] 例： `switch# bloggerd log-snapshot snapshot1`	スイッチに保存されている最新の 10 個のイベント ログのスナップショット バンドル ファイルを作成します。この操作のデフォルトのストレージは **logflash** です。 *file-name*：生成されたスナップショット ログ ファイル バンドルのファイル名。*file-name* には最大 64 文字を使用します。 （注） この変数はオプションです。設定されていない場合、システムはタイムスタンプと「_snapshot_bundle.tar」をファイル名として適用します。例： `20200605161704_snapshot_bundle.tar` **bootflash:** *file-path*：スナップショット ログ ファイル バンドルがブート フラッシュに保存されているファイル パス。次の初期パスのいずれかを選択します。 • bootflash:/// • bootflash://module-1/ • bootflash://sup-1/ • bootflash://sup-active/ • bootflash://sup-local/

	コマンドまたはアクション	目的
		logflash: *file-path*：スナップショット ログ ファイル バンドルがログ フラッシュに保存されるファイル パス。次の初期パスのいずれかを選択します。 • logflash:/// • logflash://module-1/ • logflash://sup-1/ • logflash://sup-active/ • logflash://sup-local/ **usb1:**：USB デバイス上のスナップショット ログ ファイル バンドルが保存されているファイル パス。 **size** *file-size*：メガバイト（MB）単位のサイズに基づくスナップショット ログ ファイル バンドル。範囲は 5MBー250MB です。 **time** *minutes*：最後の x 時間（分）に基づくスナップショット ログ ファイル バンドル。範囲は 1 ～ 30 分です。

例

```
switch# bloggerd log-snapshot snapshot1
Snapshot generated at logflash:evt_log_snapshot/snapshot1_snapshot_bundle.tar Please
cleanup once done.
switch#
switch# dir logflash:evt_log_snapshot
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

Usage for logflash://sup-local
759865344 bytes used
5697142784 bytes free
6457008128 bytes total
```

次の例のコマンドを使用して、同じファイルを表示します。

```
switch# dir debug:log-snapshot-user/
159098880 Dec 05 06:40:24 2019 snapshot1_snapshot_bundle.tar
159354880 Dec 05 06:40:40 2019 snapshot2_snapshot_bundle.tar

Usage for debug://sup-local
929792 bytes used
4313088 bytes free
5242880 bytes total
```

（注） ファイル名は、例の最後に示されています。個々のログ ファイルは、生成された日時によっても識別されます。

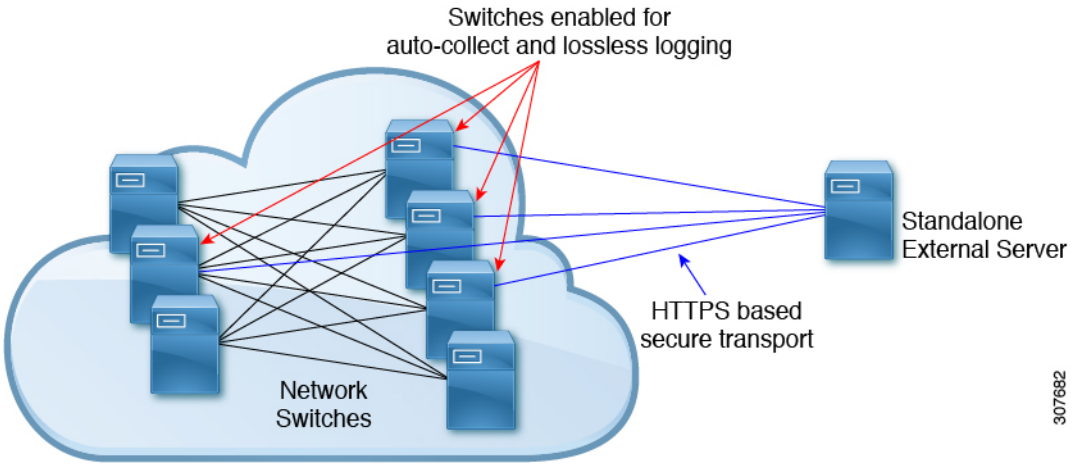
外部ログ ファイルのストレージ

外部サーバ ソリューションは、ログを安全な方法でオフスイッチに保存する機能を提供します。

（注） 外部ストレージ機能を作成するため、Cisco Technical Assistance Center（TAC）に連絡して、外部サーバ ソリューションの展開をサポートを求めてください。

次に、外部ログ ファイルの保存機能を示します。

- オンデマンドで有効
- HTTPS ベースの転送
- ストレージ要件：
  - 非モジュラ スイッチ：300 MB
  - モジュラ スイッチ：12 GB（1 日あたり、スイッチあたり）
- 通常、外部サーバには 10 台のスイッチのログが保存されます。ただし、外部サーバでサポートされるスイッチの数に厳密な制限はありません。



外部サーバ ソリューションには、次の特性があります。

- コントローラレス環境

- セキュリティ証明書の手動管理
- サポートされている 3 つの使用例：
  - 選択したスイッチからのログの継続的な収集
  - TAC のサポートによる、シスコ サーバへのログの展開とアップロード。
  - 限定的なオンプレミス処理

（注） 外部サーバでのログ ファイルの設定と収集については、Cisco TAC にお問い合わせください。

その他の参考資料

関連資料

関連項目	マニュアル タイトル
EEM コマンド	『*Cisco Nexus 3000 Series NX-OS System Management Command Reference*』

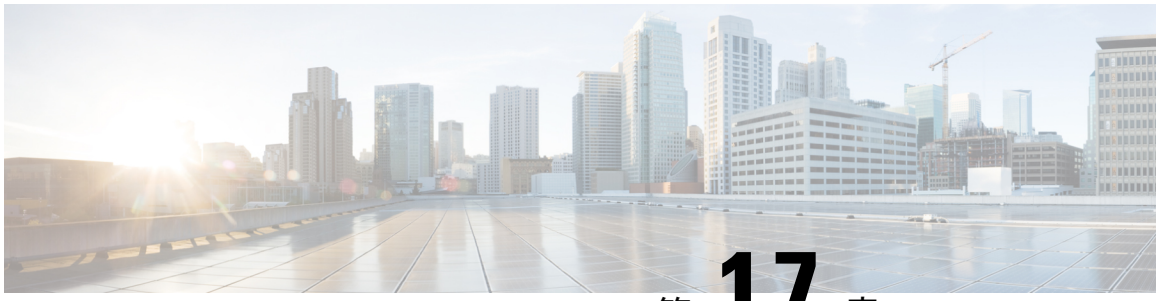
標準

この機能では、新規の標準がサポートされることも、一部変更された標準がサポートされることもありません。また、既存の標準に対するサポートが変更されることもありません。

EEM の機能の履歴

表 *29 : EEM* の機能の履歴

機能名	リリース	機能情報
組み込みイベント マネージャ（EEM）	5.0(3)U3(1)	機能が追加されました。



第 17 章

SPAN の設定

この章は、次の項で構成されています。

SPAN について

スイッチド ポート アナライザ（SPAN）機能（ポート ミラーリングまたはポート モニタリングとも呼ばれる）は、ネットワーク アナライザによる分析のためにネットワーク トラフィックを選択します。ネットワーク アナライザは、Cisco SwitchProbe またはその他のリモート モニタリング（RMON）プローブです。

SPAN ソース

SPAN 送信元とは、トラフィックをモニタリングできるインターフェイスを表します。Cisco Nexus デバイスは、SPAN 送信元として、イーサネット、ファイバ チャネル、仮想ファイバ チャネル、ポート チャネル、SAN ポート チャネル、VSAN、および VLAN をサポートします。VLAN または VSAN では、指定された VLAN または VSAN でサポートされているすべてのインターフェイスが SPAN 送信元として含まれます。イーサネット、ファイバ チャネル、および仮想ファイバ チャネルの送信元インターフェイスで、入力方向、出力方向、または両方向の SPAN トラフィックを選択できます。

- 入力送信元（Rx）：この送信元ポートを介してデバイスに入るトラフィックは、SPAN 宛先ポートにコピーされます。
- 出力送信元（Tx）：この送信元ポートを介してデバイスから出るトラフィックは、SPAN 宛先ポートにコピーされます。

VLAN アクセス コントロール リスト（VACL）を使用し、入力トラフィック（Rx）をフィルタ処理するように SPAN 送信元セッションを設定することもできます。

Cisco Nexus 34180YC プラットフォーム スイッチは、SPAN 送信元として VLAN をサポートしていません。

送信元ポートの特性

送信元ポート（モニタリング対象ポートとも呼ばれる）は、ネットワーク トラフィック分析のためにモニタリングするスイッチドインターフェイスです。スイッチは、任意の数の入力送信元ポート（スイッチで使用できる最大数のポート）と任意の数の送信元 VLAN をサポートします。

送信元ポートの特性は、次のとおりです。

- イーサネット、ポート チャネル、または VLAN ポート タイプにできます。
- ACL フィルタが設定されていない場合、方向または SPAN 宛先のいずれかが異なっていれば、複数のセッションに対して同じ送信元を設定することができます。ただし、各 SPAN RX の送信元は、ACL フィルタを使用して、1 つの SPAN セッションにのみ設定する必要があります。
- 宛先ポートには設定できません。
- モニターする方向（入力、出力、または両方）を設定できます。VLAN 送信元の場合、モニタリング方向は入力のみであり、グループ内のすべての物理ポートに適用されます。RX と TX のオプションは、VLAN の SPAN セッションでは使用できません。
- ACL を使用して入力トラフィックをフィルタし、ACL 基準に一致する情報のパケットのみがミラーリングされるようにすることができます。
- 同じまたは別の VLAN に設定できます。

SPAN 宛先

SPAN 宛先とは、送信元ポートをモニタリングするインターフェイスを表します。Cisco Nexus シリーズ デバイスは、SPAN 宛先として、イーサネット インターフェイス インターフェイスをサポートします。

宛先ポートの特性

各ローカル SPAN セッションには、送信元ポートまたは VLAN からトラフィックのコピーを受信する宛先ポート（モニタリングポートとも呼ばれる）が必要です。宛先ポートの特性は、次のとおりです。

- すべての物理ポートが可能です。送信元イーサネットおよび FCoE ポートは、宛先ポートにできません。
- 送信元ポートにはなれません。
- ポート チャネルにはできません。
- SPAN セッションがアクティブなときは、スパニングツリーに参加しません。
- 任意の SPAN セッションの送信元 VLAN に属する場合、送信元リストから除外され、モニタリングされません。
- すべてのモニタリング対象送信元ポートの送受信トラフィックのコピーを受信します。

SPAN の注意事項および制約事項

SPAN には、次の注意事項と制限事項があります。

- 同じ送信元（イーサネットまたはポートチャネル）は、複数のセッションの一部にすることができます。宛先が異なる 2 つのモニターセッションを設定することはできますが、同じ送信元 VLAN はサポートされていません。
- VLAN 送信元セッションおよびポート送信元セッションの組み合わせはサポートされていません。トラフィック ストリームが VLAN 送信元セッションに加えてポート送信元セッションとも一致する場合、2 つの宛先ポートで 2 つのコピーが必要です。ハードウェアの制限により、VLAN 送信元 SPAN と特定の宛先ポートのみが SPAN パケットを受信します。

  この制限は、次のシスコ デバイスに適用されます。

表 *30 : Cisco Nexus 3000* シリーズ スイッチ

Cisco Nexus 3048TP	Cisco Nexus 31128PQ	Cisco Nexus 3132Q

Cisco Nexus 3172PQ	Cisco Nexus 3172TQ	Cisco Nexus 3172TQ-XL

- 複数の ACL フィルタは、同じ送信元でサポートされます。
- 同じ送信元インターフェイスで 2 つの SPAN または ERSPAN セッションを 1 つのフィルタだけで設定することはできません。同じ送信元が複数の SPAN または ERSPAN セッションで使用されている場合は、すべてのセッションに異なるフィルタを設定するか、セッションにフィルタを設定しないでください。
- **show monitor session** コマンドの出力には、送信元 VLAN のすべての方向が表示されますが、フィルタ VLAN のオプションは表示されません。
- Cisco NX-OS NX-OS リリース 5.0(3)U2(2) をインストールしてからソフトウェアを以前のバージョンにダウングレードすると、SPAN 構成は失われます。

  Cisco NX-OS リリース NX-OS 5.0(3)U2(2) にアップグレードする前に設定を保存し、ダウングレード後にローカル SPAN の設定を再適用する必要があります。

  同様の ERSPAN の制約事項については、を参照してください。 ERSPAN の注意事項および制約事項 （269 ページ）
- ACL フィルタリングは、Rx SPAN に対してのみサポートされます。Tx SPAN は、送信元インターフェイスで出力されるすべてのトラフィックをミラーリングします。
- ACL フィルタリングは、TCAM（Ternary Content Addressable Memory）幅の制限により、IPv6 および MAC ACL ではサポートされていません。
- UDF-SPAN の ACL フィルタリングはソース インターフェイス rx のみをサポートします。この制限は、次のスイッチに適用されます。
  - Cisco Nexus 3048TP
  - Cisco Nexus 31108TC-V
  - Cisco Nexus 3132Q-40GX
  - Cisco Nexus 3132Q-V
  - Cisco Nexus 31108PC-V
  - Cisco Nexus 3172PQ
  - Cisco Nexus 3172TQ
  - Cisco Nexus 3164Q
  - Cisco Nexus 31128PQ-10GE
  - Cisco Nexus 3232C
  - Cisco Nexus 3264Q
- SPAN TCAM サイズは、ASIC に応じて 128 または 256 です。1 つのエントリがデフォルトでインストールされ、4 つは ERSPAN 用に予約されます。

- 同じ送信元が複数の SPAN セッションで設定されていて、各セッションに ACL フィルタが設定されている場合、送信元インターフェイスは、最初のアクティブ SPAN セッションに対してのみプログラムされます。その他のセッションの ACE にプログラムされているハードウェア エントリは、この送信元インターフェイスには含まれません。
- 許可と拒否の両方のアクセス コントロール エントリ（ACE）は、同様に処理されます。ACE と一致するパケットは、ACL の許可エントリまたは拒否エントリを含んでいるかどうかに関係なく、ミラーリングされます。

（注） 拒否 ACE により、パケットがドロップされることはありません。SPAN セッションに設定されている ACL によってのみ、パケットをミラーリングするかどうかが決まります。

- パフォーマンス向上のため、SPAN には Rx タイプの送信元トラフィックのみを使用することをお勧めします。Rx トラフィックがカットスルーであるのに対し、Tx はストア アンド フォワードであるためです。したがって、両方向（Rx および Tx）をモニターする場合、パフォーマンスは Rx のみをモニターするときほど良好になりません。両方向のトラフィックをモニターする必要がある場合は、より多くの物理ポートで Rx をモニターすると、トラフィックの両側をキャプチャすることができます。
- Cisco Nexus 34180YC プラットフォーム スイッチには次の制限が適用されます。
  - VLAN は SPAN 送信元としてサポートされていません。
  - 送信元として VLAN ポート タイプはサポートされていません。
  - VACL フィルタはサポートされていません。
  - ACL フィルタと VLAN フィルタはサポートされていません。
  - SPAN UDF ベースの ACL サポートはサポートされていません
  - 同じ送信元を複数の SPAN セッションで設定することはできません。
  - SPAN および ERSPAN では、PortChannel は宛先インターフェイスとしてサポートされていません。
  - Cisco Nexus 34180YC スイッチは、スイッチに設定されている合計で 32 セッションの SPAN および ERSPAN セッションをサポートします。32 すべてのセッションを同時にアクティブにできます。
  - **filter access-group** コマンドは、Cisco Nexus 34180YC スイッチでサポートされていません。
  - スーパーバイザに対する SPAN はサポートされていません。
- Tx SPAN のサポートは、Cisco Nexus 3132C-Z スイッチでは使用されません。
- Cisco Nexus 3500 シリーズ プラットフォーム スイッチの Cisco NX-OS リリース 10.5（3）以降、VLAN を送信元として使用して SPAN セッションを構成する場合、システムは IP

フィルタと VLAN フィルタを統合します。これにより、IP フィルタ基準に従って、指定されたユニキャストおよびマルチキャスト トラフィックのみがミラーリングされるようになります。

SPAN セッションの作成または削除

monitor session コマンドを使用してセッション番号を割り当てることによって、SPAN セッションを作成できます。セッションがすでに存在する場合、既存のセッションにさらに設定情報が追加されます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **monitor session** *session-number*	モニター コンフィギュレーション モードを開始します。既存のセッション設定に新しいセッション設定が追加されます。

Example

次に、SPAN モニター セッションを設定する例を示します。

```
switch# configure terminal
switch(config) # monitor session 2
switch(config) #
```

イーサネット宛先ポートの設定

SPAN 宛先ポートとしてイーサネット インターフェイスを設定できます。

Note SPAN 宛先ポートは、スイッチ上の物理ポートにのみ設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

	Command or Action	Purpose
ステップ 2	switch(config)# **interface ethernet** *slot*/*port*	指定されたスロットとポートでイーサネットインターフェイスのインターフェイス コンフィギュレーション モードを開始します。 **Note** 仮想イーサネット ポート上で**switchport monitor** コマンドを有効にするには、**interface vethernet** *slot*/*port* コマンドを使用できます。
ステップ 3	switch(config-if)# **switchport monitor**	指定されたイーサネット インターフェイスのモニター モードを開始します。ポートが SPAN 宛先として設定されている場合、プライオリティ フロー制御はディセーブルです。
ステップ 4	switch(config-if)# **exit**	グローバル コンフィギュレーション モードに戻ります。
ステップ 5	switch(config)# **monitor session** *session-number*	指定した SPAN セッションのモニター コンフィギュレーション モードを開始します。
ステップ 6	switch(config-monitor)# **destination interface ethernet** *slot*/*port*	イーサネット SPAN 宛先ポートを設定します。 **Note** モニター コンフィギュレーションで宛先インターフェイスとして仮想イーサネット ポートを有効にするには、**destination interface vethernet** *slot*/*port* コマンドを使用できます。

Example

次に、イーサネット SPAN 宛先ポート（HIF）を設定する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet100/1/24
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 1
switch(config-monitor)# destination interface ethernet100/1/24
switch(config-monitor)#
```

次に、仮想イーサネット（VETH）SPAN 宛先ポートを設定する例を示します。

```
switch# configure terminal
switch(config)# interface vethernet10
```

```
switch(config-if)# switchport monitor
switch(config-if)# exit
switch(config)# monitor session 2
switch(config-monitor)# destination interface vethernet10
switch(config-monitor)#
```

SPAN トラフィックのレート制限の設定

モニター セッション全体で SPAN トラフィックのレート制限を 1Gbps に設定することで、モニターされた実稼働トラフィックへの影響を回避できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface ethernet** *slot*/*port*	スロット値およびポート値による選択で指定されたイーサネット インターフェイスで、インターフェイス 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	switch(config-if)# **switchport monitor rate-limit 1G**	レート制限が 1 Gbps であることを指定します。
ステップ 4	switch(config-if)# **exit**	グローバル コンフィギュレーション モードに戻ります。

例

次に、イーサネット インターフェイス 1/2 の帯域幅を 1 Gbps に制限する例を示します。

```
switch(config)# interface ethernet 1/2
switch(config-if)# switchport monitor rate-limit 1G
switch(config-if)#
```

送信元ポートの設定

送信元ポートは、イーサネット ポートのみに設定できます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **monitor session** *session-number*	指定したモニタリング セッションのモニター 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	switch(config-monitor) # **source interface** *type slot*/*port* [**rx** | **tx** | **both**]	イーサネット SPAN の送信元ポートを追加し、パケットを複製するトラフィック方向を指定します。イーサネット、ファイバ チャネル、または仮想ファイバ チャネルのポート範囲を入力できます。複製するトラフィック方向を、入力（Rx）、出力（Tx）、または両方向（both）として指定できます。デフォルトは both です。

Example

次に、イーサネット SPAN 送信元ポートを設定する例を示します。

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# filter access-group acl1
switch(config-monitor)# source interface ethernet 1/16
switch(config-monitor)#
```

送信元ポート チャネルまたは VLAN の設定

SPANセッションに送信元チャネルを設定できます。これらのポートは、ポート チャネル、および VLAN に設定できます。モニタリング方向は入力、出力、またはその両方に設定でき、グループ内のすべての物理ポートに適用されます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **monitor session** *session-number*	指定した SPAN セッションのモニター コンフィギュレーション モードを開始します。

	Command or Action	Purpose
ステップ 3	switch(config-monitor) # **filter access-group** *access-map*	ACL リストに基づいて、送信元ポートで入力トラフィックをフィルタリングします。アクセスマップに使用されるアクセスリストと一致するパケットのみがスパニングされます。
ステップ 4	switch(config-monitor) # **source** {**interface** {**port-channel**} *channel-number* [**rx** | **tx** | **both**] | **vlan** *vlan-range*}	ポート チャネルまたは VLAN 送信元を設定します。VLAN送信元の場合、モニタリング方向は暗黙的です。

Example

次に、ポート チャネル SPAN 送信元を設定する例を示します。

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# filter access-group acl1
switch(config-monitor)# source interface port-channel 1 rx
switch(config-monitor)# source interface port-channel 3 tx
switch(config-monitor)# source interface port-channel 5 both
switch(config-monitor)#
```

次に、VLAN SPAN 送信元を設定する例を示します。

```
switch# configure terminal
switch(config)# monitor session 2
switch(config-monitor)# filter access-group acl1
switch(config-monitor)# source vlan 1
switch(config-monitor)#
```

SPAN セッションの説明の設定

参照しやすいように、SPAN セッションにわかりやすい名前を付けることができます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **monitor session** *session-number*	指定した SPAN セッションのモニター 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	switch(config-monitor) # **description** *description*	SPANセッションのわかりやすい名前を作成します。

Example

次に、SPAN セッションの説明を設定する例を示します。

```
switch# configure terminal
switch(config) # monitor session 2
switch(config-monitor) # description monitoring ports eth2/2-eth2/4
switch(config-monitor) #
```

SPAN セッションのアクティブ化

デフォルトでは、セション ステートは shut のままになります。送信元から宛先へパケットをコピーするセッションを開くことができます。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **no monitor session** {**all** | *session-number*} **shut**	指定された SPAN セッションまたはすべてのセッションを開始します。

Example

次に、SPAN セッションをアクティブにする例を示します。

```
switch# configure terminal
switch(config) # no monitor session 3 shut
```

SPAN セッションの一時停止

デフォルトでは、セッション状態は **shut** です。

Procedure

	Command or Action	Purpose
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config) # **monitor session** {**all** | *session-number*} **shut**	指定された SPAN セッションまたはすべてのセッションを一時停止します。

Example

次に、SPAN セッションを一時停止する例を示します。

```
switch# configure terminal
switch(config) # monitor session 3 shut
switch(config) #
```

SPAN 情報の表示

Procedure

	Command or Action	Purpose
ステップ 1	switch# **show monitor** [**session** {**all** | *session-number* | **range** *session-range*} [**brief**]]	SPAN 設定を表示します。

Example

次に、SPAN セッションの情報を表示する例を示します。

```
switch# show monitor
SESSION  STATE        REASON                  DESCRIPTION
-------  -----------  ----------------------  --------------------------------
2        up           The session is up
3        down         Session suspended
4        down         No hardware resource
```

次に、SPAN セッションの詳細を表示する例を示します。

```
switch# show monitor session 2
   session 2
---------------
type              : local
state             : up

source intf       :

source VLANs      :
    rx            : 100
    tx            :
    both          :
filter VLANs      : filter not specified
destination ports : Eth3/1
```

SPAN のコンフィギュレーション例

SPAN セッションのコンフィギュレーション例

SPAN セッションを設定する手順は、次のとおりです。

手順

ステップ 1 アクセス モードで宛先ポートを設定し、SPAN モニタリングをイネーブルにします。

例：

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

ステップ 2 SPAN セッションを設定します。

例：

```
switch(config)# no monitor session 3
switch(config)# monitor session 3
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# source interface port-channel 2
switch(config-monitor)# source interface sup-eth 0 both
switch(config-monitor)# source vlan 3, 6-8 rx
switch(config-monitor)# source interface ethernet 101/1/1-3
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

単一方向 SPAN セッションの設定例

単一方向 SPAN セッションを設定するには、次の手順を実行します。

手順

ステップ 1 アクセス モードで宛先ポートを設定し、SPAN モニタリングをイネーブルにします。

例：

```
switch# configure terminal
switch(config)# interface ethernet 2/5
switch(config-if)# switchport
switch(config-if)# switchport monitor
switch(config-if)# no shut
switch(config-if)# exit
switch(config)#
```

ステップ 2 SPAN セッションを設定します。

例：

```
switch(config)# no monitor session 3
switch(config)# monitor session 3 rx
switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx
switch(config-monitor)# filter vlan 3-5, 7
switch(config-monitor)# destination interface ethernet 2/5
switch(config-monitor)# no shut
switch(config-monitor)# exit
switch(config)# show monitor session 3
switch(config)# copy running-config startup-config
```

SPAN ACL の設定例

次に、SPAN ACL を設定する例を示します。

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# ip access-list match_12_pkts
switch(config-acl)# permit ip 12.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# vlan access-map span_filter 5
switch(config-access-map)# match ip address match_11_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# vlan access-map span_filter 10
switch(config-access-map)# match ip address match_12_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# monitor session 1
switch(config-erspan-src)# filter access-group span_filter
```

UDF ベース SPAN の設定例

次に、以下の一致基準を使用して、カプセル化された IP-in-IP パケットの内部 TCP フラグで照合する UDF ベース SPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定

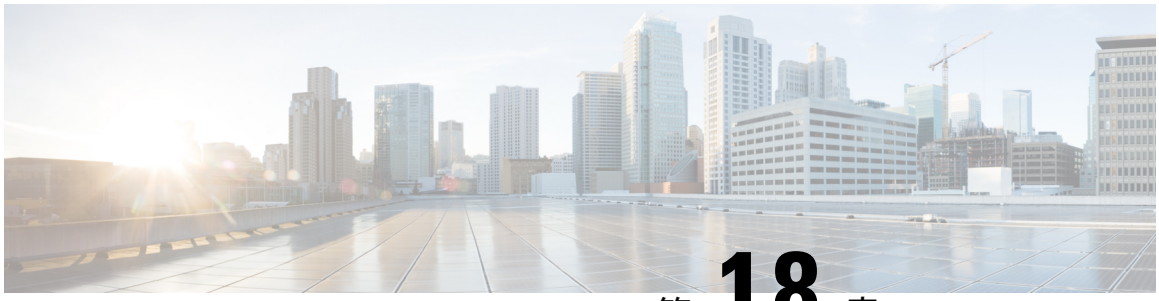
- バイト：Eth Hdr（14）＋外部 IP（20）＋内部 IP（20）＋内部 TCP（20、ただし、13 番目のバイトの TCP フラグ）
- パケットの先頭からのオフセット：14 + 20 + 20 + 13 = 67
- UDF の照合値：0x20
- UDF マスク：0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
  permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1
  source interface Ethernet 1/1
  filter access-group acl-udf
```

次に、以下の一致基準を使用して、レイヤ 4 ヘッダーの先頭から 6 バイト目のパケット署名（DEADBEEF）と通常の IP パケットを照合する UDF ベース SPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋IP（20）＋TCP（20）＋ペイロード：112233445566DEADBEEF7788
- レイヤ 4 ヘッダーの先頭からのオフセット：20 + 6 = 26
- UDF の照合値：0xDEADBEEF（2 バイトのチャンクおよび 2 つの UDF に分割）
- UDF マスク：0xFFFFFFFF

```
udf udf_pktsig_msb header outer l4 26 2
udf udf_pktsig_lsb header outer l4 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
  permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1
  source interface Ethernet 1/1
  filter access-group acl-udf-pktsig
```



