Cisco Nexus 9000 シリーズ NX-OS システム管理設定ガイド リリース 7.x

初版：2015 年 02 月 01 日

最終更新：2016 年 05 月 07 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、角カッコ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。

表記法	説明
{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、波カッコで囲み、縦棒で区切って示しています。
[x {y | z}]	角カッコまたは波カッコ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。角カッコ内の波カッコと縦棒は、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、screen 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 screen 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 9000 シリーズ スイッチの関連資料

Cisco Nexus 9000 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。

http://www.cisco.com/en/US/products/ps13386/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。

マニュアルの入手方法およびテクニカル サポート

マニュアルの入手、Cisco Bug Search Tool（BST）の使用、サービス要求の送信、追加情報の収集の詳細については、『*What's New in Cisco Product Documentation*』を参照してください。このドキュメントは、http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html から入手できます。

『*What's New in Cisco Product Documentation*』では、シスコの新規および改訂版の技術マニュアルの一覧を、RSS フィードとして購読できます。また、リーダー アプリケーションを使用して、コンテンツをデスクトップに配信することもできます。RSS フィードは無料のサービスです。

第 1 章

新機能および変更された機能に関する情報

この章では、『*Cisco Nexus 9000* シリーズ *NX-OS* システム管理コンフィギュレーション ガイド リリース *7.x*』に記載されている新機能および変更された機能に関するリリース固有の情報について説明します。

新機能および変更された機能に関する情報

次の表は、『*Cisco Nexus 9000* シリーズ *NX-OS* システム管理コンフィギュレーション ガイド リリース *7.x*』に記載されている新機能および変更機能をまとめたものです。それぞれの説明が記載されている箇所も併記されています。

表 *1*：*NX-OS* リリース *7.x* の新機能および変更された機能

機能	説明	変更されたリリース	参照先
SNMP	MIB 情報を更新しました。	7.0(3)I4(2)	SNMP 通知のイネーブル化，（188 ページ）
SNMP	Cisco Nexus 9200 シリーズ スイッチの入力パイプラインでフォワードパケット ドロップをスパンする機能が追加されました。	7.0(3)I4(1)	SNMP 通知のイネーブル化，（188 ページ）
ERSPAN	Cisco Nexus 9200 シリーズ スイッチの入力パイプラインでフォワードパケット ドロップをスパンする機能が追加されました。	7.0(3)I4(1)	ERSPAN の設定，（281 ページ）

新機能および変更された機能に関する情報

機能	説明	変更されたリリース	参照先
ERSPAN	Cisco Nexus 9200 シリーズ スイッチ用の ERSPAN ACL に **set-erspan-gre-proto** および **set-erspan-dscp** アクションが追加されました。	7.0(3)I4(1)	ERSPAN の設定，（281 ページ）
ERSPAN および SPAN	Cisco Nexus 9200 シリーズ スイッチで、ユーザ定義フィールド（UDF）ベース ERSPAN および UDF ベース SPAN のサポートが追加されました。	7.0(3)I4(1)	ERSPAN の設定，（281 ページ） および SPAN の設定，（261 ページ）
ERSPAN および SPAN	複数セッションでの同じ送信元のサポートが追加されました。	7.0(3)I4(1)	ERSPAN の設定，（281 ページ） および SPAN の設定，（261 ページ）
ERSPAN および SPAN	Cisco Nexus 9300 および 9500 シリーズ スイッチでの同じ送信元に対する複数の ACL フィルタのサポートが追加されました。	7.0(3)I4(1)	ERSPAN の設定，（281 ページ） および SPAN の設定，（261 ページ）
SNMP	IPv6 ACL を SNMPv2 コミュニティまたは SNMPv3 ユーザに割り当てて SNMP 要求をフィルタする機能が追加されました。	7.0(3)I4(1)	SNMP 要求のフィルタリング，（181 ページ）
SPAN	Cisco Nexus 9200 シリーズ スイッチでのみ、SPAN の宛先として CPU のサポートが追加されました。	7.0(3)I4(1)	SPAN の設定，（261 ページ）
LLDP	DCBXP のサポートが追加されました。	7.0(3)I3(1)	LLDP の設定，（305 ページ）
システム メッセージ ロギング	syslog メッセージでの origin ID を設定する機能が追加されました。	7.0(3)I3(1)	Syslog メッセージの Origin ID の設定，（82 ページ）

新機能および変更された機能に関する情報

機能	説明	変更されたリリース	参照先
グレースフル挿入と削除（GIR）	vPC ドメインのシャットダウンサポートおよびPIMプロトコルの分離のサポートが追加されました。	7.0(3)I2(2)	グレースフル挿入と削除の設定，（347 ページ）
LLDP	**show lldp all** コマンドが追加されました。	7.0(3)I2(2)	LLDP の設定，（305 ページ）
ERSPAN	プライオリティ フロー制御（PFC）フレームのスパニングを可能にするために、**allow-pfc** オプションが **source interface** コマンドに追加されました。	7.0(3)I2(1)	ERSPAN の設定，（281 ページ）
ERSPAN	**show monitor session** コマンドの出力に、出力インタフェース情報が追加されました。	7.0(3)I2(1)	ERSPAN の設定，（281 ページ）
ERSPAN	入力パイプラインでフォワードパケットドロップをスパンする機能が追加されました。	7.0(3)I2(1)	ERSPAN の設定，（281 ページ）
ERSPAN	ERSPAN ACL に **set-erspan-gre-proto** および **set-erspan-dscp** アクションが追加されました。	7.0(3)I2(1)	ERSPAN の設定，（281 ページ）
ERSPAN および SPAN	ユーザ定義フィールド（UDF）ベース ERSPAN およびUDF ベース SPAN のサポートが追加されました。	7.0(3)I2(1)	ERSPAN の設定，（281 ページ） および SPAN の設定，（261 ページ）
グレースフル挿入と削除（GIR）	この機能が導入されました。	7.0(3)I2(1)	グレースフル挿入と削除の設定，（347 ページ）
MPLS ストリッピング	この機能が導入されました。	7.0(3)I2(1)	TAP アグリゲーションおよび MPLS ストリッピングの設定，（331 ページ）
sFlow	この機能が導入されました。	7.0(3)I2(1)	sFlow の設定，（315 ページ）

新機能および変更された機能に関する情報

機能	説明	変更されたリリース	参照先
スイッチ プロファイル（config-sync）	この機能が導入されました。	7.0(3)I2(1)	スイッチプロファイルの設定，（15 ページ）
PortLoopback テスト	動作がオンデマンドから定期実行（30 分ごと）に変更されました。	7.0(3)I1(2)	ランタイムまたはヘルスモニタリング診断，（218 ページ）
TAP アグリゲーション	100G ポートのサポートが追加されました。	7.0(3)I1(2)	TAP アグリゲーションおよび MPLS ストリッピングの設定，（331 ページ）
高精度時間プロトコル（PTP）	100G 9408PC ライン カードおよび 100G M4PC 汎用拡張モジュール（GEM）を除く、すべての Cisco Nexus 9000 シリーズおよび 3164Q ハードウェアでのサポートが追加されました。	7.0(3)I1(2)	PTP の設定，（57 ページ）
ERSPAN	ERSPAN タイプ III および ERSPAN タイプ III パケットでの 1588 タイムスタンプのサポートが追加されました。	7.0(3)I1(1)	ERSPAN の設定，（281 ページ）
高精度時間プロトコル（PTP）	Cisco Nexus 9332PQ、9396PX、93128TX スイッチ、Cisco Nexus 9504 および 9508 スイッチでシングル X9636PQ ライン カード上の PTP ポートを装備したもの、および Cisco Nexus 3164Q スイッチ用に、この機能が導入されました。	7.0(3)I1(1)	PTP の設定，（57 ページ）
TAP アグリゲーション	この機能が導入されました。	7.0(3)I1(1)	TAP アグリゲーションおよび MPLS ストリッピングの設定，（331 ページ）

第 2 章

概要

この章では、Cisco NX-OS デバイスのモニタや管理に使用できるシステム管理機能について説明します。

この章の内容は、次のとおりです。

- TAP アグリゲーション，12 ページ
- MPLS ストリッピング，12 ページ
- sFlow，12 ページ
- SMU，12 ページ
- 仮想デバイス コンテキスト，12 ページ
- トラブルシューティング機能，12 ページ

ソフトウェア イメージ

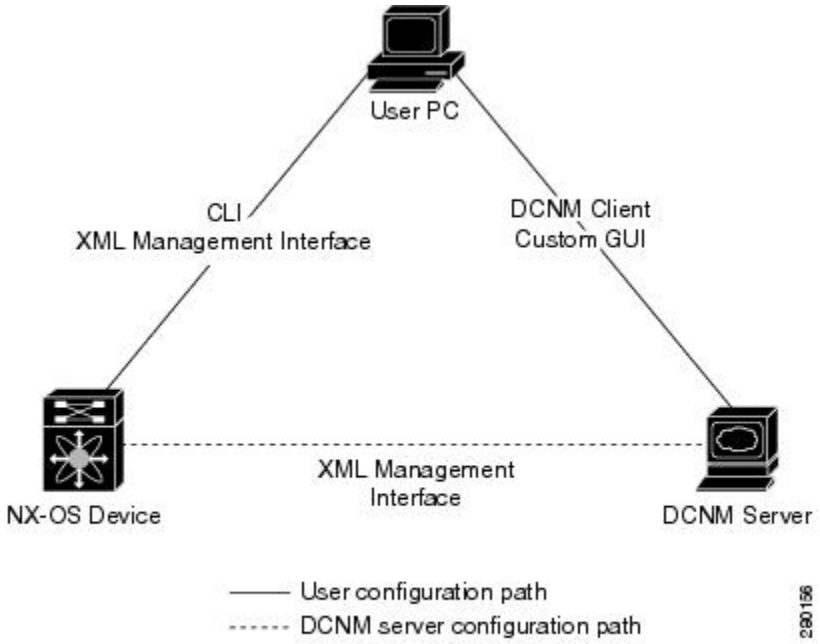
Cisco NX-OS ソフトウェアは、1 個の NXOS ソフトウェア イメージ（例：n9000-dk9.6.1.2.I1.1.bin）で構成されます。このイメージは、すべての Cisco Nexus 9000 シリーズ スイッチで実行されます。

Cisco NX-OS デバイス コンフィギュレーション方式

デバイスは、直接ネットワーク コンフィギュレーション方式または Cisco データセンター ネットワーク管理（DCNM）サーバが提供する Web サービスを使用して設定できます。

次の図は、ネットワーク ユーザが使用できるデバイスのコンフィギュレーション方式を示します。

図 *1*：*Cisco NX-OS* デバイス コンフィギュレーション方式



次の表に、コンフィギュレーション方式と詳しい説明が記載されているマニュアルを示します。

表 *2*：コンフィギュレーション方式および参考資料

コンフィギュレーション方式	マニュアル
セキュア シェル（SSH）セッション、Telnet セッション、またはコンソール ポートからの CLI	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』
XML 管理インターフェイス	『*Cisco NX-OS XML Management Interface User Guide*』
Cisco DCNM クライアント	*Cisco DCNM* 基本ガイド

CLI または XML 管理インターフェイスによる設定

次のように SSH からコマンドライン インターフェイス（CLI）または XML 管理インターフェイスを使用して、Cisco NX-OS デバイスを設定できます。

- SSH セッション、Telnet セッション、またはコンソール ポート：SSH セッション、Telnet セッション、またはコンソール ポートから CLI を使用してデバイスを設定できます。SSH ではデバイスへの安全な接続が提供されます。詳細については、『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』を参照してください。
- SSH を介して XML 管理インターフェイス：XML 管理インターフェイスを使用してデバイスを設定できます。これは、CLI 機能を補完する NETCONF プロトコルに基づくプログラム方式です。詳細については、『*Cisco NX-OS XML Management User Guide*』を参照してください。

Cisco DCNM での設定

Cisco DCNM クライアントを使用して Cisco NX-OS デバイスを設定できます。Cisco DCNM クライアントはユーザのローカル PC 上で動作し、Cisco DCNM サーバの Web サービスを使用します。Cisco DCNM サーバでは XML 管理インターフェイスを使用してデバイスを設定します。Cisco DCNM クライアントの詳細については、『*Cisco DCNM Fundamentals Guide*』を参照してください。

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、設定ミスが削減され、両方の vPC メンバーを同時に設定する必要性が解消されることから、管理上のオーバーヘッドが軽減されます。

コンフィギュレーション同期化モード（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。

ネットワーク タイム プロトコル

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、ネットワーク内のデバイスから受信するシステム ログなどの時間関連の情報を相互に関連付けることができます。

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。

Cisco Discovery Protocol

Cisco Discovery Protocol（CDP）を使用して、デバイスに直接接続されているすべてのシスコ製機器を検出し、情報を表示できます。CDPは、ルータ、ブリッジ、アクセス サーバ、コミュニケーション サーバ、スイッチを含む、シスコ製のあらゆる機器で動作します。CDPは、メディアにもプロトコルにも依存せず、ネイバー デバイスのプロトコル アドレスを収集し、各デバイスのプラットフォームを検出します。CDP の動作はデータリンク層上に限定されます。異なるレイヤ 3 プロトコルをサポートする 2 つのシステムで相互学習が可能です。

システム メッセージ

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 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 E 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 Network Operations Center 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。

ロールバック

ロールバック機能では、デバイスのコンフィギュレーションのスナップショットまたはチェックポイントを使用して、デバイスをリロードせずに、いつでもそのコンフィギュレーションを再適用できます。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。

Session Manager を使用すると、コンフィギュレーション セッションを作成し、そのセッション内のすべてのコマンドを自動的に適用できます。

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。

Scheduler

スケジューラを使用すると、データの定期的なバックアップや Quality of Service（QoS）ポリシーの変更などのジョブを作成し、管理できます。スケジューラでは、ジョブを指定された時間に一度だけ、または定期的な間隔で実行するなど、ニーズに合わせて開始できます。

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。

RMON

リモート モニタリング（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インターネット技術特別調査委員会（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。

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。

組み込まれている Event Manager

Embedded EventManager（EEM）を使用すると、重要なシステム イベントを検出して処理できます。EEMは、イベント発生時点で、またはしきい値を超えた時点でのイベントモニタリングを含め、イベントを検出して回復する機能を提供します。

オンボード障害ロギング

永続ストレージに障害データを記録するように、デバイスを設定できます。あとで記録されたデータを取得して表示し、分析できます。この On-Board Failure Logging（OBFL: オンボード障害ロギング）機能は、障害および環境情報をモジュールの不揮発性メモリに保管します。この情報は、障害モジュールの分析に役立ちます。

SPAN

イーサネットスイッチドポートアナライザ（SPAN）を設定すると、デバイスの入出力トラフィックをモニタできます。SPAN の機能を使用すると、送信元ポートから宛先ポートへのパケットを複製できます。

ERSPAN

カプセル化リモート スイッチド ポート アナライザ（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 送信元セッションを設定するには、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。

LLDP

リンク層検出プロトコル（LLDP）はベンダーに依存しない、単一方向のデバイス ディスカバリ プロトコルです。このプロトコルでは、ネットワーク上の他のデバイスにネットワーク デバイスから固有の情報をアドバタイズできます。このプロトコルはデータリンク層で動作するため、異なるネットワーク層プロトコルが稼働する 2 つのシステムで互いの情報を学習できます。LLDP はグローバルに、またはインターフェイスごとにイネーブルにすることができます。

TAP アグリゲーション

この機能は、複数の Test Access Point（TAP）の集約（アグリゲーション）ができるようにすることで、データセンターでのモニタリングやトラブルシューティング作業を支援します。TAP 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 デバイスをリンクします。

MPLS ストリッピング

MPLS ストリッピングはパケットから MPLS ラベルを削除する機能を提供するもので、これにより非 MPLS 対応のネットワーク モニタリング ツールによるパケットのモニタが可能になります。

sFlow

サンプリングされた Flow（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し、サンプル データを中央のデータ コレクタに転送できます。

SMU

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。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。

仮想デバイス コンテキスト

Cisco NX-OS では、仮想デバイスをエミュレートする Virtual Device Context（VDCs）に、OS およびハードウェア リソースを分割できます。Cisco Nexus 9000 シリーズ スイッチは、現在のところ、複数の VDC をサポートしていません。すべてのスイッチ リソースはデフォルト VDC で管理されます。

トラブルシューティング機能

Cisco NX-OS には ping、traceroute、Ethanalyzer、Blue Beacon 機能など、さまざまなトラブルシューティング ツールが揃っています。

サービスで障害が発生すると、システムは障害の原因を判定するために使用できる情報を生成します。次の情報ソースが使用可能です。

- サービスの再起動によって、LOG_ERR レベルの Syslog メッセージが生成されます。
- Smart Call Home サービスがイネーブルになっている場合は、サービスの再起動によって Smart Call Home イベントが生成されます。
- SNMP トラップがイネーブルになっている場合、サービスが再起動されると、SNMP エージェントはトラップを送信します。
- サービスの障害がローカル モジュール上で発生した場合は、そのモジュール内で **show processes log** コマンドを入力することで、イベントのログを表示できます。プロセスのログは、スーパーバイザのスイッチオーバーまたはリセット後も保持されます。
- サービスの障害が発生すると、システムのコア イメージ ファイルが生成されます。最新のコア イメージを表示するには、アクティブなスーパーバイザ上で **show cores** コマンドを入力します。スーパーバイザのスイッチオーバーおよびリセットが生じると、コア ファイルは保持されません。ただし、**system cores** コマンドを入力し、ファイル転送ユーティリティ Trivial File Transfer Protocol（TFTP）を使用して、コア ファイルを外部サーバへエクスポートするようシステムを設定できます。
- CISCO-SYSTEM-MIB には、コアのテーブルが含まれています（cseSwCoresTable）。

第 3 章

スイッチ プロファイルの設定

この章では、Cisco Nexus 9000 シリーズ スイッチ上でスイッチ プロファイルを設定する方法について説明します。

スイッチ プロファイルについて

複数のアプリケーションは、ネットワーク内のデバイス間で整合性のある設定が必要です。たとえば、仮想ポート チャネル（vPC）のコンフィギュレーションを同じにする必要があります。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。
- 既存の vPC 設定からスイッチ プロファイルへの移行が行えます。

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（config-sync）
- スイッチ プロファイル モード（config-sync-sp）
- スイッチ プロファイル インポート モード（config-sync-sp-import）

コンフィギュレーション同期化モード

コンフィギュレーション同期モード（config-sync）を使用すると、スイッチ プロファイルを作成できます。

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（config-sync-sp）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の一時バッファに、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実行されません。コマンドの構文は入力時に検証されますが、これは **commit** コマンドを入力した際にそれらのコマンドが正常に実行されるかを保証するものではありません。

スイッチ プロファイル インポート モード

スイッチプロファイルインポート モード（config-sync-sp-import）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スイッチ プロファイルにインポートし、どのコマンドをプロファイルに含めるかを指定できます。このオプションが特に有用なのは、スイッチ プロファイルが非サポートの Cisco NX-OS リリースからこれらをサポートしたリリースにアップグレードする場合です。

スイッチプロファイルのインポートモードを使用して実行コンフィギュレーションから必要な設定をインポートし、その他の追加の変更をスイッチプロファイルまたはグローバル コンフィギュレーション モードに加える前に、これらの変更をコミットしておくことを推奨します。そうしな

いと、インポートした情報の有用性が損なわれ、現在のインポート セッションを放棄して設定プロセスを再実行しなければならなくなる可能性があります。詳細については、スイッチ プロファイルのインポート, （23 ページ）を参照してください。

コンフィギュレーションの検証

2 種類の設定の有効性検査により、スイッチ プロファイルの障害を識別できます。

- 相互排除チェック
- マージ チェック

相互排除チェック

コンフィギュレーション コマンドの相互排除は、config-sync およびグローバル コンフィギュレーション モードでのコマンドの重複を避けるために適用されます。スイッチプロファイルの設定をコミットすると、相互排除（ミューテックス）のチェックがローカル スイッチおよびピア スイッチ（設定されている場合）で実施されます。両方のスイッチで障害がレポートされなければ、コミットは受領されて実行コンフィギュレーションにプッシュされます。

スイッチ プロファイルに含まれるコマンドは、スイッチ プロファイル外に設定できます。

ミューテックス チェックがエラーを識別すると、ミューテックスの障害として報告され、手動で修正する必要が生じます。詳細については、ミューテックスおよびマージ障害の手動での修正, （27 ページ）を参照してください。

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 コンフィギュレーション：インターフェイス コンフィギュレーションは、競合しない限り、スイッチプロファイルと実行コンフィギュレーションのそれぞれに部分的に含まれることができます。
- 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、検証またはコミットプロセスで実行されます。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。詳細については、ミューテックスおよびマージ障害の手動での修正, （27 ページ）を参照してください。

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

スイッチ プロファイルをサポートする Cisco NX-OS リリースから非サポートのリリースにダウングレードする場合は、スイッチ プロファイルの削除が必要です。

旧リリースからスイッチ プロファイルをサポートする Cisco NX-OS リリースにアップグレードする場合、実行コンフィギュレーションコマンドの一部をスイッチプロファイルに移動することができます。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, （16 ページ）を参照してください。

アップグレードは、バッファされた（コミットされていない）設定がある場合に実行できます。ただし、コミットされていない設定は失われます。

スイッチ プロファイルのライセンス要件

製品	ライセンス要件
Cisco NX-OS	スイッチ プロファイルにはライセンスが必要ありません。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

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 プロファイルの設定に関する注意事項および制約事項は次のとおりです。

- Cisco Nexus 9300 シリーズ スイッチのみが、スイッチ プロファイルをサポートします。Cisco Nexus 9500 シリーズ スイッチは、スイッチ プロファイルをサポートしません。
- 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。
- 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 モード（config-sync-sp）で設定できます。
- サポートされているスイッチ プロファイル コマンドは、vPC コマンドに関連します。
- 一度に進行できるのは 1 つのスイッチ プロファイル セッションのみです。別のセッションの開始を試みると失敗します。

スイッチ プロファイルの設定

- スイッチ プロファイル セッションの進行中は、グローバル コンフィギュレーション モードから実行されたコマンドの変更はブロックされます。
- **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。
- レイヤ 3 コマンドはサポートされません。

スイッチ プロファイルの設定

ローカル スイッチでスイッチ プロファイルを作成および設定した後、同期に含める 2 番めのスイッチを追加することができます。

各スイッチに同じ名前を持つスイッチ プロファイルを作成し、スイッチを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。

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
```
ピア スイッチ間の Cisco Fabric Services（CFS）配信をイネーブルにします。

ステップ 3 **config sync**

例：
```
switch(config)# 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 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 [**no**] **sync-peers destination***ip-address*

例：
```
switch(config-sync-sp)# sync-peers destination 10.1.1.1
```
スイッチ プロファイルにスイッチを追加します。宛先 IP アドレスは、同期するスイッチの IP アドレスです。

このコマンドの **no** 形式を使用すると、指定されたスイッチがスイッチ プロファイルから削除されます。

ステップ 6 Cisco Nexus 3164Q スイッチでのみ、次の手順を実行します。

a) **interface***type slot/port*

例：
```
switch(config-sync-sp)# interface ethernet 1/1
switch(config-sync-sp-if)#
```
スイッチ プロファイル 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。

b) **switchport**

例：
```
switch(config-sync-sp-if)# switchport
```
レイヤ 3 インターフェイスをレイヤ 2 インターフェイスに変更します。

c) **exit**

例：
```
switch(config-sync-sp-if)# exit
switch(config-sync-sp)#
```
スイッチ プロファイル 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。

d) **commit**

例：
```
switch(config-sync-sp)# commit
```
現在の設定をコミットします。

ステップ 7 （任意） **end**

例：
```
switch(config-sync-sp)# end
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） **show switch-profile***name***status**

例：
```
switch# show switch-profile abc status
```
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 9 （任意） **show switch-profile***name***peer***ip-address*

例：
```
switch# show switch-profile abc peer 10.1.1.1
```

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 10 （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

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。

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。

手順の概要

1. **config sync**
2. **switch-profile***name*
3. *command*
4. （任意） **show switch-profile***name***buffer**
5. **verify**
6. **commit**
7. （任意） **end**
8. （任意） **show switch-profile***name***status**
9. （任意） **copy running-config startup-config**

手順の詳細

	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。

	コマンドまたはアクション	目的

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 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*
	
	例：
	`sswitch(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-if)# exit`
	`switch(config-sync-sp)#`
		スイッチ プロファイルにコマンドを追加します。

ステップ 4	**show switch-profile***name***buffer** 例： `switch(config-sync-sp)# show switch-profile abc buffer`	（任意） 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。このコマンドは次の処理も行います。 • 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。 • スイッチ プロファイル内のいずれかのスイッチでアプリケーション障害が発生する場合は、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。

ステップ 7	**end**
	
	例：
	`switch(config-sync-sp)# end`
	`switch#`
		（任意）
		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。

	コマンドまたはアクション	目的
ステップ 8	**show switch-profile***name***status** 例： `switch# show switch-profile abc status`	（任意） 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 9	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 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。

はじめる前に

スイッチ プロファイルにコマンドをインポートする前に、スイッチ プロファイル バッファが空であることを確認します。

手順の概要

1. （任意） 手順 4 でインポートするインターフェイスを設定します。
2. **config sync**
3. **switch-profile***name*
4. **import** [**interface***interfaceport*/*slot* | **running-config**]
5. **commit**
6. （任意） **abort**
7. （任意） **end**
8. （任意） **show switch-profile**
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	手順4でインポートするインターフェイスを設定します。
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)# switchport`
	`switch(config-if)# switchport mode trunk`
	
	`switch(config-if)# switchport trunk allowed vlan 12`
	`switch(config-if)# speed 10000`
	`switch(config-if)# spanning-tree port type edge trunk`
	`switch(config)# end`
	`switch#`
		（任意）
		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	**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 3	**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 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	**import** [**interface***interfaceport*/*slot* | **running-config**]
	
	例：
	`switch(config-sync-sp)# import interface 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。次のオプションを使用できます。
		• **import** コマンドに何もオプションを付けずに入力すると、選択されたコマンドをスイッチ プロファイルに追加します。
		• **import interface** オプションは、指定したインターフェイスでサポートされるコマンドを追加します。
		• **running-config** オプションは、サポートされるシステム レベル コマンドを追加します。
		
		（注） 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。

ステップ 5	**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。

	コマンドまたはアクション	目的
ステップ 6	**abort** 例： `switch(config-sync-sp-import)# abort`	（任意） インポート プロセスを中止します。

ステップ 7	**end**
	
	例：
	`switch(config-sync-sp-import)# end`
	`switch#`
		（任意）
		スイッチプロファイルインポートモードを終了し、EXEC モードに戻ります。

ステップ 8	**show switch-profile** 例： `switch# show switch-profile`	（任意） 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 9	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 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。

vPC トポロジでの設定のインポート

2 スイッチ vPC トポロジで設定をインポートできます。

（注） 次の手順の詳細については、この章の該当する項を参照してください。

1. 両方のスイッチで、同じ名前を持つスイッチ プロファイルを設定します。
2. 両方のスイッチに設定を個別にインポートします。

（注） 両方のスイッチで、スイッチプロファイルに移動された設定が同じであることを確認します。同じでない場合、マージ チェックの障害が発生する場合があります。

3. **sync-peers destination** コマンドを入力してスイッチを設定します。
4. 適切な **show** コマンドを入力して、スイッチ プロファイルが同一であることを確認します。

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、コンフィギュレーション同期のブロック、設定のデバッグ、または config-sync 機能が同期しなくなった場合の回復の際に使用できます。

ピア スイッチを分離する際には、スイッチ プロファイルのピア接続を解除し、その後でピア スイッチをスイッチ プロファイルに追加し直す必要があります。

（注） 次の手順の詳細については、この章の該当する項を参照してください。

1. 両方のスイッチで、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. **no sync-peers destination** コマンドをスイッチプロファイルに追加して、両方のスイッチで変更をコミットします。
3. すべての必要なトラブルシューティングの設定を追加します。
4. show running switch-profile の実行結果が、両方のスイッチで同じであることを確認します。
5. **sync-peers destination***ip-address* コマンドを両方のスイッチに追加して、変更をコミットします。
6. ピアが同期していることを確認します。

スイッチ プロファイルの削除

スイッチ プロファイルを削除できます。

手順の概要

1. **config sync**
2. **no switch-profile***name* {**all-config** | **local-config**}
3. （任意） **end**
4. （任意） **copy running-config startup-config**

手順の詳細

	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	**end**
	
	例：
	`switch(config-sync-sp)# end`
	`switch#`
		（任意）
		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 4	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 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。このコマンドを入力すると、config-sync 機能がトリガーとなり、ピア スイッチでも同じ動作が行われます。

ミューテックスおよびマージ障害の手動での修正

ミューテックスおよびマージに障害が発生した場合は、手動で修正できます。

（注） ピア スイッチで競合が生じた場合は、「ピア スイッチの分離, （26 ページ）」の手順に従ってスイッチの問題を修正してください。

1. スイッチ プロファイルのインポート モードを使用して、問題を生じさせているコマンドをスイッチ プロファイルにインポートします。
2. 必要に応じて動作を変更します。

スイッチ プロファイル設定の確認

スイッチ プロファイルの情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show 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 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。
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。

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

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。これには QoS ポリシー（vPC ピア リンクおよびスイッチ プロファイル中の vPC）の設定が含まれます。

1 ローカルおよびピア スイッチで CFS 配信をイネーブルにして、スイッチの管理インターフェイスなど、同期させるスイッチの宛先 IP アドレスを設定します。

```
—Local switch-1#---
switch-1# configure terminal
switch-1(config)# cfs ipv4 distribute
switch-1(config)# interface mgmt 0
```

```
switch-1(config-if)# ip address 30.0.0.81/8

—Peer switch-2#--
switch-2# configure terminal
switch-2(config)# cfs ipv4 distribute
switch-2(config)# interface mgmt 0
switch-2(config-if)# ip address 30.0.0.82/8
```

2 ローカルおよびピア スイッチで新しいスイッチ プロファイルを作成します。

```
—Local switch-1#---
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# sync-peers destination 30.0.0.82
switch-1(config-sync-sp)# end

—Peer switch-2#--
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# sync-peers destination 30.0.0.81
switch-1(config-sync-sp)# end
```

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。

```
switch-1(config-sync-sp)# show switch-profile status

switch-profile : A
------------------------------------------------------------

Start-time: 843992 usecs after Wed Aug 19 17:00:01 2015
End-time: 770051 usecs after Wed Aug 19 17:00:03 2015

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

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

Peer information:
----------------
IP-address: 30.0.0.82
Sync-status: In sync
Status: Commit Success
Error(s):
```

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。

```
switch-1# config sync
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface port-channel 10
switch-1(config-sync-sp-if)# switchport
switch-1(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface port-channel 10
switch-1(config-sync-sp-if)# switchport mode trunk
```

```
switch-1(config-sync-sp-if)# switchport trunk allowed vlan 10
switch-1(config-sync-sp-if)# spanning-tree port type network
switch-1(config-sync-sp-if)# vpc peer-link
switch-1(config-sync-sp-if)# switch-profile switching-mode switchname
switch-1(config-sync-sp-if)# show switch-profile buffer

switch-profile : A
-----------------------------------------------------------
Seq-no Command
-----------------------------------------------------------
1 interface port-channel10
1.1 switchport mode trunk
1.2 switchport trunk allowed vlan 10
1.3 spanning-tree port type network
1.4 vpc peer-link

switch-1(config-sync-sp-if)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# switch-profile A
Switch-Profile started, Profile ID is 1
switch-1(config-sync-sp)# interface ethernet 2/1
switch-1(config-sync-sp-if)# switchport mode trunk
switch-1(config-sync-sp-if)# switchport trunk allowed vlan 10
switch-1(config-sync-sp-if)# spanning-tree port type network
switch-1(config-sync-sp-if)# channel-group 10 mode active
```

5 バッファリングされたコマンドを表示します。

```
switch-1(config-sync-sp-if)# show switch-profile buffer

switch-profile : A
-----------------------------------------------------------
Seq-no Command
-----------------------------------------------------------
1 interface Ethernet2/1
1.1 switchport mode trunk
1.2 switchport trunk allowed vlan 10
1.3 spanning-tree port type network
1.4 channel-group 10 mode active
```

6 スイッチ プロファイルのコマンドを検証します。

```
switch-1(config-sync-sp-if)# verify
Verification Successful
```

7 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。

```
—Local switch-2#--
switch-1(config-sync-sp)# commit
Verification successful...
Proceeding to apply configuration. This might take a while depending on amount of
configuration in buffer.
Please avoid other configuration changes during this time.
Commit Successful
switch-1(config-sync)# end

switch-1# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.82

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
```

```
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active


—Peer switch-2#--
switch-2# show running-config switch-profile

switch-profile A
sync-peers destination 30.0.0.81

interface port-channel10
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
vpc peer-link

interface Ethernet2/1
switchport mode trunk
switchport trunk allowed vlan 10
spanning-tree port type network
channel-group 10 mode active
```

同期ステータスの確認

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-1# show switch-profile status

switch-profile : A
-------------switch-1-----------------------------------------------

Start-time: 912776 usecs after Wed Aug 19 17:03:43 2015
End-time: 868379 usecs after Wed Aug 19 17:03:48 2015

Profile-Revision: 4
Session-type: Commit
Session-subtype: -
Peer-triggered: No
Profile-status: Sync Success

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

Peer information:
----------------
IP-address: 30.0.0.82
Sync-status: In sync
Status: Commit Success
Error(s):
```

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。

```
—— PEER SWITCH-1 ——
switch-1# show running-config switch-profile
```

```
switch-profile A
 sync-peers destination 30.0.0.82

 interface port-channel10
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 vpc peer-link

 interface Ethernet2/1
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 channel-group 10 mode active
switch-1#

─── PEER SWITCH-2 ──
switch-2# show running-config switch-profile

switch-profile A
 sync-peers destination 30.0.0.81

 interface port-channel10
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 vpc peer-link

 interface Ethernet2/1
 switchport mode trunk
 switchport trunk allowed vlan 10
 spanning-tree port type network
 channel-group 10 mode active
switch-2#
```

ローカルとピア 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 台のピア間の最初の正常な同期を表示する例を示します。

```
switch1# show switch-profile sp status

Start-time: 491815 usecs after Mon Jul 20 11:54:51 2015
End-time: 449475 usecs after Mon Jul 20 11:54:58 2015

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):


switch2# show switch-profile sp status

Start-time: 503194 usecs after Mon Jul 20 11:54:51 2015
End-time: 532989 usecs after Mon Jul 20 11:54:58 2015

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):
```

ローカルおよびピア スイッチでの確認とコミットの表示

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

```
switch1# config sync
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)# exit
switch1(config-sync-sp)# 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 Jul 20 17:51:28 2015
End-time: 676451 usecs after Wed Jul 20 17:51:43 2015

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 Mon Jul 20 16:51:28 2015
```

```
End-time: 734702 usecs after Mon Jul 20 16:51:43 2015

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):
```

ローカルおよびピア スイッチ間の成功および失敗した同期の表示

次に、ピア スイッチでスイッチプロファイルの同期ステータスを設定する例を示します。最初の例は正常な同期を示し、2 番めの例はピアの到達不能な状態を示します。

```
switch1# show switch-profile sp peer

switch1# show switch-profile sp peer 10.193.194.52
Peer-sync-status           : In Sync.
Peer-status                : Commit Success
Peer-error(s)              :
switch1#


switch1# 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)              :
```

スイッチ プロファイル バッファの表示

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。

```
switch1# config sync
switch1(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch1(config-sync-sp)# vlan 101
switch1(config-sync-sp-vlan)# ip igmp snooping querier 10.101.1.1
switch1(config-sync-sp-vlan)# exit
switch1(config-sync-sp)# mac address-table static 0000.0000.0001 vlan 101 drop
switch1(config-sync-sp)# interface Ethernet1/2
switch1(config-sync-sp-if)# switchport mode trunk
switch1(config-sync-sp-if)# switchport trunk allowed vlan 101
switch1(config-sync-sp-if)# exit
switch1(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

switch1(config-sync-sp)# buffer-move 3 1
switch1(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

switch1(config-sync-sp)# buffer-delete 1
switch1(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

switch1(config-sync-sp)# buffer-delete all
switch1(config-sync-sp)# show switch-profile sp buffer
```

設定のインポート

次に、インターフェイス コンフィギュレーションをインポートする例を示します。

```
switch# show running-config interface Ethernet1/3

!Command: show running-config interface Ethernet1/3
!Time: Wed Jul 20 18:12:44 2015

version 7.0(3)I2(1)

interface Ethernet1/3
  switchport mode trunk
  switchport trunk allowed vlan 1-100

switch# config sync
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1

switch(config-sync-sp)# import interface Ethernet1/3
switch(config-sync-sp-import)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
1       interface Ethernet1/3
1.1       switchport mode trunk
1.2       switchport trunk allowed vlan 1-100

switch(config-sync-sp-import)# verify
Verification Successful
switch(config-sync-sp-import)# commit
Commit Successful
```

次に、実行コンフィギュレーションにサポートされるコマンドをインポートする例を示します。

```
switch(config-sync)# switch-profile sp
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# import running-config
switch(config-sync-sp-import)# show switch-profile sp buffer
------------------------------------------------------------
Seq-no  Command
------------------------------------------------------------
```

```
1       logging event link-status default
2       vlan 1
3       interface port-channel 3
3.1       switchport mode trunk
3.2       vpc peer-link
3.3       spanning-tree port type network
4       interface port-channel 30
4.1       switchport mode trunk
4.2       vpc 30
4.3       switchport trunk allowed vlan 2-10
5       interface port-channel 31
5.1       switchport mode trunk
5.2       vpc 31
5.3       switchport trunk allowed vlan 11-20
6       interface port-channel 101
6.1       switchport mode fex-fabric
6.2       fex associate 101
7       interface port-channel 102
7.1       switchport mode fex-fabric
7.2       vpc 102
7.3       fex associate 102
8       interface port-channel 103
8.1       switchport mode fex-fabric
8.2       vpc 103
8.3       fex associate 103
9      interface Ethernet1/1
10      interface Ethernet1/2
11      interface Ethernet1/3
12      interface Ethernet1/4
12.1      switchport mode trunk
12.2      channel-group 3
13      interface Ethernet1/5
13.1      switchport mode trunk
13.2      channel-group 3
14      interface Ethernet1/6
14.1      switchport mode trunk
14.2      channel-group 3
15      interface Ethernet1/7
15.1      switchport mode trunk
15.2      channel-group 3
16      interface Ethernet1/8
17      interface Ethernet1/9
17.1      switchport mode trunk
17.2      switchport trunk allowed vlan 11-20
17.3      channel-group 31 mode active
18      interface Ethernet1/10
18.1      switchport mode trunk
18.2      switchport trunk allowed vlan 11-20
18.3      channel-group 31 mode active
19      interface Ethernet1/11
20      interface Ethernet1/12
...
45      interface Ethernet2/4
45.1      fex associate 101
45.2      switchport mode fex-fabric
45.3      channel-group 101
46      interface Ethernet2/5
46.1      fex associate 101
46.2      switchport mode fex-fabric
46.3      channel-group 101
47      interface Ethernet2/6
47.1      fex associate 101
47.2      switchport mode fex-fabric
47.3      channel-group 101
48      interface Ethernet2/7
48.1      fex associate 101
48.2      switchport mode fex-fabric
48.3      channel-group 101
49      interface Ethernet2/8
49.1      fex associate 101
...
89      interface Ethernet100/1/32
90      interface Ethernet100/1/33
```

```
91      interface Ethernet100/1/34
92      interface Ethernet100/1/35
93      interface Ethernet100/1/36
...
105     interface Ethernet100/1/48
```

ファブリック エクステンダ ストレート スルー トポロジ トポロジでの Cisco NX-OS リリース 7.0(3)I2(1) 以降への移行

次に、ファブリック エクステンダのアクティブ/アクティブまたはストレート スルー トポロジで Cisco NX-OS リリース 7.0(3)I2(1) 以降への移行に使用するタスクの例を示します。タスクの詳細については、この章の該当する項を参照してください。

1. 設定が両方のスイッチで同じであることを確認します。
2. 両方のスイッチで、同じ名前を持つスイッチ プロファイルを設定します。
3. 両方のスイッチのすべての vPC ポート チャネルについて、**import interface port-channel***x-y*, **port-channel***z* コマンドを入力します。
4. **show switch-profile***name***buffer** コマンドを入力し、すべての設定が両方のスイッチで正しくインポートされていることを確認します。
5. バッファを編集して不要な設定を削除します。
6. 両方のスイッチで **commit** コマンドを入力します。
7. 両方のスイッチでピア スイッチを設定するには、**sync-peers destination***ip-address* コマンドを入力します。
8. 両方のスイッチが同期されていることを確認するには、**show switch-profile***name***status** コマンドを入力します。

Cisco Nexus 9000 シリーズ スイッチの交換

Cisco Nexus 9000 シリーズ スイッチを交換した場合、交換するスイッチで次の設定手順を実行し、既存の Cisco Nexus 9000 シリーズ スイッチとの同期をとります。この手順は、ハイブリッド ファブリック エクステンダのアクティブ/アクティブ トポロジおよびファブリック エクステンダのストレート スルー トポロジで実行できます。

1. ピア リンク、vPC、アクティブ/アクティブ、ストレート スルー トポロジ ファブリック ポートは、交換スイッチには接続しないでください。
2. 交換するスイッチを起動します。スイッチは設定なしで起動します。
3. 交換するスイッチを設定します。
   - 実行コンフィギュレーションがオフラインで保存されていた場合は、手順 4 から 8 に従って設定を適用します。

- 実行コンフィギュレーションがオフラインで保存されていなかった場合で、設定同期機能がイネーブルの場合、ピア スイッチから実行コンフィギュレーションを取得できます（「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，（28 ページ）」の手順 1 および 2 を参照してください。その後、ここでの手順 9 以降を実施します）。
- いずれの条件にも当てはまらない場合は、手動で設定を追加し、ここでの手順9以降を実施します。

4 コンフィギュレーション同期機能を使用している場合は、コンフィギュレーションファイルを編集し、**sync-peer** コマンドを削除します。

5 mgmt port IP アドレスを設定し、コンフィギュレーション ファイルをダウンロードします。

6 実行コンフィギュレーションに、コンフィギュレーション ファイルをコピーします。

7 **show running-config** コマンドを使用して、設定が正しいことを確認します。

8 交換スイッチが動作していない間に、ピア スイッチでスイッチ プロファイルの設定が変更された場合、スイッチプロファイルでこれらの設定を適用して、**commit** コマンドを入力します。

9 vPC トポロジに含まれるすべてのファブリック エクステンダのストレート スルー トポロジ ポートをシャットダウンします。

10 ファブリック エクステンダのストレート スルー トポロジ ファブリック ポートを接続します。

11 ファブリック エクステンダのストレート スルー トポロジ スイッチがオンラインになるまで待機します。

12 既存スイッチの vPC のロール プライオリティが、交換スイッチよりも上位であることを確認します。

13 ピア リンク ポートをピア スイッチに接続します。

14 スイッチ vPC ポートを接続します。

15 すべてのファブリック エクステンダのストレート スルー vPC ポートで、**no shutdown** コマンドを入力します。

16 交換スイッチにあるすべての vPC スイッチおよびファブリック エクステンダがオンラインになり、トラフィックに中断がないことを確認します。

17 コンフィギュレーション同期機能を使用している場合、手順3でイネーブルにされなかった場合は、sync-peer の設定をスイッチ プロファイルに追加します。

18 コンフィギュレーション同期機能を使用している場合、**show switch-profile***name***status** コマンドを使用し、両方のスイッチが同期されるようにします。

設定の同期

Cisco Nexus 9000 シリーズ スイッチのリブート後の設定の同期化

新しい設定をスイッチプロファイルを使用してピア スイッチにコミットする間にCisco Nexus 9000シリーズスイッチがリブートした場合は、リロード後にピアスイッチを同期するように、次の手順を実行します。

1. 両方のスイッチで、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. **no sync-peers destination** コマンドをスイッチプロファイルに追加して、両方のスイッチで変更をコミットします。
3. 喪失または変更されたコマンドがあれば、すべて追加します。
4. show running switch-profile の実行結果が、両方のスイッチで同じであることを確認します。
5. **sync-peers destination***ip-address* コマンドを両方のスイッチに追加して、変更をコミットします。
6. ピアが同期していることを確認します。

mgmt0 インターフェイスの接続が失われた場合の設定の同期化

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、両方のスイッチに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、両方のスイッチが同期されます。

このシナリオで設定変更が1台のスイッチのみで実行された場合、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たときに成功します。

グローバル コンフィギュレーション モードでのレイヤ 2 からレイヤ 3 への意図しないポート モード変更の復元

config-sync モードでインポートしたポートに関連した設定は、グローバル コンフィギュレーション モードに設定するべきではありません。通常、そうした試みはconfig-sync機能によって拒否され、ミューテックスの警告が表示されます。ただし、ミューテックス チェックの制限により、config-sync モードでレイヤ 2 として設定されたポートがグローバル コンフィギュレーション モードのレイヤ 3（非スイッチポート）に変更された場合は、config-sync 機能による検出と防止は行えません。その結果として、config-syncモードとグローバル コンフィギュレーション モードの同期が失われる可能性があります。こうした場合に、変更を復元するには、以下の手順に従ってください。

1. 両方のスイッチで、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. **no sync-peers destination** コマンドをスイッチプロファイルに追加して、両方のスイッチで変更をコミットします。
3. 現在のインターフェイス設定をインポートします。
4. すべての必要な変更を行い、コミットします。
5. show running switch-profile の実行結果が、両方のスイッチで同じであることを確認します。
6. **sync-peers destination***ip-address* コマンドを両方のスイッチに追加して、変更をコミットします。
7. ピアが同期していることを確認します。

第 4 章

NTP の設定

この章では、Cisco NX-OS デバイスでネットワーク タイム プロトコル（NTP）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。

NTP について

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 では層（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。

NTP アソシエーション

N TP アソシエーションは、次のいずれかになります。

- ピア アソシエーション：デバイスが別のデバイスに同期するか、別のデバイスをそのデバイスに同期させることができます。
- サーバ アソシエーション：デバイスは、サーバに同期します。

設定する必要があるのはアソシエーションの片側だけです。他方のデバイスは自動的にアソシエーションを確立できます。

タイム サーバとしての NTP

Cisco NX-OS デバイスでは、時刻を配信するために NTP を使用できます。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。

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 などの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。

クロック マネージャを使用して、システム内のさまざまなクロックを制御するプロトコルを指定できます。いったんプロトコルを指定すると、システム クロックの更新が始まります。クロック マネージャの設定の詳細については『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』を参照してください。

ハイ アベイラビリティ

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。ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*』を参照してください。

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。VRF に関する詳細情報については、『*Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide*』を参照してください。

NTP のライセンス要件

製品	ライセンス要件
Cisco NX-OS	NTP にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

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。

- NTP サーバ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。
- サーバが 1 台だけの場合は、すべてのデバイスをそのサーバ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（サーバおよびピア）は、最大 64 です。
- VRF で NTP を設定する場合は、NTP サーバおよびピアが、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、シンメトリック 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。
- **ntp authenticate** コマンドが指定されている場合、シンメトリック 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。

NTP のデフォルト設定

次の表に、NTP パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
NTP	イネーブル

パラメータ（Parameters）	デフォルト
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 ロギング	ディセーブル

NTP の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

NTP のイネーブル化/ディセーブル化

NTP をイネーブルまたはディセーブルにできます。NTP はデフォルトでイネーブルです。

手順の概要

1. **configure terminal**
2. [**no**] **feature ntp**
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **feature ntp** 例： `switch(config)# feature ntp`	NTP をイネーブルまたはディセーブルにします。

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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サーバとして動作するよう設定し、既存のタイムサーバ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。

手順の概要

1. **configure terminal**
2. [**no**] **ntp master** [*stratum*]
3. （任意） **show running-config ntp**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **ntp master** [*stratum*] 例： `switch(config)# ntp master`	正規の NTP サーバとしてデバイスを設定します。 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	**show running-config ntp** 例： `switch(config)# show running-config ntp`	（任意） NTP コンフィギュレーションを表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 サーバおよびピアの設定

NTP サーバおよびピアを設定できます。

はじめる前に

使用している NTP サーバと、そのピアの IP アドレスまたはドメイン ネーム システム（DNS）名がわかっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **ntp server** {*ip-address* | *ipv6-address* | *dns-name*} [**key***key-id*] [**maxpoll** *max-poll*] [**minpoll***min-poll*] [**prefer**] [**use-vrf***vrf-name*]
3. [**no**] **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} [**key***key-id*] [**maxpoll** *max-poll*] [**minpoll***min-poll*] [**prefer**] [**use-vrf***vrf-name*]
4. （任意） **show ntp peers**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **ntp server** {*ip-address* | *ipv6-address* | *dns-name*} [**key***key-id*] [**maxpoll** *max-poll*] [**minpoll***min-poll*] [**prefer**] [**use-vrf***vrf-name*] 例： `switch(config)# ntp server 192.0.2.10`	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。*max-poll* および *min-poll* 引数の範囲は 4 ～ 16 秒で、デフォルト値はそれぞれ 6 秒と 4 秒です。 このサーバをデバイスの優先 NTP サーバにするには、**prefer** キーワードを使用します。 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。

	コマンドまたはアクション	目的
		（注） NTP サーバ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	[**no**] **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} [**key***key-id*] [**maxpoll** *max-poll*] [**minpoll***min-poll*] [**prefer**] [**use-vrf***vrf-name*] 例： `switch(config)# ntp peer 2001:0db8::4101`	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。 ピア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 ～ 17 秒で、デフォルト値はそれぞれ 6 秒と 4 秒です。 このピアをデバイスの優先 NTP ピアにするには、**prefer** キーワードを使用します。 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。
ステップ 4	**show ntp peers** 例： `switch(config)# show ntp peers`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

はじめる前に

この手順で指定する予定の認証キーによって、NTP サーバが設定されていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **ntp authentication-key***number***md5***md5-string*
3. **ntp server***ip-address***key***key-id*
4. （任意） **show ntp authentication-keys**
5. [**no**] **ntp trusted-key***number*
6. （任意） **show ntp trusted-keys**
7. [**no**] **ntp authenticate**
8. （任意） **show ntp authentication-status**
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **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`	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。 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** および **ntp peer** コマンドで、**key** キーワードが指定されていない場合は、いずれも認証なしで動作し続けます。
ステップ 4	**show ntp authentication-keys** 例： `switch(config)# show ntp authentication-keys`	（任意） 設定済みの NTP 認証キーを表示します。

	コマンドまたはアクション	目的
ステップ 5	[**no**] **ntp trusted-key***number* 例： `switch(config)# ntp trusted-key 42`	1 つ以上のキー（手順 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	**show ntp trusted-keys** 例： `switch(config)# show ntp trusted-keys`	（任意） 設定済みの NTP の信頼されているキーを表示します。
ステップ 7	[**no**] **ntp authenticate** 例： `switch(config)# ntp authenticate`	NTP認証機能をイネーブルまたはディセーブルにします。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	**show ntp authentication-status** 例： `switch(config)# show ntp authentication-status`	（任意） NTP 認証の状況を表示します。
ステップ 9	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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、デバイスを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。

手順の概要

1. **configure terminal**
2. [**no**] **ntpaccess-group** {**peer** | **serve** | **serve-only** | **query-only**} *access-list-name*
3. （任意） **show ntp access-groups**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **ntpaccess-group** {**peer** | **serve** | **serve-only** | **query-only**} *access-list-name* 例： `switch(config)# ntp access-group peer accesslist1`	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。 NTP がピアに設定されている拒否 ACL ルールに一致した場合、ACL の処理は停止し、次のアクセス グループ オプションに継続されません。 • **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からの NTP 制御クエリーだけを受信するようにします。
ステップ 3	**show ntp access-groups** 例： `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。

	コマンドまたはアクション	目的
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

手順の概要

1. **configure terminal**
2. [**no**] **ntp source***ip-address*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **ntp source***ip-address* 例： `switch(config)# ntp source 192.0.2.1`	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。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

手順の概要

1. **configure terminal**
2. [**no**] **ntp source-interface***interface*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **ntp source-interface***interface* 例： `switch(config)# ntp source-interface ethernet 2/1`	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。サポートされているインターフェイスのリストを表示するには、**?** キーワードを使用します。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

手順の概要

1. **configure terminal**
2. [**no**] **ntp logging**
3. （任意） **show ntp logging-status**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **ntp logging** 例： `switch(config)# 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	**show ntp logging-status** 例： `switch(config)# show ntp logging-status`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 の設定確認

NTP 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 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。
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 peers	すべての NTP ピアを表示します。
show ntp rts-update	RTS アップデートの状況を表示します。

コマンド	目的
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* | *ipv6-addr*} | **name***peer-name*}}	NTP 統計情報を表示します。
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 セッションをクリアするには、**clear ntp session** コマンドを使用します。