第 18 章

ローカル SPAN および ERSPAN の設定

この章は、次の項で構成されています。

ERSPAN に関する情報

Cisco NX-OS システムは、発信元および宛先ポートの両方で Encapsulated Remote Switching Port Analyzer（ERSPAN）機能をサポートします。ERSPAN は、IP ネットワークでミラーリングされたトラフィックを転送します。トラフィックは、送信元ルータでカプセル化され、ネットワーク間を転送されます。パケットは宛先ルータでカプセル化解除され、宛先インターフェイスに送信されます。

ERSPAN は、ERSPAN 送信元セッション、ルーティング可能な ERSPAN Generic Routing Encapsulation（GRE）カプセル化トラフィック、および ERSPAN 宛先セッションで構成されています。異なるスイッチで ERSPAN 送信元セッションおよび宛先セッションを個別に設定することができます。ACL を使用し、入力トラフィックをフィルタ処理するように ERSPAN 送信元セッションを設定することもできます。

ERSPAN 送信元

トラフィックをモニタできるモニタ元インターフェイスのことを ERSPAN 送信元と呼びます。送信元では、監視するトラフィックを指定し、さらに入力、出力、または両方向のトラフィックをコピーするかどうかを指定します。ERSPAN 送信元には次のものが含まれます。

- イーサネット ポートおよびポート チャネル。

- VLAN：VLAN が ERSPAN 送信元として指定されている場合、VLAN でサポートされているすべてのインターフェイスが ERSPAN 送信元となります。

ERSPAN 送信元ポートには、次の特性があります。

- 送信元ポートとして設定されたポートを宛先ポートとしても設定することはできません。
- ERSPAN は送信元に関係なく、スーパーバイザによって生成されるパケットをモニターしません。
- ACL を使用して送信元ポートで入力トラフィックをフィルタし、ACL 基準に一致する情報のパケットのみがミラーリングされるようにすることができます。

マルチ ERSPAN セッション

最大 18 個の ERSPAN セッションを定義できますが、同時に作動できるのは最大 4 個の ERSPAN または SPAN セッションのみです。受信ソースと送信ソースの両方が同じセッションに設定されている場合、同時に作動できるのは 2 つの ERSPAN または SPAN セッションのみです。未使用の ERSPAN セッションはシャットダウンもできます。

（注） Cisco Nexus 34180YC プラットフォーム スイッチは、スイッチに設定されている合計で 32 セッションの SPAN および ERSPAN セッションをサポートします。32 すべてのセッションを同時にアクティブにできます。

ERSPAN セッションのシャットダウンについては、ERSPAN セッションのシャットダウンまたはアクティブ化 （285 ページ）を参照してください。

高可用性

ERSPAN 機能はステートレス およびステートフル リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバー後に、実行コンフィギュレーションが適用されます。

ERSPAN の前提条件

ERSPAN の前提条件は、次のとおりです。

•所定の ERSPAN 設定をサポートするには、まず各デバイス上でポートのイーサネット インターフェイスを設定する必要があります。詳細については、お使いのプラットフォームのインターフェイス コンフィギュレーション ガイドを参照してください。

ERSPAN の注意事項および制約事項

ERSPAN 設定時の注意事項と制限事項は次のとおりです。

- 同じ送信元は、複数のセッションの一部にすることができます。
- 複数の ACL フィルタは、同じ送信元でサポートされます。
- 2 つの ERSPAN 宛先セッションは、Cisco Nexus 3000、3100、および 3200 プラットフォーム スイッチではサポートされていません。
- Cisco Nexus 34180YC プラットフォーム スイッチには次の制限が適用されます。
  - ERSPAN では、PortChannel は宛先インターフェイスとしてサポートされていません。
  - ACL フィルタと VLAN フィルタはサポートされていません。
  - ERSPAN UDF ベースの ACL サポートはサポートされていません
  - Cisco Nexus 34180YC プラットフォーム スイッチは、スイッチに設定されている合計で 32 セッションの SPAN および ERSPAN セッションをサポートします。32 すべてのセッションを同時にアクティブにできます。
  - **filter access-group** コマンドは、Cisco Nexus 34180YC プラットフォーム スイッチでサポートされていません。
  - スーパーバイザに対する ERSPAN はサポートされていません。
  - ERSPAN での IPv6 ベースのルーティングおよび IPv6 UDF はサポートされていません。
- ERSPAN は次をサポートしています。
  - 4 ～ 6 個のトンネル
  - トンネルなしパケット
  - IP-in-IP トンネル
  - IPv4 トンネル（制限あり）
  - Cisco Nexus 3000 シリーズ スイッチでは、ERSPAN 送信元セッションと一致するパケットのスパニングに汎用 GRE ERSPAN ヘッダー形式を使用します。この形式は、Cisco ERSPAN タイプ 1/2/3 ヘッダー形式に準拠していません。Cisco ASIC ベースのプラットフォームでは、Cisco ERSPAN カプセル化形式タイプに準拠した ERSPAN パケットに対してのみ ERSPAN 終端およびカプセル化解除がサポートされます。したがって、Cisco Nexus 3000 シリーズ スイッチから CISCO ASIC ベース スイッチのローカル宛先 IP アドレスに対して発信される ERSPAN パケットは ERSPAN 終端フィルタと一致しません。宛先 IP アドレスが Cisco ASIC プラットフォーム上のローカル IP アドレスでもある場合、ERSPAN パケットはソフトウェアに送信され、ソフトウェアでドロップされます。

  - ERSPAN 宛先セッション タイプ（ただし、ERSPAN パケットのカプセル化を解除するためのサポートは使用できません。カプセル化されたパケット全体は、ERSPAN 終端ポイントの前面パネル ポートにスパンされます）。

- ERSPAN パケットは、カプセル化されたミラー パケットがレイヤ 2 MTU のチェックに失敗した場合、ドロップされます。
- 出力カプセルでは 112 バイトの制限があります。この制限を超えるパケットはドロップされます。このシナリオは、トンネルとミラーリングが混在する場合に発生することがあります。
- ERSPAN セッションは複数のローカル セッションで共有されます。最大 18 セッションが設定できます。ただし、同時に動作できるのは最大 4 セッションのみです。受信ソースと送信ソースの両方が同じセッションで設定されている場合、2 セッションのみが動作できます。
- NX-OS 5.0(3)U2(2) をインストールして ERSPAN を設定し、その後でソフトウェアを以前のバージョンにダウングレードすると、ERSPAN の設定は失われます。これは、ERSPAN が NX-OS 5.0(3)U2(2) よりも前のバージョンでサポートされていないためです。

  同様の SPAN の制約事項については、SPAN の注意事項および制約事項 （253 ページ）を参照してください。
- ERSPAN および ERSPAN（ACL フィルタリングあり）は、スーパーバイザが生成したパケットではサポートされません。
- ACL フィルタリングは、Rx ERSPAN に対してのみサポートされます。Tx ERSPAN は、送信元インターフェイスで出力されるすべてのトラフィックをミラーリングします。
- ACL フィルタリングは、TCAM 幅の制限があるため、IPv6 および MAC ACL ではサポートされません。
- 同じ送信元が複数の ERSPAN セッションで設定されていて、各セッションに ACL フィルタが設定されている場合、送信元インターフェイスは、最初のアクティブ ERSPAN セッションに対してのみプログラムされます。その他のセッションに属する ACE には、この送信元インターフェイスはプログラムされません。
- 同じ送信元を使用するように ERSPAN セッションおよびローカル SPAN セッション（filter access-group および allow-sharing オプションを使用）を設定する場合は、設定を保存してスイッチをリロードすると、ローカル SPAN セッションがダウンします。
- モニター セッションの filter access-group を使用する VLAN アクセスマップ設定では、ドロップ アクションはサポートされていません。モニター セッションでドロップ アクションのある VLAN アクセスマップに filter access-group が設定されている場合、モニターセッションはエラー状態になります。
- 許可 ACE と拒否 ACE は、どちらも同様に処理されます。ACE と一致するパケットは、ACL の許可エントリまたは拒否エントリを含んでいるかどうかに関係なく、ミラーリングされます。
- ERSPAN は、管理ポートではサポートされません。

- 宛先ポートは、一度に 1 つの ERSPAN セッションだけで設定できます。
- ポートを送信元ポートと宛先ポートの両方として設定することはできません。
- 1 つの ERSPAN セッションに、次の送信元を組み合わせて使用できます。
  - イーサネット ポートまたはポート チャネル（サブ インターフェイスを除く）。
  - ポート チャネル サブインターフェイスに割り当てることのできる VLAN またはポート チャネル。
  - コントロール プレーン CPU へのポート チャネル。

（注） ERSPAN は送信元に関係なく、スーパーバイザによって生成されるパケットをモニターしません。

- 宛先ポートはスパニングツリーインスタンスまたはレイヤ3プロトコルに参加しません。
- ERSPANセッションに、送信方向または送受信方向でモニターされている送信元ポートが含まれている場合、パケットが実際にはその送信元ポートで送信されなくても、これらのポートを受け取るパケットが ERSPAN の宛先ポートに複製される可能性があります。送信元 ポート上でのこの動作の例を、次に示します。
  - フラッディングから発生するトラフィック
  - ブロードキャストおよびマルチキャスト トラフィック
- 入力と出力の両方が設定されている VLAN ERSPAN セッションでは、パケットが同じ VLAN 上でスイッチングされる場合に、宛先ポートから 2 つのパケット（入力側から 1 つ、出力側から 1 つ）が転送されます。
- VLAN ERSPAN がモニターするのは、VLAN のレイヤ 2 ポートを出入りするトラフィックだけです。
- Cisco Nexus 3000 シリーズ スイッチが ERSPAN 宛先の場合、GRE ヘッダーは、終端ポイントからミラー パケットが送信される前には削除されません。パケットは、GRE パケットである GRE ヘッダー、および GRE ペイロードである元のパケットとともに送信されます。
- ERSPAN 送信元セッションの出力インターフェイスは、**show monitor session <session-number>** CLI コマンドの出力に表示されるようになりました。出力インターフェイスには、物理ポートまたは port-channel を指定できます。ECMP の場合、ECMP メンバー内の 1 つのインターフェイスが出力に表示されます。この特定のインターフェイスがトラフィックの出力に使用されます。
- SPAN/ERSPAN ACL 統計情報は、**show monitor filter-list** コマンドを使用して表示できます。このコマンドの出力には、SPAN TCAM の統計情報とともにすべてのエントリが表示されます。ACL 名は表示されず、エントリのみ出力に表示されます。統計情報は、**clear monitor filter-list statistics** コマンドを使用してクリアできます。出力は、**show ip access-list**

コマンドの出力と同様です。Cisco Nexus 3000 シリーズ スイッチは、ACL レベルごとの統計情報をサポートしていません。この機能強化は、ローカル SPAN および ERSPAN の両方でサポートされています。

- CPU とやりとりされるトラフィックはスパニングされます。その他のインターフェイス SPAN に似ています。この機能強化は、ローカル SPAN でのみサポートされています。ACL 送信元ではサポートされていません。Cisco Nexus 3000 シリーズ スイッチは、CPU から送信される（RCPU.dest_port != 0）ヘッダー付きのパケットはスパニングしません。
- SPAN 転送ドロップ トラフィックの場合、フォワーディング プレーンにおけるさまざまな原因でドロップされるパケットのみ SPAN されます。この機能強化は、ERSPAN 送信元セッションでのみサポートされています。SPAN ACL、送信元 VLAN、および送信元インターフェイスとともにはサポートされません。SPAN のドロップ トラフィックには、3 つの ACL エントリがインストールされます。ドロップ エントリに優先度を設定して、その他のモニター セッションの SPAN ACL エントリや VLAN SPAN エントリよりも高いまたは低い優先度にすることができます。デフォルトでは、ドロップエントリの優先度の方が高くなります。
- SPAN UDF（ユーザー定義フィールド）ベースの ACL サポート
  - パケットの最初の 128 バイトのパケット ヘッダーまたはペイロード（一定の長さ制限あり）を照合できます。
  - 照合のために、特定のオフセットと長さを指定して UDF を定義できます。
  - 1 バイトまたは 2 バイトの長さのみ照合できます。
  - 最大 8 個の UDF がサポートされます。
  - 追加の UDF 一致基準が ACL に追加されます。
  - UDF 一致基準は、SPAN ACL に対してのみ設定できます。この機能強化は、その他の ACL 機能（RACL、PACL、および VACL）ではサポートされていません。
  - ACE ごとに最大 8 個の UDF 一致基準を指定できます。
  - UDF および HTTP リダイレクト設定を、同じ ACL に共存させることはできません。
  - UDF 名は、SPAN TCAM に適合している必要があります。
  - UDF は、SPAN TCAM によって認定されている場合のみ有効です。
  - UDF 定義の設定および SPAN TCAM での UDF 名の認定では、**copy r s** コマンドを使用して、リロードする必要があります。
  - UDF の照合は、ローカル SPAN と ERSPAN 送信元セッションの両方でサポートされています。
  - UDF 名の長さは最大 16 文字です。
  - UDF のオフセットは 0（ゼロ）から始まります。オフセットが奇数で指定されている場合、ソフトウェアの 1 つの UDF 定義に対して、ハードウェアで 2 つの UDF が使用

されます。ハードウェアで使用している UDF の数が 8 を超えると、その設定は拒否されます。

- UDF の照合では、SPAN TCAM リージョンが倍幅になる必要があります。そのため、その他の TCAM リージョンのサイズを減らして、SPAN の領域を確保する必要があります。
- SPAN UDF は、タップ アグリゲーション モードではサポートされていません。

- erspan-src セッションに sup-eth 送信元インターフェイスが設定されている場合、acl-span を送信元としてそのセッションに追加することはできません（その逆も同様）。
- ERSPAN 送信元および ERSPAN 宛先セッションでは、専用のループバック インターフェイスを使用する必要があります。そのようなループバック インターフェイスには、どのようなコントロール プレーン プロトコルも使用しません。
- ERSPAN マーケットパケット UDP データ ペイロードは、Cisco Nexus 3000 シリーズ スイッチで 58 バイトです。

ERSPAN のデフォルト設定

次の表に、ERSPAN パラメータのデフォルト設定を示します。

表 *31* : デフォルトの *ERSPAN* パラメータ

パラメータ	デフォルト
ERSPAN セッション	シャット ステートで作成されます。

ERSPAN の設定

ERSPAN 送信元セッションの設定

ERSPANセッションを設定できるのはローカルデバイス上だけです。デフォルトでは、ERSPAN セッションはシャット ステートで作成されます。

送信元には、イーサネット ポート、ポート チャネル、および VLAN を指定できます。単一の ERSPAN セッションには、イーサネット ポートまたは VLAN を組み合わせた送信元を使用できます。

（注） ERSPAN は送信元に関係なく、スーパーバイザによって生成されるパケットをモニタしません。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# config t`
	`switch(config)#`
		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	**monitor erspan origin ip-address** *ip-address* **global** 例： `switch(config)# monitor erspan origin ip-address 10.0.0.1 global`	ERSPAN のグローバルな送信元 IP アドレスを設定します。
ステップ 3	**no monitor session** {*session-number* | **all**} 例： `switch(config)# no monitor session 3`	指定した ERSPAN セッションの設定を消去します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 4	**monitor session** {*session-number* | **all**} **type erspan-source**
	例：
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		ERSPAN 送信元セッションを設定します。

ステップ 5	**description** *description* 例： `switch(config-erspan-src)# description erspan_src_session_3`	セッションの説明を設定します。デフォルトでは、説明は定義されません。説明には最大 32 の英数字を使用できます。
ステップ 6	**filter access-group** *acl-name* 例： `switch(config-erspan-src)# filter access-group acl1`	ACL リストに基づいて、送信元ポートで入力トラフィックをフィルタリングします。アクセス リストに一致するパケットのみがスパニングされます。*acl-name* には、IP アクセス リストを指定できますが、アクセスマップは指定できません。
ステップ 7	**source** { **interface** *type* [**rx** [**allow-pfc**] | **tx** | **both**] | **vlan** {*number* | *range*} [**rx**] | **forward-drops rx** [**priority-low**]} 例：	送信元およびパケットをコピーするトラフィックの方向を設定します。イーサネット ポート範囲、ポート チャネル、または VLAN 範囲を入力できます。

	コマンドまたはアクション	目的

	`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	例：
	`switch(config-erspan-src)# source interface port-channel 2`
	例：
	`switch(config-erspan-src)# source interface sup-eth 0 both`
	例：
	`switch(config-monitor)# source interface ethernet 101/1/1-3`
		送信元は 1 つ設定することも、またはカンマで区切った一連のエントリとして、または番号の範囲として、複数設定することもできます。最大 128 のインターフェイスを指定できます。VLAN の範囲については、『*Cisco Nexus 3000 Series NX-OS Layer 2 Switching Configuration Guide*』を参照してください。
		コピーするトラフィックの方向には、入力、出力、または両方を指定できます。デフォルトは双方向です。
		allow-pfc オプションは、ポートで受信されるプライオリティ フロー制御（PFC）フレームのスパニングを開始します。PFC フレームは、ドロップされずに入力パイプラインで許可されます。該当ポートに ERSPAN が設定されている場合、それらの PFC フレームは適切な出力インターフェイスにスパニングされます。このオプションを指定して設定されているポートは、通常のデータ トラフィックもスパニングできます。
		インターフェイスまたは VLAN を ERSPAN 送信元として設定する代わりに、入力パイプラインで可能な最大数のフォワードパケット ドロップをスパニングするように ERSPAN を設定できます。そのように設定することで、ネットワークのパケット ドロップを分析して、分離することができます。デフォルトでは、**source forward-drops rx** コマンドは、ネットワーク転送モジュールのすべてのポートのパケット ドロップをキャプチャします。**priority-low** オプションを指定すると、この ERSPAN アクセス コントロール エントリ（ACE）の一致ドロップ条件は、標準インターフェイスや VLAN ERSPAN ACL によって設定されている

	コマンドまたはアクション	目的
		その他の ERSPAN ACE よりも優先度が低くなります。
ステップ 8	（任意） ステップ 6 を繰り返して、すべての ERSPAN 送信元を設定します。	—
ステップ 9	**destination ip** *ip-address* 例： `switch(config-erspan-src)# destination ip 10.1.1.1`	ERSPAN セッションの宛先 IP アドレスを設定します。ERSPAN 送信元セッションごとに 1 つの宛先 IP アドレスのみがサポートされます。
ステップ 10	（任意） **ip ttl** *ttl-number* 例： `switch(config-erspan-src)# ip ttl 25`	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。
ステップ 11	（任意） **ip dscp** *dscp-number* 例： `switch(config-erspan-src)# ip dscp 42`	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。範囲は 0 ～ 63 です。
ステップ 12	**no shut** 例： `switch(config-erspan-src)# no shut`	ERSPAN 送信元セッションをイネーブルにします。デフォルトでは、セッションはシャットステートで作成されます。 （注） 同時に実行できる ERSPAN 送信元セッションは 2 つだけです。
ステップ 13	（任意） **show monitor session** {**all** | *session-number* | **range** *session-range*} 例： `switch(config-erspan-src)# show monitor session 3`	ERSPAN セッション設定を表示します。
ステップ 14	（任意） **show running-config monitor** 例： `switch(config-erspan-src)# show running-config monitor`	ERSPAN の実行コンフィギュレーションを表示します。
ステップ 15	（任意） **show startup-config monitor** 例： `switch(config-erspan-src)# show startup-config monitor`	ERSPAN のスタートアップ コンフィギュレーションを表示します。

	コマンドまたはアクション	目的
ステップ 16	（任意） **copy running-config startup-config** 例： `switch(config-erspan-src)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN 送信元セッションの SPAN 転送ドロップ トラフィックの設定

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# config t`
	`switch(config)#`
		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	**monitor session** {*session-number* | **all**} **type erspan-source**
	例：
	`switch(config)# monitor session 1 type erspan-source`
	`switch(config-erspan-src)#`
		ERSPAN 送信元セッションを設定します。

ステップ 3	**vrf** *vrf-name* 例： `switch(config-erspan-src)# vrf default`	ERSPAN送信元セッションがトラフィックの転送に使用する VRF を設定します。
ステップ 4	**destination ip** *ip-address* 例： `switch(config-erspan-src)# destination ip 10.1.1.1`	ERSPAN セッションの宛先 IP アドレスを設定します。ERSPAN送信元セッションごとに 1 つの宛先 IP アドレスのみがサポートされます。
ステップ 5	**source forward-drops rx** [*priority-low*] 例： `switch(config-erspan-src)# source forward-drops rx [priority-low]`	ERSPAN 送信元セッションの SPAN 転送ドロップ トラフィックを設定します。低い優先度に設定されている場合、このSPAN ACEの一致ドロップ条件は、ACL SPAN またはVLAN ACL SPANインターフェイスによって設定されているその他の SPAN ACE よりも優先度が低くなります。priority-low キーワードを指定しない場合、これらのドロップ ACE は、標準インターフェイスや VLAN SPAN ACL よりも優先度が高くなります。優

	コマンドまたはアクション	目的
		先度は、パケットの一致ドロップ ACE およびインターフェイス/VLAN SPAN ACL が設定されている場合のみ問題になります。
ステップ 6	**no shut** 例： `switch(config-erspan-src)# no shut`	ERSPAN 送信元セッションをイネーブルにします。デフォルトでは、セッションはシャットステートで作成されます。 （注） 同時に実行できる ERSPAN 送信元セッションは 2 つだけです。
ステップ 7	（任意） **show monitor session** {**all** | *session-number* | **range** *session-range*} 例： `switch(config-erspan-src)# show monitor session 3`	ERSPANセッション設定を表示します。

例

```
switch# config t
  switch(config)# monitor session 1 type erspan-source
  switch(config-erspan-src)# vrf default
  switch(config-erspan-src)# destination ip 40.1.1.1
  switch(config-erspan-src)# source forward-drops rx
  switch(config-erspan-src)# no shut
  switch(config-erspan-src)# show monitor session 1

switch# config t
  switch(config)# monitor session 1 type erspan-source
  switch(config-erspan-src)# vrf default
  switch(config-erspan-src)# destination ip 40.1.1.1
  switch(config-erspan-src)# source forward-drops rx  priority-low
  switch(config-erspan-src)# no shut
  switch(config-erspan-src)# show monitor session 1
```

ERSPAN ACL の設定

デバイスに IPv4 ERSPAN ACL を作成して、ルールを追加できます。

始める前に

DSCP 値または GRE プロトコルを変更するには、新しい宛先モニタ セッションを割り当てる必要があります。最大 4 つの宛先モニタ セッションがサポートされます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**ip access-list** *acl-name*
	例：
	`switch(config)# ip access-list erspan-acl`
	`switch(config-acl)#`
		ERSPAN ACL を作成して、IP ACL コンフィギュレーション モードを開始します。*acl-name* 引数は 64 文字以内で指定します。

ステップ 3	[*sequence-number*] {**permit** | **deny**} *protocol source destination* [**set-erspan-dscp** *dscp-value*] [**set-erspan-gre-proto** *protocol-value*] 例： `switch(config-acl)# permit ip 192.168.2.0/24 any set-erspan-dscp 40 set-erspan-gre-proto 5555`	ERSPAN ACL 内にルールを作成します。多数のルールを作成できます。 *sequence-number* 引数には、1 ～ 4294967295 の整数を指定します。 **permit** コマンドと **deny** コマンドには、トラフィックを識別するための多くの方法が用意されています。 **set-erspan-dscp** オプションは、ERSPAN 外部 IP ヘッダーに DSCP 値を設定します。DSCP 値の範囲は 0 ～ 63 です。ERSPAN ACL に設定された DSCP 値でモニター セッションに設定されている値が上書きされます。ERSPAN ACL にこのオプションを含めない場合、0 またはモニター セッションで設定されている DSCP 値が設定されます。 **set-erspan-gre-proto** オプションは、ERSPAN GRE ヘッダーにプロトコル値を設定します。プロトコル値の範囲は 0 ～ 65535 です。ERSPAN ACL にこのオプションを含めない場合、ERSPAN カプセル化パケットの GRE ヘッダーのプロトコルとしてデフォルト値の 0x88be が設定されます。 **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定されている各アクセス コントロール エントリ（ACE）は、1 つの宛先モニター セッションを使用します。ERSPAN ACL ご

	コマンドまたはアクション	目的
		とに、これらのアクションのいずれかが設定されている最大 3 つの ACE がサポートされます。たとえば、次のいずれかを設定できます。 • **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定された最大 3 つの ACE がある ACL が設定されている 1 つの ERSPAN セッション • **set-erspan-gre-proto** または set-erspan-dscp アクションと 1 つの追加のローカルまたは ERSPAN セッションが設定された 2 つの ACE がある ACL が設定されている 1 つの ERSPAN セッション • **set-erspan-gre-proto** または **set-erspan-dscp** アクションが設定された 1 つの ACE がある ACL が設定されている最大 2 つの ERSPAN セッション
ステップ 4	（任意） **show ip access-lists** *name* 例： `switch(config-acl)# show ip access-lists erpsan-acl`	ERSPAN ACL の設定を表示します。
ステップ 5	（任意） **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**] 例： `switch(config-acl)# show monitor session 1`	ERSPAN セッション設定を表示します。
ステップ 6	（任意） **copy running-config startup-config** 例： `switch(config-acl)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ユーザー定義フィールド（UDF）ベースの ACL サポートの設定

Cisco Nexus 3000 シリーズ スイッチにユーザー定義フィールド（UDF）ベースの ACL のサポートを設定できます。次の手順を参照して、UDF に基づく ERSPAN を設定します。詳細については、「ERSPAN の注意事項および制約事項」を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

ステップ 2	switch(config)# **udf** < *udf -name*> <*packet start*> <*offset*> <*length*>
	例：
	`(config)# udf udf1 packet-start 10 2`
	`(config)# udf udf2 packet-start 50 2`
		UDF を定義します。
		（注）
		複数の UDF を定義できますが、必要な UDF のみ設定することを推奨します。UDF は、TCAM カービング時（ブートアップ時）にリージョンの修飾子セットに追加されるため、この設定は、UDF を TCAM リージョンにアタッチして、ボックスを再起動した後でのみ有効になります。

ステップ 3	switch(config)# **udf** < *udf -name*> header <*Layer3/Layer4*> <*offset*> <*length*>
	例：
	`(config)# udf udf3 header outer l4 0 1`
	`(config)# udf udf3 header outer l4 10 2`
	`(config)# udf udf3 header outer l4 50 1`
		UDF を定義します。

ステップ 4	switch(config)# **hardware profile tcam region span qualify udf** <*name1*>…… <*name8*>
	例：
	`(config)# hardware profile tcam region span qualify udf udf1 udf2 udf3 udf4 udf5`
	`[SUCCESS] Changes to UDF qualifier set will be applicable only after reboot. You need to 'copy run start' and 'reload'`
	`config)#`
		SPAN TCAM に UDF 認定を設定します。TCAM カービング時（ブートアップ時）に UDF を TCAM リージョンの修飾子セットに追加します。この設定では、SPAN リージョンにアタッチできる最大 4 つの UDF を許可できます。UDF はすべて、リージョンの単一コマンドでリストされます。リージョンの新しい設定により、既存の設定が置き換わりますが、設定を有効にするには再起動する必要があります。
		UDF 修飾子が SPAN TCAM に追加されると、TCAM リージョンはシングル幅