NTP 統計情報を消去するには、**clear ntp statistics** コマンドを使用します。

NTP の設定例

次に、NTPパケット内で認証キー42を提示している時刻源とだけ同期するようデバイスを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 authentication-key 42 md5 aNiceKey
switch(config)# ntp server 192.0.2.105 key 42
switch(config)# ntp trusted-key 42
switch(config)# ntp authenticate
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
```

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
Clock Manager	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』

MIB

MIB	MIB のリンク
NTP 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するには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

第 5 章

PTP の設定

この章では、Cisco NX-OS デバイスで高精度時間プロトコル（PTP）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。

PTP について

PTP は、IEEE 1588 で定義された、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。PTP を使用すると、分散したクロックを、イーサネット ネットワークを経由して、1 マイクロ秒以下の精度で同期させることができます。その他の PTP のハードウェア タイムスタンプ機能の用途としては、ERSPAN タイプ III ヘッダーにおいて、エッジ、集約、およびコア スイッチ間でのパケット遅延の計算に使われるタイムスタンプ情報が取得されています。

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 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。シスコでは、スイッチに接続された、同期を必要とするクロックが含まれるサーバを使用して、グランドマスタークロック（10MHz）アップストリームを配置することを推奨します。

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 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。したがって、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 リスタートがサポートされています。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。ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*』を参照してください。

PTP のライセンス要件

製品	ライセンス要件
Cisco NX-OS	PTP にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

PTP の注意事項および制約事項

- 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 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。
- すべての管理メッセージは PTP がイネーブルのポートに転送されます。管理メッセージの処理はサポートされていません。
- PTP は、FEX インターフェイスで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。
- PTP をポート チャネル メンバー ポートでイネーブルにできます。
- PTP は、100G 9408PC ライン カードおよび 100G M4PC 汎用拡張モジュール（GEM）を除く、すべての Cisco Nexus 9000 シリーズおよび 3164Q ハードウェアでサポートされます。

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 *3*：デフォルトの *PTP* パラメータ

パラメータ（Parameters）	デフォルト
PTP	ディセーブル

パラメータ（Parameters）	デフォルト
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 アナウンス タイムアウト	3 アナウンス間隔
PTP 遅延要求間隔	0 ログ秒
PTP 同期間隔	-2 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 パラメータを設定できます。

手順の概要

1. **configure terminal**
2. [**no**] **feature ptp**
3. [**no**] **ptp source***ip-address* [**vrf***vrf*]
4. （任意） [**no**] **ptp domain***number*
5. （任意） [**no**] **ptp priority1***value*
6. （任意） [**no**] **ptp priority2***value*
7. （任意） **show ptp brief**
8. （任意） **show ptp clock**
9. （任意） **show ptp parent**
10. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **feature ptp** 例： `switch(config)# 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	[**no**] **ptp source***ip-address* [**vrf***vrf*] 例： `switch(config)# ptp source 10.10.10.1`	すべての PTP パケットの送信元 IPv4 アドレスを設定します。
ステップ 4	[**no**] **ptp domain***number* 例： `switch(config)# ptp domain 1`	（任意） このクロックで使用するドメイン番号を設定します。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	[**no**] **ptp priority1***value* 例： `switch(config)# ptp priority1 1`	（任意） このクロックをアドバタイズするときに使用する priority1 の値を設定します。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。

	コマンドまたはアクション	目的
ステップ 6	[**no**] **ptp priority2***value* 例： `switch(config)# ptp priority2 1`	（任意） このクロックをアドバタイズするときに使用する priority2 の値を設定します。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	**show ptp brief** 例： `switch(config)# show ptp brief`	（任意） PTP のステータスを表示します。
ステップ 8	**show ptp clock** 例： `switch(config)# show ptp clock`	（任意） 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	**show ptp parent** 例： `switch(config)# show ptp parent`	（任意） PTP の親のプロパティを表示します。
ステップ 10	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

はじめる前に

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。

手順の概要

1. **configure terminal**
2. **interface ethernet***slot/port*
3. [**no**] **ptp**
4. （任意） [**no**] **ptp announce** {**interval***log-seconds* | **timeout***count*}
5. （任意） [**no**] **ptp delay-request minimum interval***log-seconds*
6. （任意） [**no**] **ptp sync interval***log-seconds*
7. （任意） [**no**] **ptp vlan***vlan-id*
8. （任意） **show ptp brief**
9. （任意） **show ptp port interface***interface slot/port*
10. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 ethernet***slot/port*
	
	例：
	`switch(config)# interface ethernet 2/1`
	`switch(config-if)#`
		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 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**] **ptp** 例： `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。
ステップ 4	[**no**] **ptp announce** {**interval***log-seconds* | **timeout***count*} 例： `switch(config-if)# ptp announce interval 3`	（任意） 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 間隔の数を設定します。 PTP アナウンス間隔の範囲は 0 ～ 4 ログ秒で、間隔のタイムアウトの範囲は 2 ～ 4 間隔です。
ステップ 5	[**no**] **ptp delay-request minimum interval***log-seconds* 例： `switch(config-if)# ptp delay-request minimum interval -1`	（任意） ポートがマスター ステートの場合に PTP 遅延メッセージ間で許可される最小間隔を設定します。 ログ（-1）=1 フレーム/秒として、範囲はログ（-1）～ログ（6）秒です。

	コマンドまたはアクション	目的
ステップ 6	[**no**] **ptp sync interval***log-seconds* 例： `switch(config-if)# ptp sync interval 1`	（任意） インターフェイス上の PTP 同期メッセージの送信間隔を設定します。 範囲はログ（-6）～ログ（1）秒です。
ステップ 7	[**no**] **ptp vlan***vlan-id* 例： `switch(config-if)# ptp vlan 1`	（任意） 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	**show ptp brief** 例： `switch(config-if)# show ptp brief`	（任意） PTP のステータスを表示します。
ステップ 9	**show ptp port interface***interface slot/port* 例： `switch(config-if)# show ptp port interface ethernet 2/1`	（任意） PTP ポートのステータスを表示します。
ステップ 10	**copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	（任意） 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。

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 *4*：***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。

コマンド	目的
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 counters [**all** | **interface ethernet***slot/port*]	すべてのインターフェイスまたは指定され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 parent	PTP の親のプロパティを表示します。
show ptp port interfaceethernet*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 time-property	PTP クロック プロパティを表示します。
show running-config ptp [**all**]	PTP の実行コンフィギュレーションを表示します。

PTP の設定例

次に、デバイス上で PTP をグローバルに設定し、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設定する例を示します。

```
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:Mon Dec 22 14:13:24 2014
```

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

```
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 -1
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
```

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	ERSPAN の設定，（281 ページ）

MIB

MIB	MIB のリンク
PTP 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するには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

第 6 章

CDP の設定

この章では、Cisco NX-OS デバイス上で Cisco Discovery Protocol（CDP）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。

CDP について

Cisco Discovery Protocol（CDP）は、ルータ、ブリッジ、アクセス サーバ、コミュニケーション サーバ、スイッチを含め、シスコ製のあらゆる機器で動作する、メディア独立型およびプロトコル独立型のプロトコルです。CDP を使用すると、デバイスに直接接続されているすべてのシスコ デバイスの情報を検出して表示できます。

CDP はネイバーデバイスのプロトコルアドレスを収集し、各デバイスのプラットフォームを検出します。CDP の動作はデータリンク層上に限定されます。異なるレイヤ 3 プロトコルをサポートする 2 つのシステムで相互学習が可能です。

CDP が設定された各デバイスは、マルチキャストアドレスに定期的にアドバタイズメントを送信します。各デバイスは、SNMP メッセージを受信できるアドレスを少なくとも 1 つアドバタイズします。アドバタイズメントには保持時間情報も含まれます。保持時間は、受信デバイスが CDP

情報を削除するまでに保持する時間の長さを表します。アドバタイズメントまたはリフレッシュ タイマーおよびホールド タイマーを設定できます。

CDP Version-2（CDPv2）では、接続デバイスとの間でネイティブ VLAN ID またはポート デュプレックス ステートが一致していないインスタンスを追跡できます。

CDP では、次の Type-Length-Value（TLV）フィールドがアドバタイズされます。

- デバイス ID
- Address
- Port ID
- Capabilities
- Version
- Platform
- ネイティブ VLAN
- Full/Half Duplex
- MTU
- SysName
- SysObjectID
- Management Address
- Physical Location
- VTP

すべての CDP パケットに VLAN ID が含まれます。レイヤ 2 アクセス ポート上で CDP を設定した場合、そのアクセス ポートから送信される CDP パケットには、アクセス ポートの VLAN ID が含まれます。レイヤ 2 トランク ポート上で CDP を設定した場合は、そのトランク ポートから送信される CDP パケットに、トランク ポート上で許可設定されている最小の VLAN ID が含まれます。トランク ポートは、そのトランク ポートの許可 VLAN リストに指定されている VLAN ID であれば、どの VLAN ID が含まれている CDP パケットでも受信できます。VLAN の詳細については、『*Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide*』を参照してください。

VTP 機能のサポート

次の条件に当てはまる場合、CDP は VLAN トランキング プロトコル（VTP）の type-length-value（TLV）フィールドを送信します。

- CDP バージョン 2 がイネーブルになっている
- VTP 機能がイネーブルになっている
- VTP ドメイン名が設定されている

show cdp neighbors detail コマンドを使用すると、VTP 情報を参照できます。

ハイ アベイラビリティ

Cisco NX-OS は、CDP のステートフルおよびステートレス両方のリスタートとスイッチオーバーをサポートします。ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*』を参照してください。

仮想化のサポート

Cisco NX-OS は、CDP のインスタンスを 1 つサポートします。

CDP のライセンス要件

製品	ライセンス要件
Cisco NX-OS	CDP にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

CDP の注意事項と制約事項

CDP 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。

- 接続数が 256 のハブにポートを接続した場合、CDP はポートあたり最大 256 のネイバーを検出できます。
- デバイス上で CDP をイネーブルにする必要があります。イネーブルにしておかないと、インターフェイス上で CDP をイネーブルにできません。
- CDP を設定できるのは、物理インターフェイスおよびポート チャネル上に限られます。

CDP のデフォルト設定

次の表に、CDP パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
CDP	グローバルおよびすべてのインターフェイスでイネーブル
CDP version	Version 2
CDP device ID	Serial number

パラメータ（Parameters）	デフォルト
CDP timer	60 秒
CDP hold timer	180 秒

CDP の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

CDP のグローバルなイネーブルまたはディセーブル

CDP はデフォルトで有効になっています。CDP をディセーブルにしてから、もう一度イネーブルにできます。

インターフェイス上で CDP をイネーブルにするには、先にデバイス上で CDP をイネーブルにしておく必要があります。CDP がグローバルなディセーブルになっているときに、特定のインターフェイス上で CDP をイネーブルにしても、これらのインターフェイス上で CDP がアクティブになることはなく、エラー メッセージが戻ります。

手順の概要

1. **configure terminal**
2. **[no] cdp enable**
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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] cdp enable** 例： `switch(config)# cdp enable`	デバイス全体でCDP機能をイネーブルまたはディセーブルにします。デフォルトではイネーブルです。

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

インターフェイス上での CDP のイネーブルまたはディセーブル

CDPはデフォルトで、インターフェイス上でイネーブルです。インターフェイス上でCDPをディセーブルにできます。

CDP がグローバルなディセーブルになっているときに、特定のインターフェイス上で CDP をイネーブルにしても、これらのインターフェイス上でCDPがアクティブになることはなく、エラーメッセージが戻ります。

手順の概要

1. **configure terminal**
2. **interface***interface slot*/*port*
3. [**no**] **cdp enable**
4. （任意） **show cdp interface***interface slot*/*port*
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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***interface slot*/*port*
	
	例：
	`switch(config)# interface ethernet 1/2`
	`switch(config-if)#`
		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**] **cdp enable** 例： `switch(config-if)# cdp enable`	このインターフェイスでCDPをイネーブルまたはディセーブルにします。デフォルトではイネーブルです。 （注） CDP がデバイス上でグローバルにイネーブルになっていることを確認します。

	コマンドまたはアクション	目的
ステップ 4	**show cdp interface***interface slot/port* 例： `switch(config-if)# show cdp interface ethernet 1/2`	（任意） インターフェイスの CDP 情報を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

CDP オプション パラメータの設定

この手順でオプションのコマンドを使用して CDP を変更できます。

手順の概要

1. **configure terminal**
2. （任意） **cdp advertise** {**v1** | **v2**}
3. （任意） **cdp format device-id** {**mac-address** | **serial-number** | **system-name**}
4. （任意） **cdp holdtime***seconds*
5. （任意） **cdp timer***seconds*
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**cdp advertise** {**v1**	**v2**} 例： `switch(config)# cdp advertise v1`	（任意） デバイスがサポートする CDP のバージョンを設定します。 デフォルトは v2 です。

	コマンドまたはアクション	目的
ステップ 3	**cdp format device-id** {**mac-address** | **serial-number** | **system-name**} 例： `switch(config)# cdp format device-id mac-address`	（任意） CDP デバイス ID を設定します。オプションは次のとおりです。 • **mac-address**：シャーシの MAC アドレス • **serial-number**：シャーシのシリアル番号/組織固有識別子（OUI） • **system-name**：システム名または完全修飾ドメイン名 デフォルトでは **system-name** です。
ステップ 4	**cdp holdtime***seconds* 例： `switch(config)# cdp holdtime 150`	（任意） CDP ネイバー情報を削除するまでに保持する時間を設定します。範囲は 10 ～ 255 秒です。デフォルト値は 180 秒です。
ステップ 5	**cdp timer***seconds* 例： `switch(config)# cdp timer 50`	（任意） CDP がネイバーにアドバタイズメントを送信するリフレッシュ タイムを設定します。範囲は 5 ～ 254 秒です。デフォルトは 60 秒です。
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

CDP コンフィギュレーションの確認

CDP 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 cdp all	CDP がイネーブルになっているすべてのインターフェイスを表示します。
show cdp entry {**all** | **name***entry-name*}	CDP データベース エントリを表示します。
show cdp global	CDP グローバル パラメータを表示します。
show cdp interface*interfaceslot*/*port*	CDP インターフェイスのステータスを表示します。

コマンド	目的
show cdp neighbors {**device-id** | **interface***interfaceslot*/*port*} [**detail**]	CDP ネイバーのステータスを表示します。
show cdp interface*interfaceslot*/*port*	インターフェイスの CDP トラフィック統計を表示します。

インターフェイスの CDP 統計情報を消去するには、**clear cdp counters** コマンドを使用します。

1 つまたはすべてのインターフェイスの CDP キャッシュを消去するには、**clear cdp table** コマンドを使用します。

CDP のコンフィギュレーション例

CDP機能をイネーブルにして、リフレッシュ タイマーおよびホールドタイマーを設定する例を示します。

```
configure terminal
cdp enable
cdp timer 50
cdp holdtime 100
```

その他の参考資料

MIB

MIB	MIB のリンク
CDP 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するには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

第 7 章

システム メッセージ ロギングの設定

この章では、Cisco NX-OS デバイス上でシステム メッセージ ロギングを設定する方法について説明します。

この章の内容は、次のとおりです。

システム メッセージ ロギングについて

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 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。

デフォルトでは、デバイスはターミナル セッションにメッセージを出力し、ログ ファイルにシステム メッセージをログします。

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。

表 *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
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

デバイスは重大度 0、1、または 2 のメッセージのうち、最新の 100 メッセージを 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リモート システム上で動作します。IPv4 または IPv6 の Syslog サーバを最大 8 つ設定できます。

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

（注） 最初のデバイス初期化時に、メッセージが syslog サーバに送信されるのは、ネットワークの初期化後です。

システム メッセージ ロギングのライセンス要件

製品	ライセンス要件
Cisco NX-OS	システム メッセージ ロギングにライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。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。

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 ロギングには次の設定上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。
- syslog サーバが到達可能になる前に出力されたシステム メッセージ（スーパーバイザのアクティブまたはオンライン メッセージなど）は、syslog サーバに送信できません。

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。

表 *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（Parameters）	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 5 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	Seconds
Syslog サーバ ロギング	ディセーブル

パラメータ（**Parameters**）	デフォルト
Syslog サーバ設定の配布	ディセーブル

システム メッセージ ロギングの設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

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

重大度に基づいて、コンソール、Telnet、および SSH セッションにメッセージを記録するようにデバイスを設定できます。

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。

（注） コンソールのボー レートが 9600 ボー（デフォルト）の場合、現在の Critical（デフォルト）ロギング レベルが維持されます。コンソール ロギング レベルを変更しようとすると、必ずエラー メッセージが生成されます。ロギング レベルを上げる（Critical よりも上に）には、コンソールのボー レートを 38400 ボーに変更する必要があります。

手順の概要

1. **terminal monitor**
2. **configure terminal**
3. [**no**] **logging console** [*severity-level*]
4. （任意） **show logging console**
5. [**no**] **logging monitor** [*severity-level*]
6. （任意） **show logging monitor**
7. [**no**] **logging message interface type ethernet description**
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**terminal monitor** 例： `switch# terminal monitor`	デバイスがコンソールにメッセージを記録できるようにします。

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。

ステップ 3	[**no**] **logging console** [*severity-level*] 例： `switch(config)# logging console 3`	指定された重大度とそれより上位の重大度のメッセージをコンソール セッションに記録するように、デバイスを設定します。小さい値は、より高い重大度を示します。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。**no** オプションは、メッセージをコンソールにログするデバイスの機能をディセーブルにします。
ステップ 4	**show logging console** 例： `switch(config)# show logging console`	（任意） コンソール ロギング設定を表示します。
ステップ 5	[**no**] **logging monitor** [*severity-level*] 例： `switch(config)# logging monitor 3`	デバイスが指定された重大度とそれより上位の重大度のメッセージをモニタに記録できるようにします。小さい値は、より高い重大度を示します。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

	コマンドまたはアクション	目的
		• 3：エラー • 4：警告 • 5：通知 • 6：情報 • 7：デバッグ 設定は Telnet および SSH セッションに適用されます。 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。**no** オプションは、メッセージを Telnet および SSH セッションにログするデバイスの機能をディセーブルにします。
ステップ 6	**show logging monitor** 例： `switch(config)# show logging monitor`	（任意） モニタ ロギング設定を表示します。
ステップ 7	**[no] logging message interface type ethernet description** 例： `switch(config)# logging message interface type ethernet description`	システム メッセージ ログ内で、物理的なイーサネット インターフェイスおよびサブインターフェイスに対して説明を追加できるようにします。この説明は、インターフェイスで設定された説明と同じものです。 **no** オプションは、物理イーサネット インターフェイスのシステム メッセージ ログ内のインターフェイス説明の印刷をディセーブルにします。
ステップ 8	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 メッセージの Origin ID の設定

リモートの syslog サーバに送信される syslog メッセージに、ホスト名、IP アドレス、またはテキスト文字列を付けるように Cisco NX-OS を設定できます。

手順の概要

1. **configure terminal**
2. **logging origin-id** {**hostname** | **ip***ip-address* | **string***text-string*}
3. （任意） **show logging origin-id**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**logging origin-id** {**hostname** | **ip***ip-address* | **string***text-string*} 例： `switch(config)# logging origin-id string n9k-switch-abc`	リモートの syslog サーバに送信される syslog メッセージに追加する、ホスト名、IP アドレス、またはテキスト文字列を指定します。

ステップ 3	**show logging origin-id**
	
	例：
	`switch(config)# show logging origin-id`
	`Logging origin_id : enabled (string: n9k-switch-abc)`
		（任意）
		リモートの syslog サーバに送信される syslog メッセージに追加するよう設定された、ホスト名、IP アドレス、またはテキスト文字列が表示されます。

ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

ファイルへのシステム メッセージの記録

システム メッセージをファイルに記録するようにデバイスを設定できます。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。

手順の概要

1. **configure terminal**
2. [**no**] **logging logfile***logfile-name severity-level* [**size***bytes*]
3. **logging event** {**link-status** | **trunk-status**} {**enable** | **default**}
4. （任意） **show logging info**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 logfile***logfile-name severity-level* [**size***bytes*] 例： `switch(config)# logging logfile my_log 6`	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、記録する最小重大度を設定します。小さい値は、より高い重大度を示します。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 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 10485760 です。ファイル サイズは 4096 ～ 4194304 バイトです。

ステップ 3	**logging event** {**link-status**	**trunk-status**} {**enable**	**default**}
			
			例：
			`switch# logging event link-status default`
			`switch(config)#`
				インターフェイス イベントをロギングします。
				• **link-status**：すべての UP/DOWN メッセージおよび CHANGE メッセージをログに記録します。
				• **trunk-status**：すべての TRUNK ステータス メッセージをログに記録します。

	コマンドまたはアクション	目的
		• **enable**：ポート レベルのコンフィギュレーションを上書きしてロギングをイネーブルにするよう、指定します。 • **default**：ロギングが明示的に設定されてないインターフェイスで、デフォルトのロギング設定を使用するよう、指定します。
ステップ 4	**show logging info** 例： `switch(config)# show logging info`	（任意） ロギング設定を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

手順の概要

1. **configure terminal**
2. [**no**] **logging module** [*severity-level*]
3. （任意） **show logging module**
4. [**no**] **logging level***facility severity-level*
5. （任意） **show logging level** [*facility*]
6. （任意） [**no**] **logging level***ethpm*
7. [**no**] **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}
8. （任意） **show logging timestamp**
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 module** [*severity-level*] 例： `switch(config)# logging module 3`	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。**no** オプションを使用すると、モジュール ログ メッセージがディセーブルになります。
ステップ 3	**show logging module** 例： `switch(config)# show logging module`	（任意） モジュール ロギング設定を表示します。
ステップ 4	[**no**] **logging level***facility severity-level* 例： `switch(config)# logging level aaa 2`	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ング 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

	コマンドまたはアクション	目的
		• 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。 **no** オプションを使用すると、指定されたファシリティのロギング重大度がデフォルトのレベルにリセットされます。ファシリティおよび重大度を指定しなかった場合、すべてのファシリティがそれぞれのデフォルト重大度にリセットされます。
ステップ 5	**show logging level** [*facility*] 例： `switch(config)# show logging level aaa`	（任意） 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かった場合は、すべてのファシリティのレベルが表示されます。

ステップ 6	[**no**] **logging level***ethpm*
	
	例：
	`switch(config)# logging level ethpm ?`
	`<0-7>`
	`0-emerg;1-alert;2-crit;3-err;4-warn;5-notif;6-inform;7-debug`
	
	`link-down  Configure logging level for link down syslog`
		（任意）
		イーサネット ポート マネージャの link-up/link-down syslog メッセージのロギングをレベル 3 で有効にします。
		
		no 形式を使用すると、イーサネット ポート マネージャの

	コマンドまたはアクション	目的

	messages
	  link-up Configure logging level for link up syslog messages
	
	switch(config)#logging level ethpm link-down ?
	error ERRORS
	  notif NOTICE
	(config)# logging level ethpm link-down error ?
	
	<CR>
	(config)# logging level ethpm link-down notif ?
	<CR>
	switch(config)#logging level ethpm link-up ?
	error ERRORS
	  notif NOTICE
	(config)# logging level ethpm link-up error ?
	
	<CR>
	(config)# logging level ethpm link-up notif ?
	<CR>
		syslog メッセージにデフォルトのロギング レベルが使用されます。

ステップ 7	**[no] logging timestamp** {**microseconds** | **milliseconds** | **seconds**} 例： switch(config)# logging timestamp milliseconds	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。 （注） このコマンドは、スイッチ内で保持されているログに適用されます。また、外部のロギング サーバには適用されません。
ステップ 8	**show logging timestamp** 例： switch(config)# show logging timestamp	（任意） 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 9	**copy running-config startup-config** 例： switch(config)# copy running-config startup-config	（任意） 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 サーバを最大で 8 台設定できます。

（注） 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 9000 Series NX-OS Unicast Routing Configuration Guide*』を参照してください。

手順の概要

1. **configure terminal**
2. [**no**] **logging server***host* [*severity-level* [**use-vrf***vrf-name*]]
3. **logging source-interface loopback***virtual-interface*
4. （任意） **show logging server**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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* [**use-vrf***vrf-name*]]
	
	例：
	`switch(config)# logging server 192.0.2.253`
	
	例：
	`switch(config)# logging server 2001::)db*::3 5 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。**use-vrf** キーワードを使用すると、メッセージ ロギングを特定の VRF に限定できます。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
		
		デフォルトの発信ファシリティは local7 です。
		no オプションは、指定したホストのロギング サーバを削除します。

	コマンドまたはアクション	目的
		最初の例では、ファシリティ local7 のすべてのメッセージを転送します。2 番目の例では、VRF red で重大度が 5 以下のメッセージを転送します。
ステップ 3	**logging source-interface loopback***virtual-interface* 例： `switch(config)# logging source-interface loopback 5`	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。*virtual-interface* 引数の範囲は 0 ～ 1023 です。
ステップ 4	**show logging server** 例： `switch(config)# show logging server`	（任意） Syslog サーバ設定を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

UNIX または Linux システムでの Syslog サーバの設定

/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。

表 *7*：*syslog.conf* の *Syslog* フィールド

フィールド	説明
ファシリティ	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。 （注） 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。

フィールド	説明
レベル	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 @ 記号を加えたホスト名、ユーザをカンマで区切ったリスト、またはすべてのログイン ユーザを表すアスタリスク（*）を使用できます。

手順の概要

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。
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。
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。

手順の詳細

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。

例：
```
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。

例：
```
$ 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。

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

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。

手順の概要

1. **show logging last***number-lines*
2. **show logging logfile** [**start-time***yyyymmm dd hh:mm:ss*] [**end-time***yyyy mmm dd hh:mm:ss*]
3. **show logging nvram** [**last***number-lines*]
4. **clear logging logfile**
5. **clear logging nvram**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**show logging last***number-lines* 例： `switch# show logging last 40`	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	**show logging logfile** [**start-time***yyyymmm dd hh:mm:ss*] [**end-time***yyyy mmm dd hh:mm:ss*] 例： `switch# show logging logfile start-time 2013 oct 1 15:10:0`	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	**show logging nvram** [**last***number-lines*] 例： `switch# show logging nvram last 10`	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	**clear logging logfile** 例： `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	**clear logging nvram** 例： `switch# clear logging nvram`	NVRAM の記録されたメッセージをクリアします。

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 ロギングの設定情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show 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 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 ロギング重大度設定を表示します。
show logging logfile [**start-time***yyyymmmddhh:mm:ss*] [**end-time***yyyymmmddhh: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 origin-id	リモートの syslog サーバに送信される syslog メッセージに追加するよう設定された、ホスト名、IP アドレス、またはテキスト文字列が表示されます。
show logging server	Syslog サーバ設定を表示します。
show logging timestamp	ロギング タイムスタンプ単位設定を表示します。

システムメッセージロギングのコンフィギュレーション例

システム メッセージ ロギングのコンフィギュレーション例を示します。

```
configure terminal
 logging console 3
 logging monitor 3
 logging logfile my_log 6
 logging module 3
 logging level aaa 2
 logging timestamp milliseconds
 logging server 172.28.254.253
 logging server 172.28.254.254 5 facility local3
 copy running-config startup-config
```

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
システム メッセージ	『*Cisco NX-OS System Messages Reference*』

第 8 章

Smart Call Home の設定

この章では、Cisco NX-OS デバイスの Smart Call Home 機能を設定する方法について説明します。

この章の内容は、次のとおりです。

Smart Call Home の概要

Smart Call Home では、重要なシステム ポリシーに対して電子メールベースの通知が提供されます。豊富なメッセージ フォーマットから選択できるので、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の最適な互換性が得られます。この機能を使用して、ネットワーク サポート エンジニアにポケットベルで連絡したり、ネットワーク オペレーション センターに電子メールを送信したりできます。また、Cisco Smart Call Home サービスを使用して TAC のケースを自動的に生成できます。

Smart Call Home には、次の機能があります。

- 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形式のレポートに最適。
- 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 Schema Definition（XSD）を使用する、調和の取れた判読可能なフォーマット。AML XSD は Cisco.com の Web サイトで公開されています。XML フォーマットでは、TAC との通信が可能になります。

- 複数のメッセージ宛先への同時配信が可能。それぞれの宛先プロファイルには、最大 50 個の電子メール宛先アドレスを設定できます。

宛先プロファイル

宛先プロファイルには、次の情報が含まれます。

- 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）。
- メッセージ重大度：Cisco NX-OS 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 重大度。アラートの Smart Call Home 重大度が宛先プロファイルに設定されたメッセージの重大度に満たない場合、Cisco NX-OS 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 NX-OS 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。このプロファイルは、callhome@cisco.com という E メール コンタクト、最大メッセージ サイズ、およびメッセージ重大度 0 で設定済みです。このプロファイルのデフォルト情報はどれも変更できません。
- 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重大度と同じか、それ以上である場合のみ、デバイスは 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。

表 *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。
設定（Configuration）	設定に関連した定期的なイベント。	**show module** **show version**
診断	診断によって生成されたイベント。	**show diagnostic result module all detail** **show diagnostic result module***number***detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support gold** **show tech-support ha** **show tech-support platform** **show version**

アラート グループ	説明	実行されるコマンド
組み込みイベント マネージャ（EEM）	EEM によって生成されるイベント	**show diagnostic result module all detail** **show diagnostic result module***number***detail** **show module** **show tech-support gold** **show tech-support ha** **show tech-support platform**
Environmental	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 200** **show module** **show version**
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 inventory** **show license usage** **show module** **show sprom all** **show system uptime** **show version**
ライセンス	ライセンスおよびライセンス違反に関連するイベント	**show logging last 200**

アラート グループ	説明	実行されるコマンド
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 diagnostic result module***number***detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support ethpm** **show tech-support gold** **show tech-support ha** **show tech-support platform** **show version**
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 hardware** **show logging last 200** **show module** **show sprom all** **show tech-support ethpm** **show tech-support gold** **show tech-support ha** **show tech-support platform** **show version**
Syslog port group	syslog PORT ファシリティによって生成されるイベント	**show license usage** **show logging last 200**

アラート グループ	説明	実行されるコマンド
システム	装置の動作に必要なソフトウェア システムの障害によって生成されたイベント。	**show diagnostic result module all detail** **show hardware** **show logging last 200** **show module** **show sprom all** **show tech-support ethpm** **show tech-support gold** **show tech-support ha** **show tech-support platform**
Test	ユーザが作成したテスト メッセージ	**show module** **show version**

Smart Call Home は、syslog の重大度を、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 CLI **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。各定義済みまたはユーザ定義宛先プロファイルを、0（最小緊急度）～9（最大緊急度）までの Smart Call Home しきい値と関連付けることができます。デフォルトは 0（全メッセージを送信）です。

Syslog 重大度は、Smart Call Home メッセージ レベルにマッピングされています。

（注） 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 レベルを一覧表示します。

表 *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。他と関係しない、重要性の低い障害です。
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。

Smart Call Home の取得

シスコと直接サービス契約を結んでいる場合は、Smart Call Home サービスに登録できます。Smart Call Home は、Smart Call Home メッセージを分析し、背景説明と推奨措置を提供します。既知の問題、特にオンライン診断障害については、TAC に Automatic Service Request が作成されます。

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。

- Smart Call Home メッセージの分析。必要に応じて、自動サービス要求（詳細な診断情報が含まれる）が作成され、該当する TAC チームにルーティングされるため、問題解決を高速化できます。
- セキュアなメッセージが、ご使用のデバイスから直接、HTTP プロキシ サーバを経由して転送されるか、またはダウンロード可能な転送ゲートウェイ（TG）から転送されます。TG 集約ポイントは、複数のデバイスをサポートする場合またはセキュリティ要件によって、デバイスをインターネットに直接接続できない場合に使用できます。
- あらゆる Smart Call Home デバイスの Smart Call Home メッセージおよび推奨事項、インベントリ情報、設定情報への Web アクセス。この機能によって、関連する現場の注意事項、セキュリティ勧告、および廃止情報にアクセスできます。

登録には次の情報が必要です。

- デバイスの SMARTnet 契約番号
- 電子メール アドレス
- Cisco.com ID

Smart Call Home の詳細については、次の Smart Call Home のページを参照してください。 https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

データベース マージの注意事項

2 つの Smart Call Home データベースをマージする場合は、次の注意事項に従ってください。

- マージされるデータベースには、次の情報が含まれます。
  - マージ側デバイスからの全宛先プロファイルのスーパーセット。
  - 宛先プロファイルの電子メール アドレスとアラート グループ。
  - マージ側デバイスにあるその他の設定情報（メッセージスロットリング、定期的なインベントリなど）。
- 宛先プロファイル名は、マージするデバイス内で重複しないようにしてください。コンフィギュレーションが異なっても、同じ名前は使用できません。プロファイル名が重複している場合、重複するプロファイルの 1 つを削除する必要があります。そうしなければマージ処理が失敗します。

ハイ アベイラビリティ

ステートフルおよびステートレスの両方のリスタートが、Smart Call Home でサポートされます。

仮想化のサポート

Smart Call Home のインスタンスが 1 つサポートされます。次の URL から、Smart Call Home の Web サイトでお客様の連絡先を登録できます。 https://supportforums.cisco.com/community/netpro/solutions/smart_services/smartcallhome

callhome send および **callhome test** コマンドを使用して Smart Call Home をテストできます。

Smart Call Home は、仮想ルーティングおよびフォワーディング（VRF）を認識します。特定の VRF を使用して Smart Call Home SMTP サーバに接続するように Smart Call Home を設定できます。

Smart Call Home のライセンス要件

製品	ライセンス要件
Cisco NX-OS	Smart Call Home にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

Smart Call Home の前提条件

Smart Call Home には、次の前提条件があります。

- 電子メール アドレスにメッセージを送信するには、まず電子メール サーバを設定する必要があります。HTTP を使用してメッセージを送信するには、HTTPS サーバにアクセスでき、Cisco Nexus デバイスに有効な証明書がインストールされている必要があります。
- デバイスは E メール サーバまたは HTTPS サーバと IP 接続している必要があります。
- まず、コンタクト名（SNMP サーバのコンタクト）、電話番号、および住所情報を設定する必要があります。この手順は、受信メッセージの送信元を判別するために必要です。
- Smart Call Home サービスを使用する場合、設定中のデバイスに対応している現在のサービス契約が必要です。

Smart Call Home の注意事項および制約事項

Smart Call Home には、次の注意事項および制限事項があります。

- 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転送（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、デバイスは Smart Call Home メッセージを送信できません。

- Smart Call Home はあらゆる SMTP サーバで動作します。
- Smart Call Home に対して最大 5 つの SMTP サーバを設定できます。

Smart Call Home のデフォルト設定

このテーブルは、Smart Call Home パラメータのデフォルト設定を示します。

表 *10*：デフォルトの *Smart Call Home* パラメータ

パラメータ（Parameters）	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	2,500,000
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	2,500,000
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
プライオリティを指定しなかった場合の SMTP サーバのプライオリティ	50
プロファイルとアラート グループの関連付け	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
Smart Call Home メッセージ レベル	0（ゼロ）
HTTP プロキシ サーバの使用	ディセーブルであり、プロキシ サーバは設定されていない

Smart Call Home の設定

（注） Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があるので注意してください。

次の順序で Smart Call Home 設定を行うことを推奨します。

1. 連絡先情報の設定, （105 ページ）
2. 宛先プロファイルの作成, （107 ページ）
3. アラート グループと宛先プロファイルの関連付け, （112 ページ）
4. （任意） アラート グループへの show コマンドの追加, （113 ページ）
5. Smart Call Home のイネーブル化またはディセーブル化, （121 ページ）
6. （任意） Smart Call Home 設定のテスト, （122 ページ）

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。

手順の概要

1. **configure terminal**
2. **snmp-server contact***sys-contact*
3. **callhome**
4. **email-contact***email-address*
5. **phone-contact***international-phone-number*
6. **streetaddress***address*
7. （任意） **contract-id***contract-number*
8. （任意） **customer-id***customer-number*
9. （任意） **site-id***site-number*
10. （任意） **switch-priority***number*
11. **commit**
12. （任意） **show callhome**
13. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 contact***sys-contact* 例： `switch(config)# snmp-server contact personname@companyname.com`	SNMP sysContact を設定します。

ステップ 3	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**email-contact***email-address* 例： `switch(config-callhome)# email-contact admin@Mycompany.com`	デバイスの主要責任者の電子メール 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。 （注） 有効な電子メール 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	**phone-contact***international-phone-number* 例： `switch(config-callhome)# phone-contact +1-800-123-4567`	デバイスの主要責任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。 （注） 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	**streetaddress***address* 例： `switch(config-callhome)# streetaddress 123 Anystreet st. Anytown,AnyWhere`	デバイスの主要責任者の住所を空白の含まれる英数字ストリングとして設定します。 *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	**contract-id***contract-number* 例： `switch(config-callhome)# contract-id Contract5678`	（任意） サービス契約からこのデバイスの契約番号を設定します。 *contract-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。

	コマンドまたはアクション	目的
ステップ 8	**customer-id***customer-number* 例： `switch(config-callhome)# customer-id Customer123456`	（任意） サービス契約からこのデバイスの顧客番号を設定します。 *customer-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 9	**site-id***site-number* 例： `switch(config-callhome)# site-id Site1`	（任意） このデバイス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	**switch-priority***number* 例： `switch(config-callhome)# switch-priority 3`	（任意） このデバイスのスイッチ 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	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 12	**show callhome** 例： `switch(config-callhome)# show callhome`	（任意） 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 13	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

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

ユーザ定義宛先プロファイルを作成し、メッセージ フォーマットを設定できます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **destination-profile***name*
4. **destination-profile***name***format** {**XML** | **full-txt** | **short-txt**}
5. **commit**
6. （任意） **show callhome destination-profile** [**profile***name*]
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**destination-profile***name* 例： `switch(config-callhome)# destination-profile Noc101`	新しい宛先プロファイルを作成します。名前は、最大 31 文字の英数字で指定できます。
ステップ 4	**destination-profile***name***format** {**XML** | **full-txt** | **short-txt**} 例： `switch(config-callhome)# destination-profile Noc101 format full-txt`	プロファイルのメッセージ フォーマットを設定します。名前は、最大31 文字の英数字で指定できます。
ステップ 5	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 6	**show callhome destination-profile** [**profile***name*] 例： `switch(config-callhome)# show callhome destination-profile profile Noc101`	（任意） 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。

	コマンドまたはアクション	目的
ステップ 7	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

宛先プロファイルに 1 つまたは複数のアラート グループを関連付けます。

宛先プロファイルの変更

定義済みまたはユーザ定義の宛先プロファイルの次の属性を変更できます。

- 宛先電子メール 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）。
- 宛先 URL：アラートの送信先となる HTTP または HTTPS URL。
- 転送方式：電子メールまたは HTTP 転送によって、使用される宛先アドレスのタイプが決まります。
- 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 Smart Call Home メッセージの重大度
- メッセージ サイズ：この宛先プロファイルの電子メール アドレスに送信された Smart Call Home メッセージの許容長さ。

手順の概要

1. **configure terminal**
2. **callhome**
3. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **email-addr***address*
4. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **http***address*
5. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **transport-method** {**email** | **http**}
6. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-level***number*
7. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-size***number*
8. **commit**
9. （任意） **show callhome destination-profile** [**profile***name*]
10. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **email-addr***address* 例： `switch(config-callhome)# destination-profile full-txt-destination email-addr person@place.com`	ユーザ定義または定義済みの宛先プロファイルに電子メール アドレスを設定します。宛先プロファイルには、最大 50 個の電子メール アドレスを設定できます。
ステップ 4	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **http***address* 例： `switch(config-callhome)# destination-profile CiscoTAC-1 http http://site.com/service/callhome`	ユーザ定義または定義済み宛先プロファイルの HTTP または HTTPS URL を設定します。URL の最大文字数は 255 文字です。

	コマンドまたはアクション	目的
ステップ 5	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **transport-method** {**email** | **http**} 例： `switch(config-callhome)# destination-profile CiscoTAC-1 transport-method http`	ユーザ定義または定義済み宛先プロファイルに対応する電子メールまたは HTTP 転送方式を設定します。選択する転送方式のタイプによって、そのタイプに設定された宛先アドレスが決まります。
ステップ 6	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-level***number* 例： `switch(config-callhome)# destination-profile full-txt-destination message-level 5`	この宛先プロファイルの Smart Call Home メッセージの重大度を設定します。Cisco NX-OS がこのプロファイルの宛先に送信するのは、Smart Call Home の重大度が同じか、それ以上のアラートだけです。指定できる範囲は 0 ～ 9 です。9 は最大の重大度を示します。
ステップ 7	**destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-size***number* 例： `switch(config-callhome)# destination-profile full-txt-destination message-size 100000`	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。範囲は 0 ～ 5000000 です。デフォルト値は 2500000 です。
ステップ 8	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 9	**show callhome destination-profile** [**profile***name*] 例： `switch(config-callhome)# show callhome destination-profile profile full-text-destination`	（任意） 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 10	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

宛先プロファイルに 1 つまたは複数のアラート グループを関連付けます。

アラート グループと宛先プロファイルの関連付け

手順の概要

1. **configure terminal**
2. **callhome**
3. **destination-profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**}
4. **commit**
5. （任意） **show callhome destination-profile** [**profile***name*]
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**destination-profile** {*name*	**CiscoTAC-1**	**full-txt-destination**	**short-txt-destination**} **alert-group** {**All**	**Cisco-TAC**	**Configuration**	**Diagnostic**	**EEM**	**Environmental**	**Inventory**	**License**	**Supervisor-Hardware**	**Syslog-group-port**	**System**	**Test**} 例： `switch(config-callhome)# destination-profile Noc101 alert-group All`	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	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home設定コマンドをコミットします。
ステップ 5	**show callhome destination-profile** [**profile***name*] 例： `switch(config-callhome)# show callhome destination-profile profile Noc101`	（任意） 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。

	コマンドまたはアクション	目的
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

任意で **show** コマンドをアラート グループに追加し、SMTP 電子メール サーバを設定します。

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 グループにユーザ定義の CLI **show** コマンドを 5 つまで割り当てることができます。

（注） CiscoTAC-1 宛先プロファイルには、ユーザ定義の CLI **show** コマンドを追加できません。

手順の概要

1. **configure terminal**
2. **callhome**
3. **alert-group** {**Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **user-def-cmd***show-cmd*
4. **commit**
5. （任意） **show callhome user-def-cmds**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**alert-group** {**Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **user-def-cmd***show-cmd* 例： `switch(config-callhome)# alert-group Configuration user-def-cmd show ip route`	**show** コマンド出力を、このアラート グループに送信された Smart 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。
ステップ 4	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	**show callhome user-def-cmds** 例： `switch(config-callhome)# show callhome user-def-cmds`	（任意） アラート グループに追加されたすべてのユーザ定義 **show** コマンドに関する情報を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

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

電子メール サーバの設定

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

Smart Call Home に対して最大 5 つの SMTP サーバを設定できます。サーバは、プライオリティに基づいて試行されます。最もプライオリティの高いサーバが最初に試行されます。メッセージが送信できない場合、制限に達するまでリスト内の次のサーバが試行されます。2 つのサーバのプライオリティが同じ場合は、先に設定された方が最初に試行されます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **transport email mail-server***ip-address* [**port***number*] [**priority***number*] [**use-vrf***vrf-name*]
4. （任意） **transport email from***email-address*
5. （任意） **transport email reply-to***email-address*
6. **commit**
7. （任意） **show callhome transport**
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**transport email mail-server***ip-address* [**port***number*] [**priority***number*] [**use-vrf***vrf-name*] 例： `switch(config-callhome)# transport email mail-server 192.0.2.1 use-vrf Red`	ドメイン ネーム サーバ（DNS）名、IPv4 アドレス、または IPv6 アドレスのいずれかとして SMTP サーバを設定します。任意でポート番号を設定します。ポート範囲は1～65535 です。デフォルトのポート番号は 25 です。 任意で、SMTP サーバのプライオリティを設定します。プライオリティの範囲は 1 ～ 100 で、1 が最高、100 が最低のプライオリティです。プライオリティを指定しない場合、デフォルト値の 50 が使用されます。 また、この SMTP サーバと通信する際に使用するよう任意で VRF を設定します。指定された VRF は、HTTP を使用したメッセージの送信には使用されません。
ステップ 4	**transport email from***email-address* 例： `switch(config-callhome)# transport email from person@company.com`	（任意） 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。

	コマンドまたはアクション	目的
ステップ 5	**transport email reply-to***email-address* 例： `switch(config-callhome)# transport email reply-to person@company.com`	（任意） 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	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 7	**show callhome transport** 例： `switch(config-callhome)# show callhome transport`	（任意） Smart Call Home に対する転送関係のコンフィギュレーションを表示します。
ステップ 8	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

任意で、VRF を使用して HTTP で Smart Call Home メッセージを送信します。

HTTP を使用したメッセージ送信のための VRF 設定

VRF を使用して、HTTP で Smart Call Home メッセージを送信できます。HTTP VRF が設定されていない場合は、デフォルトの VRF を使用して HTTP でメッセージが転送されます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **transport http use-vrf***vrf-name*
4. **commit**
5. （任意） **show callhome**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**transport http use-vrf***vrf-name* 例： `switch(config-callhome)# transport http use-vrf Blue`	HTTP で電子メールおよび他の Smart Call Home メッセージを送信するための VRF を設定します。
ステップ 4	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	**show callhome** 例： `switch(config-callhome)# show callhome`	（任意） Smart Call Home に関する情報を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

任意で、HTTP プロキシ サーバから HTTP メッセージを送信するように Smart Call Home を設定します。

HTTP プロキシ サーバの設定

手順の概要

1. **configure terminal**
2. **callhome**
3. **transport http proxy server***ip-address* [**port***number*]
4. **transport http proxy enable**
5. **commit**
6. （任意） **show callhome transport**
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**transport http proxy server***ip-address* [**port***number*] 例： `switch(config-callhome)# transport http proxy server 192.0.2.1`	HTTP プロキシ サーバのドメイン ネーム サーバ（DNS）の名前、IPv4 アドレス、または IPv6 アドレスを設定します。任意でポート番号を設定します。ポート範囲は 1 ～ 65535 です。デフォルト ポート番号は、8080 です。
ステップ 4	**transport http proxy enable** 例： `switch(config-callhome)# transport http proxy enable`	Smart Call Home で、HTTP プロキシ サーバ経由ですべての HTTP メッセージを送信できるようにします。 （注） プロキシ サーバ アドレスが設定された後にだけ、このコマンドを実行できます。 （注） プロキシ サーバを経由してメッセージを転送するために使用する VRF は、**transport http use-vrf** コマンドを使用して設定したものと同じです。
ステップ 5	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。

	コマンドまたはアクション	目的
ステップ 6	**show callhome transport** 例： `switch(config-callhome)# show callhome transport`	（任意） Smart Call Home に対する転送関係のコンフィギュレーションを表示します。
ステップ 7	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

任意で、定期的にインベントリ通知を送信するようにデバイスを設定します。

定期的なインベントリ通知の設定

デバイス上で現在イネーブルにされて動作しているすべてのソフトウェア サービスのインベントリとともに、ハードウェア インベントリ情報を示すメッセージを定期的に送信するように、デバイスを設定できます。デバイスは、2 種類の Smart Call Home 通知を生成します。定期的コンフィギュレーション メッセージと定期的インベントリ メッセージです。

手順の概要

1. **configure terminal**
2. **callhome**
3. **periodic-inventory notification** [**interval***days*] [**timeofday***time*]
4. **commit**
5. （任意） **show callhome**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**periodic-inventory notification** [**interval***days*] [**timeofday***time*] 例： `switch(config-callhome)# periodic-inventory notification interval 20`	定期的なインベントリ メッセージを設定します。間隔の範囲は 1 ～ 30 日で、デフォルトは 7 です。*time* 引数は HH:MM の形式です。これは、*X* 日ごとに更新が送信される日の時間を定義します（ここで*X*は更新間隔です）。
ステップ 4	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	**show callhome** 例： `switch(config-callhome)# show callhome`	（任意） Smart Call Home に関する情報を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

任意で重複メッセージ スロットリング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、デバイス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、デバイスは同じアラート タイプの以降のメッセージを廃棄します。

手順の概要

1. **configure terminal**
2. **callhome**
3. **no duplicate-message throttle**
4. **commit**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**no duplicate-message throttle** 例： `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	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

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

担当者情報を設定した場合、Smart Call Home 機能をイネーブルにできます。

手順の概要

1. **configure terminal**
2. **callhome**
3. [**no**] **enable**
4. **commit**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	[**no**] **enable** 例： `switch(config-callhome)# 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	**commit** 例： `switch(config-callhome)# commit`	Smart Call Home 設定コマンドをコミットします。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次の作業

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

テスト メッセージを生成して Smart Call Home 通信をテストできます。

手順の概要

1. **configure terminal**
2. **callhome**
3. **callhome send** [**configuration** | **diagnostic**]
4. **callhome test**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**callhome**
	
	例：
	`switch(config)# callhome`
	`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	**callhome send** [**configuration**	**diagnostic**] 例： `switch(config-callhome)# callhome send diagnostic`	設定されたすべての宛先に、指定された Smart Call Home テスト メッセージを送信します。
ステップ 4	**callhome test** 例： `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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

Smart Call Home 設定の確認

Smart Call Home 設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show callhome	Smart Call Home 設定を表示します。

コマンド	目的
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 transport	Smart Call Home に対する転送関係のコンフィギュレーションを表示します。
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 [**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。

Smart Call Home の設定例

Noc101 という宛先プロファイルを作成し、コンフィギュレーションのアラート グループをこのプロファイルに関連付けて、連絡先情報と電子メールの情報を設定した後で、HTTP を介して Smart Call Home メッセージを送信するための VRF を指定する例を示します。

```
configure terminal
snmp-server contact person@company.com
callhome
distribute
email-contact admin@Mycompany.com
phone-contact +1-800-123-4567
streetaddress 123 Anystreet st. Anytown,AnyWhere
destination-profile Noc101 format full-txt
destination-profile full-text-destination email-addr person@company.com
destination-profile full-text-destination message-level 5
destination-profile Noc101 alert-group Configuration
alert-group Configuration user-def-cmd show ip route
transport email mail-server 192.0.2.10 priority 1
transport http use-vrf Blue
enable
commit
```

次に、Smart Call Home メッセージに対して複数の SMTP サーバを設定する例を示します。

```
configure terminal
callhome
transport email mail-server 192.0.2.10 priority 4
transport email mail-server 172.21.34.193
transport email smtp-server 10.1.1.174
transport email mail-server 64.72.101.213 priority 60
transport email from person@company.com
transport email reply-to person@company.com
commit
```

上記のコンフィギュレーションに基づいて、SMTP サーバはこの順序で試行されます。

10.1.1.174（プライオリティ 0）

192.0.2.10（プライオリティ 4）

172.21.34.193（プライオリティ 50、デフォルト）

64.72.101.213（プライオリティ 60）

（注） **transport email smtp-server** コマンドのプライオリティは、最大の 0 です。このコマンドで指定されたサーバは最初に試行され、次に、**transport email mail-server** コマンドで指定されたサーバが、プライオリティの順に試行されます。

次に、HTTP プロキシ サーバからの HTTP メッセージを送信するように、Smart Call Home を設定する例を示します。

```
configure terminal
callhome
transport http proxy server 10.10.10.1 port 4
transport http proxy enable
commit
```

その他の参考資料

イベント トリガー

次の表に、イベント トリガーおよび Smart Call Home メッセージの重大度を示します。

アラート グループ	Event Name	説明	Smart Call Home 重大度
設定（Configuration）	PERIODIC_CONFIGURATION	定期的コンフィギュレーション アップデート メッセージ	2
診断	DIAGNOSTIC_MAJOR_ALERT	GOLDが生成したメジャーアラート	7
	DIAGNOSTIC_MINOR_ALERT	GOLDが生成したマイナーアラート	4
	DIAGNOSTIC_NORMAL_ALERT	Smart Call Home が生成した通常の診断アラート	2

アラート グループ	Event Name	説明	Smart Call Home 重大度
環境および CISCO_TAC	FAN_FAILURE	冷却ファンが障害になりました。	5
	POWER_SUPPLY_ALERT	電源モジュールに関する警告の発生	6
	POWER_SUPPLY_FAILURE	電源モジュールの故障	6
	POWER_SUPPLY_SHUTDOWN	電源モジュールのシャットダウン	6
	TEMPERATURE_ALARM	温度センサーの障害	6
	TEMPERATURE_MAJOR_ALARM	温度が動作メジャーしきい値を超えたことを示す温度センサーの表示	6
	TEMPERATURE_MINOR_ALARM	温度が動作マイナーしきい値を超えたことを示す温度センサーの表示	4
インベントリおよび CISCO_TAC	COLD_BOOT	スイッチの電源が投入され、コールド ブート シーケンスにリセットされます。	2
	HARDWARE_INSERTION	シャーシへの新しいハードウェア コンポーネントの追加	2
	HARDWARE_REMOVAL	シャーシからのハードウェアの取り外し	2
	PERIODIC_INVENTORY	定期的インベントリ メッセージの作成	2
ライセンス	LICENSE_VIOLATION	使用中の機能にライセンスがなく、猶予期間を経てオフになった場合	6
Line module Hardware および CISCO_TAC	LINEmodule_FAILURE	モジュールの動作障害	7
スーパーバイザ ハードウェアおよび CISCO_TAC	SUP_FAILURE	スーパーバイザ モジュールの動作障害	7

アラート グループ	Event Name	説明	Smart Call Home 重大度
Syslog グループ ポート	PORT_FAILURE	ポート ファシリティに対応する syslog メッセージの生成	6
	SYSLOG_ALERT	syslog アラート メッセージの生成	5
システムおよび CISCO_TAC	SW_CRASH	ステートレス リスタートによるソフトウェア プロセス障害、つまりサービスの停止スーパーバイザ モジュールでのプロセス クラッシュに対してメッセージが送信されます。	5
	SW_SYSTEM_INCONSISTENT	ソフトウェアまたはファイル システムにおける不整合の検出	5
テストおよび CISCO_TAC	TEST	ユーザが作成したテストの発生	2

メッセージ フォーマット

Smart 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

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。

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	エラー レベル（システム メッセージに適用されるエラー レベルなど）

共通のイベント メッセージ フィールド

次の表では、フル テキストまたは XML メッセージに共通するイベント メッセージ フィールドの最初のセットについて説明します。

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）
Timestamp	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。	/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
重大度	メッセージの重大度。	/aml/header/level
送信元 ID	ルーティング製品タイプ（Cisco Nexus 9000 シリーズ スイッチなど）。	/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 から取得した製品モデル番号です。 • @ 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：N9K-C9508@C@12345678	/aml/ header/deviceId

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）
Customer 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
契約 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
Server ID	デバイスからメッセージが生成された場合、この ID 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 から取得した製品モデル番号です。 • @ 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：N9K-C9508@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

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）
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
Contact email	このユニットの連絡先として識別される担当者の電子メールアドレス。	/aml/body/sysContactEmail
Contact phone number	このユニットの連絡先である人物の電話番号。	/aml/body/sysContactPhone Number
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
Serial number	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

アラート グループ メッセージ フィールド

次の表に、フル テキストおよび XML のアラート グループ メッセージに固有のフィールドについて説明します。1 つのアラート グループに対して複数の CLI コマンドが実行される場合は、これらのフィールドが繰り返されることがあります。

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）
コマンド出力名	実行された CLI コマンドの正確な名前	/aml/attachments/attachment/name
添付タイプ	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

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）
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

リアクティブおよびプロアクティブ イベント メッセージのフィールド

次の表では、フル テキストまたは XML メッセージのリアクティブおよびプロアクティブ イベント メッセージ形式について説明します。

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 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 の名前	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
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 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

インベントリ イベント メッセージのフィールド

次の表では、フル テキストまたは XML メッセージのインベントリ イベント メッセージ形式について説明します。

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 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 name	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 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

ユーザが作成したテスト メッセージのフィールド

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

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
Process state	プロセスの状態（実行中、中止など）。	/aml/body/process/processState
Process exception	原因コードの例外。	/aml/body/process/exception

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。