	コマンドまたはアクション	目的
		から倍幅に拡大します。拡大に使用できる十分な空き領域（128 以上のシングル幅エントリ）があることを確認します。十分な領域がない場合、コマンドは拒否されます。未使用リージョンの TCAM 領域を削減して領域を確保したら、コマンドを再入力します。**no hardware profile tcam region span qualify udf <name1> ..<name8>** コマンドを使用して UDF が SPAN/TCAM リージョンからデタッチされると、SPAN TCAM リージョンはシングル幅エントリであると見なされます。

ステップ 5	switch(config)# **permit** …… <*regular ACE match criteria*> **udf** <*name1*> < *val* > <*mask*> .....<*name8*> < *val* > <*mask*>
	例：
	`(config)# ip access-list test`
	`10 permit ip any any udf udf1 0x1234 0xffff udf3 0x56 0xff`
	`30 permit ip any any dscp af11 udf udf5 0x22 0x22`
	`config)#`
		UDF と一致する ACL を設定します。

ステップ 6	switch(config)# **show monitor session** <*session-number*>
	例：
	`(config)# show monitor session 1`
	`session 1`
	`---------------`
	`type : erspan-source`
	`state : up`
	`vrf-name : default`
	`destination-ip : 40.1.1.1`
	`ip-ttl : 255`
	`ip-dscp : 0`
	`acl-name : test`
	`origin-ip : 100.1.1.10 (global)`
	`source intf :`
	`rx : Eth1/20`
	`tx : Eth1/20`
	`both : Eth1/20`
	`source VLANs :`
	`filter VLANs : filter not specified`
	`rx :`
	`source fwd drops :`
	`egress-intf : Eth1/23`
	`switch#`
	`config)#`
		show monitor session <session-number> コマンドを使用して、ACL を表示します。BCM SHELL コマンドを使用して、SPAN TCAM リージョンがカービングされているかどうかを確認できます。

ERSPAN での IPv6 ユーザー定義フィールド（UDF）の設定

Cisco Nexus 3000 シリーズ スイッチでは ERSPAN で IPv6 ユーザー定義フィールド（UDF）を設定できます。次の手順を参照して、IPv6 UDF に基づく ERSPAN を設定します。詳細については、「ERSPAN の注意事項および制約事項」を参照してください。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。

ステップ 2	switch(config)# **udf** < *udf -name*> <*packet start*> <*offset*> <*length*>
	例：
	`(config)# udf udf1 packet-start 10 2`
	`(config)# udf udf2 packet-start 50 2`
		UDF を定義します。
		（注）
		複数の UDF を定義できますが、必要な UDF のみ設定することを推奨します。UDF は、TCAM カービング時（ブートアップ時）にリージョンの修飾子セットに追加されるため、この設定は、UDF を TCAM リージョンにアタッチして、ボックスを再起動した後でのみ有効になります。

ステップ 3	switch(config)# **udf** < *udf -name*> header <*Layer3/Layer4*> <*offset*> <*length*>
	例：
	`(config)# udf udf3 header outer l4 0 1`
	`(config)# udf udf3 header outer l4 10 2`
	`(config)# udf udf3 header outer l4 50 1`
		UDF を定義します。

ステップ 4	switch(config)# **hardware profile tcam region ipv6-span-l2 512**
	例：
	`(config)# hardware profile tcam region ipv6-span-l2 512`
	`Warning: Please save config and reload the system for the configuration to take effect.`
	`config)#`
		レイヤ 2 ポートの UDF で IPv6 を設定します。リージョンの新しい設定により既存の設定が置き換わりますが、設定を有効にするにはスイッチを再起動する必要があります。

ステップ 5	switch(config)# **hardware profile tcam region ipv6-span 512** 例： `(config)# hardware profile tcam region ipv6-span 512`	レイヤ 3 ポートの UDF で IPv6 を設定します。リージョンの新しい設定により既存の設定が置き換わりますが、設定を有効にするにはスイッチを再起動する必要があります。

	コマンドまたはアクション	目的
	Warning: Please save config and reload the system for the configuration to take effect. config)#	
ステップ 6	switch(config)# **hardware profile tcam region span spanv6 qualify udf** *<name1>…… <name8>* 例： (config)# **hardware profile tcam region spanv6 qualify udf udf1** [SUCCESS] Changes to UDF qualifier set will be applicable only after reboot. You need to 'copy run start' and 'reload' config)#	レイヤ 3 ポートの SPAN に UDF 認定を設定します。これにより、ipv6-span TCAM リージョンの UDF 照合が有効になります。TCAM カービング時（ブートアップ時）に UDF を TCAM リージョンの修飾子セットに追加します。この設定では、SPAN リージョンにアタッチできる最大 2 つの IPv6 UDF を許可できます。UDF はすべて、リージョンの単一コマンドでリストされます。リージョンの新しい設定により、既存の設定が置き換わりますが、設定を有効にするには再起動する必要があります。
ステップ 7	switch(config)# **hardware profile tcam region span spanv6-12 qualify udf** *<name1>…… <name8>* 例： (config)# **hardware profile tcam region spanv6-l2 qualify udf udf1** [SUCCESS] Changes to UDF qualifier set will be applicable only after reboot. You need to 'copy run start' and 'reload' config)#	レイヤ 2 ポートの SPAN に UDF 認定を設定します。これにより、ipv6-span-12 TCAM リージョンの UDF 照合が有効になります。TCAM カービング時（ブートアップ時）に UDF を TCAM リージョンの修飾子セットに追加します。この設定では、SPAN リージョンにアタッチできる最大 2 つの IPv6 UDF を許可できます。UDF はすべて、リージョンの単一コマンドでリストされます。リージョンの新しい設定により、既存の設定が置き換わりますが、設定を有効にするには再起動する必要があります。
ステップ 8	switch (config-erspan-src)# **filter** …… *ipv6 access-group….<aclname>….<allow-sharing>* 例： (config-erspan-src)# **ipv6 filter access-group test** (config)#	SPAN および ERSPAN モードで IPv6 ACL を設定します。1 つのモニターセッションには「filter ip access-group」または「filter ipv6 access-group」 のいずれか 1 つだけを設定できます。同じ送信元インターフェイスが IPv4 と IPv6 ERSPAN ACL モニターセッションの一部である場合は、モニターセッションの設定で「allow-sharing」に「filter

	コマンドまたはアクション	目的
		[ipv6] access-group」を設定する必要があります。

ステップ 9	switch(config)# **permit** …… <*regular ACE match criteria*> **udf** <*name1*> < *val* > <*mask*> .....<*name8*> < *val* > <*mask*>
	例：
	`(config-erspan-src)# ipv6 access-list test`
	`(config-ipv6-acl)# permit ipv6 any any udf udf1 0x1 0x0`
		UDF と一致する ACL を設定します。

ステップ 10	switch(config)# **show monitor session** <*session-number*>
	例：
	`(config)# show monitor session 1`
	`session 1`
	`---------------`
	`type : erspan-source`
	`state : up`
	`vrf-name : default`
	`destination-ip : 40.1.1.1`
	`ip-ttl : 255`
	`ip-dscp : 0`
	`acl-name : test`
	`origin-ip : 100.1.1.10`
	`(global)`
	`source intf :`
	`rx : Eth1/20`
	`tx : Eth1/20`
	`both : Eth1/20`
	`source VLANs :`
	`filter VLANs : filter not specified`
	`rx :`
	`source fwd drops :`
	`egress-intf : Eth1/23`
	`switch#`
	`config)#`
		show monitor session <session-number> コマンドを使用して、ACL を表示します。

ERSPAN セッションのシャットダウンまたはアクティブ化

ERSPANセッションをシャットダウンすると、送信元から宛先へのパケットのコピーを切断できます。同時に実行できる ERSPAN セッション数は限定されているため、あるセッションをシャットダウンしてハードウェア リソースを解放することによって、別のセッションが使用できるようになります。デフォルトでは、ERSPAN セッションはシャット ステートで作成されます。

ERSPANセッションをイネーブルにすると、送信元から宛先へのパケットのコピーをアクティブ化できます。すでにイネーブルになっていて、動作状況がダウンの ERSPAN セッションをイネーブルにするには、そのセッションをいったんシャットダウンしてから、改めてイネーブ

ルにする必要があります。ERSPAN セッション ステートをシャットダウンおよびイネーブルにするには、グローバルまたはモニタ コンフィギュレーション モードのいずれかのコマンドを使用できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configuration terminal**
	例：
	`switch# configuration terminal`
	`switch(config)#`
		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	**monitor session** {*session-range* | **all**} **shut** 例： `switch(config)# monitor session 3 shut`	指定の ERSPAN セッションをシャットダウンします。セッションの範囲は 1 ～ 18 です。デフォルトでは、セッションはシャット ステートで作成されます。単方向の 4 つのセッション、または双方向の 2 つのセッションを同時にアクティブにすることができます。 （注） • Cisco Nexus 5000 および 5500 プラットフォームでは、2 つのセッションを同時に実行できます。 • Cisco Nexus 5600 および 6000 プラットフォームでは、16 のセッションを同時に実行できます。
ステップ 3	**no monitor session** {*session-range* | **all**} **shut** 例： `switch(config)# no monitor session 3 shut`	指定の ERSPAN セッションを再開（イネーブルに）します。セッションの範囲は 1 ～ 18 です。デフォルトでは、セッションはシャットステートで作成されます。単方向の 4 つのセッション、または双方向の 2 つのセッションを同時にアクティブにすることができます。 （注） モニターセッションがイネーブルで動作状況がダウンの場合、セッションをイネーブルにするには、最初に **monitor session shut** コマンドを指定してから、**no monitor session shut** コマンドを続ける必要があります。

	コマンドまたはアクション	目的

ステップ 4	**monitor session** *session-number* **type erspan-source**
	例：
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		ERSPAN 送信元タイプのモニタ コンフィギュレーション モードを開始します。新しいセッション コンフィギュレーションは、既存のセッション コンフィギュレーションに追加されます。

ステップ 5	**monitor session** *session-number* **type erspan-destination** 例： `switch(config-erspan-src)# monitor session 3 type erspan-destination`	ERSPAN 宛先タイプのモニター コンフィギュレーション モードを開始します。
ステップ 6	**shut** 例： `switch(config-erspan-src)# shut`	ERSPAN セッションをシャットダウンします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 7	**no shut** 例： `switch(config-erspan-src)# no shut`	ERSPAN セッションをイネーブルにします。デフォルトでは、セッションはシャット ステートで作成されます。
ステップ 8	（任意） **show monitor session all** 例： `switch(config-erspan-src)# show monitor session all`	ERSPAN セッションのステータスを表示します。
ステップ 9	（任意） **show running-config monitor** 例： `switch(config-erspan-src)# show running-config monitor`	ERSPAN の実行コンフィギュレーションを表示します。
ステップ 10	（任意） **show startup-config monitor** 例： `switch(config-erspan-src)# show startup-config monitor`	ERSPAN のスタートアップ コンフィギュレーションを表示します。
ステップ 11	（任意） **copy running-config startup-config** 例： `switch(config-erspan-src)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ERSPAN 設定の確認

ERSPAN の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show monitor session {**all** | *session-number* | **range** *session-range*}	ERSPAN セッション設定を表示します。
show running-config monitor	ERSPAN の実行コンフィギュレーションを表示します。
show startup-config monitor	ERSPAN のスタートアップ コンフィギュレーションを表示します。

ERSPAN の設定例

ERSPAN 送信元セッションの設定例

次に、ERSPAN 送信元セッションを設定する例を示します。

```
switch# config t
switch(config)# interface e14/30
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# monitor erspan origin ip-address 3.3.3.3 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# filter access-group acl1
switch(config-erspan-src)# source interface e14/30
switch(config-erspan-src)# ip ttl 16
switch(config-erspan-src)# ip dscp 5
switch(config-erspan-src)# vrf default
switch(config-erspan-src)# destination ip 9.1.1.2
switch(config-erspan-src)# no shut
switch(config-erspan-src)# exit
switch(config)# show monitor session 1
```

ERSPAN ACL の設定例

次に、ERSPAN ACL を設定する例を示します。

```
switch# configure terminal
switch(config)# ip access-list match_11_pkts
switch(config-acl)# permit ip 11.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# ip access-list match_12_pkts
switch(config-acl)# permit ip 12.0.0.0 0.255.255.255 any
switch(config-acl)# exit
switch(config)# vlan access-map erspan_filter 5
switch(config-access-map)# match ip address match_11_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
```

```
switch(config)# vlan access-map erspan_filter 10
switch(config-access-map)# match ip address match_12_pkts
switch(config-access-map)# action forward
switch(config-access-map)# exit
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# filter access_group erspan_filter
```

UDF ベース ERSPAN の設定例

次に、以下の一致基準を使用して、カプセル化された IP-in-IP パケットの内部 TCP フラグで照合する UDF ベース ERSPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋外部 IP（20）＋内部 IP（20）＋内部 TCP（20、ただし、13 番目のバイトの TCP フラグ）
- パケットの先頭からのオフセット：14 + 20 + 20 + 13 = 67
- UDF の照合値：0x20
- UDF マスク：0xFF

```
udf udf_tcpflags packet-start 67 1
hardware access-list tcam region racl qualify udf udf_tcpflags
copy running-config startup-config
reload
ip access-list acl-udf
  permit ip 10.0.0.2/32 any udf udf_tcpflags 0x20 0xff
monitor session 1 type erspan-source
  source interface Ethernet 1/1
  filter access-group acl-udf
```

次に、以下の一致基準を使用して、レイヤ 4 ヘッダーの先頭から 6 バイト目のパケット署名（DEADBEEF）と通常の IP パケットを照合する UDF ベース ERSPAN を設定する例を示します。

- 外部送信元 IP アドレス：10.0.0.2
- 内部 TCP フラグ：緊急 TCP フラグを設定
- バイト：Eth Hdr（14）＋IP（20）＋TCP（20）＋ペイロード：112233445566DEADBEEF7788
- レイヤ 4 ヘッダーの先頭からのオフセット：20 + 6 = 26
- UDF の照合値：0xDEADBEEF（2 バイトのチャンクおよび 2 つの UDF に分割）
- UDF マスク：0xFFFFFFFF

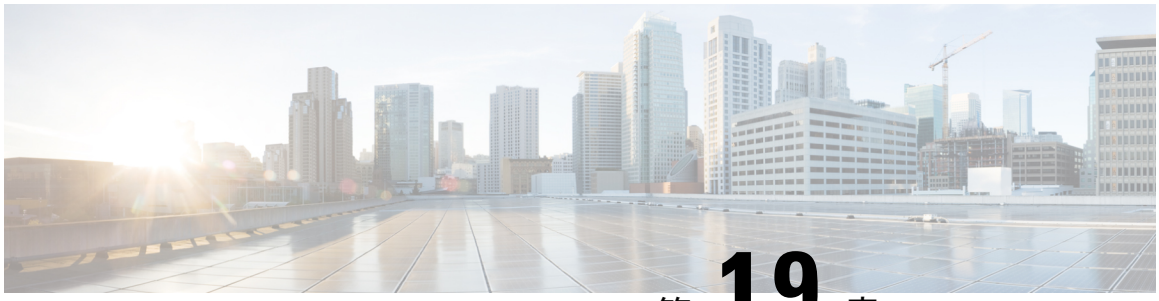
```
udf udf_pktsig_msb header outer l3 26 2
udf udf_pktsig_lsb header outer l3 28 2
hardware access-list tcam region racl qualify udf udf_pktsig_msb udf_pktsig_lsb
copy running-config startup-config
reload
ip access-list acl-udf-pktsig
```

```
  permit udf udf_pktsig_msb 0xDEAD 0xFFFF udf udf_pktsig_lsb 0xBEEF 0xFFFF
monitor session 1 type erspan-source
  source interface Ethernet 1/1
  filter access-group acl-udf-pktsig
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
ERSPAN コマンド：コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト、使用上の注意事項、および例	ご使用プラットフォームの『*Cisco Nexus NX-OS System Management Command Reference*』。



第 19 章

DNS の設定

この章は、次の項で構成されています。

DNS クライアントに関する情報

自分で名前の割り当てを管理していないネットワーク内のデバイスとの接続を、ネットワーク デバイスが必要とする場合は、DNS を使用して、ネットワーク間でデバイスを特定する一意のデバイス名を割り当てることができます。DNS は、階層方式を使用して、ネットワーク ノードのホスト名を確立します。これにより、クライアントサーバー方式によるネットワークのセグメントのローカル制御が可能となります。DNS システムは、デバイスのホスト名をその関連する IP アドレスに変換することで、ネットワーク デバイスを検出できます。

インターネット上のドメインは、組織のタイプや場所に基づく一般的なネットワークのグループを表す命名階層ツリーの一部です。ドメイン名は、ピリオド（.）を区切り文字として使用して構成されています。たとえば、シスコは、インターネットでは com ドメインで表される営利団体であるため、そのドメイン名は cisco.com です。このドメイン内の特定のホスト名、たとえばファイル転送プロトコル（FTP）システムは ftp.cisco.com で識別されます。

ネーム サーバ

ネーム サーバはドメイン名の動向を把握し、自身が完全な情報を持っているドメイン ツリーの部分を認識しています。ネーム サーバは、ドメイン ツリーの他の部分の情報を格納している場合もあります。Cisco NX-OS 内の IP アドレスにドメイン名をマッピングするには、最初にホスト名を示し、その後にネーム サーバーを指定して、DNS サービスをイネーブルにする必要があります。

Cisco NX-OS では、スタティックに IP アドレスをドメイン名にマッピングできます。また、1 つ以上のドメイン ネーム サーバーを使用してホスト名の IP アドレスを見つけるよう、Cisco NX-OS を設定することもできます。

DNS の動作

ネームサーバは、次に示すように、特定のゾーン内でローカルに定義されるホストのDNSサーバに対してクライアントが発行したクエリーを処理します。

- 権限ネーム サーバは、その権限ゾーン内のドメイン名を求める DNS ユーザ照会に、自身のホストテーブル内にキャッシュされた永久的なエントリを使用して応答します。照会で求められているのが、自身の権限ゾーン内であるが、設定情報が登録されていないドメイン名の場合、権限ネーム サーバはその情報が存在しないと応答します。
- 権限ネーム サーバとして設定されていないネーム サーバは、以前に受信した照会への返信からキャッシュした情報を使用して、DNSユーザ照会に応答します。ゾーンの権限ネーム サーバとして設定されたルータがない場合は、ローカルに定義されたホストを求める DNS サーバへの照会には、正規の応答は送信されません。

ネーム サーバは、特定のドメインに設定された転送パラメータおよびルックアップ パラメータに従って、DNS 照会に応答します（着信 DNS 照会を転送するか、内部的に生成された DNS 照会を解決します）。

高可用性

Cisco NX-OS は、DNS クライアントのステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバーの後、Cisco NX-OS は実行コンフィギュレーションを適用します。

DNS クライアントの前提条件

DNS クライアントには次の前提条件があります。

- ネットワーク上に DNS ネーム サーバが必要です。

DNS クライアントのデフォルト設定

次の表に、DNS クライアント パラメータのデフォルト設定を示します。

パラメータ	デフォルト
DNS クライアント	有効（Enabled）

DNS 送信元インターフェイスの設定

特定のインターフェイスを使用するように DNS を設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **ip dns source-interface** *type slot/port*	すべての DNS パケットの送信元インターフェイスを設定します。次のリストに、*interface* として有効な値を示します。 • ethernet • loopback • mgmt • port-channel • vlan （注） DNS の送信元インターフェイスを設定する場合、サーバーから開始される SCP コピー操作は失敗します。サーバーからの SCP コピー操作を実行するには、DNS 送信元インターフェイスの設定を削除します。
ステップ 3	switch(config)# **show ip dns source-interface**	設定済みの DNS 送信元インターフェイスを表示します。

例

次に、DNS 送信元インターフェイスを設定する例を示します。

```
switch(config)# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# ip dns source-interface ethernet 1/8
switch(config)# show ip dns source-interface
VRF Name                                Interface
default                                 Ethernet1/8
```

DNS クライアントの設定

ネットワーク上の DNS サーバを使用するよう、DNS クライアントを設定できます。

始める前に

- ネットワーク上にドメイン ネーム サーバがあることを確認します。

手順

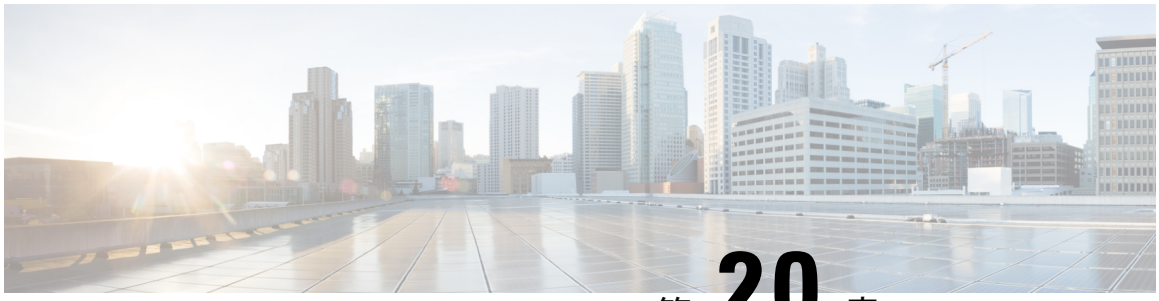
	コマンドまたはアクション	目的
ステップ 1	switch# **configuration terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# vrf context managment	設定可能な仮想およびルーティング（VRF）名を指定します。
ステップ 3	switch(config)# **{ip | ipv6} host** *name ipv/ipv6 address1 [ip/ipv6 address2... ip/ipv6 address6]*	ホスト名キャッシュに、6 つまでのスタティック ホスト名/アドレス マッピングを定義します。
ステップ 4	（任意） switch(config)# **ip domain name** *name* [**use-vrf** *vrf-name*]	Cisco NX-OS が非完全修飾ホスト名に使用するデフォルトのドメイン ネーム サーバーを定義します。このドメイン名を設定した VRF でこのドメイン ネーム サーバーを解決できない場合は、任意で、Cisco NX-OS がこのドメイン ネーム サーバーを解決するために使用する VRF を定義することもできます。 Cisco NX-OS は、ドメイン名ルックアップを開始する前に、完全なドメイン名を含まないあらゆるホスト名にデフォルト ドメイン名を追加します。
ステップ 5	（任意） switch(config)# **ip domain-list** *name* [**use-vrf** *vrf-name*]	Cisco NX-OS が非完全修飾ホスト名に使用できる追加のドメイン ネーム サーバーを定義します。このドメイン名を設定した VRF でこのドメイン ネーム サーバーを解決できない場合は、任意で、Cisco NX-OS がこのドメイン ネーム サーバーを解決するために使用する VRF を定義することもできます。

	コマンドまたはアクション	目的
		Cisco NX-OS はドメイン リスト内の各エントリを使用して、ドメイン名ルックアップを開始する前に、完全なドメイン名を含まないあらゆるホスト名にこのドメイン名を追加します。Cisco NX-OS は、一致するものが見つかるまで、ドメイン リストの各エントリにこれを実行します。
ステップ 6	（任意） switch(config)# **ip name-server** *ip/ipv6 server-address1 [ip/ipv6 server-address2... ip/ipv6 server-address6]* **[use-vrf** *vrf-name*]	最大 6 台のネーム サーバを定義します。使用可能なアドレスは、IPv4 アドレスまたは IPv6 アドレスです。 このネーム サーバを設定した VRF でこのネーム サーバに到達できない場合は、任意で、Cisco NX-OS がこのネーム サーバに到達するために使用する VRF を定義することもできます。
ステップ 7	（任意） switch(config)# **ip domain-lookup**	DNSベースのアドレス変換をイネーブルにします。この機能は、デフォルトでイネーブルにされています。
ステップ 8	（任意） switch(config)# **show hosts**	DNS に関する情報を表示します。
ステップ 9	switch(config)# **exit**	コンフィギュレーション モードを終了し、EXEC モードに戻ります。
ステップ 10	（任意） switch# **copy running-config startup-config**	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

例

次に、デフォルト ドメイン名を設定し、DNSルックアップをイネーブルにする例を示します。

```
switch# config t
switch(config)# vrf context management
switch(config)# ip domain-name mycompany.com
switch(config)# ip name-server 172.68.0.10
switch(config)# ip domain-lookup
```



第 20 章

sFlow の設定

この章は、次の項で構成されています。

sFlow について

sFlowを使用すると、スイッチやルータを含むデータ ネットワーク内のリアルタイム トラフィックをモニターできます。sFlow では、トラフィックをモニターするためにスイッチやルータ上の sFlow エージェント ソフトウェアでサンプリング メカニズムを使用して、入力および出力ポート上のサンプルデータを中央のデータ コレクタ（sFlowアナライザとも呼ばれる）に転送します。

sFlow の詳細については、RFC 3176 を参照してください。

sFlow エージェント

Cisco NX-OS ソフトウェアに組み込まれている sFlow エージェントは、サンプリングされるパケットのデータ ソースに関連付けられたインターフェイス カウンタを定期的にサンプリングまたはポーリングします。このデータ ソースは、イーサネット インターフェイス、EtherChannel インターフェイス、ある範囲に属するイーサネット インターフェイスのいずれかです。sFlow エージェントは、イーサネット ポート マネージャにクエリーを送信して対応する EtherChannel メンバーシップ情報を確認するほか、イーサネット ポート マネージャからもメンバーシップの変更の通知を受信します。

Cisco NX-OS ソフトウェアで sFlow サンプリングをイネーブルにすると、サンプリング レートとハードウェア内部の乱数に基づいて、入力パケットと出力パケットが sFlow でサンプリングされたパケットとして CPU に送信されます。sFlow エージェントはサンプリングされたパケットを処理し、sFlow アナライザに sFlow データグラムを送信します。sFlow データグラムには、元のサンプリングされたパケットに加えて、入力ポート、出力ポート、および元のパケット長に関する情報が含まれます。sFlow データグラムには、複数の sFlow サンプルを含めることができます。

前提条件

sFlow を設定するには、feature sflow コマンドを使用して sFlow 機能をイネーブルにする必要があります。

sFlow の注意事項および制約事項

sFlow 設定時の注意事項および制約事項は次のとおりです。

- インターフェイスの sFlow をイネーブルにすると、入力と出力の両方に対してイネーブルになります。入力だけまたは出力だけの sFlow をイネーブルにできません。
- マルチキャスト、ブロードキャスト、または未知のユニキャストパケットの sFlow の出力のサンプリングはサポートされません。
- システムの sFlow の設定およびトラフィックに基づいてサンプリング レートを設定する必要があります。
- Cisco Nexus 3000 シリーズは、1 つの sFlow コレクタだけをサポートします。

sFlow のデフォルト設定

表 *32 :* デフォルトの *sFlow* パラメータ

パラメータ	デフォルト
sFlow sampling-rate	4096
sFlow sampling-size	128
sFlow max datagram-size	1400
sFlow collector-port	6343
sFlow counter-poll-interval	20

sFLow の設定

sFlow 機能のイネーブル化

スイッチの sFlow を設定する前に sFlow 機能をイネーブルにする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **feature sflow**	sFlow 機能をイネーブルにします。
ステップ 3	（任意） **show feature**	イネーブルおよびディセーブルにされた機能を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、sFlow 機能をイネーブルにする例を示します。

```
switch# configure terminal
switch(config)# feature sflow
switch(config)# copy running-config startup-config
```

サンプリング レートの設定

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow sampling-rate** *sampling-rate*	パケットの sFlow のサンプリング レートを設定します。

	コマンドまたはアクション	目的
		sampling-rate には 4096 ～ 1000000000 の整数を指定できます。デフォルト値は 4096 です。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、サンプリング レートを 50,000 に設定する例を示します。