```
Severity Level:5
Series:Nexus9000
Switch Priority:0
Device Id:N9K-C9508@C@TXX12345678
Server Id:N9K-C9508C@TXX12345678
Time of Event:2013-05-17 16:31:33 GMT+0000 Message Name:
```

```
Message Type:syslog
System Name:dc3-test
Contact Name:Jay Tester
Contact Email:contact@example.com
Contact Phone:+91-80-1234-5678
Street Address:#1 Any Street
Event Description:SYSLOG_ALERT 2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error
 (0x20) while communicating with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP
 (for:RID_PORT: Ethernet3/1)

syslog_facility:ETHPORT
start chassis information:
Affected Chassis:N9K-C9508
Affected Chassis Serial Number:TXX12345678 Affected Chassis Hardware Version:0.405 Affected
 Chassis Software Version:6.1(2) Affected Chassis Part No:11-11111-11 end chassis information:
start attachment
   name:show logging logfile | tail -n 200
   type:text
   data:
   2013 May 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages) cleared
 by user
   2013 May 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
   2013 May 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
   2013 May 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
 argument: - sshd[14484]
   2013 May 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
   2013 May 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: "System Manager
(gsync controller)" (PID 12000) has finished with error code
SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL (12).
   2013 May 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
   2013 May 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 3504) hasn't
 caught signal 9 (no core).
   2013 May 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero.
   2013 May 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23210)
hasn't caught signal 9 (no core).
   2013 May 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero.
   2013 May 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23294)
hasn't caught signal 9 (no core).
   2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is
becoming active (pre-start phase).
   2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
 active.
   2013 May 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed -
 device_test
   2013 May 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
 message from MRIB. Major type 1807
   2013 May 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN
   2013 May 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]

   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
   2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
   2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
   2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
   2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
   2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
```

```
   2013 May 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
   2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
   2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
   2013 May 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP
   2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
   2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
   2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but
all AC inputs are not connected, ac-redundancy might be affected
   2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but
all AC inputs are not connected, ac-redundancy might be affected
   2013 May 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
  2013 May 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
  2013 May 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
  2013 May 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
  2013 May 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
   2013 May 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
  2013 May 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 4820) hasn't
 caught signal 9 (no core).
   2013 May 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24239)
hasn't caught signal 9 (no core).
   2013 May 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24401)
hasn't caught signal 9 (no core).
   2013 May 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
   2013 May 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message
Core not generated by system for eltm(0). WCOREDUMP(9) returned zero .
   2013 May 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24407)
hasn't caught signal 9 (no core).
  2013 May 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
  2013 May 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
  2013 May 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command <more>
 return code <14>
   2013 May 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
   2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
 with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)
 end attachment start attachment
   type:text
   data:

   dc3-test interfaces:
        Ethernet3/1    Ethernet3/2    Ethernet3/3
        Ethernet3/4    Ethernet3/5    Ethernet3/6
        Ethernet3/7    Ethernet3/8    Ethernet3/9
        Ethernet3/10   Ethernet3/11   Ethernet3/12
        Ethernet3/13   Ethernet3/14   Ethernet3/15
        Ethernet3/16   Ethernet3/17   Ethernet3/18
        Ethernet3/19   Ethernet3/20   Ethernet3/21
        Ethernet3/22   Ethernet3/23   Ethernet3/24
        Ethernet3/25   Ethernet3/29   Ethernet3/30
        Ethernet3/31   Ethernet3/32   Ethernet3/33
        Ethernet3/34   Ethernet3/35   Ethernet3/36
        Ethernet3/37   Ethernet3/38   Ethernet3/39
        Ethernet3/40   Ethernet3/41   Ethernet3/42
        Ethernet3/43   Ethernet3/44   Ethernet3/45
        Ethernet3/46   Ethernet3/47   Ethernet3/48
end attachment
start attachment
   type:text
```

```
   data:
end attachment
start attachment
   name:show license usage
   type:text
   data:
   Feature Ins Lic Status Expiry Date Comments
             Count
   --------------------------------------------------------------------------------
   LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
   --------------------------------------------------------------------------------
end attachment
```

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。

```
<?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.cisco.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.cisco.com/neddce/services/DDCEService</aml-session:To>
<aml-session:Path>
<aml-session:Via>http://www.cisco.com/appliance/uri</aml-session:Via>
</aml-session:Path>
<aml-session:From>http://www.cisco.com/appliance/uri</aml-session:From>
<aml-session:MessageId>1004:TXX12345678:478F82E6</aml-session:MessageId>
</aml-session:Session>
</soap-env:Header>
<soap-env:Body>
<aml-block:Block xmlns:aml-block="http://www.cisco.com/2004/01/aml-block">
<aml-block:Header>
<aml-block:Type>http://www.cisco.com/2005/05/callhome/syslog</aml-block:Type>
<aml-block:CreationDate>2013-05-17 16:31:33 GMT+0000</aml-block:CreationDate>
<aml-block:Builder> <aml-block:Name>DC3</aml-block:Name>
<aml-block:Version>4.1</aml-block:Version>
</aml-block:Builder>
<aml-block:BlockGroup>
<aml-block:GroupId>1005:TXX12345678:478F82E6</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>5</aml-block:Severity>
</aml-block:Header>
<aml-block:Content>
<ch:CallHome xmlns:ch="http://www.cisco.com/2005/05/callhome" version="1.0">
<ch:EventTime>2013-05-17 16:31:33 GMT+0000</ch:EventTime> <ch:MessageDescription>SYSLOG_ALERT
 2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)
 </ch:MessageDescription>
<ch:Event> <ch:Type>syslog</ch:Type> <ch:SubType></ch:SubType> <ch:Brand>Cisco</ch:Brand>
<ch:Series>Nexus9000</ch:Series> </ch:Event> <ch:CustomerData> <ch:UserData>
<ch:Email>contact@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:DeviceId>N9K-C9508@C@TXX12345678</ch:DeviceId>
</ch:ContractData>
<ch:SystemInfo>
<ch:Name>dc3-test</ch:Name>
<ch:Contact>Jay Tester</ch:Contact> <ch:ContactEmail>contact@example.com</ch:ContactEmail>
<ch:ContactPhoneNumber>+91-80-1234-5678</ch:ContactPhoneNumber>
<ch:StreetAddress>#1, Any Street</ch:StreetAddress> </ch:SystemInfo> </ch:CustomerData>
<ch:Device> <rme:Chassis xmlns:rme="http://www.cisco.com/rme/4.1">
<rme:Model>N9K-C9508</rme:Model>
<rme:HardwareVersion>0.405</rme:HardwareVersion>
```

```
<rme:SerialNumber>TXX12345678</rme:SerialNumber>
</rme:Chassis>
</ch:Device>
</ch:CallHome>
</aml-block:Content>
<aml-block:Attachments>
<aml-block:Attachment type="inline">
<aml-block:Name>show logging logfile | tail -n 200</aml-block:Name> <aml-block:Data
encoding="plain">
2013 May 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages) cleared by
 user
2013 May 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16: Invalid
argument: - sshd[14484]
2013 May 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 12:11:18 dc3-test %SYSMGR-STANDBY-5-SUBPROC_TERMINATED: \"System Manager (gsync
 controller)\" (PID 12000) has finished with error code SYSMGR_EXITCODE_GSYNCFAILED_NONFATAL
 (12).
2013 May 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2013 May 17 16:28:44 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 3504)
hasn&amp;apos;t caught signal 9 (no core).
2013 May 17 16:29:08 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23210)
hasn&amp;apos;t caught signal 9 (no core).
2013 May 17 16:29:17 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2579 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23294)
hasn&amp;apos;t caught signal 9 (no core).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is becoming
 active (pre-start phase).
2013 May 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is becoming
active.
2013 May 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed -
device_test
2013 May 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR: netstack [4336] Unrecognized
message from MRIB. Major type 1807
2013 May 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN
2013 May 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0 - ntpd[19045]
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET: netstack [4336] HA client filter
recovery failed (0)
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19072]
2013 May 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19073]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19079]
2013 May 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: Telnet disabled, removing -
dcos-xinetd[19105]
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 2 present but all
```

```
 AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:35 dc3-test %PLATFORM-2-PS_AC_IN_MISSING: Power supply 3 present but all
 AC inputs are not connected, ac-redundancy might be affected
2013 May 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2013 May 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:30:47 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 4820)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:02 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24239)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:14 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24401)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
2013 May 17 16:31:23 dc3-test %SYSMGR-3-BASIC_TRACE: core_copy: PID 2630 with message Core
 not generated by system for eltm(0). WCOREDUMP(9) returned zero .
2013 May 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24407)
hasn&apos;t caught signal 9 (no core).
2013 May 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2013 May 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR: netstack [4336] (null)
2013 May 17 16:31:33 dc3-test %ETHPORT-2-IF_SEQ_ERROR: Error (0x20) while communicating
with component MTS_SAP_ELTM opcode:MTS_OPC_ETHPM_PORT_PHY_CLEANUP (for:RID_PORT: Ethernet3/1)
  </aml-block:Data> </aml-block:Attachment> <aml-block:Attachment type="inline">
<aml-block:Name> <aml-block:Data encoding="plain">
dc3-test interfaces:
   Ethernet3/1     Ethernet3/2     Ethernet3/3
   Ethernet3/4     Ethernet3/5     Ethernet3/6
   Ethernet3/7     Ethernet3/8     Ethernet3/9
   Ethernet3/10    Ethernet3/11    Ethernet3/12
   Ethernet3/13    Ethernet3/14    Ethernet3/15
   Ethernet3/16    Ethernet3/17    Ethernet3/18
   Ethernet3/19    Ethernet3/20    Ethernet3/21
   Ethernet3/22    Ethernet3/23    Ethernet3/24
   Ethernet3/25    Ethernet3/26    Ethernet3/27
   Ethernet3/28    Ethernet3/29    Ethernet3/30
   Ethernet3/31    Ethernet3/32    Ethernet3/33
   Ethernet3/34    Ethernet3/35    Ethernet3/36
   Ethernet3/37    Ethernet3/38    Ethernet3/39
   Ethernet3/40    Ethernet3/41    Ethernet3/42
   Ethernet3/43    Ethernet3/44    Ethernet3/45
   Ethernet3/46    Ethernet3/47    Ethernet3/48



</aml-block:Data>
</aml-block:Attachment>
<aml-block:Attachment type="inline">
<aml-block:Name> <aml-block:Data encoding="plain"> <!> </aml-block:Data>
</aml-block:Attachment> <aml-block:Attachment type="inline"> <aml-block:Name>show license
usage</aml-block:Name> <aml-block:Data encoding="plain">
Feature Ins Lic Status Expiry Date Comments
                    Count
--------------------------------------------------------------------------------
LAN_ENTERPRISE_SERVICES_PKG Yes - Unused Never -
--------------------------------------------------------------------------------

</aml-block:Data>
```

```
</aml-block:Attachment>
</aml-block:Attachments>
</aml-block:Block>
</soap-env:Body>
</soap-env:Envelope>
```

MIB

MIB	MIB のリンク
Smart Call Home 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するには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

第 9 章

ロールバックの設定

この章では、Cisco NX-OS デバイスでロールバックを設定する方法について説明します。

この章の内容は、次のとおりです。

ロール バックについて

ロールバックを使用すると、Cisco NX-OS 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 チェックポイントを使用して、デバイス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デバイスに再適用できます。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。

Cisco NX-OS は、システムのチェックポイントを自動的に作成します。ユーザまたはシステムのチェックポイントのいずれかを使用して、ロールバックを実行できます。

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、次のロールバック タイプを発生させることができます。

- atomic：エラーが発生しなかった場合に限り、ロールバックを実装します。
- best-effort：ロールバックを実装し、エラーがあってもスキップします。
- stop-at-first-failure：エラーが発生した場合は中止されるロールバックを実装します。

デフォルトのロールバック タイプは atomic です。

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可能になった時点で、現在の実行コンフィギュレーションに適用される変更を確認してから、ロールバック操作にコミットできます。ロールバック操作時にエラーが発生した場合は、操作を取り消すか、またはエラーを無視してロールバック操作を続行するかを選択できます。操作を取り消した場合、Cisco NX-OS はエラーが発生する前に適用した変更のリストを提示します。これらの変更は手動で処理する必要があります。

自動的に生成されるシステム チェックポイント

Cisco NX-OS ソフトウェアは、コンフィギュレーション情報が消失しないよう、システム チェックポイントを自動的に生成します。システム チェックポイントは次のイベントによって生成されます。

- **no feature** コマンドで、イネーブルになっている機能をディセーブルにする
- **no router bgp** コマンドや **no ip pim sparse-mode** コマンドで、レイヤ 3 プロトコルのインスタンスを削除する
- 機能のライセンスの有効期限が切れる

これらのイベントのいずれかによってシステム コンフィギュレーションの変更が生じると、この機能ソフトウェアによって、システム チェックポイントが作成されます。これを使用すると、以前のシステム コンフィギュレーションへロールバックできます。システムで生成されたチェックポイント ファイルの名前は「system-」で始まり、機能名が含まれています。たとえば、EIGRP 機能を最初にディセーブルにすると、システムは、system-fm-__inst_1__eigrp という名前のチェックポイントを作成します。

ハイ アベイラビリティ

checkpoint または checkpoint checkpoint_name コマンドを使用してチェックポイントが作成されるときは必ず、チェックポイントはスタンバイ ユニットと同期されます。

ロールバックではチェックポイント操作の状況を記憶しています。このためチェックポイント操作が中断された場合、およびシステムが不整合の状態になった場合には、ロールバック操作を続行する前に、ロールバックでチェックポイント操作（スタンバイ ユニットへのチェックポイントの同期化）を完了できます。

チェックポイント ファイルは、プロセスのリスタート後またはスーパーバイザのスイッチオーバー後も引き続き使用できます。プロセスの再起動中またはスーパーバイザのスイッチオーバー中に中断された場合でも、操作を続行する前にチェックポイントが正常に完了します。スーパーバイザのスイッチオーバーでは、チェックポイントは新しいアクティブ ユニットで完了します。

ロールバック操作中にプロセスの再起動またはスーパーバイザのスイッチオーバーが生じた場合は、再起動またはスイッチオーバーが完了した後で、ロールバックが以前の状態から再開し、正常に終了します。

仮想化のサポート

Cisco NX-OS は実行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。異なるチェックポイント コピーを作成できます。

ロールバックのライセンス要件

製品	ライセンス要件
Cisco NX-OS	ロールバック機能にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

ロール バックの前提条件

ロール バックを設定するには、network-admin のユーザ権限が必要です。

ロールバックの注意事項と制約事項

ロールバック設定時の注意事項と制限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。
- チェックポイント ファイル名の長さは、最大 80 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。

- システムで **write erase** または **reload** コマンドを実行すると、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。
- 異なるソフトウェア バージョン間でのチェックポイントのロールバックはサポートされていませんが、ユーザは自己判断でロールバックを実行し、best-effort モードでエラーから回復できます。
- 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。
- **checkpoint** および **checkpoint***checkpoint_name* コマンドを使用して作成されるチェックポイントは、スイッチオーバーの直後に出現します。
- チェック ポイントは、リロードの前に **write erase** コマンドを発行しない限り、リロードの直後に出現します。
- **checkpoint***checkpoint_name* コマンドを使用して作成されたファイルで、その他の ASCII タイプのファイルではない場合に限り、ブートフラッシュ時のファイルへのロールバックがサポートされます。
- 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。

ロールバックのデフォルト設定

次の表に、ロールバック パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
ロールバック タイプ	アトミック

ロールバックの設定

（注） Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があるので注意してください。

チェックポイントの作成

設定には、最大 10 個のチェック ポイントを作成できます。

手順の概要

1. [**no**] **checkpoint** {[*cp-name*] [**description***descr*] | **file***file-name*}
2. （任意） **show checkpoint***cp-name* [**all**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	[**no**] **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。 **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 2	**show checkpoint***cp-name* [**all**] 例： `switch# show checkpoint stable`	（任意） 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。

手順の概要

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*}
2. **rollback running-config** {**checkpoint***cp-name* | **file***cp-file*} [**atomic** | **best-effort** | **stop-at-first-failure**]

手順の詳細

	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： `switch# show diff rollback-patch checkpoint stable running-config`	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。
ステップ 2	**rollback running-config** {**checkpoint***cp-name* | **file***cp-file*} [**atomic** | **best-effort** | **stop-at-first-failure**] 例： `switch# rollback running-config checkpoint stable`	指定されたチェックポイント名またはファイルへのロールバックを作成します。次のロールバック タイプを実装できます。 • atomic：エラーが発生しなかった場合に限り、ロールバックを実装します。 • best-effort：ロールバックを実装し、エラーがあってもスキップします。 • stop-at-first-failure：エラーが発生した場合は中止されるロールバックを実装します。 デフォルトは atomic です。 次に、ユーザ チェックポイント名に対するロールバックを実装する例を示します。

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情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show 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**]	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**]	すべてのチェック ポイントの一覧を表示します。表示されるチェックポイントを、ユーザ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。

コマンド	目的
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。

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。

ロールバックのコンフィギュレーション例

次に、チェックポイント ファイルを作成して、ユーザ チェックポイント名に対する best-effort ロールバックを実装する例を示します。

```
checkpoint stable
rollback running-config checkpoint stable best-effort
```

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
コンフィギュレーション ファイル	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』

第 10 章

Session Manager の設定

この章では、Cisco NX-OS デバイスで Session Manager を設定する方法について説明します。

この章の内容は、次のとおりです。

Session Manager について

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 はコンフィギュレーション全体を確認して、デバイスに対する変更を実行します。障害が発生した場合、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。

ハイ アベイラビリティ

Session Manager セッションは、スーパーバイザのスイッチオーバー後も引き続き使用できます。セッションはソフトウェア リロード後までは維持されません。

Session Manager のライセンス要件

製品	ライセンス要件
Cisco NX-OS	Session Manager にライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

Session Manager の前提条件

使用する予定の Session Manager コマンドをサポートする権限があることを確認してください。

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）および Quality of Service（QoS）機能だけをサポートします。
- 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。
- 複数のコンフィギュレーション セッションまたはコンフィギュレーション ターミナル モードで、コンフィギュレーション コマンドを同時に実行することはできません。パラレル コンフィギュレーション（1 つのコンフィギュレーション セッションと 1 つのコンフィギュレーション ターミナルのようなもの）は、コンフィギュレーション セッションで確認または検証が失敗する原因になることがあります。
- コンフィギュレーションセッションであるインターフェイスを設定中に、そのインターフェイスをリロードすると、そのときにインターフェイスがデバイス上になくても Session Manager コマンドを受け取ることができます。

Session Manager の設定

（注） Cisco NX-OS コマンドは Cisco IOS コマンドと異なる場合があるので注意してください。

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。

手順の概要

1. **configure session***name*
2. （任意） **show configuration session** [*name*]
3. （任意） **save***location*

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure session***name*
	
	例：
	`switch# configure session myACLs`
	`switch(config-s)#`
		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	**show configuration session** [*name*] 例： `switch(config-s)# show configuration session myACLs`	（任意） セッションの内容を表示します。
ステップ 3	**save***location* 例： `switch(config-s)# save bootflash:sessions/myACLs`	（任意） セッションをファイルに保存します。保管場所には bootflash:、slot0:、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。

手順の概要

1. **configure session***name*
2. **ip access-list***name*
3. （任意） **permit***protocol source destination*
4. **interface***interface-type number*
5. **ip access-group***name* {**in** | **out**}
6. （任意） **show configuration session** [*name*]

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure session***name*
	
	例：
	`switch# configure session myacls`
	`switch(config-s)#`
		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	**ip access-list***name*
	
	例：
	`switch(config-s)# ip access-list acl1`
	`switch(config-s-acl)#`
		ACL を作成し、その ACL のコンフィギュレーション モードを開始します。

ステップ 3	**permit***protocol source destination* 例： `switch(config-s-acl)# permit tcp any any`	（任意） ACL に許可文を追加します。

ステップ 4	**interface***interface-type number*
	
	例：
	`switch(config-s-acl)# interface ethernet 2/1`
	`switch(config-s-if)#`
		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	**ip access-group***name* {**in**	**out**} 例： `switch(config-s-if)# ip access-group acl1 in`	アクセス グループを適用するトラフィックの方向を指定します。
ステップ 6	**show configuration session** [*name*] 例： `switch(config-s-if)# show configuration session myacls`	（任意） 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 モードで次のコマンドを使用して、セッションを確認します。

コマンド	目的
verify [**verbose**] 例： `switch(config-s)# verify`	既存のハードウェアおよびソフトウェアのコンフィギュレーションおよびリソースに基づいて、コンフィギュレーション全体を確認します。Cisco NX-OS は、設定がこの確認で合格しなかった場合にエラーを返します。

セッションのコミット

セッション モードで次のコマンドを使用して、セッションをコミットします。

コマンド	目的
commit [**verbose**] 例： `switch(config-s)# commit`	現在のセッションで行われたコンフィギュレーションの変更を検証し、有効な変更をデバイスに適用します。検証に失敗した場合、Cisco NX-OS は元の設定に戻ります。

セッションの保存

セッション モードで次のコマンドを使用して、セッションを保存します。

コマンド	目的
save*location* 例： `switch(config-s)# save bootflash:sessions/myACLs`	（任意）セッションをファイルに保存します。保管場所には bootflash:、slot0:、または volatile: を指定できます。

セッションの廃棄

セッション モードで次のコマンドを使用して、セッションを廃棄します。

コマンド	目的

abort
例：
`switch(config-s)# abort`
`switch#`
	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 設定の確認

Session Managerのコンフィギュレーション情報を表示するには、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。

Session Manager のコンフィギュレーション例

Session Managerを使用してACLコンフィギュレーションを作成し、コミットする例を示します。

```
switch# configure session ACL_tcp_in
Config Session started, Session ID is 1
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-s)# ip access-list ACL1
switch(config-s-acl)# permit tcp any any
switch(config)# interface e 7/1
switch(config-if)# ip access-group ACL1 in
switch(config-if)# exit
switch(config)# exit
switch# config session ACL_tcp_in
Config Session started, Session ID is 1
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-s)# verify
Verification Successful
switch(config-s)# commit
Commit Successful
switch#
```

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
コンフィギュレーション ファイル	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』

関連資料

第 11 章

スケジューラの設定

この章では、Cisco NX-OS デバイス上でスケジューラ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。

スケジューラについて

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。

- Quality of Service（QoS）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、特定の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、ジョブの実行が定期的な間隔で繰り返されます。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。
  - One-time mode：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を認証します。リモート認証で得たユーザ クレデンシャルは短時間しか保有されないため、スケジューリングされたジョブをサポートできません。ジョブを作成するユーザ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。

ログ

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。

ハイ アベイラビリティ

スケジューリングされたジョブは、スーパーバイザのスイッチオーバーまたはソフトウェアのリロード後も使用可能です。

スケジューラのライセンス要件

製品	ライセンス要件
Cisco NX-OS	スケジューラにはライセンスは不要です。ライセンス パッケージに含まれていない機能はnx-osイメージにバンドルされており、無料で提供されます。Cisco 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。

スケジューラの前提条件

スケジューラの前提条件は次のとおりです。

- 条件付き機能をイネーブルにしてからでなければ、ジョブでそれらの機能を設定できません。
- ライセンスの必要な機能をジョブで設定するには、各機能の有効なライセンスをインストールしておく必要があります。
- スケジューリングされたジョブを設定するには、network-admin ユーザ特権が必要です。

スケジューラの注意事項および制約事項

スケジューラ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。

- 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。
- 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、ジョブは開始しません。
- 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**、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。

スケジューラのデフォルト設定

この表は、スケジューラのデフォルト設定を示します。

パラメータ（Parameters）	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またはディセーブル化

ジョブを設定してスケジュールできるようにスケジューラ機能をイネーブルにすることができ、または、スケジューラをイネーブルにした後にスケジューラ機能をディセーブルにすることもできます。

手順の概要

1. **configure terminal**
2. **[no] feature scheduler**
3. （任意） **show scheduler config**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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] feature scheduler** 例： `switch(config)# feature scheduler`	スケジューラをイネーブルまたはディセーブルにします。

ステップ 3	**show scheduler config**
	
	例：
	`switch(config)# show scheduler config`
	`config terminal`
	`feature scheduler`
	`scheduler logfile size 16`
	`end`
		（任意）
		スケジューラ設定を表示します。

	コマンドまたはアクション	目的
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

スケジューラ ログ ファイル サイズの定義

ジョブ、スケジュール、およびジョブ出力をキャプチャするログ ファイルのサイズを設定できます。

手順の概要

1. **configure terminal**
2. **scheduler logfile size***value*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**scheduler logfile size***value* 例： `switch(config)# scheduler logfile size 1024`	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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

リモート ユーザ認証の設定

ジョブの設定およびスケジューリングを行うユーザにリモート認証を使用するように、スケジューラを設定できます。

（注） リモート ユーザは、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。

（注） **show running-config** コマンドの出力では、リモート ユーザ 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。

手順の概要

1. **configure terminal**
2. **scheduler aaa-authentication password** [**0** | **7**] *password*
3. **scheduler aaa-authentication username***name***password** [**0** | **7**] *password*
4. （任意） **show running-config | include "scheduler aaa-authentication"**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**scheduler aaa-authentication password** [**0**	**7**] *password* 例： `switch(config)# scheduler aaa-authentication password X12y34Z56a`	現在ログインしているユーザ用のクリア テキスト パスワードを設定します。
ステップ 3	**scheduler aaa-authentication username***name***password** [**0**	**7**] *password* 例： `switch(config)# scheduler aaa-authentication username newuser password Z98y76X54b`	リモート ユーザのクリア テキスト パスワードを設定します。

	コマンドまたはアクション	目的
ステップ 4	**show running-config | include "scheduler aaa-authentication"** 例： `switch(config)# show running-config | include "scheduler aaa-authentication"`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

ジョブの定義

ジョブを定義して、ジョブ名とコマンド シーケンスを指定することができます。

注意 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。

手順の概要

1. **configure terminal**
2. **scheduler job name***string*
3. *command1* ;[*command2* ;*command3* ;...]
4. （任意） **show scheduler job** [**name***name*]
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**scheduler job name***string*
	
	例：
	`switch(config)# scheduler job name backup-cfg`
	`switch(config-job)`
		ジョブを作成し、ジョブ コンフィギュレーション モードを開始します。
		backup-cfg という名前のスケジューラ ジョブを作成する例を示します。

ステップ 3	*command1* ;[*command2* ;*command3* ;...]
	
	例：
	`switch(config-job)# cli var name timestamp $(TIMESTAMP) ;copy running-config bootflash:/$(SWITCHNAME)-cfg.$(timestamp) ;copy bootflash:/$(SWITCHNAME)-cfg.$(timestamp) tftp://1.2.3.4/ vrf management`
	`switch(config-job)#`
		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（「;」）で区切る必要があります。
		実行コンフィギュレーションを bootflash 内のファイルに保存し、ファイルを bootflash から TFTP サーバにコピーするスケジューラ ジョブを作成する例を示します。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されます。

ステップ 4	**show scheduler job** [**name***name*] 例： `switch(config-job)# show scheduler job`	（任意） ジョブ情報を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

ジョブの削除

スケジューラからジョブを削除できます。

手順の概要

1. **configure terminal**
2. **no scheduler job name***string*
3. （任意） **show scheduler job** [**name***name*]
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 scheduler job name***string*
	
	例：
	`switch(config)# no scheduler job name configsave`
	`switch(config-job)`
		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。

ステップ 3	**show scheduler job** [**name***name*] 例： `switch(config-job)# show scheduler job name configsave`	（任意） ジョブ情報を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

タイムテーブルの定義

1 つまたは複数のジョブで使用するタイムテーブルをスケジューラで定義できます。

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 2013 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 2013 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。

手順の概要

1. **configure terminal**
2. **scheduler schedule name***string*
3. **job name***string*
4. **time daily***time*
5. **time weekly** [[*dow*:] *HH*:]*MM*
6. **time monthly** [[*dm*:]*HH*:]*MM*
7. **time start** {**now repeat***repeat-interval* | *delta-time* [**repeat***repeat-interval*]}
8. （任意） **show scheduler config**
9. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**scheduler schedule name***string*
	
	例：
	`switch(config)# scheduler schedule name weekendbackupqos`
	`switch(config-schedule)#`
		スケジュールを作成し、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。

ステップ 3	**job name***string* 例： `switch(config-schedule)# job name offpeakZoning`	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。
ステップ 4	**time daily***time* 例： `switch(config-schedule)# time daily 23:00`	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。

	コマンドまたはアクション	目的
ステップ 5	**time weekly** [[*dow*:] *HH*:]*MM* 例： `switch(config-schedule)# time weekly Sun:23:00`	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（dow）は次のいずれかの方法で指定されます。 • 曜日を表す整数。たとえば 1 = 日曜日、2 = 月曜日。 • 曜日の省略形。たとえば Sun = Sunday。 引数全体の最大長は 10 です。
ステップ 6	**time monthly** [[*dm*:]*HH*:]*MM* 例： `switch(config-schedule)# time monthly 28:23:00`	ジョブが月の特定の日（dm）に開始することを意味します。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	**time start** {**now repeat***repeat-interval* | *delta-time* [**repeat***repeat-interval*]} 例： `switch(config-schedule)# time start now repeat 48:00`	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**：ジョブを今すぐ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。 この例では、ただちにジョブが開始され、48 時間間隔で反復されます。
ステップ 8	**show scheduler config** 例： `switch(config)# show scheduler config`	（任意） スケジューラ設定を表示します。
ステップ 9	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

スケジューラ ログ ファイルの消去

スケジューラ ログ ファイルを消去できます。

手順の概要

1. **configure terminal**
2. **clear scheduler logfile**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**clear scheduler logfile** 例： `switch(config)# clear scheduler logfile`	スケジューラ ログ ファイルの消去

スケジューラの設定確認

スケジューラの設定情報を表示するには、次のタスクのいずれかを行います。

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***string*]	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***string*]	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

次に、実行中のコンフィギュレーションを bootflash 内のファイルに保存し、ファイルを bootflash から TFTP サーバにコピーするスケジューラ ジョブを作成する例を示します（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されます）。

```
switch# configure terminal
switch(config)# scheduler job name backup-cfg
switch(config-job)# cli var name timestamp $(TIMESTAMP) ;copy running-config
bootflash:/$(SWITCHNAME)-cfg.$(timestamp) ;copy bootflash:/$(SWITCHNAME)-cfg.$(timestamp)
tftp://1.2.3.4/ 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-if)# job name backup-cfg
switch(config-if)# time daily 1:00
switch(config-if)# 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 2013
Last Completion Time: Fri Jan 2 1:00:01 2013
Execution count : 2
-----------------------------------------------
Job Name Last Execution Status
-----------------------------------------------
back-cfg Success (0)
switch#
```

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 2013
-------------------------------- Job Output ---------------------------------
`cli var name timestamp 2013-01-01-01.00.00`
`copy running-config bootflash:/$(HOSTNAME)-cfg.$(timestamp)`
`copy bootflash:/switch-cfg.2013-01-01-01.00.00 tftp://1.2.3.4/ vrf management `
copy: cannot access file '/bootflash/switch-cfg.2013-01-01-01.00.00'
==============================================================================
Job Name : back-cfg Job Status: Success (0)
Schedule Name : daily User Name : admin
Completion time: Fri Jan 2 1:00:01 2013
-------------------------------- Job Output ---------------------------------
`cli var name timestamp 2013-01-02-01.00.00`
`copy running-config bootflash:/switch-cfg.2013-01-02-01.00.00`
`copy bootflash:/switch-cfg.2013--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#
```

第 12 章

SNMP の設定

この章では、Cisco NX-OS デバイス上で SNMP 機能を設定する方法について説明します。

この章の内容は、次のとおりです。

SNMP の概要

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。
- 管理情報ベース（Management Information Base）：SNMPエージェントの管理対象オブジェクトのコレクション

SNMP は、RFC 3411 ～ 3418 で規定されています。

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。

Cisco NX-OS は IPv6 による SNMP をサポートしています。

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。応答要求は、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。デバイスは、トラップが受信されたかどうかを判断できません。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。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度送信できます。

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。

次の表は、デフォルトでイネーブルになっている SNMP トラップを示します。

トラップ タイプ	説明
generic	: coldStart
generic	: warmStart
entity	: entity_mib_change
entity	: entity_module_status_change
entity	: entity_power_status_change
entity	: entity_module_inserted
entity	: entity_module_removed

トラップ タイプ	説明
entity	: entity_unrecognised_module
entity	: entity_fan_status_change
entity	: entity_power_out_change
link	: linkDown
link	: linkUp
link	: extended-linkDown
link	: extended-linkUp
link	: cieLinkDown
link	: cieLinkUp
link	: delayed-link-state-change
rf	: redundancy_framework
license	: notify-license-expiry
license	: notify-no-license-for-feature
license	: notify-licensefile-missing
license	: notify-license-expiry-warning
upgrade	: UpgradeOpNotifyOnCompletion
upgrade	: UpgradeJobStatusNotify
rmon	: risingAlarm
rmon	: fallingAlarm
rmon	: hcRisingAlarm
rmon	: hcFallingAlarm
entity	: entity_sensor

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。次の表に、セキュリティ モデルとレベルの組み合わせの意味を示します。

表 *11*：*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	No	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	No	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	No	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）標準に基づいた認証を提供します。

ユーザベースのセキュリティ モデル

SNMPv3 ユーザベース 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。
- メッセージ発信元の認証：受信データを発信したユーザのアイデンティティが確認されたことを保証します。
- 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** オプションおよび **aes-128** トークンは、128 ビットの AES キーを生成するためのプライバシー パスワードであることを示します。AES のプライバシー パスワードは最小で 8 文字です。パスフレーズをクリア テキストで指定する場合は、大文字と小文字を区別して、最大 64 文字の英数字を指定できます。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。

（注） 外部の AAA サーバを使用して SNMPv3 を使う場合、外部 AAA サーバのユーザ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、次のようにユーザ設定を同期化します。

- **snmp-server user** コマンドで指定された認証パスフレーズが CLI ユーザ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が SNMP ユーザの認証およびプライバシー 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。
- CLI から行ったロール変更（削除または変更）は、SNMP と同期します。

（注） 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 はユーザ情報（パスワードやロールなど）を同期させません。

Cisco NX-OS はデフォルトで、同期したユーザ設定を 60 分間維持します。

グループベースの SNMP アクセス

（注） 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。各グループは読み取りアクセス権または読み取りと書き込みアクセス権を指定して定義します。

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 および EEM

Embedded Event Manager（EEM）機能は、SNMP MIB オブジェクトを含むイベントをモニタし、これらのイベントに基づいてアクションを開始します。SNMP 通知の送信もアクションの 1 つです。EEM は SNMP 通知として、CISCO-EMBEDDED-EVENT-MGR-MIB の cEventMgrPolicyEvent を送信します。

マルチインスタンス サポート

デバイスは、プロトコル インスタンスや仮想ルーティングおよびフォワーディング（VRF）インスタンスなどの論理ネットワーク エンティティの複数のインスタンスをサポートできます。大部分の既存 MIB は、これら複数の論理ネットワーク エンティティを識別できません。たとえば、元々の OSPF-MIB ではデバイス上のプロトコル インスタンスが 1 つであることが前提になりますが、現在はデバイス上で複数の OSPF インスタンスを設定できます。

SNMPv3 ではコンテキストを使用して、複数のインスタンスを識別します。SNMP コンテキストは管理情報のコレクションであり、SNMP エージェントを通じてアクセスできます。デバイスは、さまざまな論理ネットワーク エンティティの複数のコンテキストをサポートできます。SNMP コンテキストによって、SNMP マネージャはさまざまな論理ネットワーク エンティティに対応するデバイス上でサポートされる、MIB モジュールの複数のインスタンスの 1 つにアクセスできます。

Cisco NX-OS は、SNMP コンテキストと論理ネットワーク エンティティ間のマッピングのために、CISCO-CONTEXT-MAPPING-MIB をサポートします。SNMP コンテキストは VRF、プロトコル インスタンス、またはトポロジに関連付けることができます。

SNMPv3 は、SNMPv3 PDU の contextName フィールドでコンテキストをサポートします。この contextName フィールドを特定のプロトコル インスタンスまたは VRF にマッピングできます。

SNMPv2c の場合は、SNMP-COMMUNITY-MIB の snmpCommunityContextName MIB オブジェクトを使用して、SNMP コミュニティをコンテキストにマッピングできます（RFC 3584）。さらに CISCO-CONTEXT-MAPPING-MIB または CLI を使用すると、この snmpCommunityContextName を特定のプロトコル インスタンスまたは VRF にマッピングできます。

SNMP のハイ アベイラビリティ

Cisco NX-OS は、SNMP 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。

SNMP の仮想化サポート

Cisco NX-OS は、SNMP のインスタンスを 1 つサポートします。SNMP は複数の MIB モジュール インスタンスをサポートし、それらを論理ネットワーク エンティティにマッピングします。

SNMP も VRF を認識します。特定の VRF を使用して、SNMP 通知ホスト レシーバに接続するように SNMP を設定できます。通知が発生した VRF に基づいて、SNMP ホスト レシーバへの通知をフィルタリングするように SNMP を設定することもできます。

SNMP のライセンス要件

製品	ライセンス要件
Cisco NX-OS	SNMP にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

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、一部の SNMP 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 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。
ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

• Cisco NX-OS では、SNMPv3 noAuthNoPriv セキュリティ レベルはサポートされていません。

SNMP のデフォルト設定

次の表に、SNMP パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
ライセンス通知	イネーブル

SNMP の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

SNMP ユーザの設定

SNMP ユーザを設定できます。

手順の概要

1. **configure terminal**
2. **snmp-server user***name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID***id*] [**localizedkey**]]
3. （任意） **show snmp user**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 user***name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID***id*] [**localizedkey**]]	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。

	コマンドまたはアクション	目的
	例： `switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh`	**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	**show snmp user** 例： `switch(config) # show snmp user`	（任意） 1 人または複数の SNMP ユーザに関する情報を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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 パラメータを使用しているすべての SNMPv3 PDU 要求に対して、許可エラーで応答します。

手順の概要

1. **configure terminal**
2. **snmp-server user***name***enforcePriv**
3. **snmp-server globalEnforcePriv**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**snmp-server user***name***enforcePriv** 例： `switch(config)# snmp-server user Admin enforcePriv`	このユーザに対して SNMP メッセージ暗号化を適用します。
ステップ 3	**snmp-server globalEnforcePriv** 例： `switch(config)# snmp-server globalEnforcePriv`	すべてのユーザに対して SNMP メッセージ暗号化を適用します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

SNMPv3 ユーザに対する複数のロールの割り当て

SNMP ユーザを作成した後で、そのユーザに複数のロールを割り当てることができます。

（注） 他のユーザ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だけです。

手順の概要

1. **configure terminal**
2. **snmp-server user***namegroup*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**snmp-server user***namegroup* 例： `switch(config)# snmp-server user Admin superuser`	この SNMP ユーザと設定されたユーザ ロールをアソシエートします。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

手順の概要

1. **configure terminal**
2. **snmp-server community***name* {**group***group* | **ro** | **rw**}
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 community***name* {**group***group*	**ro**	**rw**} 例： `switch(config)# snmp-server community public ro`	SNMP コミュニティ ストリングを作成します。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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 SNMPv2 コミュニティまたは SNMPv3 ユーザに割り当てて、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）

手順の概要

1. **configure terminal**
2. **snmp-server community***name* [**use-ipv4acl***acl-name* | **use-ipv6acl***acl-name*]
3. **snmp-server user***username* [**use-ipv4acl***acl-name* | **use-ipv6acl***acl-name*]
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 community***name* [**use-ipv4acl***acl-name*	**use-ipv6acl***acl-name*] 例： `switch(config)# snmp-server community public use-ipv4acl myacl`	SNMPv2 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。 （注） Cisco NX-OS リリース 7.0(3)I4(1) 以降では、IPv6 ACL が SNMPv2 コミュニティでサポートされます。 （注） Cisco NX-OS リリース 7.0(3)I4(1) よりも前のリリースでは、この CLI コマンドで使用するのは **use-ipv4acl** ではなく **use-acl** です。

	コマンドまたはアクション	目的
ステップ 3	**snmp-server user***username* [**use-ipv4acl***acl-name* | **use-ipv6acl***acl-name*] 例： `switch(config)# snmp-server user user1 use-ipv4acl myacl`	SNMPv3 ユーザに IPv4 ACL または IPv6 ACL を割り当てて SNMP 要求をフィルタします。 （注） Cisco NX-OS リリース 7.0(3)I4(1) 以降では、IPv6 ACL が SNMPv3 ユーザでサポートされます。 （注） Cisco NX-OS リリース 7.0(3)I4(1) よりも前のリリースでは、この CLI コマンドで使用するのは **use-ipv4acl** ではなく **use-acl** です。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

手順の概要

1. **configure terminal**
2. **snmp-server host***ip-address***traps version 1***community* [**udp_port***number*]
3. **snmp-server host***ip-address* {**traps** | **informs**} **version 2c***community* [**udp_port***number*]
4. **snmp-server host***ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port***number*]
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 host***ip-address***traps version 1***community* [**udp_port***number*] 例： `switch(config)# snmp-server host 192.0.2.1 traps version 1 public`	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。*community* 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。

	コマンドまたはアクション	目的
ステップ 3	**snmp-server host***ip-address* {**traps** | **informs**} **version 2c***community* [**udp_port***number*] 例： `switch(config)# snmp-server host 192.0.2.1 informs version 2c public`	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。*community* 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。
ステップ 4	**snmp-server host***ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port***number*] 例： `switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS`	SNMPv3 トラップまたは応答要求のホスト 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。*username* 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。 （注） SNMP マネージャは、SNMPv3 メッセージを認証して復号化するために、Cisco NX-OS デバイスの SNMP エンジン ID に基づいてユーザ クレデンシャル（authKey/PrivKey）を調べる必要があります。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 通知用の発信元インターフェイスの設定

通知の送信元 IP アドレスとしてインターフェイスの IP アドレスを使用するよう、SNMP を設定できます。通知が生成される場合、送信元 IP アドレスは、この設定済みインターフェイスの IP アドレスに基づいています。

次のように発信元インターフェイスを設定できます。

- すべての通知が、すべての SNMP 通知レシーバへ送信される。
- すべての通知が、特定の SNMP 通知レシーバへ送信される。このコンフィギュレーションは、グローバル発信元インターフェイスのコンフィギュレーションよりも優先されます。

（注） 発信トラップ パケットの送信元インターフェイス IP アドレスを設定すると、デバイスがトラップの送信に同じインターフェイスを使用することが保証されません。送信元インターフェイス IP アドレスは、SNMP トラップの内部で送信元アドレスを定義し、出力インターフェイス アドレスを送信元として接続が開きます。

手順の概要

1. **configure terminal**
2. **snmp-server host***ip-address***source-interface***if-type if-number* [**udp_port***number*]
3. **snmp-server source-interface** {**traps** | **informs**} *if-type if-number*
4. **show snmp source-interface**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 host***ip-address***source-interface***if-type if-number* [**udp_port***number*] 例： `switch(config)# snmp-server host 192.0.2.1 source-interface ethernet 2/1`	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。**?** を使用して、サポートされているインターフェイス タイプを特定します。UDP ポート番号の範囲は 0 ～ 65535 です。 このコンフィギュレーションは、グローバル発信元インターフェイスのコンフィギュレーションよりも優先されます。
ステップ 3	**snmp-server source-interface** {**traps** | **informs**} *if-type if-number* 例： `switch(config)# snmp-server source-interface traps ethernet 2/1`	SNMPv2c トラップまたは応答要求を送信するよう発信元インターフェイスを設定します。**?** を使用して、サポートされているインターフェイス タイプを特定します。
ステップ 4	**show snmp source-interface** 例： `switch(config)# 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。

通知対象ユーザの設定

SNMPv3 インフォーム通知を通知ホスト レシーバに送信するには、デバイスに通知ターゲット ユーザを設定する必要があります。

Cisco NX-OS は通知ターゲット ユーザのクレデンシャルを使用して、設定された通知ホスト レシーバへの SNMPv3 インフォーム通知メッセージを暗号化します。

（注） 受信した INFORM PDU を認証して復号化する場合、Cisco NX-OS で設定されているのと同じ、応答要求を認証して解読するユーザ クレデンシャルが通知ホスト レシーバに必要です。

手順の概要

1. **configure terminal**
2. **snmp-server user***name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID***id*]
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 user***name* [**auth** {**md5** | **sha**} *passphrase* [**auto**] [**priv** [**aes-128**] *passphrase*] [**engineID***id*] 例： `switch(config)# snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID 00:00:00:63:00:01:00:10:20:15:10:03`	通知ホスト レシーバのエンジン ID を指定して、通知ターゲット ユーザを設定します。エンジン ID 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

VRF を使用する SNMP 通知レシーバの設定

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。

ホスト レシーバに到達するように設定した VRF を使用したり、または通知が発生した VRF に基づいて通知をフィルタするように Cisco NX-OS を設定できます。

手順の概要

1. **configure terminal**
2. [**no**] **snmp-server host***ip-address***use-vrf***vrf-name* [**udp_port***number*]
3. [**no**] **snmp-server host***ip-address***filter-vrf***vrf-name* [**udp_port***number*]
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **snmp-server host***ip-address***use-vrf***vrf-name* [**udp_port***number*] 例： `switch(config)# snmp-server host 192.0.2.1 use-vrf Blue`	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-address* 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。 このコマンドの **no** 形式は、設定されたホストの VRF 到達可能性情報を削除し、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable からエントリを削除します。 （注） このコマンドによってホスト設定は削除されません。
ステップ 3	[**no**] **snmp-server host***ip-address***filter-vrf***vrf-name* [**udp_port***number*] 例： `switch(config)# snmp-server host 192.0.2.1 filter-vrf Red`	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-address* 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。 このコマンドの **no** 形式は、設定されたホストの VRF フィルタ情報を削除し、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable からエントリを削除します。 （注） このコマンドによってホスト設定は削除されません。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

帯域内ポートを使用してトラップを送信するための SNMP 設定

帯域内ポートを使用してトラップを送信するよう SNMP を設定できます。このようにするには、（グローバルまたはホスト レベルで）発信元インターフェイスを設定し、トラップを送信するための VRF を設定します。

手順の概要

1. **configure terminal**
2. **snmp-server source-interface traps***if-type if-number*
3. （任意） **show snmp source-interface**
4. **snmp-server host***ip-address***use-vrf***vrf-name* [**udp_port***number*]
5. （任意） **show snmp host**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 source-interface traps***if-type if-number* 例： `switch(config)# snmp-server source-interface traps ethernet 1/2`	SNMP トラップを送信するための発信元インターフェイスをグローバルに設定します。? を使用して、サポートされているインターフェイス タイプを特定します。 グローバル レベルまたはホスト レベルで発信元インターフェイスを設定できます。発信元インターフェイスをグローバルに設定すると、新しいホストコンフィギュレーションはグローバルなコンフィギュレーションを使用してトラップを送信します。 （注） 発信元インターフェイスをホスト レベルで設定するには、**snmp-server host***ip-address***source-interface***if-type if-number* コマンドを使用します。
ステップ 3	**show snmp source-interface** 例： `switch(config)# show snmp source-interface`	（任意） 設定した発信元インターフェイスの情報を表示します。

	コマンドまたはアクション	目的
ステップ 4	**snmp-server host***ip-address***use-vrf***vrf-name* [**udp_port***number*] 例： `switch(config)# snmp-server host 171.71.48.164 use-vrf default`	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-address* 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。 （注） デフォルトでは、SNMP は管理 VRF を使用してトラップを送信します。管理 VRF を使用しない場合は、このコマンドを使用して対象の VRF を指定する必要があります。
ステップ 5	**show snmp host** 例： `switch(config)# show snmp host`	（任意） 設定した SNMP ホストの情報を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

（注） **snmp-server enable traps** コマンドを使用すると、設定されている通知ホスト レシーバに応じて、トラップおよび応答要求の両方がイネーブルになります。

次の表に、Cisco NX-OS MIB の通知をイネーブルにするコマンドを示します。

表 *12*：*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-AAA-SERVER-MIB	**snmp-server enable traps aaa** **snmp-server enable traps aaa server-state-change**

MIB	関連コマンド
CISCO-BGP4-MIB	**snmp-server enable traps bgp**
CISCO-STP-BRIDGE-MIB	**snmp-server enable traps bridge** **snmp-server enable traps bridge newroot** **snmp-server enable traps bridge topologychange**
CISCO-CALLHOME-MIB	**snmp-server enable traps callhome** **snmp-server enable traps callhome event-notify** **snmp-server enable traps callhome smtp-send-fail**
CISCO-CONFIG-MAN-MIB	**snmp-server enable traps config** **snmp-server enable traps config ccmCLIRunningConfigChanged**
CISCO-EIGRP-MIB	**snmp-server enable traps eigrp**[*tag*]
CISCO-ERR-DISABLE-MIB	**snmp-server enable traps show interface status**
ENTITY-MIB, CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity entity_fan_status_change** **snmp-server enable traps entity entity_mib_change** **snmp-server enable traps entity entity_module_inserted** **snmp-server enable traps entity entity_module_removed** **snmp-server enable traps entity entity_module_status_change** **snmp-server enable traps entity entity_power_out_change** **snmp-server enable traps entity entity_power_status_change** **snmp-server enable traps entity entity_unrecognised_module**

MIB	関連コマンド
CISCO-FEATURE-CONTROL-MIB	**snmp-server enable traps feature-control** **snmp-server enable traps feature-control FeatureOpStatusChange**
CISCO-HSRP-MIB	**snmp-server enable traps hsrp** **snmp-server enable traps hsrp state-change**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license** **snmp-server enable traps license notify-license-expiry** **snmp-server enable traps license notify-license-expiry-warning** **snmp-server enable traps license notify-licensefile-missing** **snmp-server enable traps license notify-no-license-for-feature**
IF-MIB	**snmp-server enable traps link** **snmp-server enable traps link IETF-extended-linkDown** **snmp-server enable traps link IETF-extended-linkUp** **snmp-server enable traps link cisco-extended-linkDown** **snmp-server enable traps link cisco-extended-linkUp** **snmp-server enable traps link linkDown** **snmp-server enable traps link Up**
OSPF-MIB, OSPF-TRAP-MIB	**snmp-server enable traps ospf**[*tag*] **snmp-server enable traps ospf lsa** **snmp-server enable traps ospf rate-limit***rate*

MIB	関連コマンド
CISCO-RF-MIB	**snmp-server enable traps rf** **snmp-server enable traps rf redundancy_framework**
CISCO-RMON-MIB	**snmp-server enable traps rmon** **snmp-server enable traps rmon fallingAlarm** **snmp-server enable traps rmon hcFallingAlarm** **snmp-server enable traps rmon hcRisingAlarm** **snmp-server enable traps rmon risingAlarm**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-STPX-MIB	**snmp-server enable traps stpx** **snmp-server enable traps stpx inconsistency** **snmp-server enable traps stpx loop-inconsistency** **snmp-server enable traps stpx root-inconsistency**
CISCO-SWITCH-QOS-MIB	**snmp-server enable traps show hardware internal ns buffer info pkt-stats** **snmp-server enable traps show hardware internal ns buffer info pkt-stats input**
CISCO-SYSTEM-EXT-MIB	**snmp-server enable traps sysmgr** **snmp-server enable traps sysmgr cseFailSwCoreNotifyExtended**

MIB	関連コマンド
UPGRADE-MIB	**snmp-server enable traps upgrade** **snmp-server enable traps upgrade UpgradeJobStatusNotify** **snmp-server enable traps upgrade UpgradeOpNotifyOnCompletion**
VTP-MIB	**snmp-server enable traps vtp** **snmp-server enable traps vtp notifs** **snmp-server enable traps vtp vlancreate** **snmp-server enable traps vtp vlandelete**
CISCO-PORT-STORM-CONTROL-MIB	**storm-control action trap**

指定した通知をイネーブルにするには、表示される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
snmp-server enable traps 例： `switch(config)# snmp-server enable traps`	すべての SNMP 通知をイネーブルにします。
snmp-server enable traps aaa[**server-state-change**] 例： `switch(config)# snmp-server enable traps aaa`	AAA SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **server-state-change**：AAA サーバの状態変化通知をイネーブルにします。
snmp-server enable traps bgp 例： `switch(config)# snmp-server enable traps bgp`	ボーダー ゲートウェイ プロトコル（BGP）SNMP 通知をイネーブルにします。