```
switch# configure terminal
switch(config)# sflow sampling-rate 50000
switch(config)# copy running-config startup-config
```

最大サンプリング サイズの設定

サンプリングされたパケットからコピーする最大バイト数を設定できます。

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow max-sampled-size** *sampling-size*	sFlow の最大サンプリング サイズ パケットを設定します。 *sampling-size* の範囲は 64～256 バイトです。デフォルト値は 128 です。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、sFlow エージェントの最大サンプリング サイズを設定する例を示します。

```
switch# configure terminal
switch(config)# sflow max-sampled-size 200
switch(config)# copy running-config startup-config
```

カウンタのポーリング間隔の設定

データ ソースに関連するカウンタの継続的なサンプル間の最大秒数を設定できます。サンプリング間隔 0 は、カウンタのサンプリングをディセーブルにします。

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow counter-poll-interval** *poll-interval*	インターフェイスの sFlow のポーリング間隔を設定します。*poll-interval* の範囲は 0～2147483647 秒です。デフォルト値は 20 です。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、インターフェイスの sFlow のポーリング間隔を設定する例を示します。

```
switch# configure terminal
switch(config)# sflow counter-poll-interval 100
switch(config)# copy running-config startup-config
```

最大データグラム サイズの設定

1 つのサンプル データグラムで送信できるデータの最大バイト数を設定できます。

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow max-datagram-size** *datagram-size*	sFlow の最大データグラム サイズを設定します。 *datagram-size* の範囲は 200～9000 バイトです。デフォルト値は 1400 です。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、sFlow の最大データグラム サイズを設定する例を示します。

```
switch# configure terminal
switch(config)# sflow max-datagram-size 2000
switch(config)# copy running-config startup-config
[########################################] 100%
```

sFlow アナライザのアドレスの設定

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow collector-ip** *IP-address vrf-instance*	sFlow アナライザの IPv4 アドレスを設定します。

	コマンドまたはアクション	目的
		vrf-instance は、次のいずれかになります。 • ユーザー定義の VRF 名：最大 32 文字の英数字を指定できます。 • **vrf management**：sFlow データ コレクタが管理ポートに接続されたネットワークに存在する場合は、このオプションを使用する必要があります。 • **vrf default**：sFlow データ コレクタが前面パネルのポートに接続されたネットワークに存在する場合は、このオプションを使用する必要があります。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、管理ポートに接続されている sFlow データ コレクタの IPv4 アドレスを設定する例を示します。

```
switch# configure terminal
switch(config)# sflow collector-ip 192.0.2.5 vrf management
switch(config)# copy running-config startup-config
```

sFlow アナライザ ポートの設定

sFlow データグラムの宛先ポートを設定できます。

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow collector-port** *collector-port*	sFlow アナライザの UDP ポートを設定します。 *collector-port* の範囲は 0～65535 です。デフォルト値は 6343 です。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、sFlow データグラムの宛先ポートを設定する例を示します。

```
switch# configure terminal
switch(config)# sflow collector-port 7000
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

sFlow エージェント アドレスの設定

始める前に

sFlow 機能がイネーブルになっていることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	[**no**] **sflow agent-ip** *ip-address*	sFlow エージェントの IPv4 アドレスを設定します。 デフォルトの *ip-address* は 0.0.0.0 です。つまり、すべてのサンプリングがスイッチでディセーブルであることを示します。sFlow 機能をイネーブルにするに

	コマンドまたはアクション	目的
		は、有効な IP アドレスを指定する必要があります。 （注） この IP アドレスは、コレクタに sFlow データグラムを送信するための送信元 IP アドレスとは限りません。
ステップ 3	（任意） **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、sFlow エージェントの IPv4 アドレスを設定する例を示します。

```
switch# configure terminal
switch(config)# sflow agent-ip 192.0.2.3
switch(config)# copy running-config startup-config
```

sFlow サンプリング データ ソースの設定

sFlow のサンプリング データ ソースには、イーサネット ポート、イーサネット ポートの範囲、またはポート チャネルを指定できます。

始める前に

- sFlow 機能がイネーブルになっていることを確認します。
- データ ソースとしてポート チャネルを使用する場合は、すでにポート チャネルを設定して、ポート チャネル番号がわかっていることを確認してください。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# [**no**] **sflow data-source interface** [**ethernet** *slot*/*port*[-*port*] | **port-channel** *channel-number*]	sFlow のサンプリング データ ソースを設定します。 イーサネットのデータ ソースの場合、*slot* はスロット番号、*port* は 1 つのポー

	コマンドまたはアクション	目的
		ト番号または *port-port* で指定されたポートの範囲です。
ステップ 3	（任意） switch(config)# **show sflow**	sFlow 情報を表示します。
ステップ 4	（任意） switch(config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

例

次に、sFlow のサンプラーのイーサネット ポート 5～12 を設定する例を示します。

```
switch# configure terminal
switch(config)# sflow data-source interface ethernet 1/5-12
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

次に、sFlow のサンプラーのポート チャネル 100 を設定する例を示します。

```
switch# configure terminal
switch(config)# sflow data-source interface port-channel 100
switch(config)# copy running-config startup-config
[########################################] 100%
switch(config)#
```

sFlow 設定の確認

sFlow の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show sflow	sFlow のグローバル コンフィギュレーションを表示します。
show sflow statistics	sFlow の統計情報を表示します。
clear sflow statistics	sFlow 統計情報をクリアします。
show running-config sflow [**all**]	現在実行中の sFlow コンフィギュレーションを表示します。

sFlow の設定例

次に sFlow を設定する例を示します。

```
feature sflow
sflow sampling-rate 5000
sflow max-sampled-size 200
sflow counter-poll-interval 100
sflow max-datagram-size 2000
sflow collector-ip 192.0.2.5 vrf management
sflow collector-port 7000
sflow agent-ip 192.0.2.3
sflow data-source interface ethernet 1/5
```

sFlow に関する追加情報

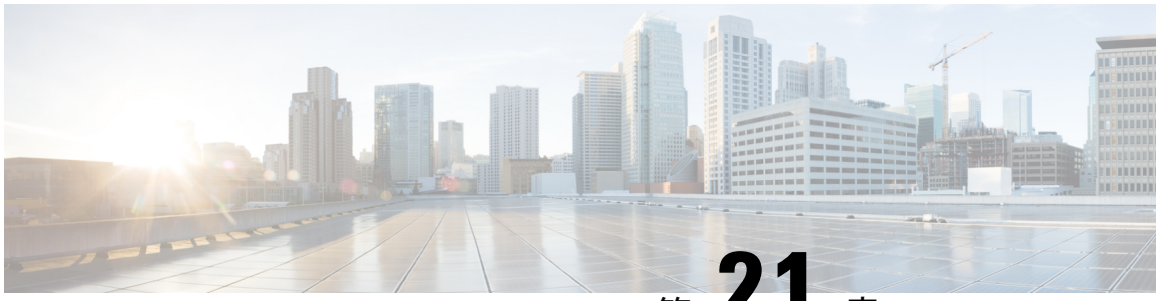
表 *33 : sFlow* の関連資料

関連項目	マニュアル タイトル
sFlow CLI コマンド	『*Cisco Nexus 3000 Series NX-OS System Management Command Reference*』
RFC 3176	sFlow のパケット形式と SNMP MIB を定義します。 http://www.sflow.org/rfc3176.txt

sFlow の機能の履歴

この表には、機能の追加や変更によるリリースの更新内容のみが記載されています。

機能名	リリース	機能情報
sFlow	5.0(3)U4(1)	この機能が導入されました。



第 21 章

タップアグリゲーションおよびMPLSストリッピングの設定

この章は、次の項で構成されています。

タップ アグリゲーションに関する情報

ネットワーク タップ

さまざまなメソッドを使用して、パケットをモニターできます。1つのメソッドでは、物理ハードウェア タップが使用されます。

ネットワーク タップは、ネットワークを通過するデータへの直接インライン アクセスが可能なので、トラフィックのモニタリングに非常に役立ちます。多くの場合、サード パーティがネットワーク内の2ポイント間のトラフィックをモニターするのに適しています。ポイントAとBの間のネットワークが物理ケーブルで構成されている場合、ネットワーク タップがこのモニタリングを実現する最良の方法になります。ネットワーク タップには、少なくとも3つのポート（Aポート、Bポート、およびモニター ポート）があります。AポートとBポートの間に挿入されるタップは、すべてのトラフィックをスムーズに通過させますが、同じデータをそのモニター ポートにもコピーするため、サード パーティがリッスンできるようになります。

タップには次の利点があります。

- 全二重データ伝送を処理可能
- 目立たず、ネットワークによって検出されることがなく、物理または論理アドレッシングが不要

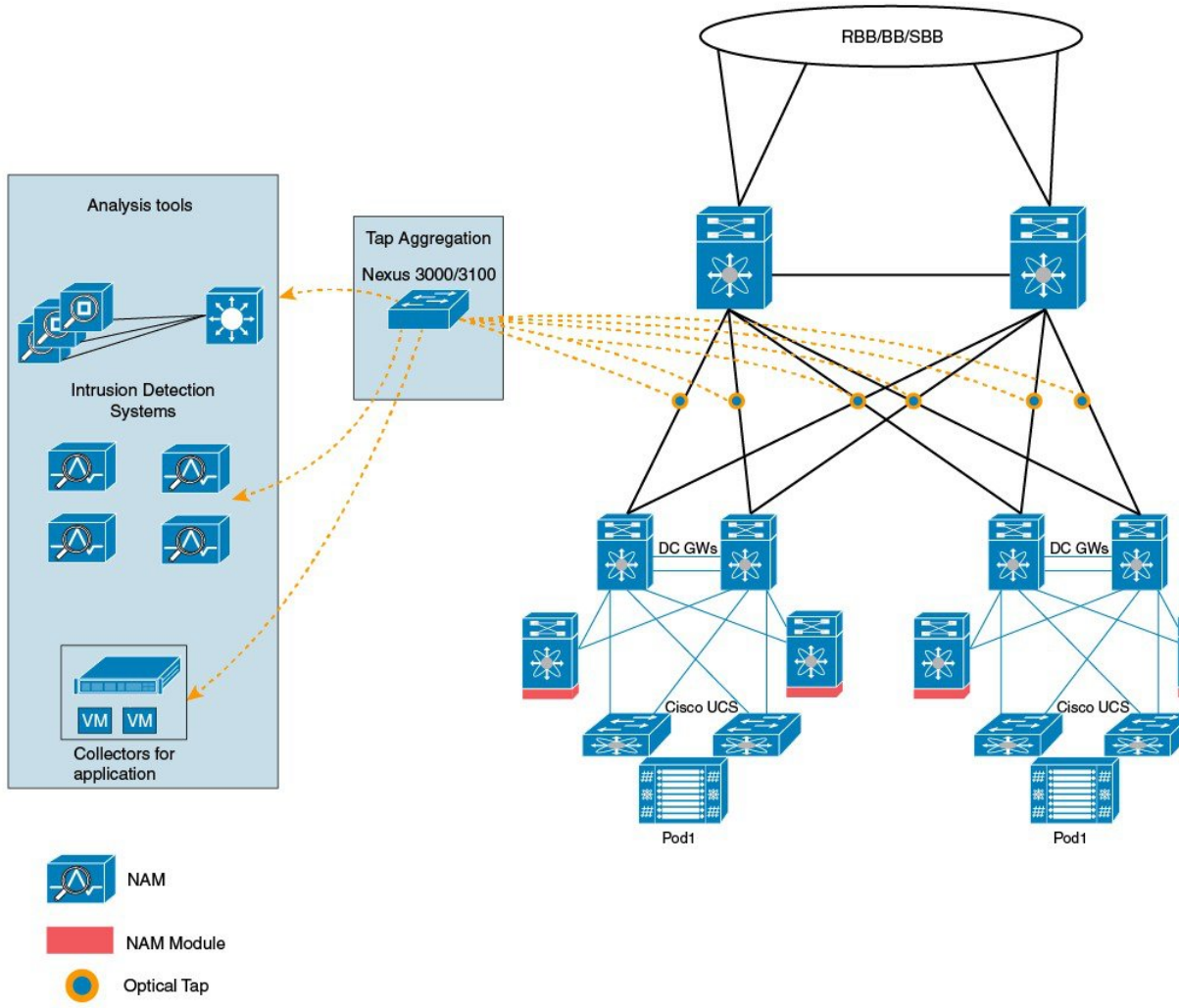
- 一部のタップは、分散タップを構築する機能のあるフル インライン パワーをサポート

ネットワークのエッジまたは仮想エッジにおけるサーバー間データ通信に対する可視性を確保しようとする場合、またはネットワークのインターネット エッジで侵入防御システム（IPS）アプライアンスにトラフィックのコピーを提供する場合でも、ネットワーク タップは、環境内のほぼすべての場所で使用できます。ただし、大規模環境にネットワーク タップを導入する場合、多くのコストがかかり、運用の複雑さが増し、ケーブル配線の問題が生じます。

タップ アグリゲーション

データセンターにおけるモニタリングおよびトラブルシューティング タスクに役立つ代替ソリューションは、複数タップの集約を可能にし、複数のモニタリング システムに接続するためだけに指定されているデバイスを使用するソリューションです。このソリューションは、タップ アグリゲーションと呼ばれます。タップ アグリゲーション スイッチは、監視する必要があるパケットを処理するネットワーク ファブリック内の特定のポイントにすべてのモニタリング デバイスを直接リンクします。

図 *1:* タップ アグリゲーション スイッチ ソリューション



タップ アグリゲーション スイッチ ソリューションでは、Cisco Nexus 3000 または Cisco Nexus 3100 シリーズ スイッチは、パケットのモニタリングに都合の良い、ネットワーク内のさまざまなポイントに接続されます。各ネットワーク要素から、スイッチド ポート アナライザ（SPAN）または光タップを使用して、このタップ アグリゲーション スイッチにトラフィック フローを直接送信できます。タップ アグリゲーション スイッチ自体は、ネットワーク ファブリック内のイベントをモニターするために使用されるすべての分析ツールに直接接続されます。これらのモニタリング デバイスには、リモート モニタリング（RMON）プローブ、アプリケーション ファイアウォール、IPS デバイス、およびパケット スニファ ツールが含まれます。

ネットワーク要素に接続されている特定のポートのセットを介して、トラフィックのスイッチへの到達を許可する設定を指定して、タップ アグリゲーション スイッチを動的にプログラミングできます。特定のトラフィックをフィルタ処理して、1 つ以上のツールにリダイレクトする、複数の一致条件とアクションも設定できます。

タップ アグリゲーションの注意事項と制約事項

タップ アグリゲーションに関する注意事項と制約事項は次のとおりです。

- Cisco Nexus 3000 シリーズ スイッチでは、MPLS タグでの TAP アグリゲーション フィルタはサポートされていません。
- タップ アグリゲーション ポリシーとともに適用されるインターフェイスは、レイヤ 2 にある必要があります。レイヤ 3 インターフェイスはポリシーを指定して設定できますが、そのポリシーは機能しなくなります。
- 各ルールは、1 つの固有の一致基準とのみ関連付ける必要があります。
- すべてのタップ アグリゲーション インターフェイスが、同じ ACL を共有する必要があります。一致基準には入力インターフェイスが含まれているため、複数のインターフェイス間に複数の ACL は必要ありません。
- アクション **vlan-set** と **vlan-strip** は必ず **redirect** アクションの後に指定する必要があります。そうしないと、エントリが無効であるとして拒否されます。
- 拒否ルールでは、**redirect**、**vlan-set**、および **vlan-strip** などのアクションはサポートされません。
- ポリシー用インターフェイスのリストなどの入力リストを入力する場合は、スペースではなくカンマでエントリを区切る必要があります。例：port-channel50,ethernet1/12,port-channel20。
- ポリシーにターゲット インターフェイスを指定する場合、短縮形ではなく、完全なインターフェイス タイプを入力する必要があります。例、eth1/1 ではなく ethernet1/1、po50 ではなく port-channel 50 と入力します。

MPLS ストリッピングに関する情報

MPLS の概要

マルチプロトコル ラベル スイッチング（MPLS）では、レイヤ 2 スイッチングのパフォーマンスおよびトラフィック管理機能と、レイヤ 3 ルーティングの拡張性、柔軟性、およびパフォーマンスが統合されています。

MPLS アーキテクチャには、次の利点があります。

- データは、レイヤ 2 テクノロジーの任意の組み合わせを使用して転送できます。
- サポートは、すべてのレイヤ 3 プロトコルに対して提供されています。
- 今日のネットワークで提供される最も優れた拡張性を備えています。

MPLS ヘッダー ストリッピング

Cisco Nexus 3172 の入力ポートは、さまざまな MPLS パケット タイプを受信します。MPLS ネットワークの各データ パケットには、1 つ以上のラベル ヘッダーがあります。これらのパケットはリダイレクト ACL に基づいてリダイレクトされます。

ラベルは、Forwarding Equivalence Class（FEC）を特定するために使用される短い 4 バイトの固定長のローカルで有効な識別子です。特定のパケットに設定されているラベルは、そのパケットが割り当てられている FEC を表します。次のコンポーネントがあります。

- Label：ラベルの値（非構造化）、20 ビット
- Exp：試験的使用、3 ビット、現在、サービス クラス（CoS）フィールドとして使用
- S：スタックの一番下、1 ビット
- TTL：存続可能時間、8 ビット

MPLS ラベルはレイヤ 2 ヘッダーとレイヤ 3 ヘッダーの間に適用されるため、そのヘッダーとデータは、標準のバイト オフセットには含まれません。標準のネットワーク モニタリング ツールでは、このトラフィックのモニタリングと分析はできません。標準のネットワーク モニタリング ツールでこのトラフィックをモニタリングできるようにするには、単一ラベルのパケットから MPLS ラベル ヘッダーを削除して、T キャッシュ デバイスにリダイレクトします。

複数のラベル ヘッダーがある MPLS パケットは、MPLS ヘッダーが削除されずに、ディープ パケット インスペクション（DPI）デバイスに送信されます。

MPLS ストリッピングに関する注意事項と制限事項

MPLS ストリッピングに関する注意事項と制約事項は次のとおりです。

- MPLS ストリッピングを有効にする前に、すべてのレイヤ 3 および vPC 機能を無効にします。
- グローバル タップ アグリゲーション モードが有効であることを確認します。
- MPLS ストリッピングに関係する入力および出力インターフェイスで、**mode tap-aggregation** が有効になっている必要があります。
- 目的の宛先にパケットを転送するためには、入力インターフェイスのリダイレクト アクションを使用してタップ アグリゲーション ACL を設定する必要があります。
- システムでは 1 つのタップ ACL のみサポートされます。
- 削除されたパケットが出力される出力インターフェイスは、許可 VLAN としての VLAN 1 が存在するインターフェイスである必要があります。出力インターフェイスは、デフォルトですべての VLAN が許可されるトランクとして設定することを推奨します。
- MPLS ストリッピングを有効にするには、MPLS のコントロール プレーン ポリシング（CoPP）クラス（copp-s-mpls）を設定する必要があります。

- MPLS ストリッピング パケットの場合、port-channel ロード バランシングがサポートされます。
- レイヤ 3 ヘッダー ベースのハッシュおよびレイヤ 4 ヘッダー ベースのハッシュはサポートされていますが、レイヤ 2 ヘッダー ベースのハッシュはサポートされていません。
- MPLS ストリッピング時、VLAN では MPLS ラベルも削除されます。
- MPLS ストリッピングは、Cisco Nexus 3100 シリーズ スイッチでのみサポートされています。

タップ アグリゲーションの設定

タップ アグリゲーションの有効化

タップ アグリゲーションを有効にしたら、**copy running-config startup-config** コマンドを実行して、スイッチをリロードしてください。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch (config)# **[no] hardware profile tap-aggregation [l2drop]**	タップ アグリゲーションを有効にし、VLAN タギングに必要なエントリをインターフェイス テーブルに予約します。 **l2drop** オプションは、タップ インターフェイス上で IP 以外のトラフィック入力をドロップします。 このコマンドの **no** 形式を使用すると、この機能が無効化されます。
ステップ 3	switch (config)# **copy running-config startup-config**	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。
ステップ 4	switch (config)# **reload**	Cisco NX-OS ソフトウェアをリロードします。

例

次に、スイッチ上でタップ アグリゲーションをグローバルに設定する例を示します。

```
switch# configure terminal
switch(config)# hardware profile tap-aggregation
switch(config)# copy running-config startup-config
switch(config)# reload
```

タップ アグリゲーション ポリシーの設定

IP アクセス コントロール リスト（ACL）または MAC ACL で、TAP アグリゲーション ポリシーを設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	• switch(config)# **ip access-list** *access-list-name* • switch(config)# **mac access-list** *access-list-name*	IP ACL を作成して IP アクセス リスト コンフィギュレーション モードを開始するか、あるいは MAC ACL を作成して MAC アクセス リスト コンフィギュレーション モードを開始します。 （注） リリース 7.0(3)I5(1) 以降の Cisco Nexus 3000 シリーズ スイッチでは、IPv6 ACL のサポートが追加されます。IPv6 ACL ではリダイレクト アクションがサポートされます。リダイレクト アクションでは、現在 IPv6 PACL でサポートされているすべての match オプションがサポートされています。
ステップ 3	switch(config-acl)# **statistics per-entry**	各エントリで許可または拒否されるパケット数の統計情報の記録を開始します。
ステップ 4	switch(config-acl)# **[no] permit** *protocol source destination match-criteria action*	条件に一致するトラフィックを許可する、IP アクセス コントロール リスト（ACL）のルールを作成します。 このコマンドの **no** バージョンは、ポリシーから許可ルールを削除します。

	コマンドまたはアクション	目的
		match-criteria は、次のいずれかになります。 • **ingress-intf** （注） 入力インターフェイスはレイヤ 2 のみの一致基準（EtherType またはポート チャネル）になります。 • **vlan** • **vlan-priority** （注） 各ポリシーには、一意の一致条件と関連付けられた 1 つのルールのみ設定できます。 *action* は、次のいずれかになります。 • **redirect** • **priority** • **set-vlan** IP 以外の Ethertype で一致するタップ ACL には、0 よりも大きい優先度を指定する必要があります。
ステップ 5	switch(config-acl)# **[no] deny** *protocol source destination match-criteria action*	条件に一致するトラフィックを拒否する、IP アクセス コントロール リスト（ACL）のルールを作成します。 このコマンドの **no** バージョンは、ポリシーから拒否ルールを削除します。 **redirect**、および **vlan-set** アクションはサポートしていません。

例

次に、タップ アグリゲーション ポリシーを設定する例を示します。

```
switch# configure terminal
switch(config)# ip access-list test
switch(config-acl)# statistics per-entry
switch(config-acl)# permit ip any any ingress-intf Ethernet1/4 redirect Ethernet1/8
switch(config-acl)# permit ip any any ingress-intf Ethernet1/6 redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
switch(config-acl)# permit tcp any eq www any ingress-intf Ethernet1/10 redirect
```

```
port-channel4
switch(config-acl)# deny ip any any
```

タップ アグリゲーション ポリシーのインターフェイスへのアタッチ

タップアグリゲーションポリシーをインターフェイスにアタッチするには、タップアグリゲーション モードを開始し、タップ アグリゲーションが設定された ACL をインターフェイスに適用します。ポリシーをアタッチするインターフェイスがレイヤ2インターフェイスであることを確認します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **interface** *type slot/port*	指定したインターフェイスのインターフェイス 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	switch (config-if)# **[no] mode tap-aggregation**	ACL と一致基準とアクション基準のアタッチメントを許可します。 このコマンドの **no** 形式は、タップ アグリゲーション ポリシーを設定した ACL のインターフェイスへのアタッチメントを禁止します。インターフェイスから ACL を削除するには、**no ip port access-group** コマンドを使用します。
ステップ 4	switch(config-if)# **[no] ip port access-group** *access-list-name* **in**	IPv4 アクセス コントロール リスト（ACL）をポート ACL としてインターフェイスに適用します。 このコマンドの **no** 形式は、インターフェイスから ACL を削除します。

例

次に、タップアグリゲーションポリシーをインターフェイスにアタッチする例を示します。

```
switch# configure terminal
switch(config)# interface ethernet1/2
switch (config-if)# mode tap-aggregation
switch(config-if)# ip port access-group test in
```

タップ アグリゲーションの設定の確認

コマンド	目的
show ip access-list *access-list-name*	すべての IPv4 アクセス コントロール リスト（ACL）または特定の IPv4 ACL を表示します。

例

次に、IPv4 ACL を表示する例を示します。

```
switch(config)# show ip access-list test
IPV4 ACL test
        10 permit ip any any ethertype 0x800 ingress-intf Ethernet1/4 redirect E
thernet1/8
        20 permit ip any any ingress-intf Ethernet1/6 redirect Ethernet1/1,Ether
net1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
        30 permit tcp any eq www any ethertype 0x800 ingress-intf Ethernet1/10 r
edirect port-channel4
        40 deny ip any any
```

MPLS ストリッピングの設定

MPLS ストリッピングの有効化

MPLS ストリッピングをグローバルに有効にできます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **[no] mpls strip**	MPLS ストリッピングをグローバルに有効にします。 このコマンドの **no** 形式を使用すると、MPLS ストリッピングが無効化されます。

例

次に、MPLS ストリッピングを有効にする例を示します。

```
switch# configure terminal
switch(config)# mpls strip
```

MPLS ラベルの追加と削除

デバイスは、フレームがモード タップ インターフェイスで不明なラベルを受信するたびにラベルを動的に学習できます。また、次のコマンドを使用して、スタティック MPLS ラベルを追加または削除できます。

始める前に

- タップ アグリゲーションの有効化
- タップ アグリゲーション ポリシーの設定
- タップ アグリゲーション ポリシーのインターフェイスへのアタッチ

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **mpls strip label** *label*	指定したスタティック MPLS ラベルを追加します。 ラベルの値の範囲は 1 ～ 1048575 です。
ステップ 3	switch(config)# **no mpls strip label** *label* | **all**	指定したスタティック MPLS ラベルを削除します。 **all** オプションは、すべてのスタティック MPLS ラベルを削除します。

例

次に、スタティック MPLS ラベルを追加する例を示します。

```
switch# configure terminal
switch(config)# mpls strip label 100
switch(config)# mpls strip label 200
switch(config)# mpls strip label 300
```

次に、スタティック MPLS ラベルを削除する例を示します。

```
switch# configure terminal
switch(config)# no mpls strip label 200
```

次に、すべてのスタティック MPLS ラベルを削除する例を示します。

```
switch# configure terminal
switch(config)# no mpls strip label all
```

ラベル エントリのクリア

次のコマンドを使用して、MPLS ラベル テーブルからダイナミック ラベル エントリをクリアできます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **clear mpls strip label dynamic**	MPLS ラベル テーブルからダイナミック ラベル エントリをクリアします。

例

次に、ダイナミック ラベル エントリをクリアする例を示します。

```
switch# clear mpls strip label dynamic
```

MPLS ストリッピング カウンタのクリア

すべてのソフトウェアおよびハードウェア MPLS ストリッピング カウンタをクリアできます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **clear counters mpls strip**	すべての MPLS ストリッピング カウンタをクリアします。

例

次に、すべての MPLS ストリッピング カウンタをクリアする例を示します。

```
switch# clear counters mpls strip
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 15000
    Static     : 2
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
--------------------------------------------------------------------------------
```

```
     Label     Interface      Idle-Age    SW-Counter       HW-Counter
--------------------------------------------------------------------------------
      4096     Eth1/44              15             0                0
      8192     Eth1/44              17             0                0
     12288     Eth1/44              15             0                0
     16384     Eth1/44              39             0                0
     20480     Eth1/44              47             0                0
     24576     Eth1/44               7             0                0
     28672     Eth1/44               5             0                0
     36864     Eth1/44               7             0                0
     40960     Eth1/44              19             0                0
     45056     Eth1/44               9             0                0
     49152     Eth1/44              45             0                0
     53248     Eth1/44               9             0                0
```

MPLS ラベル エージングの設定

使用されていないダイナミック MPLS ラベルがエージ アウトする時間を定義できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **mpls strip label-age** *age*	ダイナミック MPLS ラベルがエージ アウトする時間を指定します。

例

次に、ダイナミック MPLS ラベルのラベル エージを設定する例を示します。

```
switch# configure terminal
switch(config)# mpls strip label-age 300
```

宛先 MAC アドレスの設定

削除された出力フレームの宛先 MAC アドレスを設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	グローバル構成モードを開始します。
ステップ 2	switch(config)# **mpls strip dest-mac** *mac-address*	ヘッダーが削除された出力フレームの宛先 MAC アドレスを指定します。 MAC アドレスは、次の 4 つのいずれかの形式で指定できます。

	コマンドまたはアクション	目的
		• E.E.E • EE-EE-EE-EE-EE-EE • EE:EE:EE:EE:EE:EE • EEEE.EEEE.EEEE

例

次に、出力フレームの宛先 MAC アドレスを設定する例を示します。

```
switch# configure terminal
switch(config)# mpls strip dest-mac 1.1.1
```

MPLS ラベルの設定の確認

次のコマンドを使用して、MPLS ラベルの設定を表示します。

コマンド	目的
show mpls strip labels [*label* | **all** | **dynamic** |**static**]	MPLS ラベルに関する情報を表示します。次のオプションを指定できます。 • *label*：表示するラベル • **all**：すべてのラベルを表示することを指定します。これがデフォルトのオプションです。 • **dynamic**：ダイナミック ラベルのみ表示することを指定します。 • **static**：スタティック ラベルのみ表示することを指定します。

例

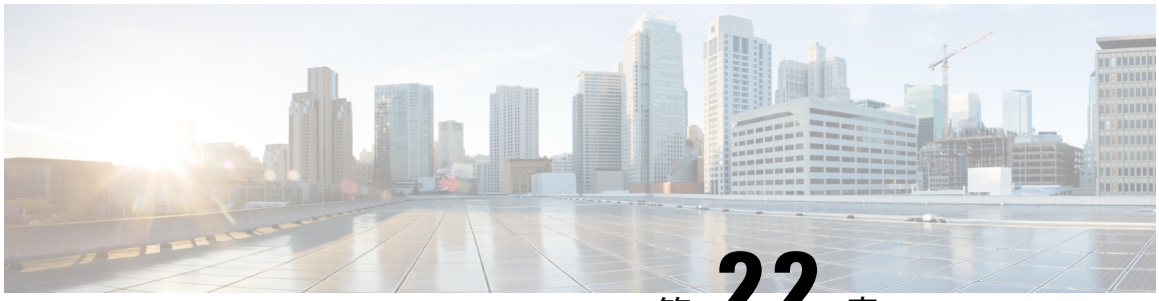
次に、すべての MPLS ラベルを表示する例を示します。

```
switch# show mpls strip labels
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
--------------------------------------------------------------------------------
     Label    Interface      Idle-Age   SW-Counter       HW-Counter
```

```
--------------------------------------------------------------------------------
     4096    Eth1/53/1            15            1             210
     4097    Eth1/53/1            15            1             210
     4098    Eth1/53/1            15            1             210
     4099    Eth1/53/1             7            2             219
     4100    Eth1/53/1             7            2             219
     4101    Eth1/53/1             7            2             219
     4102    Eth1/53/1            39            1             206
     4103    Eth1/53/1            39            1             206
     4104    Eth1/53/1            39            1             206
     4105    Eth1/53/1             1            1             217
     4106    Eth1/53/1             1            1             217
     4107    Eth1/53/1             1            1             217
     4108    Eth1/53/1            15            1             210
*   25000    None <User>          39            1             206
*   20000    None <User>          39            1             206
*   21000    None <User>           1            1             217
```

次に、スタティック MPLS ラベルのみ表示する例を示します。

```
switch(config)# show mpls strip labels static
MPLS Strip Labels:
    Total      : 3005
    Static     : 5
Legend:     * - Static Label
    Interface - where label was first learned
    Idle-Age  - Seconds since last use
    SW-Counter- Packets received in Software
    HW-Counter- Packets switched in Hardware
--------------------------------------------------------------------------------
     Label    Interface      Idle-Age   SW-Counter       HW-Counter
--------------------------------------------------------------------------------
*     300    None <User>         403            0                0
*     100    None <User>         416            0                0
*   25000    None <User>         869            0                0
*   20000    None <User>         869            0                0
*   21000    None <User>         869            0                0
```



第 22 章

一時キャプチャ バッファの設定

一時キャプチャ バッファについて

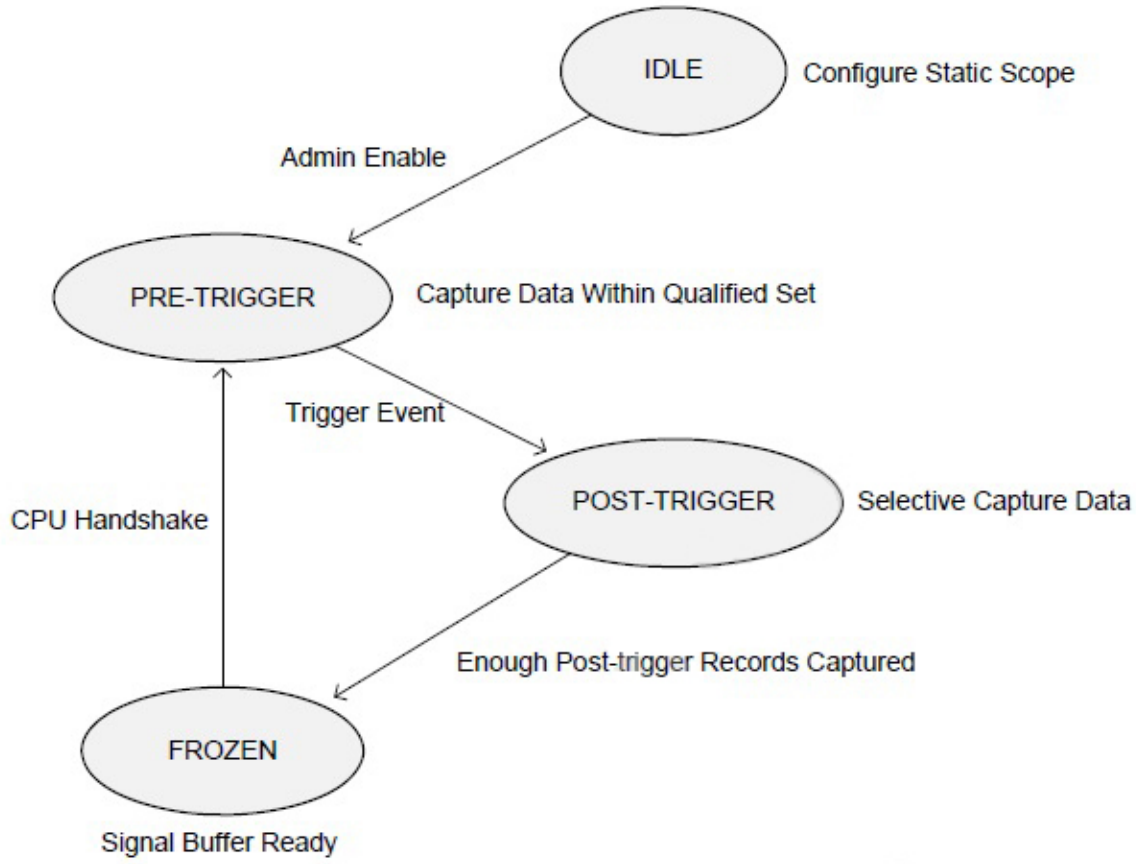
一時キャプチャ バッファ（TCB）は、パケット ドロップ イベントをモニターするデバッグ機能です。TCB により、パケット ドロップの周辺にあるトランザクションがよく見えるようになります。この機能は、予期しない珍しいパケット ドロップのデバッグを目的としています。

TCB は以下で構成されています。

- **TCB バッファ（循環バッファ）**：特定のドロップ イベントの周辺にあるメモリ管理ユニット（MMU）リソースのセットでトランザクションをキャプチャするために使用します。
  - パケット メタデータ（送信元/宛先ポート、タイムスタンプ、ユニキャスト キュー番号、ユニキャスト キューの項目数、サービス プールの深さなど）
  - raw パケット データ（パケットの最初から 80 バイト）
- **イベント バッファ（FIFO バッファ）**：次の目的で使用します。
  - ドロップ パケット メタデータの記録
  - ドロップの原因特定

次の図に、TCB のワークフローを示します。

図 *2:*一時キャプチャ バッファのフェーズ ワークフロー



トリガー後のフェーズでは、キャプチャ範囲の他のキューで発生するドロップがイベントバッファに保存されます。このバッファには、パケットのメタデータが保存されます。raw パケット情報は失われます。

TCB の設定属性を次に示します。

- キャプチャ範囲：
  - モニター範囲タイプ：TCB がモニターする範囲タイプを決定します。サポートされている範囲は次のとおりです。
    - ユニキャスト キュー（UCQ）
    - 入力ポート
    - 出力ポート
  - モニター範囲エンティティ：モニター範囲タイプと一貫性がある必要があります。サポートされているエンティティは次のとおりです。
    - UCQ ID

- ポート番号

- ドロップ イベント トリガー：トリガーを引き起こす可能性のあるメカニズムをドロップします。サポートされているトリガーは次のとおりです。
  - 入力アドミッション ドロップ
  - 出力アドミッション ドロップ
  - 重み付けランダム早期検出（WRED）ドロップ
- トリガー前フェーズのサンプル確率：トリガー前フェーズのパケット サンプリング確率（1/16 ～ すべて）
- トリガー後フェーズのサンプル確率：トリガー後フェーズのパケット サンプリング確率（1/16 ～ すべて）
- 凍結条件：TCB ステートマシンは、以下の凍結条件のいずれかに達したときに凍結フェーズに入ります。
  - 凍結前キャプチャ数：ドロップ イベント トリガーと凍結フェーズの間でキャプチャされたパケットの数
  - 凍結前キャプチャ時間：ドロップイベントトリガーから凍結フェーズまでの時間（マイクロ秒）
- しきい値プロファイル：TCB インスタンスごとに使用できる 8 個のしきい値プロファイル。開始しきい値および停止しきい値があります。開始しきい値は、停止しきい値よりも大きい必要があります。
- しきい値プロファイル マップ：TCB スコープ内の各 UCQ は 1 つのしきい値プロファイルにマッピングでき、異なる UCQ を 1 つのしきい値プロファイルにマッピングすることもできます。サポートされているマップは次のとおりです。
  - 出力アドミッション ドロップ
  - 重み付けランダム ドロップ

注意事項と制約事項

一時キャプチャ バッファのガイドラインと制限事項は以下のとおりです。

- 一時キャプチャ バッファ機能は、Cisco Nexus 3132C-Z および Cisco Nexus 3264C-E スイッチでのみサポートされます
- 一度に設定できるキャプチャ範囲（UC キュー、入力ポート、または出力ポートなど）は 1 つだけです。
- カットスルー パケットはキャプチャされません。

- TCB 機能はパケット ドロップが多数ある状況には適していない可能性があります。

一時キャプチャバッファ範囲およびエンティティ情報の設定

一時キャプチャ バッファ範囲およびエンティティの設定方法

キャプチャ エンティティ パラメータは、周辺で TCB が機能するポートを指定します。エンティティには、範囲に応じて、ポートまたはポート内の特定の qos-group を指定できます。

次の 3 つの範囲で TCB を設定する手順を以下に示します。

- **ユニキャスト**：キュー単位でキャプチャ範囲を指定する場合に使用します。一時キャプチャ バッファ ユニキャスト範囲の設定 （328 ページ）を参照してください。
- **入力**：キャプチャ範囲を入力として指定する場合に使用します。一時キャプチャ バッファ入力範囲の設定 （329 ページ）を参照してください。
- **出力**：キャプチャ範囲を出力として指定する場合に使用します。「一時キャプチャ バッファ出力範囲の設定 （329 ページ）」を参照してください。

一時キャプチャ バッファ ユニキャスト範囲の設定

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCB を設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **source unicast-queue interface** *interface* **qos-group** *qos-group* 例： `switch(config-pkt-drop)# source unicast-queue interface ethernet 1/1 qos-group 1`	キュー単位でキャプチャ範囲を指定します。 • *interface* は、イーサネット IEEE 802.3z エンティティ インターフェイスです • *qos-group* は、インターフェイスに関連付けられているキューです

一時キャプチャ バッファ入力範囲の設定

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCBを設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **source ingress interface ethernet** *interface* 例： `switch(config-pkt-drop)# source ingress interface ethernet 1/1`	キャプチャ範囲を入力として指定します。ここで、*interface* はイーサネット IEEE 802.3z エンティティ インターフェイスです。

一時キャプチャ バッファ出力範囲の設定

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCBを設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **source egress interface ethernet** *interface* 例： `switch(config-pkt-drop)# source egress interface ethernet 1/1`	キャプチャ範囲を出力として指定します。ここで、*interface* はイーサネット IEEE 802.3z エンティティ インターフェイスです。

一時キャプチャ バッファ範囲の設定サンプル

各タイプの範囲について、TCB 設定のサンプルを次に示します。

ユニキャスト範囲

```
hardware profile packet-drop
  source unicast-queue interface Ethernet1/49 qos-group 0
  timer 300
  count 200
  drop-trigger ingress-admission
  sampling-rate pre-trigger 10 post-trigger 10
  no shutdown
```

入力範囲

```
hardware profile packet-drop
  source ingress interface eth1/9
  timer 300
  count 200
  drop-trigger ingress-admission
  profile acme
    start-threshold 1500
    stop-threshold 1000
    interface Ethernet1/49 qos-group 2
    interface Ethernet1/49 qos-group 0
  sampling-rate pre-trigger 10 post-trigger 10
  no shutdown
```

出力範囲

```
hardware profile packet-drop
  source egress interface eth1/49
  timer 300
  count 200
  drop-trigger egress-admission
  profile acme
    start-threshold 1500
    stop-threshold 1000
    interface Ethernet1/49 qos-group 2
    interface Ethernet1/49 qos-group 0
  no shutdown
```

一時キャプチャ バッファ プロファイルの設定

最大7つのプロファイルを、モニタリング用のそれぞれの開始および停止しきい値とともに作成できます。設定するインターフェイスは、ハードウェアの対応するプロファイルにマッピングされます。入力範囲と出力範囲の場合にのみ必要です。

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCBを設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **profile test**	TCB プロファイルを作成できるレベルに移動します。
ステップ 3	switch(config-pkt-drop-profile)# **start-threshold** *parameter* 例： `switch(config-pkt-drop-profile)# start-threshold 512`	start-threshold パラメータを設定します。ここで、*parameter* はバイト単位のパラメータです。

	コマンドまたはアクション	目的
ステップ 4	switch(config-pkt-drop-profile)# **stop-threshold** *parameter* 例： `switch(config-pkt-drop-profile)# stop-threshold 256`	stop-threshold パラメータを設定します。ここで、*parameter* はバイト単位のパラメータです。
ステップ 5	switch(config-pkt-drop-profile)# **interface <if_list> {[qos-group <ucastqos-grp>]}** 例： `switch(config-pkt-drop-profile)# interface ethernet 1/1 qos-grop 1`	キャプチャ範囲のパラメータを設定します。

一時キャプチャ バッファのグローバル パラメータ

TCB 設定レベルに移動するには、次のコマンドを実行します。

```
switch(config)# hardware profile packet-drop
switch(config-pkt-drop)#
```

次のオプションは、このレベルで使用できます。

オプション	目的
count	キャプチャされるトランザクション数を設定します。これは省略可能なパラメータです。
drop-trigger	drop-trigger パラメータを設定します。
no	コマンドを無効にします。
profile	パケット ドロップ プロファイルの情報を提供します。
sampling-rate	sampling-rate パラメータを設定します。これは省略可能なパラメータです。
show	実行中のシステム情報を表示します。
shutdown	一時キャプチャ バッファを有効にします。
source	パケット ドロップ範囲を設定します。
timer	パケット ドロップ タイマー パラメータを設定します。これは省略可能なパラメータです。
end	EXEC モードに移行します。
exit	コマンド インタープリタを終了します。

オプション	目的
pop	スタックからモードをポップするか、名前から復元します。
push	現在のモードをスタックにプッシュするか、名前で保存します。
where	どの CLI コンテキストにいるかを表示します。

一時キャプチャ バッファ トリガー イベントの設定

ステート マシンが循環バッファで修飾セットをキャプチャできるようにするトリガー イベントを指定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCB を設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **drop-trigger** *trigger-event*	ステート マシンが循環バッファで修飾セットをキャプチャできるようにするトリガーイベントを設定します。ここで、*trigger-event* は次のいずれかです。 • **egress-admission**：出力アドミッション ドロップ。 • **ingress-admission**：入力アドミッション ドロップ。 • **wred**：重み付けランダム早期廃棄ドロップ。

一時キャプチャ バッファ サンプリング レートの設定

ドロップの前後にキャプチャする必要があるパケットのサンプリング レートを追加できます。これは省略可能なパラメータです。

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCB を設定できるレベルに移動します。

	コマンドまたはアクション	目的
ステップ 2	switch(config-pkt-drop)# **sampling-rate pre-trigger** *pre-trig-params* **post-trigger** *post-trig-params* 例： `switch(config-pkt-drop)# sampling-rate pre-trigger 11 post-trigger 12`	ドロップの前後にキャプチャする必要があるパケットのサンプリング レートを追加します。 • *pre-trig-params*：16 のサンプルから、ドロップの前にキャプチャするトランザクションの数を指定します。有効なオプションは 1 ～ 16 です。 • *post-trig-params*：16 のサンプルから、ドロップの後にキャプチャするトランザクションの数を指定します。有効なオプションは 1 ～ 16 です。

一時キャプチャ バッファ タイマーの設定

期限が切れるとステートマシンが凍結になり、バッファの開始までのポインタがソフトウェアに通知される、TCBタイマー間隔を設定することができます。これは省略可能なパラメータです。

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCBを設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **timer** *timer*	タイマー間隔を設定します。ここで、*timer* はマイクロ秒（usec）単位のキャプチャ タイマー間隔です。有効なオプションはスイッチによって異なります。 • Cisco Nexus 3132C-Z スイッチの場合、キャプチャ タイマー間隔の有効なオプションは 1 ～ 429 です。 • Cisco Nexus 3264C-E スイッチの場合、キャプチャ タイマー間隔の有効なオプションは 1 ～ 385 です。

一時キャプチャ バッファ キャプチャ数の設定

ドロップ後にキャプチャするトランザクションの最小数を設定できます。これに達するとステートマシンが凍結になり、バッファの開始までのポインタがソフトウェアに通知されます。これは省略可能なパラメータです。

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **hardware profile packet-drop**	TCBを設定できるレベルに移動します。
ステップ 2	switch(config-pkt-drop)# **count** *transactions*	ドロップ後にキャプチャするトランザクションの最小数を設定します。ここで、*transactions* は 2 ～ 1024 です。

一時キャプチャ バッファ設定の確認

TCB の実行コンフィギュレーションの確認

TCB の実行コンフィギュレーションを表示するには、**show running-config ipqos** コマンドを使用します。出力は、設定した TCB 範囲とエンティティ設定によって異なります。

- 入力範囲とエンティティ設定では、次のような出力が表示されます。

```
switch# show running config ipqos
hardware profile packet-drop
  source ingress interface eth1/9
  timer 300
  count 200
  drop-trigger ingress-admission
  profile arvinth
    start-threshold 1500
    stop-threshold 1000
    interface Ethernet1/49 qos-group 2
    interface Ethernet1/49 qos-group 0
  sampling-rate pre-trigger 10 post-trigger 10
  no shutdown
```

- 出力範囲とエンティティ設定では、次のような出力が表示されます。

```
switch# show running config ipqos
hardware profile packet-drop
  source egress interface eth1/49
  timer 300
  count 200
  drop-trigger egress-admission
  profile arvinth
    start-threshold 1500
    stop-threshold 1000
```

```
    interface Ethernet1/49 qos-group 2
    interface Ethernet1/49 qos-group 0
  no shutdown
```

- ユニキャスト範囲とエンティティ設定では、次のような出力が表示されます。

```
switch# show running config ipqos
hardware profile packet-drop
  source unicast-queue interface Ethernet1/49 qos-group 0
  timer 300
  count 200
  drop-trigger ingress-admission
  sampling-rate pre-trigger 10 post-trigger 10
  no shutdown
```

パケット ドロップ情報の確認

TCB のパケット ドロップ情報を表示するには、**show hardware profile packet-drop** *option* コマンドを使用します。ここで、*option* は次のとおりです。

- **data**：パケット ドロップの循環バッファのデータを表示します。
- **event**：パケット ドロップのイベントバッファのデータを表示します。
- **status**：パケット ドロップのステータスを表示します。

以下に、さまざまなコマンドオプションでパケット ドロップの情報を表示する例を示します。

- **show hardware profile packet-drop data** を使用してキャプチャされたデータの例を次に示します（以下の出力例は実際の完全な出力のスニペットです）。

```
switch# show hardware profile packet-drop data
Details of Instance : 1
=========================
Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3362736 bytes
Payload :
18809011ad5701a10088106096804528lea000040637d4961a8971a803c03c0000000050200277100012345678 9abcdef101112131415

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3362736 bytes
Payload :
18809011ad5701a10088106096804528lea000040637d4961a8971a803c03c0000000050200277100012345678 9abcdef101112131415

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3362736 bytes
Payload :
18809011ad5701a10088106096804528lea000040637d4961a8971a803c03c0000000050200277100012345678 9abcdef101112131415

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3362736 bytes
Payload :
18809011ad5701a10088106096804528lea000040637d4961a8971a803c03c0000000050200277100012345678 9abcdef101112131415
```

- **show hardware profile packet-drop data instance** *instance-number* を使用してキャプチャされたデータの例を次に示します。ここで、*instance-number* は 1 ～ 5 の値です。

```
switch# show hardware profile packet-drop data instance 1
Details of Instance : 1
=========================
Fri Apr 30 10-59-16 1971 , Src_port : Ethernet1/9
Dst_port : Ethernet1/49 , Qos-group : 0 , Queue_depth : 3440112 bytes
Payload :
18809011b0b1010940028045006edee900fffd8b7faaa11e1e1e1000000000000000000000000000000000000000000000000000
```

- **show hardware profile packet-drop event** を使用してキャプチャされたデータの例を次に示します（以下の出力例は実際の完全な出力のスニペットです）。

```
switch# show hardware profile packet-drop event
Details of Instance : 1
=========================
Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3375216 bytes, Drop_reason :
 Egress-Admission

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3375216 bytes, Drop_reason :
 Egress-Admission

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3375216 bytes, Drop_reason :
 Egress-Admission

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3375216 bytes, Drop_reason :
 Egress-Admission

Src_port : Ethernet1/10
Dst_port : Ethernet1/1 , Qos-group : 1 , Queue_depth : 3375216 bytes, Drop_reason :
 Egress-Admission
```

- **show hardware profile packet-drop event instance** *instance-number* を使用してキャプチャされたデータの例を次に示します。ここで、*instance-number* は 1 ～ 5 の値です。