コマンド	目的
snmp-server enable traps bridge[**newroot**][**topologychange**] 例： `switch(config)# snmp-server enable traps bridge`	STP ブリッジ SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **newroot**：STP の新しいルート ブリッジ通知をイネーブルにします。 • **topologychange**：STP ブリッジのトポロジ変更通知をイネーブルにします。
snmp-server enable traps callhome [**event-notify**] [**smtp-send-fail**] 例： `switch(config)# snmp-server enable traps callhome`	Call Home 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **event-notify**：Call Home の外部イベント通知をイネーブルにします。 • **smtp-send-fail**：簡易メール転送プロトコル（SMTP）メッセージの送信失敗通知をイネーブルにします。
snmp-server enable traps config [**ccmCLIRunningConfigChanged**] 例： `switch(config)# snmp-server enable traps config`	コンフィギュレーションの変更に対して SNMP 通知をイネーブルにします。 • **ccmCLIRunningConfigChanged**：実行中または起動時のコンフィギュレーションで、コンフィギュレーションの変更に対して SNMP 通知をイネーブルにします。
snmp-server enable traps eigrp [*tag*] 例： `switch(config)# snmp-server enable traps eigrp`	CISCO-EIGRP-MIB SNMP 通知をイネーブルにします。

コマンド	目的
snmp-server enable traps entity [**entity_fan_status_change**] [**entity_mib_change**] [**entity_module_inserted**] [**entity_module_removed**] [**entity_module_status_change**] [**entity_power_out_change**] [**entity_power_status_change**] [**entity_unrecognised_module**] 例： `switch(config)# snmp-server enable traps entity`	ENTITY-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **entity_fan_status_change**：エンティティ ファンの状態変化通知をイネーブルにします。 • **entity_mib_change**：エンティティ MIB 変更通知をイネーブルにします。 • **entity_module_inserted**：エンティティ モジュール挿入通知をイネーブルにします。 • **entity_module_removed**：エンティティ モジュール削除通知をイネーブルにします。 • **entity_module_status_change**：エンティティ モジュール ステータス変更通知をイネーブルにします。 • **entity_power_out_change**：エンティティの出力パワー変更通知をイネーブルにします。 • **entity_power_status_change**：エンティティのパワー ステータス変更通知をイネーブルにします。 • **entity_unrecognised_module**：エンティティの未確認モジュール通知をイネーブルにします。
snmp-server enable traps feature-control[**FeatureOpStatusChange**] 例： `switch(config)# snmp-server enable traps feature-control`	機能制御 SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **FeatureOpStatusChange**：機能操作の状態変化通知をイネーブルにします。
snmp-server enable traps hsrp[**state-change**] 例： `switch(config)# snmp-server enable traps hsrp`	CISCO-HSRP-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **state-change**：HSRP の状態変化通知をイネーブルにします。

コマンド	目的
snmp-server enable traps license [**notify-license-expiry**] [**notify-license-expiry-warning**] [**notify-licensefile-missing**] [**notify-no-license-for-feature**] 例： `switch(config)# snmp-server enable traps license`	ENTITY-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **notify-license-expiry**：ライセンス失効通知をイネーブルにします。 • **notify-license-expiry-warning**：ライセンス失効の警告通知をイネーブルにします。 • **notify-licensefile-missing**：ライセンス ファイル不明通知をイネーブルにします。 • **notify-no-license-for-feature**：no-license-installed-for-feature 通知をイネーブルにします。
snmp-server enable traps link [**IETF-extended-linkDown**] [**IETF-extended-linkUp**] [**cisco-extended-linkDown**] [**cisco-extended-linkUp**] [**linkDown**] [**linkUp**] 例： `switch(config)# snmp-server enable traps link`	IF-MIB リンク通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **IETF-extended-linkDown**：インターネット技術特別調査委員会（IETF）の拡張リンク ステート ダウン通知をイネーブルにします。 • **IETF-extended-linkUp**：IETF の拡張リンク ステート アップ通知をイネーブルにします。 • **cisco-extended-linkDown**：Cisco 拡張リンク ステート ダウン通知をイネーブルにします。 • **cisco-extended-linkUp**：Cisco 拡張リンク ステート アップ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。

コマンド	目的
snmp-server enable traps ospf[*tag*] [**lsa**] 例： `switch(config)# snmp-server enable traps ospf`	Open Shortest Path First（OSPF）通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **lsa**：OSPF リンク ステート アドバタイズメント（LSA）通知をイネーブルにします。
snmp-server enable traps rf[**redundancy-framework**] 例： `switch(config)# snmp-server enable traps rf`	冗長フレームワーク（RF）SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **redundancy-framework**：RF スーパーバイザ スイッチオーバー MIB 通知をイネーブルにします。
snmp-server enable traps rmon [**fallingAlarm**] [**hcFallingAlarm**] [**hcRisingAlarm**] [**risingAlarm**] 例： `switch(config)# snmp-server enable traps rmon`	リモートモニタリング（RMON）SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **fallingAlarm**：RMON 下限アラーム通知をイネーブルにします。 • **hcFallingAlarm**：RMON high-capacity 下限アラーム通知をイネーブルにします。 • **hcRisingAlarm**：RMON high-capacity 上限アラーム通知をイネーブルにします。 • **risingAlarm**：RMON 上限アラーム通知をイネーブルにします。
snmp-server enable traps snmp [**authentication**] 例： `switch(config)# snmp-server enable traps snmp`	一般的な SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **authentication**：SNMP 認証通知をイネーブルにします。

コマンド	目的
snmp-server enable traps stpx [**inconsistency**] [**loop-inconsistency**] [**root-inconsistency**] 例： `switch(config)# snmp-server enable traps stpx`	リモートモニタリング（RMON）SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **inconsistency**：SNMP STPX MIB 不一致アップデート通知をイネーブルにします。 • **loop-inconsistency**：SNMP STPX MIB ループ不一致アップデート通知をイネーブルにします。 • **root-inconsistency**：SNMP STPX MIB ルート不一致アップデート通知をイネーブルにします。
snmp-server enable traps sysmgr [**cseFailSwCoreNotifyExtended**] 例： `switch(config)# snmp-server enable traps sysmgr`	ソフトウェア変更通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **cseFailSwCoreNotifyExtended**：ソフトウェア コア通知をイネーブルにします。
snmp-server enable traps upgrade [**UpgradeJobStatusNotify**] [**UpgradeOpNotifyOnCompletion**] 例： `switch(config)# snmp-server enable traps upgrade`	アップグレード通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **UpgradeJobStatusNotify**：アップグレード ジョブ ステータス通知をイネーブルにします。 • **UpgradeOpNotifyOnCompletion**：アップグレード グローバル ステータス通知をイネーブルにします。
snmp-server enable traps vtp [**notifs**] [**vlancreate**] [**vlandelete**] 例： `switch(config)# snmp-server enable traps vtp`	VTP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **notifs**：VTP 通知をイネーブルにします。 • **vlancreate**：VLAN 作成の通知をイネーブルにします。 • **vlandelete**：VLAN 削除の通知をイネーブルにします。

コマンド	目的
storm-control action trap 例： `switch(config-if)# storm-control action trap`	トラフィック ストーム制御の限界に到達すると、トラフィック ストーム制御の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。フラッピング インターフェイス（Up と Down の間を頻繁に切り替わるインターフェイス）で、この制限通知を使用できます。

手順の概要

1. **configure terminal**
2. **interface***typeslot*/*port*
3. **no snmp trap link-status**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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***typeslot*/*port* 例： `switch(config)# interface ethernet 2/2`	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。このコマンドは、デフォルトでイネーブルになっています。
ステップ 3	**no snmp trap link-status** 例： `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。このコマンドは、デフォルトでイネーブルになっています。
ステップ 4	**copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	（任意） 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。

インターフェイスの SNMP ifIndex の表示

SNMP ifIndex は、関連するインターフェイス情報をリンクするために複数の SNMP MIB にわたって使用されます。

手順の概要

1. **show interface snmp-ifindex**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**show interface snmp-ifindex**
	
	例：
	`switch# show interface snmp-ifindex | grep -i Eth12/1`
	`Eth12/1 441974784 (0x1a580000)`
		すべてのインターフェイスについて、IF-MIB から永続的な SNMP ifIndex 値を表示します。任意で、| キーワードと **grep** キーワードを使用すると、出力で特定のインターフェイスを検索できます。

TCP による SNMP の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。

手順の概要

1. **configure terminal**
2. **snmp-server tcp-session** [**auth**]
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 tcp-session** [**auth**] 例： `switch(config)# snmp-server tcp-session`	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。デフォルトではディセーブルになっています。

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 デバイスの連絡先およびロケーション情報の割り当て

32 文字までの長さで（スペースを含まない）デバイスのコンタクト情報とデバイスのロケーションを指定できます。

手順の概要

1. **configure terminal**
2. **snmp-server contact***name*
3. **snmp-server location***name*
4. （任意） **show snmp**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 contact***name* 例： `switch(config)# snmp-server contact Admin`	SNMP コンタクト名として sysContact を設定します。
ステップ 3	**snmp-server location***name* 例： `switch(config)# snmp-server location Lab-7`	SNMP ロケーションとして sysLocation を設定します。
ステップ 4	**show snmp** 例： `switch(config)# show snmp`	（任意） 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。

	コマンドまたはアクション	目的
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

はじめる前に

論理ネットワーク エンティティのインスタンスを決定します。VRF およびプロトコル インスタンスの詳細については、『*Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide*』または『*Cisco Nexus 9000 Series NX-OS Multicast Routing Configuration Guide*』を参照してください。

手順の概要

1. **configure terminal**
2. [**no**] **snmp-server context***context-name* [**instance***instance-name*] [**vrf***vrf-name*] [**topology***topology-name*]
3. （任意） **snmp-server mib community-map***community-name***context***context-name*
4. （任意） **show snmp context**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **snmp-server context***context-name* [**instance***instance-name*] [**vrf***vrf-name*] [**topology***topology-name*] 例： `switch(config)# snmp-server context public1 vrf red`	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。 **no** オプションは、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。

	コマンドまたはアクション	目的
		（注） 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。
ステップ 3	**snmp-server mib community-map***community-name***context***context-name* 例： `switch(config)# snmp-server mib community-map public context public1`	（任意） 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	**show snmp context** 例： `switch(config)# show snmp context`	（任意） 1 つまたは複数の SNMP コンテキストに関する情報を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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

デバイスの SNMP をディセーブルにできます。

手順の概要

1. **configure terminal**
2. **no snmp-server protocol enable**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 snmp-server protocol enable** 例： `switch(config)# no snmp-server protocol enable`	SNMP をディセーブルにします。SNMP はデフォルトでイネーブルになっています。

AAA 同期時間の変更

同期したユーザ設定を Cisco NX-OS に維持させる時間の長さを変更できます。

手順の概要

1. **configure terminal**
2. **snmp-server aaa-user cache-timeout***seconds*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 aaa-user cache-timeout***seconds* 例： `switch(config)# snmp-server aaa-user cache-timeout 1200`	ローカル キャッシュで AAA 同期ユーザ設定を維持する時間を設定します。値の範囲は 1 ～ 86400 秒です。デフォルトは 3600 です。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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	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。 （注） **snmp-server mib community-map** コマンド中の SNMP コンテキストの名前が 11 文字を超過した場合、**show snmp community** コマンドの出力は、表形式の代わりに垂直形式で表示されます。
show snmp context	SNMP コンテキスト マッピングを表示します。
show snmp engineID	SNMP engineID を表示します。
show snmp group	SNMP ロールを表示します。
show snmp host	設定した SNMP ホストの情報を表示します。
show snmp session	SNMP セッションを表示します。
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。
show snmp user	SNMPv3 ユーザを表示します。

SNMP の設定例

次に、Blue VRF を使用して、ある通知ホスト レシーバに Cisco linkUp または Down 通知を送信するよう Cisco NX-OS を設定し、Admin と NMS という 2 つの SNMP ユーザを定義する例を示します。

```
configure terminal
snmp-server contact Admin@company.com
snmp-server user Admin auth sha abcd1234 priv abcdefgh
snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID
00:00:00:63:00:01:00:22:32:15:10:03
snmp-server host 192.0.2.1 informs version 3 auth NMS
snmp-server host 192.0.2.1 use-vrf Blue
snmp-server enable traps link cisco
```

次に、ホスト レベルで設定された帯域内ポートを使用してトラップを送信するよう、SNMP を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)# snmp-server host 171.71.48.164 version 2c public
switch(config)# snmp-server host 171.71.48.164 source-interface ethernet 1/2
switch(config)# show snmp host
-------------------------------------------------------------------
Host Port Version Level Type SecName
-------------------------------------------------------------------
171.71.48.164 162 v2c noauth trap public
Source interface: Ethernet 1/2
-------------------------------------------------------------------
switch(config)# snmp-server host 171.71.48.164 use-vrf default
switch(config)# show snmp host
-------------------------------------------------------------------
Host Port Version Level Type SecName
-------------------------------------------------------------------
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-------------------------------------------------------------------
```

次に、グローバルに設定した帯域内ポートを使用してトラップを送信するよう SNMP を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)# snmp-server source-interface traps ethernet 1/2
switch(config)# show snmp source-interface
-------------------------------------------------------------------
Notification source-interface
-------------------------------------------------------------------
trap Ethernet1/2
inform -
-------------------------------------------------------------------
switch(config)# snmp-server host 171.71.48.164 use_vrf default
switch(config)# show snmp host
-------------------------------------------------------------------
Host Port Version Level Type SecName
-------------------------------------------------------------------
171.71.48.164 162 v2c noauth trap public
Use VRF: default
Source interface: Ethernet 1/2
-------------------------------------------------------------------
```

VRF red を SNMPv2c のパブリック コミュニティ ストリングにマッピングする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# vrf context red
switch(config-vrf)# exit
switch(config)# snmp-server context public1 vrf red
switch(config)# snmp-server mib community-map public context public1
```

OSPF インスタンス Enterprise を同じ SNMPv2c パブリック コミュニティ ストリングにマッピングする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature ospf
switch(config)# router ospf Enterprise
switch(config-router)# exit
switch(config)# snmp-server context public1 instance Enterprise
switch(config)# snmp-server mib community-map public context public1
```

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
IP ACL および AAA	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
MIB	『*Cisco Nexus 7000 Series and 9000 Series NX-OS MIB Quick Reference*』

RFC

RFC	Title
RFC 3414	『*User-based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3)*』
RFC 3415	『*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*』

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するには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

第 13 章

RMON の設定

この章では、Cisco NX-OS デバイスでのリモート モニタリング（RMON）機能を設定する方法について説明します。

この章の内容は、次のとおりです。

RMON について

RMON は、各種ネットワーク 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、簡易ネットワーク管理プロトコル（SNMP）インターネット技術特別調査委員会（IETF）の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 NX-OS では、RMON はデフォルトでイネーブルですが、アラームは設定されていません。RMON アラームを設定するには、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。指定するオブジェクトは、標準のドット付き表記で表した既存の SNMP MIB オブジェクトでなければなりません（たとえば、1.3.6.1.2.1.2.2.1.14 は ifInOctets.14 を表します）。

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：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場合の値
- 下限しきい値：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場合の値
- イベント：アラーム（上限または下限）の発生時にデバイスが実行するアクション

（注） 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。

（注） 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。

（注） デフォルトの RMON イベント テンプレート設定の使用を選択することも、これらのエントリを削除して新しい RMON イベントを作成することもできます。RMON アラーム設定を作成するまで、これらの設定によってトリガーされるアラームはありません。

RMON のハイ アベイラビリティ

Cisco NX-OS は、RMON 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。

RMON の仮想化サポート

Cisco NX-OS は、RMON のインスタンスを 1 つサポートします。

RMON は Virtual Routing and Forwarding（VRF）を認識します。特定の VRF を使用して RMON SMTP サーバに接続するように RMON を設定できます。

RMON のライセンス要件

製品	ライセンス要件
Cisco NX-OS	RMON にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

RMON の注意事項と制約事項

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 オブジェクトにのみ、RMON アラームを設定できます。
- RMON アラームを設定する場合は、オブジェクト ID がインデックスで 1 オブジェクトだけを示すようになっている必要があります。たとえば、1.3.6.1.2.1.2.2.1.14 は cpmCPUTotal5minRev に対応し、.1 は cpmCPUTotalIndex インデックスに対応し、オブジェクト ID の 1.3.6.1.2.1.2.2.1.14.1 を作成しす。

RMON のデフォルト設定

次の表に、RMON パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
RMON	イネーブル
アラーム	未設定

RMON の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

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。

はじめる前に

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。

手順の概要

1. **configure terminal**
2. **rmon alarm***index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold***value* [*event-index*] **falling-threshold***value* [*event-index*] [**owner***name*]
3. **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*]
4. （任意） **show rmon** {**alarms** | **hcalarms**}
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**rmon alarm***index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold***value* [*event-index*] **falling-threshold***value* [*event-index*] [**owner***name*] 例： `switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.1 2900 delta rising-threshold 1500 1 falling-threshold 0 owner test`	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	**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*] 例： `switch(config)# rmon alarm 20 1.3.6.1.2.1.2.2.1.14.16777216 2900 delta rising-threshold-high 15 rising-threshold-low 151 falling-threshold-high 0 falling-threshold-low 0 owner test`	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	**show rmon** {**alarms** | **hcalarms**} 例： `switch(config)# show rmon alarms`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

はじめる前に

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。

手順の概要

1. **configure terminal**
2. **rmon event***index* [**description***string*] [**log**] [**trap***string*] [**owner***name*]
3. （任意） **show rmon events**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**rmon event***index* [**description***string*] [**log**] [**trap***string*] [**owner***name*] 例： `switch(config)# rmon event 1 trap trap1`	RMON イベントを設定します。説明のストリング、トラップ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にすることができます。
ステップ 3	**show rmon events** 例： `switch(config)# show rmon events`	（任意） RMON イベントに関する情報を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

RMON 設定の確認

RMON 設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show 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。

コマンド	目的
show rmon logs	RMON ログに関する情報を表示します。

RMON の設定例

ifInOctets.14 にデルタ上限アラームを作成し、このアラームに通知イベントを関連付ける方法の例を示します。

```
configure terminal
rmon alarm 20 1.3.6.1.2.1.2.2.1.14.1 2900 delta rising-threshold 1500 1 falling-threshold
0 owner test
rmon event 1 trap trap1
```

その他の参考資料

MIB

MIB	MIB のリンク
RMON 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するには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

第 14 章

オンライン診断の設定

この章では、デバイス上で汎用オンライン診断（GOLD）機能を設定する方法について説明します。

この章の内容は、次のとおりです。

オンライン診断について

オンライン診断機能を使用すると、デバイスをアクティブ ネットワークに接続したまま、デバイスのハードウェア機能をテストして確認できます。

オンライン診断機能には、さまざまなハードウェア コンポーネントを検査し、データ パスと制御信号を確認するテストが組み込まれています。中断を伴うオンライン診断テスト（破壊モードのループバック テストなど）、および中断を伴わないオンライン診断テスト（ASIC レジスタ検査など）は、起動時、ライン モジュールの活性挿抜（OIR）時、およびシステム リセット時に実行されます。中断を伴わないオンライン診断テストは、バックグラウンドヘルスモニタリングの一部として実行され、これらのテストはオンデマンドで実行できます。

オンライン診断は、起動、ランタイムまたはヘルスモニタリング診断、およびオンデマンド診断に分類されます。起動診断は起動時に、ヘルスモニタリング テストはバックグラウンドで、オンデマンド診断はアクティブ ネットワークにデバイスが接続されたときに 1 回だけ、またはユーザが指定した間隔で実行されます。

ブートアップ診断

起動診断は起動中に実行され、Cisco NX-OS がモジュールをオンラインにする前に、障害ハードウェアが検出されます。たとえば、デバイスに障害モジュールを搭載した場合、起動診断でモジュールがテストされ、デバイスがそのモジュールをトラフィックの転送に使用しないうちに、モジュールがオフラインにされます。

起動診断では、スーパーバイザとモジュール ハードウェア間、およびすべての ASIC のデータ パスと制御パス間の接続も検査されます。次の表では、モジュールおよびスーパーバイザの起動診断テストについて説明します。

表 *13*：ブートアップ診断

診断	説明
Module	
OBFL	オンボード障害ロギング（OBFL）フラッシュの整合性を確認します。
スーパーバイザ	
USB	中断を伴わないテスト。モジュールにおける USB コントローラの初期化を検査
ManagementPortLoopback	中断を伴うテストで、オンデマンド型テストではありません。モジュールの管理ポートでループバックをテスト
EOBCPortLoopback	中断を伴うテストで、オンデマンド型テストではありません。イーサネット帯域外。
OBFL	オンボード障害ロギング（OBFL）フラッシュの整合性を確認します。

起動診断テストはエラーを Onboard Failure Logging（OBFL）および syslog に記録し、診断の LED 表示（オン、オフ、合格、失敗）を開始します。

起動診断テストをバイパスするように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。

ランタイムまたはヘルス モニタリング診断

ランタイム診断はヘルス モニタリング（HM）診断ともいいます。これらの診断テストによって、アクティブ デバイスの状態に関する情報が得られます。ランタイム ハードウェア エラー、メモリ エラー、ハードウェア モジュールの経時的劣化、ソフトウェア障害、およびリソース不足が検出されます。

アクティブ ネットワーク トラフィックを処理するデバイスの状態を確認するヘルス モニタリング診断テストは、中断を伴わず、バックグラウンドで実行されます。ヘルスモニタリングテストはイネーブルまたはディセーブルにできます。また、ランタイム インターバルの変更が可能です。

次の表に、モジュールおよびスーパーバイザのヘルスモニタリング診断とテストIDを示します。

表 *14*： ヘルス モニタリングの無停止での診断

診断	デフォルトのインターバル	デフォルト設定	説明
Module			
ACT2	30 分	active	モジュール上のセキュリティ デバイスの整合性を確認します。
ASICRegisterCheck	1 分	active	モジュール上の ASIC のレジスタをスクラッチするための読み取りと書き込みアクセス権を確認します。
PrimaryBootROM	30 分	active	モジュール上のプライマリ ブート デバイスの完全性を確認します。
SecondaryBootROM	30 分	active	モジュール上のセカンダリ ブート デバイスの完全性を確認します。
PortLoopback	オン デマンド（Cisco NX-OS 7.0(3)I1(2) より以前のリリース） 30 分（Cisco NX-OS リリース 7.0(3)I1(2) 以降）	active	診断情報がポート単位またはすべての管理ダウン ポートでチェックされます。
RewriteEngineLoopback	1 分	active	1 エンジン ASIC デバイスまでのすべてのポートの無停止ループバックの整合性を確認します。
AsicMemory	ブートアップ時のみ	ブートアップ時のみ - inactive	ASIC の Mbist ビットを使用して AsicMemory の一貫性がチェックされます。

診断	デフォルトのインターバル	デフォルト設定	説明
Module			
FpgaRegTest	30 秒	ヘルス モニタリング テスト - 30 秒ごと - active	FPGA への読み取りと書き込みにより FPGA の状態がテストされます。
スーパーバイザ			
NVRAM	5 分	active	スーパーバイザの NVRAM ブロックの健全性を確認します。
RealTimeClock	5 分	active	スーパーバイザ上のリアルタイム クロックが時を刻んでいるかどうかを確認します。
PrimaryBootROM	30 分	active	スーパーバイザ上のプライマリ ブート デバイスの完全性を確認します。
SecondaryBootROM	30 分	active	スーパーバイザ上のセカンダリ ブート デバイスの完全性を確認します。
BootFlash	30 分	active	ブートフラッシュ デバイスへのアクセスを確認します。
USB	30 分	active	USB デバイスへのアクセスを確認します。
SystemMgmtBus	30 秒	active	システム管理バスの使用可能性を確認します。
Mce	30 分	ヘルス モニタリング テスト - 30 分 - active	このテストは mcd_dameon を使用して、カーネルのレポートするすべてのマシン チェック エラーを報告します。
Pcie	ブートアップ時のみ	ブートアップ時のみ - inactive	PCIe ステータス レジスタを読み込んで PCIe デバイスのエラーがチェックされます。

診断	デフォルトのインターバル	デフォルト設定	説明
Module			
コンソール	ブートアップ時のみ	ブートアップ時のみ - inactive	ブートアップ時の管理ポートでポート ループバック テストを実行して一貫性がチェックされます。
FpgaRegTest	30 秒	ヘルス モニタリング テスト - 30 秒ごと - active	FPGA への読み取りと書き込みにより FPGA の状態がテストされます。

オンデマンド診断

オンデマンドテストは、障害の場所を特定するのに役立ちます。通常は、次のような状況で必要です。

- 障害の分離など、発生したイベントに対処する場合。
- リソース使用限度の超過などのイベントの発生が予測される場合。

すべてのヘルスモニタリングテストをオンデマンドで実行できます。即時実行するオンデマンド診断テストをスケジューリングできます。

ヘルス モニタリング テストのデフォルト インターバルも変更可能です。

ハイ アベイラビリティ

ハイアベイラビリティの重要な要素は、アクティブネットワークでデバイスが動作しているときに、ハードウェア障害を検出して対策を取ることです。ハイ アベイラビリティのオンライン診断では、ハードウェア障害を検出して、スイッチオーバーを判断するためにハイ アベイラビリティ ソフトウェアにフィードバックします。

Cisco NX-OS は、オンライン診断のステートレス 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。

仮想化のサポート

オンライン診断機能は Virtual Routing and Forwarding（VRF）を認識します。特定の VRF を使用してオンライン診断 SMTP サーバに接続するようにオンライン診断機能を設定できます。

オンライン診断機能のライセンス要件

製品	ライセンス要件
Cisco NX-OS	オンライン診断機能にライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

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。
- 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。

パラメータ（**Parameters**）	デフォルト
起動時診断レベル	complete
中断を伴わないテスト	active

オンライン診断の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

起動診断レベルの設定

一連のすべてのテストを実行するように起動診断機能を設定することも、またはモジュールが短時間で起動するように、すべての起動診断テストをバイパスするように設定することもできます。

（注） 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。

手順の概要

1. **configure terminal**
2. **diagnostic bootup level** {**complete** | **bypass**}
3. （任意） **show diagnostic bootup level**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**diagnostic bootup level** {**complete** | **bypass**} 例： `switch(config)# diagnostic bootup level complete`	デバイスの起動に続いて診断テストが開始されるように、起動診断レベルを設定します。 • **complete**：すべての起動診断テストを実行します。complete がデフォルトです。 • **bypass**：起動診断テストを実行しません。

	コマンドまたはアクション	目的
ステップ 3	**show diagnostic bootup level** 例： `switch(config)# show diagnostic bootup level`	（任意） 現在、デバイスで実行されている起動診断レベル（bypass または complete）を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

診断テストのアクティブ化

診断テストをアクティブに設定し、任意でテストの実行間隔（時間、分、秒単位）を変更できます。

手順の概要

1. **configure terminal**
2. **diagnostic monitor interval module***slot***test** [*test-id* | *name* | **all**] **hour***hour***min***minute***second***second***
3. [**no**] **diagnostic monitor module***slot***test** [*test-id* | *name* | **all**]
4. （任意） **show diagnostic content module** {*slot* | **all**}