```
switch# show hardware profile packet-drop event instance 1
Details of Instance : 1
=========================
Fri Apr 30 20-57-24 1971 , Src_port : Ethernet1/9
Dst_port : Ethernet1/49 , Qos-group : 0 ,  Queue_depth : 3452592 bytes, Drop_reason
 : EADMIN
```

- **show hardware profile packet-drop status** を使用してキャプチャされたデータの例を次に示します。

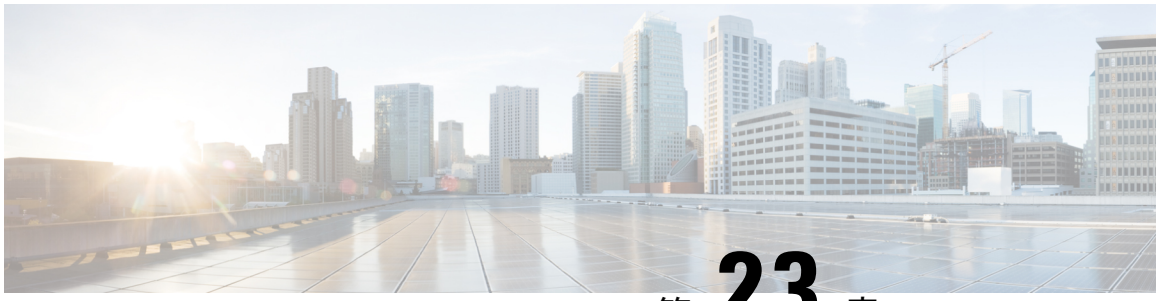
```
switch# show hardware profile packet-drop status
TCB Enabled : FALSE
TCB State : IDLE
Capture Scope : ingress
Drop Trigger : wred
Capture Transactions : 304
Capture Timer : 385
```

一時キャプチャ バッファ情報のクリア

パケットドロップ データ/イベント情報のすべてのインスタンスをクリアするには、このセクションの情報を使用します。

手順

	コマンドまたはアクション	目的
ステップ 1	switch(config)# **clear hardware profile packet-drop file_instance**	



第 23 章

グレースフル挿入と削除の設定

この章では、Cisco Nexus 3000 シリーズ スイッチでグレースフル挿入と削除（GIR）を設定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。

グレースフル挿入と削除について

グレースフル挿入と削除を使用してスイッチを正常に取り出し、そのスイッチをネットワークから分離して、デバッグ操作やアップグレード操作を実行することができます。スイッチは、最小限のトラフィックの中断だけで、通常の転送パスから取り外されます。デバッグ操作やアップグレード操作の実行が終了したら、グレースフル挿入を使用して、そのスイッチを完全な運用（通常）モードに戻すことができます。

グレースフル削除では、すべてのプロトコルと vPC ドメインが正常に停止し、スイッチはネットワークから分離されます。グレースフル挿入では、すべてのプロトコルと vPC ドメインが復元されます。

次のプロトコルは、IPv4 と IPv6 両方のアドレス ファミリでサポートされます。

- Border Gateway Protocol（BGP）
- Enhanced Interior Gateway Routing Protocol（EIGRP）

- Intermediate System-to-Intermediate System（ISIS）
- Open Shortest Path First（OSPF）
- Protocol Independent Multicast（PIM）
- Routing Information Protocol（RIP）

（注） グレースフル挿入と削除の場合、PIMプロトコルはvPC環境にのみ適用できます。グレースフル削除の間、vPC転送ロールがマルチキャストトラフィックのすべてのノースバウンド送信元に対する vPC ピアに転送されます。

プロファイル

デフォルトでは、すべての有効なプロトコルは、グレースフル削除中に分離され、グレースフル挿入時に復元されます。プロトコルは、定義済みの順序で分離および復元されます。

プロトコルを個別に分離、シャットダウン、または復元する（あるいは追加の設定を実施する）場合は、グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、プロファイルを作成できます。ただし、プロトコルの順序が正しいことを確認し、すべての依存関係を考慮する必要があります。

スイッチは、次のプロファイルをサポートしています。

- メンテナンス モード プロファイル：スイッチがメンテナンス モードになったときに、グレースフル削除中に実行されるすべてのコマンドが含まれます。
- 通常モードプロファイル：スイッチが通常モードに戻ったときに、グレースフル挿入中に実行されるすべてのコマンドが含まれます。

プロファイルでは、次のコマンド（および任意の設定コマンド）がサポートされています。

（注） ルーティング プロトコル インスタンスまたはメンテナンスモード プロファイルで **shutdown** と **isolate** の両方が設定されている場合、**shutdown** コマンドが優先されます。

コマンド	説明
isolate	プロトコルをスイッチから分離し、プロトコルをメンテナンスモードにします。
no isolate	プロトコルを復元し、プロトコルを通常モードにします。

コマンド	説明
shutdown	プロトコルまたは vPC ドメインをシャットダウンします。
no shutdown	プロトコルまたは vPC ドメインを起動します。
system interface shutdown [**exclude fex-fabric**]	システム インターフェイスをシャットダウンします（管理インターフェイスを除く）。
no system interface shutdown [**exclude fex-fabric**]	システム インターフェイスを起動します。
sleep instance *instance-number seconds*	指定の秒数だけコマンドの実行を遅延させます。コマンドの複数のインスタンスを遅延できます。 *instance-number* および *seconds* 引数の範囲は、0 ～ 2177483647 です。
python instance *instance-number uri* [*python-arguments*] 例：**python instance 1 bootflash://script1.py**	Python スクリプトの呼び出しをプロファイルに設定します。コマンドの複数の呼び出しをプロファイルに追加できます。 Python 引数には最大 32 文字の英数字を入力できます。

（注） Cisco NX-OS リリース 9.3(5) 以降、**isolate** コマンドは **include-local** オプションとともに提供されます。これは、**router bgp** にのみ適用されます。

このオプションを使用すると、BGP はピアからすべてのルートを取り消します。このオプションを使用しない場合、BGP はリモートで学習したルートのみを撤回し、集約、注入、ネットワーク、再頒布などのローカルで生成されたルートは、eBGP ピアへの最大の Multi-Exit Discriminator (MED) と iBGP ピアへの最小のローカル プリファレンスで引き続きアドバタイズされます。

スナップショット

Cisco NX-OS では、スナップショットは選択した機能の実行状態をキャプチャし、永続ストレージ メディアに保存するプロセスです。

スナップショットは、グレースフル削除前とグレースフル挿入後のスイッチの状態を比較する場合に役立ちます。スナップショット プロセスは、次の 3 つの部分で構成されます。

- 事前に選択したスイッチの一部機能の状態のスナップショットを作成し、永続ストレージメディアに保存する
- さまざまな時間間隔で取得したスナップショットを一覧にして、管理する
- スナップショットを比較して、機能間の相違を表示する

メンテナンス モード（GIR）のワークフロー

グレースフル挿入と削除（GIR）のワークフローを完了する手順は、次のとおりです。

1. （任意）メンテナンス モード プロファイルを作成します（メンテナンス モード プロファイルの設定 （343 ページ） を参照）。
2. （任意）通常モード プロファイルを作成します（通常モード プロファイルの設定 （345 ページ） を参照）。
3. グレースフル削除をトリガーする前のスナップショットを取得します（スナップショットの作成 （346 ページ） を参照）。
4. グレースフル削除をトリガーして、スイッチをメンテナンスモードにします（グレースフル削除のトリガー （350 ページ） を参照）。
5. グレースフル挿入をトリガーして、スイッチを通常モードに戻します（グレースフル挿入のトリガー （352 ページ） を参照）。
6. グレースフル挿入をトリガーした後のスナップショットを取得します（スナップショットの作成 （346 ページ） を参照）。
7. show snapshots compare コマンドを使用して、グレースフル削除と挿入の前後のスイッチの運用データを比較して、すべてが想定どおりに動作していることを確認します（GIR 設定の確認 （355 ページ） を参照）。

プロファイル

デフォルトでは、すべての有効なプロトコルは、グレースフル削除中に分離され、グレースフル挿入時に復元されます。プロトコルは、定義済みの順序で分離および復元されます。

プロトコルを個別に分離、シャットダウン、または復元する（あるいは追加の設定を実施する）場合は、グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、プロファイルを作成できます。ただし、プロトコルの順序が正しいことを確認し、すべての依存関係を考慮する必要があります。

スイッチは、次のプロファイルをサポートしています。

- メンテナンス モード プロファイル：スイッチがメンテナンス モードになったときに、グレースフル削除中に実行されるすべてのコマンドが含まれます。

- 通常モードプロファイル：スイッチが通常モードに戻ったときに、グレースフル挿入中に実行されるすべてのコマンドが含まれます。

プロファイルでは、次のコマンド（および任意の設定コマンド）がサポートされています。

コマンド	説明
isolate	プロトコルをスイッチから分離し、プロトコルをメンテナンスモードにします。
no isolate	プロトコルを復元し、プロトコルを通常モードにします。
shutdown	プロトコルをシャットダウンします。
no shutdown	プロトコルを起動します。
system interface shutdown [**exclude fex-fabric**]	システム インターフェイスをシャットダウンします（管理インターフェイスを除く）。
no system interface shutdown [**exclude fex-fabric**]	システム インターフェイスを起動します。
sleep instance *instance-number seconds*	指定の秒数だけコマンドの実行を遅延させます。コマンドの複数のインスタンスを遅延できます。 *instance-number* および *seconds* 引数の範囲は、0 ～ 2177483647 です。
python instance *instance-number uri* [*python-arguments*] 例：**python instance 1 bootflash://script1.py**	Python スクリプトの呼び出しをプロファイルに設定します。コマンドの複数の呼び出しをプロファイルに追加できます。 Python 引数には最大 32 文字の英数字を入力できます。

メンテナンス モード プロファイルの設定

グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、メンテナンス モード プロファイルを作成できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure maintenance profile maintenance-mode**
	例：
	`switch# configure maintenance profile maintenance-mode`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch(config-mm-profile)#`
		メンテナンス モード プロファイルのコンフィギュレーション セッションを開始します。
		設定しているプロトコルに応じて、プロトコルを停止する適切なコマンドを入力する必要があります。サポートされるコマンドの一覧については、プロファイル（342 ページ）を参照してください。

ステップ 2	**end**
	例：
	`switch(config-mm-profile)# end`
	`switch#`
		メンテナンス モード プロファイルを終了します。

ステップ 3	**show maintenance profile maintenance-mode** 例： `switch# show maintenance profile maintenance-mode`	メンテナンス モード プロファイルの詳細を表示します。

例

次に、メンテナンス モード プロファイルを作成する例を示します。

```
switch# configure maintenance profile maintenance-mode
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# shutdown
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# shutdown
switch(config-mm-profile)# system interface shutdown
switch(config-mm-profile)# end
Exit maintenance profile mode.
switch# show maintenance profile maintenance-mode
[Maintenance Mode]
router bgp 100
  shutdown
router eigrp 10
  shutdown
  address-family ipv6 unicast
    shutdown
system interface shutdown
```

通常モード プロファイルの設定

グレースフル削除またはグレースフル挿入時に適用できる設定コマンドを使用して、通常モード プロファイルを作成できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure maintenance profile normal-mode**
	例：
	`switch# configure maintenance profile normal-mode`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`switch(config-mm-profile)#`
		通常モードプロファイルのコンフィギュレーション セッションを開始します。
		設定しているプロトコルに応じて、プロトコルを起動する適切なコマンドを入力する必要があります。サポートされるコマンドの一覧については、プロファイル（342 ページ）を参照してください。

ステップ 2	**end**
	例：
	`switch(config-mm-profile)# end`
	`switch#`
		通常モードプロファイルを終了します。

ステップ 3	**show maintenance profile normal-mode** 例： `switch# show maintenance profile normal-mode`	通常モード プロファイルの詳細を表示します。

例

次に、メンテナンス モード プロファイルを作成する例を示します。

```
switch# configure maintenance profile normal-mode
switch(config-mm-profile)# no system interface shutdown
switch(config-mm-profile)# router eigrp 10
switch(config-mm-profile-router)# no shutdown
switch(config-mm-profile-router)# address-family ipv6 unicast
switch(config-mm-profile-router-af)# no shutdown
switch(config-mm-profile)# router bgp 100
switch(config-mm-profile-router)# no shutdown
switch(config-mm-profile-router)# end
Exit maintenance profile mode.
switch# show maintenance profile normal-mode
[Normal Mode]
no system interface shutdown
router eigrp 10
  no shutdown
  address-family ipv6 unicast
    no shutdown
router bgp 100
```

```
no shutdown
```

スナップショットの作成

選択した機能の実行状態のスナップショットを作成できます。

手順

	コマンドまたはアクション	目的

ステップ 1	**snapshot create** *snapshot-name description*
	例：
	`switch# snapshot create snap_before_maintenance Taken before maintenance`
	`Executing 'show interface'... Done`
	`Executing 'show ip route summary vrf all'... Done`
	`Executing 'show ipv6 route summary vrf all'... Done`
	`Executing 'show bgp sessions vrf all'... Done`
	`Executing 'show ip eigrp topology summary'... Done`
	`Executing 'show ipv6 eigrp topology summary'... Done`
	`Feature 'vpc' not enabled, skipping...`
	`Executing 'show ip ospf vrf all'... Done`
	`Feature 'ospfv3' not enabled, skipping...`
	`Feature 'isis' not enabled, skipping...`
	`Feature 'rip' not enabled, skipping...`
	`Snapshot 'snap_before_maintenance' created`
		選択した機能の実行状態または運用データをキャプチャし、データを永続ストレージ メディアに保存します。
		最大 64 文字の英数字のスナップショット名と最大 254 文字の英数字の説明を入力できます。
		すべてのスナップショットまたは特定のスナップショットを削除するには、**snapshot delete** {**all** | *snapshot-name*} コマンドを使用します。

ステップ 2	**show snapshots**
	例：
	`switch# show snapshots`
	`Snapshot Name Time Description`
	`---`
	`snap_before_maintenance Wed Aug 19 13:53:28 2015 Taken before maintenance`
		スイッチ上に存在するスナップショットを表示します。

ステップ 3	**show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**] 例： `switch# show snapshots compare snap_before_maintenance snap_after_maintenance`	2 つのスナップショットの比較を表示します。 **summary** オプションは、2 つのスナップショット間の全体的な変更を確認するのに十分な情報のみ表示します。

	コマンドまたはアクション	目的
		ipv4routes および **ipv6routes** オプションは、2 つのスナップショット間の IPv4 および IPv6 ルートの変更を表示します。

例

次に、2 つのスナップショット間の変更の概要の例を示します。

```
switch# show snapshots compare snapshot1 snapshot2 summary
feature                             snapshot1       snapshot2       changed
basic summary
    # of interfaces                 16              12              *
    # of vlans                      10              4               *
    # of ipv4 routes                33              3               *
    ……

interfaces
    # of eth interfaces             3               0               *
    # of eth interfaces up          2               0               *
    # of eth interfaces down        1               0               *
    # of eth interfaces other       0               0

    # of vlan interfaces            3               1               *
    # of vlan interfaces up         3               1               *
    # of vlan interfaces down       0               0
    # of vlan interfaces other      0               1               *
    ……
```

次に、2 つのスナップショット間の IPv4 ルートの変更の例を示します。

```
switch# show snapshots compare snapshot1 snapshot2 ipv4routes
metric                          snapshot1       snapshot2       changed
# of routes                     33              3               *
# of adjacencies                10              4               *

Prefix              Changed Attribute
------              -----------------
23.0.0.0/8          not in snapshot2
10.10.10.1/32       not in snapshot2
21.1.2.3/8          adjacency index has changed from 29 (snapshot1) to 38 (snapshot2)
……

There were 28 attribute changes detected
```

スナップショットへの show コマンドの追加

スナップショットでキャプチャされる追加の **show** コマンドを指定できます。それらの **show** コマンドは、ユーザ指定のスナップショット セクションで定義されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**snapshot section add** *section "show-command" row-id element-key1* [*element-key2*] 例： `switch# snapshot section add myshow “show ip interface brief” ROW_intf intf-name`	ユーザ指定のセクションをスナップショットに追加します。*section* は、**show** コマンドの出力に名前を付けるために使用されます。任意の単語を使用して、セクションに名前を付けることができます。 **show** コマンドは、引用符で囲む必要があります。**show** 以外のコマンドは拒否されます。 *row-id* 引数では、**show** コマンドの XML 出力の各行エントリのタグを指定します。*element-key1* および *element-key2* 引数では、行エントリ間を区別するために使用されるタグを指定します。ほとんどの場合、行エントリ間を区別するために指定する必要があるのは *element-key1* 引数だけです。 （注） スナップショットからユーザ指定のセクションを削除するには、**snapshot section delete** *section* コマンドを使用します。
ステップ 2	**show snapshots sections** 例： `switch# show snapshots sections`	ユーザー指定のスナップショット セクションを表示します。
ステップ 3	**show snapshots compare** *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**] 例： `switch# show snapshots compare snap1 snap2`	2 つのスナップショットの比較を表示します。 **summary** オプションは、2 つのスナップショット間の全体的な変更を確認するのに十分な情報のみ表示します。 **ipv4routes** および **ipv6routes** オプションは、2 つのスナップショット間の IPv4 および IPv6 ルートの変更を表示します。

例

次に、**show ip interface brief** コマンドを myshow スナップショット セクションに追加する例を示します。この例では、2 つのスナップショット（snap1 および snap2）が比較され、両方のスナップショットにユーザ指定のセクションが表示されます。

```
switch# snapshot section add myshow "show ip interface brief" ROW_intf intf-name
switch# show snapshots sections
user-specified snapshot sections
--------------------------------
[myshow]
  cmd:  show ip interface brief
  row:  ROW_intf
  key1: intf-name
  key2: -

[sect2]
  cmd:  show ip ospf vrf all
  row:  ROW_ctx
  key1: instance_number
  key2: cname

switch# show snapshots compare snap1 snap2
================================================================================
Feature              Tag                  snap1                snap2
================================================================================

[bgp]
--------------------------------------------------------------------------------
.............

[interface]
--------------------------------------------------------------------------------

       [interface:mgmt0]
                     vdc_lvl_in_pkts      692310             **692317**
                     vdc_lvl_in_mcast     575281             **575287**
                     vdc_lvl_in_bcast     77209              **77210**
                     vdc_lvl_in_bytes     63293252           **63293714**
                     vdc_lvl_out_pkts     41197              **41198**
                     vdc_lvl_out_ucast    33966              **33967**
                     vdc_lvl_out_bytes    6419714            **6419788**
.............

[ospf]
--------------------------------------------------------------------------------
.............

[myshow]
--------------------------------------------------------------------------------

       [interface:Ethernet1/1]
                     state                up                 **down**
                     admin_state          up                 **down**
.............
```

グレースフル削除のトリガー

デバッグ操作やアップグレード操作を実行するために、スイッチのグレースフル削除をトリガーして、スイッチを取り出し、ネットワークからそのスイッチを分離できます。

始める前に

作成するメンテナンスモードプロファイルをシステムに使用させる場合は、メンテナンスモード プロファイルの設定 （343 ページ）を参照してください。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		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	**system mode maintenance** [**dont-generate-profile** | **timeout** *value* | **shutdown** | **on-reload reset-reason** *reason*]
	例：
	`switch(config)# system mode maintenance`
	`Following configuration will be applied:`
	
	`router bgp 65502`
	`isolate`
	`router ospf p1`
	`isolate`
	`router ospfv3 p1`
	`isolate`
	
	`Do you want to continue (y/n)? [no] y`
	
	`Generating a snapshot before going into maintenance mode`
	
	`Starting to apply commands...`
	
	`Applying : router bgp 65502`
	`Applying : isolate`
	`Applying : router ospf p1`
	`Applying : isolate`
	`Applying : router ospfv3 p1`
	`Applying : isolate`
	
	`Maintenance mode operation successful.`
		すべての有効なプロトコルをメンテナンス モードにします（**isolate** コマンドを使用）。
		次のオプションを使用できます。
		• **dont-generate-profile**：有効なプロトコルの動的な検索が回避され、メンテナンス モード プロファイルに設定されているコマンドが実行されます。作成したメンテナンス モード プロファイルをシステムに使用させる場合は、このオプションを使用します。
		• **timeout** *value*：指定した分数の間、スイッチをメンテナンス モードのままにします。範囲は 5 ～ 65535 です。設定した時間が経過すると、スイッチは自動的に通常モードに戻ります。**no system mode maintenance timeout** コマンドは、タイマーを無効にします。
		• **shutdown**：すべてのプロトコルおよび管理インターフェイスを除くインターフェイスをシャットダウンします（**shutdown** コマンドを使

	コマンドまたはアクション	目的
		用）。このオプションを指定すると中断が発生しますが、デフォルト（**isolate** コマンドを使用）の場合、中断は発生しません。 • **on-reload reset-reason** *reason*：指定されているシステム クラッシュが発生した場合、スイッチは自動的にメンテナンスモードで起動します。**no system mode maintenance on-reload reset-reason** コマンドを使用すると、システム クラッシュ時にスイッチがメンテナンス モードで起動するのを回避できます。 メンテナンス モードのリセット理由は次のとおりです。 • HW_ERROR：ハードウェア エラー • SVC_FAILURE：重大なサービス障害 • KERN_FAILURE：カーネル パニック • WDOG_TIMEOUT：ウォッチドッグ タイムアウト • FATAL_ERROR：致命的なエラー • LC_FAILURE：ライン カード障害 • MATCH_ANY：上記のいずれかの理由 続行を促すプロンプトが表示されます。続行する場合は**y**、プロセスを終了する場合は **n** を入力します。

ステップ 3	（任意） **show system mode**
	例：
	`switch(config)# show system mode`
	`System Mode: Maintenance`
		現在のシステム モードを表示します。
		スイッチはメンテナンス モードになっています。スイッチに対する目的のデ

	コマンドまたはアクション	目的
		バッグ操作やアップグレード操作を実行できます。
ステップ 4	（任意） **copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。このコマンドは、再起動後にメンテナンス モードを維持する場合に必要です。

例

次に、スイッチのすべてのプロトコルおよびインターフェイスをシャットダウンする例を示します。

```
switch(config)# system mode maintenance shutdown

Following configuration will be applied:

   router bgp 65502
     shutdown
   router ospf p1
     shutdown
   router ospfv3 p1
     shutdown
   system interface shutdown

Do you want to continue (y/n)? [no] y

Generating a snapshot before going into maintenance mode

Starting to apply commands...

Applying : router bgp 65502
Applying :   shutdown
Applying : router ospf p1
Applying :   shutdown
Applying : router ospfv3 p1
Applying :   shutdown

Maintenance mode operation successful.
```

次に、致命的なエラーが発生した場合に、スイッチを自動的にメンテナンス モードで起動する例を示します。

```
switch(config)# system mode maintenance on-reload reset-reason fatal_error
```

グレースフル挿入のトリガー

デバッグ操作やアップグレード操作の実行が終了したら、グレースフル挿入をトリガーして、すべてのプロトコルを復元できます。

始める前に

作成する通常モード プロファイルをシステムに使用させる場合は、メンテナンス モード プロファイルの設定 （343 ページ）を参照してください。

手順

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	例：
	`switch# configure terminal`
	`switch(config)#`
		グローバル設定モードを開始します。

ステップ 2	**no system mode maintenance** [**dont-generate-profile**]
	例：
	`switch(config)# no system mode maintenance dont-generate-profile`
	`Following configuration will be applied:`
	`router bgp 65502`
	`no isolate`
	`router ospf p1`
	`no isolate`
	`router ospfv3 p1`
	`no isolate`
	`Do you want to continue (y/n)? [no] y`
	`Starting to apply commands...`
	`Applying : router bgp 65502`
	`Applying : no isolate`
	`Applying : router ospf p1`
	`Applying : no isolate`
	`Applying : router ospfv3 p1`
	`Applying : no isolate`
	`Maintenance mode operation successful.`
	`Generating Current Snapshot`
		すべての有効なプロトコルを通常モードにします（**no isolate** コマンドを使用）。
		dont-generate-profile オプションを指定すると、有効なプロトコルの動的な検索が回避され、通常モード プロファイルに設定されているコマンドが実行されます。作成した通常モード プロファイルをシステムに使用させる場合は、このオプションを使用します。
		続行を促すプロンプトが表示されます。続行する場合は**y**、プロセスを終了する場合は **n** を入力します。

ステップ 3	（任意） **show system mode**
	例：
	`switch(config)# show system mode`
	`System Mode: Normal`
		現在のシステム モードを表示します。スイッチは通常モードになっていて、完全に機能しています。

メンテナンス モードの強化

リリース 7.0(3)I5(1) 以降、メンテナンス モードの次の機能拡張が Cisco Nexus 3000 シリーズ スイッチに追加されました。

- システム メンテナンス シャットダウン モードで次のメッセージが追加されます。

  ```
  NOTE: The command system interface shutdown will shutdown all interfaces excluding
  mgmt 0.
  ```

- CLI コマンドを入力すると、**system mode maintenance** によって孤立ポートがチェックされ、アラートが送信されます。

- 隔離モードで vPC が設定されると、次のメッセージが追加されます。

  ```
  NOTE: If you have vPC orphan interfaces, please ensure vpc orphan-port suspend is
  configured under them, before proceeding further.
  ```

- カスタム プロファイル設定：新しい CLI コマンド、**system mode maintenance always-use-custom-profile** がカスタム プロファイル設定に追加されます。新しい CLI コマンド、**system mode maintenance non-interactive** は Cisco Nexus 9000 シリーズ スイッチのみの #ifdef 下に追加されます。

  （メンテナンスまたは通常モードで）カスタムプロファイルを作成すると、次のメッセージが表示されます。

  ```
  Please use the command system mode maintenance always-use-custom-profile
   if you want to always use the custom profile.
  ```

- after_maintenance スナップショットが取得される前に遅延が追加されました。**no system mode maintenance** コマンドは、通常モードのすべての設定が適用され、モードが通常モードに変更され、after_maintenance スナップショットを取得するためのタイマーが開始されると終了します。タイマーの期限が切れると、after_maintenance スナップショットがバックグラウンドで取得され、スナップショットが完了すると新しい警告 Syslog、MODE_SNAPSHOT_DONE が送信されます。

  CLI コマンド **no system mode maintenance** の最終出力は、after_maintenance スナップショットが生成されるタイミングを示します。

  ```
  The after_maintenance snapshot will be generated in <delay> seconds.
  After that time, please use show snapshots compare before_maintenance
  after_maintenance to check the health of the system. The timer delay
  for the after_maintenance snapshot is defaulted to 120 seconds but
  it can be changed by a new configuration command.
  ```

  after_maintenance snapshot のタイマー遅延を変更する新しい設定コマンドは、**system mode maintenance snapshot-delay <seconds>** です。この設定は、デフォルト設定の 120 秒を 0 ～ 65535 の任意の値に上書きします。これは ASCII 設定で表示されます。

  現在のスナップショット遅延の値を表示する新しい show コマンド、**show maintenance snapshot-delay** も追加されています。この新しい show コマンドでは、XML 出力がサポートされています。

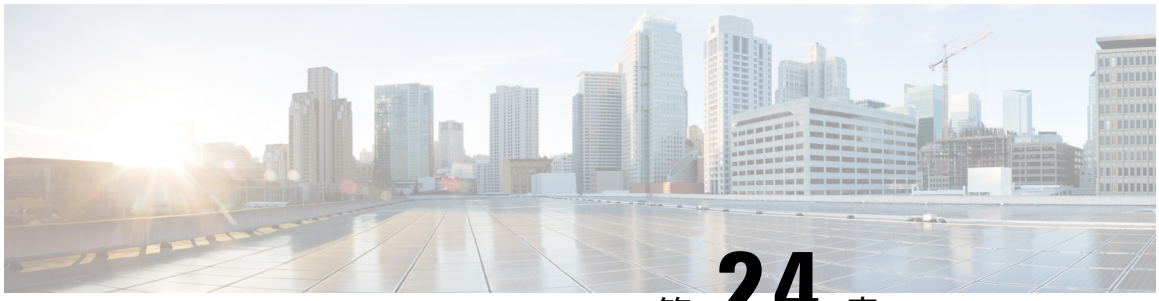
- システムがメンテナンス モードであるときに表示される CLI インジケータが追加されました（例：`switch(m-mode)#`）。
- CLI リロードまたはシステム リセットによってデバイスがメンテナンス モードから通常モードおよびその逆に移行するときの SNMP トラップのサポートが追加されました。**snmp-server enable traps mmode cseMaintModeChangeNotify** トラップは、メンテナンス モードのトラップ通知の変更を有効にするために追加されました。**snmp-server enable traps mmode cseNormalModeChangeNotify** は、通常モードへのトラップ通知の変更を有効にするために追加されました。デフォルトでは両方のトラップが無効になっています。

GIR 設定の確認

GIR の設定を表示するには、次のいずれかの作業を行います。

コマンド	目的
show interface brief	インターフェイスの要約情報を表示します。
show maintenance on-reload reset-reasons	スイッチがメンテナンス モードで起動されることになる、リセット理由を表示します。メンテナンス モードのリセット理由の説明については、グレースフル削除のトリガー （350 ページ）を参照してください。
show maintenance profile [**maintenance-mode** | **normal-mode**]	メンテナンス モードまたは通常モードのプロファイルの詳細を表示します。
show maintenance timeout	メンテナンス モードのタイムアウト期間を表示します。この期間後、スイッチは自動的に通常モードに戻ります。
show {**running-config** | **startup-config**} **mmode** [**all**]	実行コンフィギュレーションまたはスタートアップ コンフィギュレーションのメンテナンス モードのセクションを表示します。**all** オプションには、デフォルト値が含まれます。
show snapshots	スイッチ上に存在するスナップショットを表示します。

コマンド	目的
show snapshots compare *snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]	2 つのスナップショットの比較を表示します。 **summary** オプションは、2 つのスナップショット間の全体的な変更を確認するのに十分な情報のみ表示します。 **ipv4routes** および **ipv6routes** オプションは、2 つのスナップショット間の IPv4 および IPv6 ルートの変更を表示します。
show snapshots dump *snapshot-name*	スナップショットの取得時に生成された各ファイルの内容を表示します。
show snapshots sections	ユーザー指定のスナップショットセクションを表示します。
show system mode	現在のシステム モードを表示します。



第 24 章

ソフトウェア メンテナンス アップグレード（SMU）の実行

この章では、Cisco Nexus 3000 シリーズ スイッチでソフトウェア メンテナンス アップグレード（SMU）を実行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。

SMU について

ソフトウェア メンテナンス アップグレード（SMU）は、特定の障害の修正を含むパッケージ ファイルです。SMU は、直近の問題に対処するために作成され、新しい機能は含まれていません。通常、SMU がデバイスの動作に大きな影響を及ぼすことはありません。SMU のバージョンは、アップグレードするパッケージのメジャー、マイナー、およびメンテナンス バージョンに同期されます。

SMU の影響は次のタイプによって異なります。

- プロセスの再起動 SMU：アクティベーション時にプロセスまたはプロセスのグループの再起動を引き起こします。
- リロード SMU：スーパーバイザおよびライン カードのパラレル リロードを引き起こします。

SMU は、メンテナンス リリースの代わりになるものではありません。直近の問題に対する迅速な解決策を提供します。SMU で修正された障害は、メンテナンス リリースにすべて統合されます。

デバイスを新しい機能やメンテナンス リリースにアップグレードする詳細については、『*Cisco Nexus 3000 Series NX-OS Software Upgrade and Downgrade Guide*』を参照してください。

（注） SMU をアクティブにすると、以前の SMU、または SMU が適用されるパッケージが自動的に非アクティブ化されることはありません。

（注） Cisco NX-OS リリース 7.0(3)I2(1) 以降、SMU パッケージ ファイルの拡張子は .rpm です。以前のファイルの拡張子は .bin です。

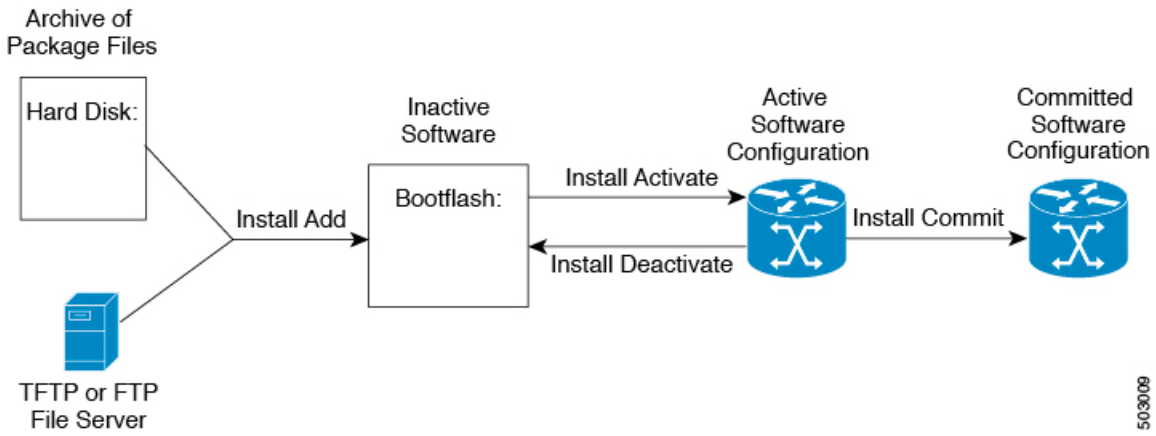
パッケージ管理

デバイスでの SMU パッケージの追加およびアクティブ化の一般的な手順は次のとおりです。

1. パッケージ ファイルをローカル ストレージ デバイスまたはファイル サーバにコピーします。
2. **install add** コマンドを使用してデバイス上でパッケージを追加します。
3. **install activate** コマンドを使用して、デバイス上でパッケージをアクティブ化します。
4. **install commit** コマンドを使用して、現在のパッケージのセットをコミットします。
5. （オプション）パッケージをアクティブでなくし、除去します。

次の図は、パッケージの管理プロセスの主要な手順について説明します。

図 *3: SMU* パッケージを追加、アクティブ化およびコミットするプロセス



SMU の前提条件

アクティブ化または非アクティブ化するパッケージでは、これらの前提条件が満たされている必要があります。

- 適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。
- すべてのライン カードが取り付けられ、正常に動作していることを確認します。たとえば、ライン カードのブート中、ライン カードのアップグレード中または交換中、または自動スイッチオーバーアクティビティが予想される場合は、パッケージのアクティブ化や非アクティブ化はできません。

SMU の注意事項と制約事項

SMU に関する注意事項および制約事項は次のとおりです。

- パッケージによっては、他のパッケージのアクティブ化または非アクティブ化が必要です。SMU に相互に依存関係がある場合は、前の SMU をまずアクティブにしないとそれらをアクティブ化できません。
- アクティブ化するパッケージは、現在のアクティブなソフトウェアのセットと互換性がある必要があります。
- 1 つのコマンドで複数の SMU をアクティブにできません。
- パッケージの互換性が確認できた場合に限り、アクティブ化が実行されます。競合がある場合は、エラー メッセージが表示されます。
- ソフトウェア パッケージをアクティブ化する間、その他の要求はすべての影響のあるノードで実行できません。これと同様のメッセージが表示されると、パッケージのアクティブ化は完了します。

```
Install operation 1 completed successfully at Thu Jan 9 01:19:24 2014
```

- 各 CLI インストール要求には要求 ID が割り当てられます。これは後でイベントを確認するのに使用できます。
- ソフトウェア メンテナンス アップグレードを実行後、デバイスを新しい Cisco Nexus 3000 ソフトウェア リリースにアップグレードする場合、新しいイメージで以前の Cisco Nexus 3000 リリースと SMU パッケージ ファイルの両方が上書きされます。

Cisco NX-OS のソフトウェア メンテナンス アップグレードの実行

パッケージ インストールの準備

SMU パッケージのインストールの準備に関する情報を収集するには、複数の **show** コマンドを使用する必要があります。

始める前に

ソフトウェアの変更が必要かどうかを確認します。

使用中のシステムで新しいパッケージがサポートされていることを確認する。ソフトウェア パッケージによっては、他のパッケージまたはパッケージバージョンをアクティブにする必要があり、特定のライン カードのみをサポートするパッケージもあります。

そのリリースに関連する重要な情報についてリリース ノートを確認し、そのパッケージとデバイス設定の互換性の有無を判断する。

システムの動作が安定していて、ソフトウェアの変更に対応できることを確認する。

手順

	コマンドまたはアクション	目的
ステップ 1	**show install active** 例： `switch# show install active`	デバイス上のアクティブなソフトウェアを表示します。デバイスに追加する必要があるソフトウェアを決定するため、またインストール操作完了後にアクティブなソフトウェアのレポートと比較するために、このコマンドを使用します。
ステップ 2	**show module** 例： `switch# show module`	すべてのモジュールが安定状態であることを確認します。
ステップ 3	**show clock** 例： `switch# show clock`	システム クロックが正しいことを確認します。ソフトウェア操作は、デバイス クロックの時刻に基づいて証明書を使用します。

例

次に、システム全体のアクティブなパッケージを表示する例を示します。この情報を使用して、ソフトウェアの変更が必要かどうかを判断します。

```
switch# show install active
Active Packages:
Active Packages on Module #3:

Active Packages on Module #6:

Active Packages on Module #7:
Active Packages on Module #22:

Active Packages on Module #30:
```

次に、現在のシステム クロックの設定を表示する例を示します。

```
switch# show clock
02:14:51.474 PST Wed Jan 04 2014
```

ローカルストレージデバイスまたはネットワークサーバへのパッケージ ファイルのコピー

デバイスがアクセスできるローカル ストレージ デバイスまたはネットワーク ファイル サーバに SMU パッケージ ファイルをコピーする必要があります。この作業が完了したら、パッケージをデバイスに追加しアクティブにできます。

デバイスにパッケージ ファイルを保存する必要がある場合は、ハード ディスクにファイルを保存することを推奨します。ブートデバイスは、パッケージを追加しアクティブするローカル ディスクです。デフォルトのブート デバイスは bootflash: です。

ヒント　ローカル ストレージ デバイスにパッケージ ファイルをコピーする前に、**dir** コマンドを使用して、必要なパッケージ ファイルがデバイスに存在するかどうかを確認します。

SMU パッケージ ファイルがリモート TFTP、FTP、または SFTP サーバにある場合、ローカル ストレージ デバイスにファイルをコピーできます。ファイルがローカル ストレージ デバイスに置かれた後、パッケージをそのストレージデバイスからデバイスに追加しアクティブにできます。次のサーバ プロトコルがサポートされます。

- TFTP：ネットワークを介して、あるコンピュータから別のコンピュータへファイルを転送できるようにします。通常は、クライアント認証（たとえば、ユーザ名およびパスワード）を使用しません。これは FTP の簡易版です。

（注） パッケージ ファイルによっては、大きさが 32 MB を超える場合もありますが、一部のベンダーにより提供される TFTP サービスではこの大きさのファイルがサポートされていない場合があります。32 MB を超えるファイルをサポートする TFTP サーバにアクセスできない場合は、FTP を使用してファイルをダウンロードします。

- ファイル転送プロトコル：FTP は TCP/IP プロトコル スタックの一部であり、ユーザ名とパスワードが必要です。
- SSH ファイル転送プロトコル：SFTP は、セキュリティ パッケージの SSHv2 機能の一部で、セキュアなファイル転送を提供します。

SMU パッケージ ファイルをネットワーク ファイル サーバまたはローカル ストレージ デバイスに転送した後に、ファイルを追加しアクティブ化することができます。

パッケージの追加とアクティブ化

ローカル ストレージ デバイスまたはリモート TFTP、FTP、SFTP サーバーに保存されている SMU パッケージ ファイルをデバイスに追加できます。

（注） アクティブ化する SMU パッケージは、現在アクティブで動作可能なソフトウェアと互換性がなければなりません。アクティブ化が試行されると、システムは自動互換性チェックを実行し、パッケージがデバイス上でアクティブなその他のソフトウェアと互換性があることを確認します。競合がある場合は、エラー メッセージが表示されます。アクティブ化が実行されるのは、すべての互換性が確認できた場合だけです。

（注） この手順では、Cisco NX-OS CLI コマンドを使用して、RPM パッケージ ファイルを追加して有効化します。YUM コマンドを使用する場合は、『Cisco Nexus 3000 Series NX-OS Programmability Guide』の「Installing RPMs from Bash」の手順に従ってください。

手順

	コマンドまたはアクション	目的
ステップ 1	**install add** *filename* [**activate**] 例： `switch# install add bootflash: nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`	ローカル ストレージ デバイスまたはネットワーク サーバからパッケージ ソフトウェア ファイルを解凍してブートフラッシュおよびデバイスにインストー

	コマンドまたはアクション	目的
		ルされているすべてのアクティブ スーパーバイザおよびスタンバイ スーパーバイザに追加します。 *filename* 引数は、次の形式をとることができます。 • **bootflash:***filename* • **tftp://***hostname-or-ipaddress/directory-path/filename* • **ftp://***username***:***password***@***hostname-or-ipaddress/directory-path/filename* • **sftp://***hostname-or-ipaddress/directory-path/filename*
ステップ 2	（任意） **show install inactive** **例**： `switch# show install inactive`	デバイス上の非アクティブなパッケージを表示します。前述の手順で追加されたパッケージが表示に出ることを確認します。

ステップ 3	必須: **install activate** *filename* [**test**]
	例：
	`switch# install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`
	例：
	`switch# install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm Install operation 1 completed successfully at Wed Mar 16 00:42:12 2016`
	例：
	`switch# install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm Install operation 2 !!WARNING!! This patch will get activated only after a reload of the switch. at Wed Mar 16 00:42:12 2016`
		デバイスに追加されたパッケージをアクティブにします。SMU パッケージは、アクティブにされるまで無効のままです。（**install add activate** コマンドを使用して、パッケージが前にアクティブにされた場合は、この手順を省略します。）
		（注）
		パッケージ名を部分的に入力してから **?** を押すと、アクティブ化に使用できるすべての候補が表示されます。候補が 1 つしかない場合に **Tab** キーを押すと、パッケージ名の残りの部分が自動入力されます。

ステップ 4	すべてのパッケージがアクティブ化されるまで手順 3 を繰り返します。	必要に応じて他のパッケージもアクティブ化します。
ステップ 5	（任意） **show install active** **例**： `switch# show install active`	すべてのアクティブなパッケージを表示します。このコマンドを使用して、正しいパッケージがアクティブであるかどうかを判断します。

アクティブなパッケージ セットのコミット

SMUパッケージがデバイス上でアクティブになると、それは現在の実行コンフィギュレーションの一部になります。パッケージのアクティブ化をシステム全体のリロード間で持続させるには、デバイス上でパッケージをコミットする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**install commit** *filename* 例： `switch# install commit nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`	現在のパッケージのセットをコミットして、デバイスが再起動したときにこれらのパッケージが使用されるようにします。
ステップ 2	（任意） **show install committed** 例： `switch# show install committed`	コミットされたパッケージを表示します。

パッケージの非アクティブ化と削除

パッケージを非アクティブ化すると、そのデバイスではアクティブではなくなりますが、パッケージ ファイルはブート ディスクに残ります。パッケージ ファイルは、後で再アクティブ化できます。また、ディスクから削除もできます。

（注） この手順では、Cisco NX-OS CLI コマンドを使用して、RPM パッケージ ファイルを非アクティブ化して削除します。YUM コマンドを使用する場合は、『Cisco Nexus 3000 Series NX-OS Programmability Guide』の「Erasing an RPM」の手順に従ってください。

手順

	コマンドまたはアクション	目的
ステップ 1	**install deactivate** *filename* 例： `switch# install deactivate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`	デバイスに追加されたパッケージを非アクティブ化し、ラインカードのパッケージ機能をオフにします。 （注） パッケージ名を部分的に入力してから **?** を押すと、非アクティブ化に使用できるすべての候補が表示されます。候補が 1 つしかない場合に **Tab** キーを押す

	コマンドまたはアクション	目的
		と、パッケージ名の残りの部分が自動入力されます。
ステップ 2	（任意） **show install inactive** **例**： `switch# show install inactive`	デバイス上の非アクティブなパッケージを表示します。
ステップ 3	（任意） **install commit** **例**： `switch# install commit`	現在のパッケージのセットをコミットして、デバイスが再起動したときにこれらのパッケージが使用されるようにします。 **（注）** パッケージを削除できるのは、非アクティブ化操作がコミットされた場合だけです。

ステップ 4	（任意） **install remove** {*filename* | **inactive**}
	例：
	`switch# install remove nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm`
	`Proceed with removing nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm? (y/n)? [n] y`
	例：
	`switch# install remove inactive`
	`Proceed with removing? (y/n)? [n] y`
		非アクティブなパッケージを削除します。
		• 削除できるのは非アクティブなパッケージだけです。
		• パッケージは、デバイスのすべてのライン カードから非アクティブにされた場合にのみ削除できます。
		• パッケージの非アクティブ化はコミットする必要があります。
		• ストレージ デバイスから特定の非アクティブなパッケージを削除するには、**install remove** コマンドに *filename* 引数を指定して使用します。
		• システムのすべてのノードから非アクティブなパッケージをすべて削除するには、**install remove** コマンドと **inactive** キーワードを使用します。

機能 RPM のダウングレード

インストールされている機能RPMを基本機能RPMにダウングレードするには、この手順を実行します。

（注） この手順では、Cisco NX-OS CLI コマンドを使用して、機能 RPM をダウングレードします。YUM コマンドを使用する場合は、『Cisco Nexus 3000 Series NX-OS Programmability Guide』の「Downgrading an RPM」の手順に従ってください。

手順

	コマンドまたはアクション	目的

ステップ 1	（任意） **show install packages**
	例：
	`switch# show install packages`
	`ntp.lib32_n9000 1.0.1-7.0.3.I2.2e installed`
		デバイス上の機能 RPM パッケージを表示します。

ステップ 2	必須: **run bash**
	例：
	`switch# run bash`
	`bash-4.2$`
		Bash をロードします。

ステップ 3	必須: **ls ****feature****
	例：
	`bash-4.2$ ls *ntp*`
	`ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm`
		指定された機能の RPM を一覧表示します。

ステップ 4	必須: **cp** *filename* **/bootflash** 例： `bash-4.2$ cp ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm /bootflash`	基本機能 RPM をブートフラッシュにコピーします。
ステップ 5	必須: **exit** 例： `bash-4.2$ exit`	Bash を終了します。

ステップ 6	必須: **install add bootflash:***filename* **activate downgrade**
	例：
	`switch# install add bootflash:ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm activate downgrade`
	`Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [############ ] 60%`
	`Adding the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) [####################] 100%`
		機能 RPM をダウングレードします。
		（注）
		デバイスのリロードを要求されたら、**Y** を入力します。リロードは、NTP および SNMP 機能 RPM をダウングレードする場合にのみ必要です。

	コマンドまたはアクション	目的
	Install operation 11 completed successfully at Thu Sep 8 15:35:35 2015 Activating the patch (/ntp-1.0.0-7.0.3.I2.2e.lib32_n9000.rpm) This install operation requires system reload. Do you wish to continue (y/n)?: [n] y [217.975959] [1473348971] writing reset reason 132, System reset due to reload patch(es) activation [217.991166] [1473348971]\ufffd\ufffd CISCO SWITCH Ver7.51 Device detected on 0:6:0 after 0 msecs Device detected on 0:1:1 after 0 msecs Device detected on 0:1:0 after 0 msecs MCFrequency 1333Mhz Relocated to memory	
ステップ 7	（任意） **show install packages | i** *feature* 例： switch# show install packages | i ntp ntp.lib32_n9000 1.0.0-7.0.3.I2.2e installed	デバイス上の基本機能 RPM を表示します。

インストール ログ情報の表示

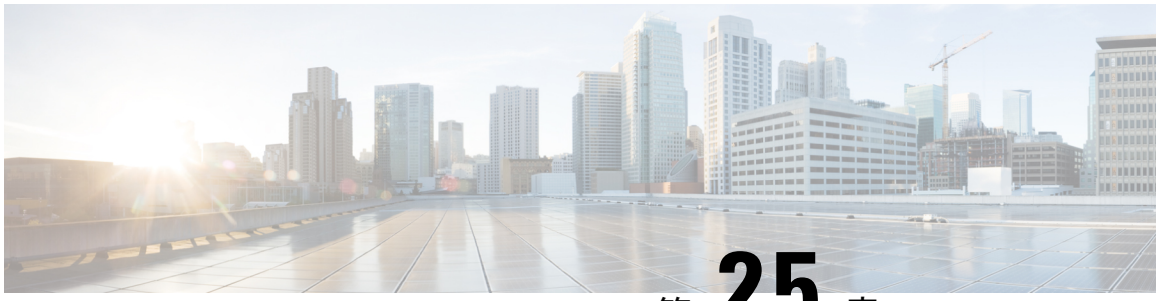
インストール ログは、インストール動作の履歴についての情報を提供します。インストール動作が実行されるたびに、その動作に対して番号が割り当てられます。

- **show install log** コマンドを使用して、インストール動作の成功および失敗の両方について情報を表示します。
- 引数を指定しない **show install log** コマンドを使用して、すべてのインストール動作のサマリーを表示します。ある動作に固有の情報を表示するには、*request-id* 引数を指定します。ファイルの変更、リロードできなかったノード、その他プロセスに影響する操作など、特定の操作の詳細を表示するには、**detail** キーワードを使用します。

次に、すべてのインストール要求の情報を表示する例を示します。

```
switch# show install log
Wed Mar 16 01:26:09 2016
Install operation 1 by user 'admin' at Wed Mar 16 01:19:19 2016
Install add bootflash: nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 1 completed successfully at Wed Mar 16 01:19:24 2016
----------------------------------------
Install operation 2 by user 'admin' at Wed Mar 16 01:19:29 2016
Install activate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 2 completed successfully at Wed Mar 16 01:19:45 2016
----------------------------------------
```

```
Install operation 3 by user 'admin' at Wed Mar 16 01:20:05 2016
Install commit nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 3 completed successfully at Wed Mar 16 01:20:08 2016
----------------------------------------
Install operation 4 by user 'admin' at Wed Mar 16 01:20:21 2016
Install deactivate nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 4 completed successfully at Wed Mar 16 01:20:36 2016
----------------------------------------
Install operation 5 by user 'admin' at Wed Mar 16 01:20:43 2016
Install commit nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 5 completed successfully at Wed Mar 16 01:20:46 2016
----------------------------------------
Install operation 6 by user 'admin' at Wed Mar 16 01:20:55 2016
Install remove nxos.CSCab00001_TOR-1.0.0-7.0.3.I2.2a.lib32_n9000.rpm
Install operation 6 completed successfully at Wed Mar 16 01:20:57 2016
```



第 25 章

コンフィギュレーションの置換の実行

この章は、次の項で構成されています。

コンフィギュレーションの置換とコミットタイムアウトについて

コンフィギュレーションの置換機能を使用すると、デバイスをリロードすることなく Cisco Nexus スイッチの実行コンフィギュレーションをユーザ指定のコンフィギュレーションに置換できます。コンフィギュレーション自体でリロードが必要な場合にのみ、デバイスのリロードが必要になることがあります。ユーザが提供する実行コンフィギュレーションファイルは、実行ファイルのコピーを使用して取得する必要があります。**copy file: to running** と異なり、コンフィギュレーションの置換機能はマージ操作ではありません。この機能では、実行コンフィギュレーション全体が、ユーザによって提供される新しいコンフィギュレーションに置換されます。コンフィギュレーションの置換に障害がある場合は、元のコンフィギュレーションがスイッチで復元されます。Cisco NX-OS リリース 9.3(1) から、**best-effort** オプションが導入されました。このオプションを使用すると、コマンドでエラーが発生した場合でも、設定の置換によって完全なパッチが実行され、元の設定はスイッチに復元されません。

コミットタイムアウト機能を使用すると、コンフィギュレーションの置換操作の実行に成功した後に以前のコンフィギュレーションにロールバックすることができます。コミットタイマーの期限が切れると、ロールバック操作は自動的に開始されます。

（注）
- Cisco NX-OS デバイスで受信済みの有効な実行コンフィギュレーションを提供する必要があります。部分コンフィギュレーションにすることはできません。

概要

設定置換機能には、次の操作手順があります。

- コンフィギュレーションの置換では、Cisco Nexus スイッチの現在の実行コンフィギュレーションとユーザ指定のコンフィギュレーションとの間の違いをインテリジェントに計算し、2 ファイルの差異のパッチ ファイルを生成します。コンフィギュレーション コマンドのセットが含まれているこのパッチ ファイルは表示できます。
- コンフィギュレーションの置換では、実行中のコマンドと同様にパッチ ファイルのコンフィギュレーション コマンドが適用されます。
- コンフィギュレーションは、次の状況下で以前の実行コンフィギュレーションにロールバックまたは復元されます。
  - パッチ ファイルが適用された後、コンフィギュレーションに不一致がある場合。
  - コミット タイムアウトを使用してコンフィギュレーション操作を実行し、コミット タイマーが期限切れになった場合。
- ベストエフォート オプションが使用されている場合、設定は以前の実行コンフィギュレーションにロールバックされず、復元もされません。このオプションを使用すると、コマンドでエラーが発生した場合でも、設定の置換によって完全なパッチが実行され、以前の設定にロールバックされません。
- **show config-replace log exec** コマンドを使用すると、エラーが発生したコンフィギュレーションそのものを表示できます。
- スイッチを元のコンフィギュレーションに復元するときにエラーが発生しても復元操作は中断されません。復元操作は、残りのコンフィギュレーションを続行します。復元操作中にエラーが発生したコマンドを一覧表示するには、**show config-replace log exec** コマンドを使用します。
- タイマーの期限が切れる前に **configure replace commit** コマンドを入力した場合、コミット タイマーは停止し、コンフィギュレーションの置換機能によって適用されているユーザ指定のコンフィギュレーションでスイッチが稼働します。
- コミット タイマーの期限が切れると、以前のコンフィギュレーションへのロールバックは自動的に開始されます。
- Cisco NX-OS リリース 9.3(1) では、セマンティック検証のサポートが設定の置換に追加されました。このセマンティック検証は、設定置換の事前チェックの一部として実行されます。パッチは、セマンティック検証が成功した場合にのみ適用されます。パッチファイル

を適用すると、コンフィギュレーションの置換によって検証プロセスがトリガーされます。コンフィギュレーションの置換は、検証プロセスで、実行コンフィギュレーションとユーザー構成ファイルを比較します。不一致がある場合、デバイスは元のコンフィギュレーションに復元されます。

コンフィギュレーションの置換と実行コンフィギュレーションへのファイルのコピーとの違いは、次のとおりです。

コンフィギュレーションの置換	ファイルのコピー
configure replace <*target-url*> コマンドでは、現在の実行コンフィギュレーションにのみ含まれ、置換ファイルには存在しないコマンドは削除されます。また、現在の実行コンフィギュレーションに追加する必要があるコマンドも追加されます。	**copy** <*source-url*> **running-config** コマンドはマージ動作であり、ソース ファイルと現在の実行コンフィギュレーションの両方のコマンドがすべて保持されます。このコマンドでは、現在の実行コンフィギュレーションにのみ含まれ、ソース ファイルには存在しないコマンドが削除されることはありません。
configure replace <*target-url*> コマンドの交換ファイルには、完全な Cisco NX-OS コンフィギュレーション ファイルを使用する必要があります。	**copy** <*source-url*> **running-config** コマンドのコピー元ファイルとして、部分コンフィギュレーション ファイルを使用できます。

コンフィギュレーションの置換の利点

コンフィギュレーションの置換の利点は次のとおりです。

- スイッチをリロードしたり、CLI で実行コンフィギュレーション ファイルに加えた変更を手動で元に戻したりすることなく、現在の実行コンフィギュレーション ファイルをユーザ指定のコンフィギュレーション ファイルと置換できます。その結果、システムのダウンタイムが減少します。
- 保存済みの Cisco NX-OS コンフィギュレーションの状態に戻すことができます。
- 追加や削除が必要なコマンドだけが影響を受ける場合、デバイスに完全なコンフィギュレーション ファイルを適用することができるため、コンフィギュレーションの変更が簡素化されます。その他のサービスおよび変更されていないコンフィギュレーションには影響しません。
- コミットタイムアウト機能を設定すると、コンフィギュレーションの置換操作が成功したときでも以前のコンフィギュレーションにロールバックすることができます。

コンフィギュレーションの置換に関する注意事項と制限事項

コンフィギュレーションの置換機能には、コンフィギュレーションに関する次のガイドラインと制限事項があります。

- 設定置換機能は、Cisco Nexus 3000 シリーズおよび Cisco Nexus 9000 シリーズ スイッチでサポートされています。
- コンフィギュレーションの置換、チェックポイント、ロールバック操作、または実行コンフィギュレーションからスタートアップコンフィギュレーションへのコピーを同時に実行できるのは、1 ユーザだけです。複数の Telnet、SSH または NX-API セッション経由の操作などのパラレル操作はサポートされていません。複数のコンフィギュレーションの置換またはロールバック要求はシリアル化され、たとえば、最初の要求の完了後にのみ、2 番目の要求の処理が開始されます。
- コミットタイマーの実行中に別のコンフィギュレーションの置換操作を開始することはできません。**configure replace commit** コマンドを使用してタイマーを停止するか、またはコミットタイマーの期限が切れるまで待機してから別のコンフィギュレーションの置換操作を開始する必要があります。
- **system default switchport shutdown** または **no system default switchport shutdown** を **configure replace bootflash:target_config_file** コマンドとともに使用する場合、ユーザーは、すべてのスイッチポートインターフェイスの target_config_file に目的のポート ステート（shutdown または no shutdown）ステートメントが存在することを確認する必要があります。
- Cisco NX-OS Release 9.3（6）以降では、**service exclude-bootconfig** の設定によって**boot nxos** イメージ設定を、**show running-config**、**show startup-config**、 **copy running-config** *filename*、および **copy startup-config** *filename* コマンドで除外できます。
- コミットタイムアウト機能は、コミットタイムアウトを使用してコンフィギュレーションの置換操作を実行する場合にのみ開始されます。タイマーの値の範囲は 30 ～ 3600 秒です。
- ユーザ指定のコンフィギュレーション ファイルは、Cisco NX-OS デバイスから取得（copy run file）された有効な show running-configuration の出力である必要があります。このコンフィぎゅーレーションは部分コンフィギュレーションにすることはできず、user admin などの必須コマンドが含まれている必要があります。
- ソフトウェア バージョン違いで生成されたコンフィギュレーション ファイルでコンフィギュレーションの置換操作を実行することは、操作が失敗する可能性があるため推奨されません。ソフトウェアバージョンの変更があるたびに新しいコンフィギュレーション ファイルを再生成する必要があります。
- Multichassis EtherChannel トランク（MCT）設定を仮想ピアリンク設定と置き換えようとした場合、コンフィギュレーションの置換操作はサポートされません。物理 MCT はイーサ

ネットを介したCFS配信モードを使用し、仮想ピアリンクはIPを介したCFS配信モードを使用するため、この操作は許可されません。

- コンフィギュレーションの置換操作が進行中の場合、他のセッションからはコンフィギュレーションを変更しないことを推奨します。操作が失敗する可能性があります。
- コンフィギュレーションの置換機能については、次の点に注意してください。
  - コンフィギュレーションの置換機能は、リロードを必要とする機能をサポートしていません。このような機能の1例は、system vlan reserveです。
  - Cisco NX-OSリリース9.3(5)以降では、FEXインターフェイス コンフィギュレーションの設定置換（CR）がサポートされています。FEXのプロビジョニングはCRではサポートされていません。プロビジョニングされたFEXインターフェイスの設定は、CRを使用して変更できます。

（注） このガイドラインは、FEXがサポートされていないCisco Nexus 3000シリーズ プラットフォーム スイッチには適用されません。

  - Cisco NX-OSリリース9.3 (5)以降では、設定置換機能がポート プロファイルでサポートされています。
  - コンフィギュレーションの置換機能は、configure terminalモード コマンドで**のみ**サポートされます。configure profile、configure jobs、およびその他のモードはサポートされていません。
  - Cisco NX-OSリリース9.3(5)以降では、ジョブの設定モードがサポートされています。スケジューラ ジョブ コマンドを含むコンフィギュレーション ファイルは、コンフィギュレーションの置換に使用できます。
  - Cisco NX-OSリリース9.3(4)以降では、ブレークアウト インターフェイス コンフィギュレーションの設定置換機能がサポートされています。
  - 実行コンフィギュレーションに**feature-set mpls**または**mpls static range**コマンドが含まれていて、MPLSなしでコンフィギュレーションに移動しようとしたり、ラベルの範囲を変更する場合、コンフィギュレーションの置換機能が失敗することがあります。
  - コンフィギュレーションの置換機能は、自動設定をサポートしていません。
- コンフィギュレーションの置換機能が適用されるライン カードがオフラインである場合、コンフィギュレーションの置換操作は失敗します。
- 設定置換機能を使用してITDを変更する前に、ITDサービスをシャットダウンする必要があります（**shutdown**）。
- ユーザ コンフィギュレーションからのメンテナンス モードへの移行はサポートされていません。

- メンテナンス モードから **configure replace** コマンドを使用すると、次の警告でユーザの確認が求められます。

```
Warning: System is in maintenance mode. Please ensure user config won't inadvertently
 revert back config in maintenance mode profile.
            Do you wish to proceed anyway? (y/n)  [n]
```

- <*non-interactive*> オプションを使用してメンテナンス モードから **configure replace** コマンドを使用することはサポートされています。デフォルトでは、*yes* のユーザ確認を受けてから進行します。
- コンフィギュレーションを適用するために Cisco NX-OS デバイスをリロードする必要がある場合、これらのコンフィギュレーションをリロードしてからコンフィギュレーションの置換操作を行う必要があります。
- ユーザ指定のコンフィギュレーション ファイルでのコマンドの順序は、Cisco Nexus スイッチの実行コンフィギュレーションでのこれらのコマンドと同じにする必要があります。
- CR を使用してスイッチの実行コンフィギュレーションを置き換える必要があるユーザ コンフィギュレーション ファイルは、新しいコマンドを設定した後、スイッチの実行コンフィギュレーションから生成する必要があります。ユーザ コンフィギュレーション ファイルは、CLI コマンドを使用して手動で編集しないでください。また、コンフィギュレーション コマンドのシーケンスを変更しないでください。
- セマンティック検証は、4 ギガビット メモリ プラットフォームではサポートされていません。
- 異なるバージョンの機能が実行コンフィギュレーションとユーザ コンフィギュレーションに存在する場合（VRRPv2 と VRRPv3 など）、セマンティック検証オプションが期待どおりに機能しません。この問題は既知の制限です。

コンフィギュレーションの置換の推奨ワークフロー

コンフィギュレーションの置換の推奨されるワークフローを次に示します。

1. Cisco Nexus シリーズ デバイスで最初にコンフィギュレーションを適用してコンフィギュレーション ファイルを生成してから、コンフィギュレーション ファイルとして **show running-configuration** 出力を使用します。このファイルを使用して、必要に応じてコンフィギュレーションを変更します。次に、この生成または更新されたコンフィギュレーション ファイルを使用して、コンフィギュレーションの置換を実行します。

（注） ソフトウェア バージョンの変更があるたびにコンフィギュレーション ファイルを再生成する必要があります。異なるソフトウェア バージョンで生成されたコンフィギュレーション ファイルを使用してコンフィギュレーションの置換操作を実行することは推奨されません。

2. **configure replace** <*file*> **show-patch** コマンドを実行してパッチ ファイルを表示し、確認します。この手順は任意です。

3. 構成の置換ファイルを実行するか、**commit-timeout** <*time*>機能をスキップします。要件に基づいて、次の手順のいずれかを実行できます。
   - コンフィギュレーションの置換で実行されるコマンドをコンソールに表示するには、**configure replace** <*file*> **verbose** を実行します。
   - **configure replace** [**bootflash**/**scp**/**sftp**] <*user-configuration-file*> **verbose commit-timeout** <*time*> コマンドを実行して、コミット時間を構成します。
4. **configure replace commit** コマンドを実行し、コミット タイマーを停止します。この手順は、コミットタイムアウト機能でコンフィギュレーションの置換操作を実行している場合に必要です。
5. コンフィギュレーションのセマンティック検証を含むプレチェックをコンフィギュレーションの置換で実行します。エラーがある場合、コンフィギュレーションの置換操作は失敗します。失敗したコンフィギュレーションの詳細を表示するには、**show config-replace log verify** コマンドを使用します。パッチファイルを適用すると、コンフィギュレーションの置換によって検証プロセスがトリガーされます。コンフィギュレーションの置換は、検証プロセスで、実行コンフィギュレーションとユーザー構成ファイルを比較します。不一致がある場合、デバイスは元のコンフィギュレーションに復元されます。不一致のコンフィギュレーションを表示するには、**show config-replace log verify** コマンドを使用します。
6. Cisco NX-OS リリース9.3(1) では、次のコンフィギュレーションの置換操作を実行できます。
   - セマンティック検証およびベストエフォートモードなしのコンフィギュレーションの置換。
   - セマンティック検証なし、ベストエフォートモードありのコンフィギュレーションの置換。
   - セマンティック検証あり、ベストエフォートモードなしのコンフィギュレーションの置換。
   - セマンティック検証およびベストエフォートモードありのコンフィギュレーションの置換。

コンフィギュレーションの置換の実行

コンフィギュレーションの置換を実行するには、次の操作を行います。

始める前に

現在の構成ファイルと候補構成ファイルの IP アドレスに競合がないことを確認します。IP アドレスの競合の例は、現在の構成ファイルの eth インターフェイス 1/53 で 172.16.0.1/24 を構成し、候補構成ファイル内の eth 1/53 で 172.16.0.1/24 と 192.168.0.1/24 を使用してポートチャネ

ル 30 を構成したとします。候補構成ファイルの構成置換を実行すると、IP アドレスの競合が発生します。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure replace** { < *uri_local* > | < *uri_remote* > } [**verbose** | **show-patch**]	コンフィギュレーションの置換を実行します。コンフィギュレーションの置換の進行中にセッションを通じてコンフィギュレーションを変更すると、コンフィギュレーションの置換操作は失敗します。1 つのコンフィギュレーション要求がすでに進行中であるときにコンフィギュレーションの置換要求を送信すると、要求はシリアル化されます。
ステップ 2	**configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **show-patch**	実行コンフィギュレーションとユーザ指定のコンフィギュレーションの違いを表示します。
ステップ 3	**configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose**	スイッチのコンフィギュレーションを、ユーザが提供する新しいユーザコンフィギュレーションに置換します。コンフィギュレーションの置換は常にアトミックです。
ステップ 4	**configure replace** <*user-configuration-file*> [**best-effort**]	スイッチの設定を新しいユーザ設定に置き換え、セマンティック検証による設定の置き換えを有効にします。 best-effort オプションを使用すると、コマンドでエラーが発生した場合でも設定の置換によって完全なパッチが実行され、以前の設定がロールバックされないようになります。 Cisco NX-OS リリース 10.5(1)F 以降、コンフィギュレーション置換機能は、Cisco Nexus 9300-FX2/FX3/GX シリーズ スイッチのバッチ ACL コンフィギュレーションをサポートします。 **ベストエフォート** モードが有効になっている場合、バッチ構成内で障害が発生すると、その特定のバッチ内の構成セット全体がスキップされます。

	コマンドまたはアクション	目的
ステップ 5	**configure replace** <*user-configuration-file*> [**verify-and-commit**]	スイッチの設定を新しいユーザ設定に置き換え、セマンティック検証による設定の置き換えを有効にします。 verify-and-commit オプションは、セマンティック検証を有効にするために使用されます。パッチは、完全なパッチのセマンティック検証に合格した場合にのみ実行されます。 ベストエフォート オプション、verify-and-commit オプション、または両方のオプションを同時に使用できます。
ステップ 6	**configure replace** <*user-configuration-file*> [**verify-only**]	パッチのみを表示し、パッチでセマンティック検証を実行し、結果を表示します。パッチはシステムに適用されません。
ステップ 7	（任意） **configure replace** [**bootflash** / **scp** / **sftp**] < *user-configuration-file* > **verbose commit-timeout** <*time*>	コミット時間を秒単位で設定します。タイマーは、コンフィギュレーションの置換操作が正常に完了した後に開始されます。
ステップ 8	（任意） **configure replace** [**commit**]	コミット タイマーを停止し、コンフィギュレーションの置換設定を続行します。 （注） この手順は、コミットタイムアウト機能を設定している場合にのみ適用されます。 （注） 以前のコンフィギュレーションにロールバックするには、コミット タイマーの期限が切れるまで待機する必要があります。タイマーの期限が切れると、スイッチは自動的に以前のコンフィギュレーションにロールバックされます。
ステップ 9	（任意） **configure replace** [**bootflash**/**scp**/**sftp**] <*user-configuration-file*> *non-interactive*	メンテナンス モードでは、ユーザ プロンプトはありません。デフォルトでは、**yes** のユーザ確認を受けてからロールバックが進行します。非インタラクティブ オプションは、メンテナンス モードでのみ使用できます。

コンフィギュレーションの置換の確認

コンフィギュレーションの置換とそのステータスをチェックして確認するには、表に記載されているコマンドを使用します。

表 *34 :* コンフィギュレーションの置換の確認

コマンド	目的
configure replace [**bootflash**/**scp**/**sftp**]<*user-configuration-file*] **show-patch**	実行コンフィギュレーションとユーザ指定のコンフィギュレーションの違いを表示します。
show config-replace log exec	実行したすべてのコンフィギュレーションと失敗したコンフィギュレーションのログを表示します。エラーの場合、そのコンフィギュレーションに対してエラー メッセージが表示されます。
show config-replace log verify	失敗したコンフィギュレーションをエラーメッセージとともに表示します。成功したコンフィギュレーションは表示されません。
show config-replace status	コンフィギュレーションの置換操作のステータス（進行中、成功、失敗など）を表示します。コミットタイムアウト機能を設定している場合、コミットとタイマーのステータスに加え、コミット タイムアウトの残り時間も表示されます。

コンフィギュレーションの置換の例

以下のコンフィギュレーションの置換の設定例を参照してください。

- **configure replace bootflash:** <*file*> **show-patch** CLI コマンドを使用して、実行コンフィギュレーションとユーザ指定のコンフィギュレーションの違いを表示します。

```
switch(config)# configure replace bootflash:<file> show-patch
Collecting Running-Config
Converting to checkpoint file
#Generating Rollback Patch
!!
no role name abc
```

- **configure replace bootflash:** <*file*> **verbose** CLI コマンドを使用して、スイッチの実行コンフィギュレーション全体をユーザコンフィギュレーションに置換します。

```
switch(config)# configure replace bootflash:<file> verbose
Collecting Running-Config
```

```
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
==========================================================
config t
no role name abc
==========================================================
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

Sample Example with adding of BGP configurations.
 switch(config)# sh run | section bgp
 switch(config)# sh file bootflash:file | section bgp
feature bgp
router bgp 1
    address-family ipv4 unicast
    neighbor 1.1.1.1
switch(config)#
switch(config)# configure replace bootflash:file verbose
Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Note: Applying config parallelly may fail Rollback verification
Collecting Running-Config
#Generating Rollback Patch
Executing Rollback Patch
==========================================================
config t
feature bgp
router bgp 1
address-family ipv4 unicast
neighbor 1.1.1.1
==========================================================
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)# sh run | section bgp
feature bgp
router bgp 1
  address-family ipv4 unicast
  neighbor 1.1.1.1

Sample Example with ACL
 switch(config)# configure replace bootflash:run_1.txt
 Collecting Running-Config
 Generating Rollback patch for switch profile
 Rollback Patch is Empty
 Note: Applying config parallelly may fail Rollback verification
 Collecting Running-Config
 #Generating Rollback Patch
 Executing Rollback Patch
 ==========================================================
 config t
 no ip access-list nexus-50-new-xyz
 ip access-list nexus-50-new-xyz-jkl-abc
 10 remark Newark
 20 permit ip 17.31.5.0/28 any
```

```
30 permit ip 17.34.146.193/32 any
40 permit ip 17.128.199.0/27 any
50 permit ip 17.150.128.0/22 any
==========================================================
Generating Running-config for verification
Generating Patch for verification

Rollback completed successfully.

switch(config)#


switch(config)# show run aclmgr | sec nexus-50-new-xyz-jkl-abc
ip access-list nexus-50-new-xyz-jkl-abc
  10 remark Newark
  20 permit ip 17.31.5.0/28 any
  30 permit ip 17.34.146.193/32 any
  40 permit ip 17.128.199.0/27 any
  50 permit ip 17.150.128.0/22 any
```

- **configure replace bootflash:user-config.cfg verify-only** CLI コマンドを使用して、パッチを意味的に生成および確認します。

```
switch(config)# configure replace bootflash:user-config.cfg verify-only

Version match between user file and running configuration.
Pre-check for User config PASSED
Collecting Running-Config
Converting to checkpoint file
Generating Rollback Patch
Validating Patch
==========================================================
`config t `
`interface Ethernet1/1`
`shutdown`
`no switchport trunk allowed vlan`
`no switchport mode`
`no switchport`
`exit`
Skip non dme command for CR validation
`interface Vlan1`
`shutdown`
`interface Ethernet1/1`
`shutdown`
`no switchport`
`ip address 1.1.1.1/24`
`exit`
Skip non dme command for CR validation
==========================================
Patch validation completed successful
switch(config)#
```

- パッチでセマティック検証を実行した後、**configure replace bootflash:user-config.cfg best-effort verify-and-commit** CLI コマンドを使用して、スイッチの実行コンフィギュレーションを特定のユーザ コンフィギュレーションに置き換えます。

```
switch(config)# configure replace bootflash:user-config.cfg best-effort
verify-and-commit

 Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
```

```
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
Generating Rollback Patch

Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch

Configure replace completed successfully. Please run 'show config-replace log exec'
 to see if there is any configuration that requires reload to take effect.

switch(config)#
```

- **show config-replace log exec** CLI コマンドを使用して、実行したコンフィギュレーションと、存在する場合はエラーをすべて確認します。

```
switch(config)# show config-replace log exec
Operation            : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme               : tmp
Rollback done By     : admin
Rollback mode        : atomic
Verbose              : enabled
Start Time           : Wed, 06:39:34 25 Jan 2017
--------------------------------------------------------------------------------

time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
End Time             : Wed, 06:39:47 25 Jan 2017
Rollback Status      : Success

Executing Patch:
----------------
switch#config t
switch#no role name abc
```

- **show config-replace log verify** CLI コマンドを使用して、存在する場合は失敗したコンフィギュレーションを確認します。

```
switch(config)# show config-replace log verify
Operation            : Rollback to Checkpoint File
Checkpoint file name : .replace_tmp_28081
Scheme               : tmp
Rollback done By     : admin
Rollback mode        : atomic
Verbose              : enabled
Start Time           : Wed, 06:39:34 25 Jan 2017
End Time             : Wed, 06:39:47 25 Jan 2017
Status               : Success

Verification patch contains the following commands:
--------------------------------------------------
!!
! No changes
--------------------------------------------------------------------------------
```

```
time: Wed, 06:39:47 25 Jan 2017
Status: SUCCESS
```

- **show config-replace status** CLI コマンドを使用して、コンフィギュレーションの置換のステータスを確認します。

```
switch(config)# show config-replace status
Last operation : Rollback to file
Details:
  Rollback type: atomic replace_tmp_28081
  Start Time: Wed Jan 25 06:39:28 2017
  End Time: Wed Jan 25 06:39:47 2017
  Operation Status: Success
switch(config)#
```

スイッチから生成された設定の代わりに手動で作成された設定を使用すると、[置換の設定（Configure Replace）] が失敗することがあります。失敗の原因として考えられるのは、show running configurationに示されていないデフォルト設定の潜在的な違いです。次の例を参照してください。

power redundancy コマンドがデフォルトのコマンドである場合、デフォルトの設定では表示されません。ただし、**show run all** コマンドを使用すると表示されます。次の例を参照してください。

```
switch# show run all

!Command: show running-config all
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:16:09 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
no hardware module boot-order reverse
no license grace-period
<snip>
hostname n9k13
```

電源冗長コマンドは、show running configuration コマンド出力には表示されません。次の例を参照してください。

```
!Command: show running-config
!Running configuration last done at: Tue Nov 12 11:07:44 2019
!Time: Tue Nov 12 11:17:24 2019

version 9.3(1) Bios:version 05.39
hostname n9k13
```

設定置換のユーザ コンフィギュレーションに **power redundancy-mode ps-redundant** コマンドが追加された場合。検証/コミットが失敗する可能性があります。次の例を参照してください。

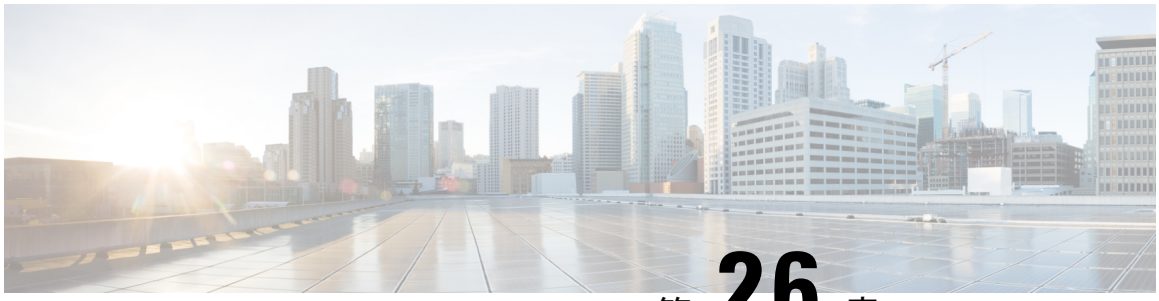
```
switch# show file bootflash:test

!Command: show running-config
!Running configuration last done at: Tue Nov 12 10:56:49 2019
!Time: Tue Nov 12 11:04:57 2019

version 9.3(1) Bios:version 05.39
power redundancy-mode ps-redundant
hostname n9k13
```

power redundancy-mode ps-redundant コマンドは、設定置換の後の show running には表示されません。したがって、「欠落」と見なされ、CR は失敗します。次に例を示します。

```
switch# config replace bootflash:test verify-and-commit

Version match between user file and running configuration.
Pre-check for User config PASSED
ADVISORY: Config Replace operation started...
Modifying running configuration from another VSH terminal in parallel
is not recommended, as this may lead to Config Replace failure.

Collecting Running-Config
Generating Rollback patch for switch profile
Rollback Patch is Empty
Collecting Running-Config
.Generating Rollback Patch

Validating Patch
Patch validation completed successful
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Rollback Patch
Executing Rollback Patch
During CR operation,will retain L3 configuration
when vrf member change on interface
Generating Running-config for verification
Generating Patch for verification
Verification failed, Rolling back to previous configuration
Collecting Running-Config
Cleaning up switch-profile buffer
Generating Rollback patch for switch profile
Executing Rollback patch for switch profiles. WARNING - This will change the
configuration of switch profiles and will also affect any peers if configured
Collecting Running-Config
Generating Rollback Patch
Rollback Patch is Empty
Rolling back to previous configuration is successful

Configure replace failed. Use 'show config-replace log verify' or 'show config-replace
log exec' to see reasons for failure

n9k13# show config-replace log verify
Operation : Config-replace to user config
Checkpoint file name : .replace_tmp_31849
Scheme : tmp
Cfg-replace done By : agargula
Cfg-replace mode : atomic
Verbose : disabled
Start Time : Tue, 11:20:59 12 Nov 2019
Start Time UTC : Tue, 10:20:59 12 Nov 2019
----------------------------------------
End Time : Tue, 11:21:28 12 Nov 2019
End Time UTC : Tue, 10:21:28 12 Nov 2019
Status : Failed

Verification patch contains the following commands:
--------------------------------------------------
!!
Configuration To Be Added Missing in Running-config
==================================================
!
power redundancy-mode ps-redundant
```

```
Undo Log
--------------------------------------------------------------------------------
End Time : Tue, 11:21:32 12 Nov 2019
End Time UTC : Tue, 10:21:32 12 Nov 2019
Status : Success
n9k13#
```

上記の例では、CR は欠落しているデフォルトのコマンドを考慮します。



第 26 章

ロールバックの設定

この章は、次の項で構成されています。

ロールバックについて

ロールバック機能を使用すると、Cisco NX-OS のコンフィギュレーションのスナップショットまたはユーザーチェックポイントを使用して、スイッチをリロードしなくても、いつでもそのコンフィギュレーションをスイッチに再適用できます。権限のある管理者であれば、チェックポイントで設定されている機能について専門的な知識がなくても、ロールバック機能を使用して、そのチェックポイント コンフィギュレーションを適用できます。

いつでも、現在の実行コンフィギュレーションのチェックポイント コピーを作成できます。Cisco NX-OS はこのチェックポイントを ASCII ファイルとして保存するので、将来、そのファイルを使用して、実行コンフィギュレーションをチェックポイント コンフィギュレーションにロールバックできます。複数のチェックポイントを作成すると、実行コンフィギュレーションのさまざまなバージョンを保存できます。

実行コンフィギュレーションをロールバックするとき、atomic ロールバックを発生させることができます。atomic ロールバックでは、エラーが発生しなかった場合に限り、ロールバックを実行します。

ロールバックの注意事項と制約事項

ロールバックに関する設定時の注意事項および制約事項は、次のとおりです。

- 作成できるチェックポイント コピーの最大数は 10 です。
- あるスイッチのチェックポイント ファイルを別のスイッチに適用することはできません。

- チェックポイント ファイル名の長さは、最大 75 文字です。
- チェックポイントのファイル名の先頭を system にすることはできません。
- チェックポイントのファイル名の先頭を auto にすることができます。
- チェックポイントのファイル名を、summary または summary の略語にすることができます。
- チェックポイント、ロールバック、または実行コンフィギュレーションからスタートアップ コンフィギュレーションへのコピーを同時に実行できるのは、1 ユーザだけです。
- **write erase** および **reload** コマンドを入力すると、チェック ポイントが削除されます。clear checkpoint database コマンドを使用すると、すべてのチェックポイント ファイルを削除できます。
- ブートフラッシュでチェックポイントを作成した場合、ロールバックの実行前は実行システム コンフィギュレーションとの違いは実行できず、「変更なし」と報告されます。
- チェック ポイントはスイッチに対してローカルです。
- **checkpoint** および **checkpoint** *checkpoint_name* コマンドを使用して作成されたチェックポイントは、すべてのスイッチの 1 つのスイッチオーバーに対して存在します。
- ブートフラッシュ時のファイルへのロールバックは、**checkpoint** *checkpoint_name* コマンドを使用して作成されたファイルでのみサポートされます。他の ASCII タイプのファイルではサポートされません。
- チェックポイントの名前は一意にする必要があります。以前に保存したチェックポイントを同じ名前で上書きすることはできません。
- ロールバックは自動設定のコンテキストではサポートされません。チェックポイントは自動設定を保存しません。したがって、ロールバックを実行した後、対応する自動設定は存在しないことになります。
- Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があります。

チェックポイントの作成

1 台のスイッチで作成できるコンフィギュレーションの最大チェックポイント数は 10 です。

手順

	コマンドまたはアクション	目的
ステップ 1	switch# **checkpoint** { [*cp-name*] [**description** *descr*] | **file** *file-name* **例：** `switch# checkpoint stable`	ユーザ チェックポイント名またはファイルのいずれかに対して、実行中のコンフィギュレーションのチェックポイントを作成します。チェックポイント名には

	コマンドまたはアクション	目的
		最大 80 文字の任意の英数字を使用できますが、スペースを含めることはできません。チェックポイント名を指定しなかった場合、Cisco NX-OS はチェックポイント名を user-checkpoint-<number> に設定します。ここで number は 1 ～ 10 の値です。 description には、スペースも含めて最大 80 文字の英数字を指定できます。
ステップ 2	（任意） switch# **no checkpoint***cp-name* 例： `switch# no checkpoint stable`	**checkpoint** コマンドの **no** 形式を使用すると、チェックポイント名を削除できます。 **delete** コマンドを使用して、チェックポイント ファイルを削除できます。
ステップ 3	（任意） switch# **show checkpoint***cp-name* 例： [**all**] `switch# show checkpoint stable`	チェックポイント名の内容を表示します。

ロールバックの実装

チェックポイント名またはファイルにロールバックを実装できます。ロールバックを実装する前に、現在のコンフィギュレーションまたは保存されているコンフィギュレーションを参照しているソースと宛先のチェックポイント間の差異を表示できます。

（注） atomic ロールバック中に設定を変更すると、ロールバックは失敗します。

手順

	コマンドまたはアクション	目的
ステップ 1	**show diff rollback-patch** { **checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} { **checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*} 例：	ソースと宛先のチェックポイント間の差異を表示します。

	コマンドまたはアクション	目的
	`switch# show diff rollback-patch checkpoint stable running-config`	
ステップ 2	**rollback running-config** { **checkpoint** *cp-name* | **file** *cp-file*} **atomic** 例： `switch# rollback running-config checkpoint stable`	エラーが発生しなければ、指定されたチェックポイント名またはファイルへの atomic ロール バックを作成します。

例

チェックポイント ファイルを作成し、次に、ユーザー チェックポイント名への atomic ロール バックを実装する例を以下に示します。

```
switch# checkpoint stable
switch# rollback running-config checkpoint stable atomic
```

ロールバック コンフィギュレーションの確認

ロールバックの設定を確認するには、次のコマンドを使用します。

コマンド	目的
show checkpoint *name* [**all**]	チェックポイント名の内容を表示します。
show checkpoint all [**user** | **system**]	現行のスイッチ内のすべてのチェックポイントの内容を表示します。表示されるチェックポイントを、ユーザーまたはシステムで生成されるチェックポイントに限定できます。
show checkpoint summary [**user** | **system**]	現在のスイッチ内のすべてのチェックポイントのリストを表示します。表示されるチェックポイントを、ユーザーまたはシステムで生成されるチェックポイントに限定できます。
show diff rollback-patch { **checkpoint** *src-cp-name* | **running-config** | **startup-config** | **file** *source-file*} { **checkpoint** *dest-cp-name* | **running-config** | **startup-config** | **file** *dest-file*}	ソースと宛先のチェックポイント間の差異を表示します。
show rollback log [**exec** | **verify**]	ロールバック ログの内容を表示します。

（注） すべてのチェックポイント ファイルを削除するには、**clear checkpoint database** コマンドを使用します。

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。