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**diagnostic monitor interval module***slot***test** [*test-id*	*name*	**all**] **hour***hour***min***minute***second***second*** 例： `switch(config)# diagnostic monitor interval module 6 test 3 hour 1 min 0 second 0`	指定されたテストを実行するインターバルを設定します。インターバルを設定しなかった場合は、過去に設定されたインターバルまたはデフォルトのインターバルでテストが実行されます。 引数の範囲は次のとおりです。 • *slot*：範囲は 1 ～ 10 • *test-id*：範囲は 1 ～ 14

	コマンドまたはアクション	目的
		• *name*：最大 32 の英数字を使用できます。大文字と小文字は区別されます。 • *hour*：範囲は 0 ～ 23 時間 • *minute*：範囲は 0 ～ 59 分 • *second*：範囲は 0 ～ 59 秒
ステップ 3	[**no**] **diagnostic monitor module***slot***test** [*test-id* | *name* | **all**] 例： `switch(config)# diagnostic monitor interval module 6 test 3`	指定されたテストをアクティブにします。 引数の範囲は次のとおりです。 • *slot*：範囲は 1 ～ 10 • *test-id*：範囲は 1 ～ 14 • *name*：最大 32 の英数字を使用できます。大文字と小文字は区別されます。 このコマンドの [**no**] 形式は、指定されたテストを非アクティブにします。非アクティブにしたテストでは、現在の設定が維持されますが、スケジュール上のインターバルではテストは実行されません。
ステップ 4	**show diagnostic content module** {*slot* | **all**} 例： `switch(config)# show diagnostic content module 6`	（任意） 診断テストおよび対応する属性の情報を表示します。

オンデマンド診断テストの開始または中止

オンデマンド診断テストを開始または中止できます。任意で、このテストを繰り返す回数の変更や、テストが失敗した場合のアクションの変更を行えます。

スケジューリングされたネットワーク メンテナンス期間内に、破壊モードの診断テストを開始する場合は、手動での開始に限定することを推奨します。

手順の概要

1. （任意） **diagnostic ondemand iteration** *number*
2. （任意） **diagnostic ondemand action-on-failure** {**continue failure-count***num-fails* | **stop**}
3. **diagnostic start module***slot***test** [*test-id* | *name* | **all** | **non-disruptive**] [**port***port-number* | **all**]
4. **diagnostic stop module***slot***test** [*test-id* | *name* | **all**]
5. （任意） **show diagnostic status module***slot*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**diagnostic ondemand iteration** *number* 例： `switch# diagnostic ondemand iteration 5`	（任意） オンデマンド テストの実行回数を設定します。範囲は 1 ～ 999 です。デフォルトは 1 です。
ステップ 2	**diagnostic ondemand action-on-failure** {**continue failure-count***num-fails*	**stop**} 例： `switch# diagnostic ondemand action-on-failure stop`	（任意） オンデマンド テストが失敗した場合のアクションを設定します。*num-fails* の範囲は 1 ～ 999 です。デフォルトは 1 です。
ステップ 3	**diagnostic start module***slot***test** [*test-id*	*name*	**all**	**non-disruptive**] [**port***port-number*	**all**] 例： `switch# diagnostic start module 6 test all`	モジュール上で 1 つまたは複数の診断テストを開始します。モジュール スロットの範囲は 1 ～ 10 です。*test-id* の範囲は 1 ～ 14 です。テスト名は大文字と小文字を区別し、最大 32 の英数字を使用できます。ポート範囲は 1 ～ 48 です。
ステップ 4	**diagnostic stop module***slot***test** [*test-id*	*name*	**all**] 例： `switch# diagnostic stop module 6 test all`	モジュール上で 1 つまたは複数の診断テストを中止します。モジュール スロットの範囲は 1 ～ 10 です。*test-id* の範囲は 1 ～ 14 です。テスト名は大文字と小文字を区別し、最大 32 の英数字を使用できます。
ステップ 5	**show diagnostic status module***slot* 例： `switch# show diagnostic status module 6`	（任意） 診断テストがスケジューリングされていることを確認します。

診断結果のシミュレーション

診断テスト結果のシミュレーションが可能です。

手順の概要

1. **diagnostic test simulation module***slot***test***test-id* {**fail** | **random-fail** | **success**} [**port***number* | **all**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**diagnostic test simulation module***slot***test***test-id* {**fail**	**random-fail**	**success**} [**port***number*	**all**] 例： `switch# diagnostic test simulation module 2 test 2 fail`	テスト結果のシミュレーションを行います。*test-id* の範囲は 1 ～ 14 です。ポート範囲は 1 ～ 48 です。

診断結果の消去

診断テスト結果を消去できます。

手順の概要

1. **diagnostic clear result module** [*slot* | **all**] **test** {*test-id* | **all**}
2. **diagnostic test simulation module***slot***test***test-id* clear

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**diagnostic clear result module** [*slot*	**all**] **test** {*test-id*	**all**} 例： `switch# diagnostic clear result module 2 test all`	指定されたテストのテスト結果を消去します。 引数の範囲は次のとおりです。 • *slot*：範囲は 1 ～ 10 • *test-id*：範囲は 1 ～ 14
ステップ 2	**diagnostic test simulation module***slot***test***test-id* clear 例： `switch# diagnostic test simulation module 2 test 2 clear`	シミュレーションしたテスト結果を消去します。*test-id* の範囲は 1 ～ 14 です。

オンライン診断設定の確認

オンライン診断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 diagnostic bootup level	起動診断に関する情報を表示します。
show diagnostic content module {*slot* | **all**}	モジュールの診断テスト内容に関する情報を表示します。
show diagnostic description module*slot***test** [*test-name* | **all**]	診断テストの説明を表示します。
show diagnostic events [**error** | **info**]	診断イベントをエラーおよび情報イベントタイプ別に表示します。
show diagnostic ondemand setting	オンデマンド診断に関する情報を表示します。
show diagnostic result module*slot* [**test** [*test-name* | **all**]] [**detail**]	診断結果に関する情報を表示します。
show diagnostic simulation module*slot*	シミュレーションした診断テストに関する情報を表示します。
show diagnostic status module*slot*	モジュールのすべてのテストについて、テスト状況を表示します。
show hardware capacity [**eobc** | **forwarding** | **interface** | **module** | **power**]	ハードウェアの機能、およびシステムによる現在のハードウェア使用率の情報を表示します。
show module	オンライン診断テストの状況を含むモジュール情報を表示します。

オンライン診断のコンフィギュレーション例

この例は、モジュール 6 ですべてのオンデマンド テストを開始する方法を示しています。

```
diagnostic start module 6 test all
```

この例は、モジュール 6 でテストテスト 2 をアクティブにして、テスト インターバルを設定する方法を示しています。

```
configure terminal
diagnostic monitor module 6 test 2
diagnostic monitor interval module 6 test 2 hour 3 min 30 sec 0
```

第 15 章

Embedded Event Manager の設定

この章では、Embedded Event Manager（EEM）を設定して Cisco NX-OS デバイス上のクリティカル イベントを検出し、対処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。

EEM について

EEMはデバイス上で発生するイベントをモニタし、設定に基づいて各イベントの回復またはトラブルシューティングのためのアクションを実行します。

EEM は次の 3 種類の主要コンポーネントからなります。

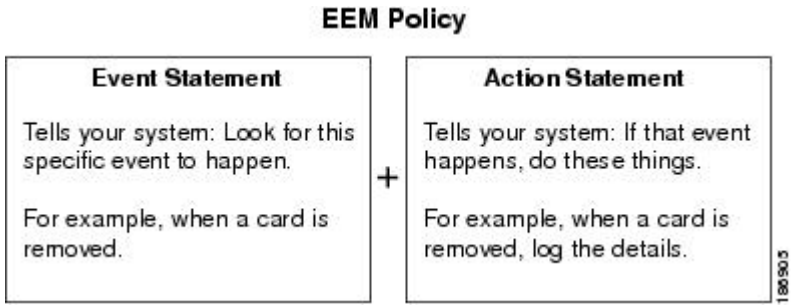
- イベント文：別の Cisco NX-OS コンポーネントからモニタし、アクション、回避策、または通知が必要になる可能性のあるイベント。
- アクション文：電子メールの送信、インターフェイスのディセーブル化など、イベントから回復するために EEM が実行できるアクション。
- ポリシー：イベントのトラブルシューティングまたはイベントからの回復を目的とした 1 つまたは複数のアクションとペアになったイベント。

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。

この図は、EEM ポリシーの基本的な 2 種類の文を示します。

図 *2* : *EEM* ポリシー文



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。スーパーバイザ上で EEM ポリシーを設定すると、EEM がイベント タイプに基づいて、正しいモジュールにポリシーをプッシュします。EEM はモジュール上でローカルに、またはスーパーバイザ上で（デフォルトのオプション）、発生したイベントに対応するアクションを実行します。

EEM はスーパーバイザ上でイベント ログを維持します。

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。

使用するネットワークに合わせてユーザ ポリシーを作成できます。ユーザ ポリシーを作成すると、そのポリシーと同じイベントに関連するシステム ポリシー アクションが EEM によって発生したあと、ユーザ ポリシーで指定したアクションが行われます。

一部のシステム ポリシーは上書きすることもできます。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。イベントまたはアクションの上書きが可能です。

設定済みのシステム ポリシーを表示して、上書き可能なポリシーを判断するには、**show event manager system-policy** コマンドを使用します。

（注） **show running-config eem** コマンドを使用して、各ポリシーのコンフィギュレーションを確認してください。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。

（注） 上書きポリシーには、必ずイベント文を指定します。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。

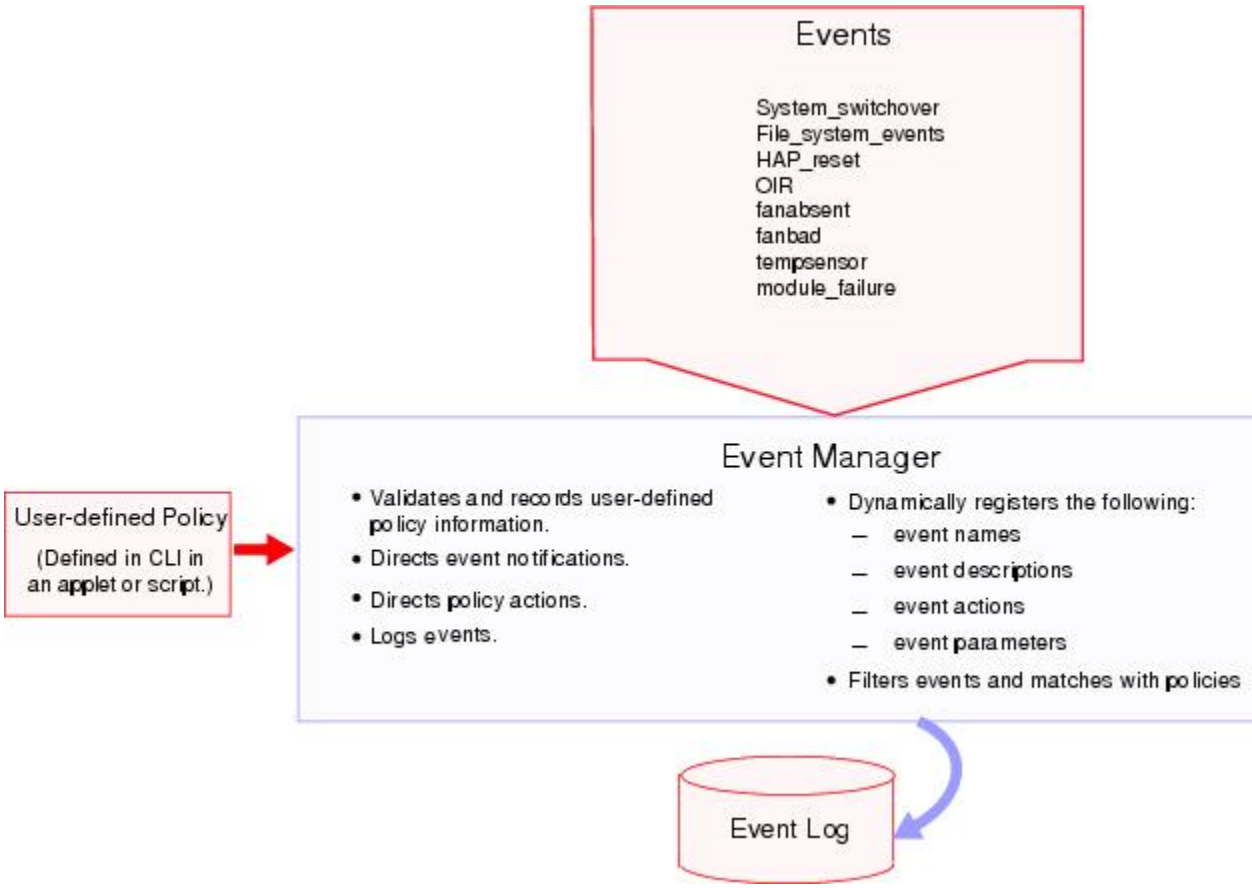
イベント文

イベントは、回避、通知など、何らかのアクションが必要なデバイス アクティビティです。これらのイベントは通常、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。

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。

この図は、EEM によって処理されたイベントを示します。

図 *3*：*EEM* の概要



イベント文では、ポリシー実行のトリガーになるイベントを指定します。複数イベント トリガーを設定できます。

EEM はイベント文に基づいてポリシーをスケジューリングし、実行します。EEM はイベントおよびアクション コマンドを検証し、定義に従ってコマンドを実行します。

（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、event-default アクション文を許可して EEM ポリシーを設定する必要があります。

アクション ステートメント

アクション文では、ポリシーによって実行されるアクションを記述します。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。

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。
- 例外の記録。
- モジュールの強制的シャットダウン
- デバイスのリロード。
- 電力のバジェット超過による特定モジュールのシャットダウン。
- Syslog メッセージの生成。
- Call Home イベントの生成。
- 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。

（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、match 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 CLI コマンドを実行できません。

（注） ユーザ ポリシーまたは上書きポリシーの中に、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アクション文がないかどうかを確認してください。

VSH スクリプト ポリシー

テキストエディタを使用し、VSHスクリプトでポリシーを作成することもできます。このようなポリシーにも、他のポリシーと同様、イベント文およびアクション文（複数可）を使用します。また、これらのポリシーでシステムポリシーを補うことも上書きすることもできます。VSHスクリプト ポリシーを書き込んだ後、デバイスにコピーしてアクティブにします。

環境変数

すべてのポリシーに使用できる、EEMの環境変数を定義できます。環境変数は、複数のポリシーで使用できる共通の値を設定する場合に便利です。たとえば、外部電子メールサーバのIPアドレスに対応する環境変数を作成できます。

パラメータ置換フォーマットを使用することによって、アクション文で環境変数を使用できます。

この例では、「EEM action」というリセット理由を指定し、モジュール 1 を強制的にシャットダウンするアクション文の例を示します。

```
switch (config-eem-policy)# action 1.0 forceshut module 1 reset-reson “EEM action.”
```

シャットダウンの理由に default-reason という環境変数を定義すると、次の例のように、リセット理由を環境変数に置き換えることができます。

```
switch (config-eem-policy)# action 1.0 foreshut module 1 reset-reason $default-reason
```

この環境変数は、任意のポリシーで再利用できます。

EEM イベント相関

イベントの組み合わせに基づいて EEM ポリシーをトリガーできます。まず、**tag** キーワードを使用してEEMポリシーに複数のイベントを作成し区別します。次に、一連のブール演算子（**AND**、**OR**、**ANDNOT**）を使用して、回数および時間をもとに、カスタム処理をトリガーするこれらのイベントの組み合わせを定義できます。

ハイ アベイラビリティ

Cisco NX-OS は、EEM 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。

仮想化のサポート

アクションまたはイベントがすべて表示されるわけではありません。ポリシーを設定するには、network-admin の権限が必要です。

EEM のライセンス要件

製品	ライセンス要件
Cisco NX-OS	EEM にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

EEM の前提条件

EEM の前提条件は、次のとおりです。

- EEM を設定するには、network-admin のユーザ権限が必要です。

EEM の注意事項と制約事項

EEM 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。

- 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、match 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 CLI コマンドを実行できません。
- 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。
- 通常のコマンド式に適用できるルールは、すべてのキーワードを拡張する必要があること、そして * 記号のみが引数の置換に使用できることです。
- EEM イベント相関はスーパーバイザ モジュールだけでサポートされます。
- EEM イベント相関は、単一ポリシー内の別のモジュール間ではサポートされません。
- 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、モジュール、モジュール障害、oir、snmp、syslog 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。
- Python から EEM を呼び出すことができます。Python の詳細については、『*Cisco Nexus 9000 Series NX-OS Programmability Guide*』を参照してください。

EEM のデフォルト設定

次の表に、EEM パラメータのデフォルト設定を示します。

パラメータ（**Parameters**）	デフォルト
システム ポリシー	Active

EEM の設定

システム ポリシーに基づいて実行されるアクションを含むポリシーを作成できます。システム ポリシーに関する情報を表示するには、**show event manager system-policy** コマンドを使用します。

環境変数の定義

EEM ポリシーでパラメータとして機能する変数を定義できます。

手順の概要

1. **configure terminal**
2. **event manager environment***variable-name variable-value*
3. （任意） **show event manager environment** {*variable-name* | **all**}
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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* 例： `switch(config)# event manager environment emailto "admin@anyplace.com"`	EEM 用の環境変数を作成します。*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* には最大 39 文字の英数字を引用符で囲んで使用できます。
ステップ 3	**show event manager environment** {*variable-name* | **all**} 例： `switch(config)# show event manager environment all`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

CLI によるユーザ ポリシーの定義

CLI を使用して、デバイスにユーザ ポリシーを定義できます。

手順の概要

1. **configure terminal**
2. **event manager applet***applet-name*
3. （任意） **description***policy-description*
4. **event***event-statement*
5. （任意） **tag***tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] {**happens***occurs in seconds*}
6. **action***number*[*.number2*] *action-statement*
7. （任意） **show event manager policy-state***name* [**module***module-id*]
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**description***policy-description* 例： `switch(config-applet)# description “Monitors interface shutdown.”`	（任意） ポリシーの説明になるストリングを設定します。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。イベント文が複数ある場合、このステップを繰り返します。イベント文の設定，（238 ページ）を参照してください。
ステップ 5	**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`	（任意） 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。アクション文の設定，（243 ページ）を参照してください。
ステップ 7	**show event manager policy-state***name* [**module***module-id*] 例： `switch(config-applet)# show event manager policy-state monitorShutdown`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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 モードで次のいずれかのコマンドを使用します。

コマンド	目的
event application [**tag***tag*] **sub-system***sub-system-id***type***event-type* 例： `switch(config-applet)# event application sub-system 798 type 1`	イベントの指定がサブシステム ID およびアプリケーション イベント タイプに一致する場合に、イベントを発生させます。 *sub-system-id* と *event-type* の範囲は 1 ～ 4294967295 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。 （注） このコマンドを使用するには、まず **feature evmed** コマンドをイネーブルにして一般的なイベント ディテクタをイネーブルにする必要があります。
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 秒です。0 は無制限を示します。
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。*entry* および *exit* の値の範囲は 0 ～ 2147483647 です。

コマンド	目的
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。*repeats* の範囲は 10 ～ 64000 です。
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。*repeats* の範囲は 10 ～ 64000 です。
event fib {**adjacency extra** | **resource tcam usage** | **route** {**extra** | **inconsistent** | **missing**}} 例： `switch(config-applet)# event fib adjacency extra`	次のいずれかに対するイベントを発生させます。 • **adjacency extra**：ユニキャスト FIB に追加のルートがある場合。 • **resource tcam usage**：TCAM 使用率がいずれかの方向で 5 の倍数になるごとに。 • **route** {**extra** | **inconsistent** | **missing**}：ユニキャスト FIB でルートが追加、変更、または削除される場合。
event gold module {*slot* | **all**} **test***test-name* [**severity** {**major** | **minor** | **moderate**}] **testing-type** {**bootup** | **monitoring** | **ondemand** | **scheduled**} **consecutive-failure***count* 例： `switch(config-applet)# event gold module 2 test ASICRegisterCheck testing-type ondemand consecutive-failure 2`	名前で指定されたオンライン診断テストが、設定された回数だけ連続して、設定された重大度で失敗した場合に、イベントを発生させます。*slot* の範囲は 1 ～ 10 です。*test-name* は設定されたオンライン診断テストの名前です。*count* の範囲は 1 ～ 1000 です。
event interface [**tag***tag*] {**name***interface slot*/*port***parameter**} 例： `switch(config-applet)# event interface ethernet 2/2 parameter`	カウンタが指定のインターフェイスに対して超えた場合に、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。 （注） このコマンドを使用するには、まず **feature evmed** コマンドをイネーブルにして一般的なイベント ディテクタをイネーブルにする必要があります。
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。メモリのしきい値の設定，（248 ページ）も参照してください。

コマンド	目的
event module [**tag***tag*] **status** {**online** | **offline** | **any**} **module** {**all** | *module-num*} 例： `switch(config-applet)# event module status offline module all`	指定したモジュールが選択された状態になったとき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。
event module-failure [**tag***tag*] **type***failure-type***module** {*slot* | **all**} **count***repeats* [**time***seconds*] 例： `switch(config-applet)# event module-failure type lc-failed module 3 count 1`	モジュールが設定された障害タイプになった場合に、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。 *repeats* の範囲は 0 ～ 4294967295 です。*seconds* の範囲は 0 ～ 4294967295 秒です。0 は無制限を示します。
event none 例： `switch(config-applet)# event none`	手動で指定されたイベントがないポリシーイベントを実行します。 （注） このコマンドを使用するには、まず **feature evmed** コマンドをイネーブルにして一般的なイベント ディテクタをイネーブルにする必要があります。
event oir [**tag***tag*] {**fan** | **module** | **powersupply**} {**anyoir** | **insert** | **remove**} [*number*] 例： `switch(config-applet)# event oir fan remove 4`	設定されたデバイス構成要素（ファン、モジュール、または電源モジュール）がデバイスに取り付けられた場合、またはデバイスから取り外された場合に、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。 任意で、ファン、モジュール、または電源モジュールの具体的な番号を設定できます。*number* の範囲は次のとおりです。 • ファン番号：モジュール依存 • モジュール番号：デバイス依存 • 電源モジュール番号：範囲は 1 ～ 3 です。

コマンド	目的
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。
event poweroverbudget 例： `switch(config-applet)# event poweroverbudget`	電力バジェットが設定された電源モジュールの容量を超えた場合に、イベントを発生させます。
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。 *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 1 ～ 2147483647 秒です。
event storm-control 例： `switch(config-applet)# event storm-control`	ポート上のトラフィックが設定されたストーム制御しきい値を超えた場合に、イベントを発生させます。
event syslog [**occurs***count*] {**pattern***string* | **period***time* | **priority***level* | **tag***tag*} 例： `switch(config-applet)# event syslog period 500`	指定した syslog のしきい値を超えた場合にイベントを発生させます。カウントの範囲は 1 ～ 65000 で、時間の範囲は 1 ～ 4294967295 です。プライオリティの範囲は 0 ～ 7 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。
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。パーセンテージの範囲は 1 ～ 99 です。

コマンド	目的
event sysmgr switchover count*count***time***interval* 例： `switch(config-applet)# event sysmgr switchover count 10 time 1000`	指定した switchover count が、指定した time interval を超えた場合にイベントを発生させます。switchover count の範囲は 1 ～ 65000 です。time interval の範囲は 0 ～ 2147483647 です。
event temperature [**module***slot*] [*sensor-number*] **threshold** {**any** | **major** | **minor**} 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。
event timer {**absolute time***time***name***name* | **countdown time***time***name***name* | **cron cronentry***string* | **tag***tag* | **watchdog time***time***name***name*} 例： `switch(config-applet)# event timer absolute time 100 name abtimer`	指定した時間に到達した場合に、イベントを発生させます。時間の範囲は 1 ～ 4294967295 です。 • **absolute time**：指定された絶対時刻が発生した場合に、イベントを発生させます。 • **countdown time**：指定された時間がゼロにカウントダウンされたときに、イベントを発生させます。タイマーはリセットされません。 • **cron cronentry**：CRON 文字列の指定が現在時刻に一致する場合に、イベントを発生させます。 • **watchdog time**：指定された時間がゼロにカウントダウンされたときに、イベントを発生させます。タイマーは、初期値に自動的にリセットされ、カウント ダウンが続行されます。 **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。 （注） このコマンドを使用するには、まず **feature evmed** コマンドをイネーブルにして一般的なイベント ディテクタをイネーブルにする必要があります。

コマンド	目的
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。

アクション文の設定

アクション文を設定するには、EEM コンフィギュレーション モードで次のいずれかのコマンドを使用します。

コマンド	目的
action*number*[.*number2*] **cli***command1* [*command2...*][**local**] 例： `switch(config-applet)# action 1.0 cli "show interface e 3/1"`	設定された CLI 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* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
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* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 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。*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。
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* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。

コマンド	目的
action*number*[.*number2*] **forceshut** [**module***slot* | **xbar***xbar-number*] **reset-reason***seconds* 例： `switch(config-applet)# action 1.0 forceshut module 2 reset-reason "flapping links"`	モジュール、クロスバー、またはシステム全体を強制的にシャットダウンします。アクション ラベルのフォーマットはnumber1.number2 です。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 リセット理由は、引用符で囲んだ最大 80 文字の英数字ストリングです。
action*number*[.*number2*] **overbudgetshut** [**module***slot*[-*slot*]] 例： `switch(config-applet)# action 1.0 overbudgetshut module 3-5`	電力バジェット超過の問題により、1 つまたは複数のモジュールまたはシステム全体を強制的にシャットダウンします。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
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* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action*number*[.*number2*] **publish-event** 例： `switch(config-applet)# action 1.0 publish-event`	アプリケーション固有のイベントの発行を強制します。アクション ラベルのフォーマットは number1.number2 です。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action*number*[.*number2*] **reload** [**module***slot*[-*slot*]] 例： `switch(config-applet)# action 1.0 reload module 3-5`	1 つまたは複数のモジュールまたはシステム全体を強制的にリロードします。 *number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。
action*number*[.*number2*] **snmp-trap** {[**intdata1***data* [**intdata2***data*]] [**strdata***string*]} 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。*number* は 16 桁までの任意の数値にできます。*number2* の範囲は 0 ～ 9 です。 *data* 引数には、最大 80 桁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。

コマンド	目的
action*number*[.*number2*] **syslog** [**priority***prio-val*] **msg***error-message* 例： `switch(config-applet)# action 1.0 syslog priority notifications msg "cpu high"`	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。*number* は 16 桁までの任意の数値にできます。*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。

（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、match 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 CLI 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、CLI が一致するすべての EEM ポリシーで、CLI コマンドを実行できます。

VSH スクリプトによるポリシーの定義

VSH スクリプトを使用してポリシーを定義できます。

はじめる前に

管理者の権限でログインしていることを確認します。

スクリプト名がスクリプト ファイル名と同じ名前であることを確認します。

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

VSH スクリプト ポリシーの登録およびアクティブ化

VSH スクリプトで定義したポリシーを登録してアクティブにできます。

手順の概要

1. **configure terminal**
2. **event manager policy***policy-script*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 の英数字を使用できます。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

ポリシーの上書き

システム ポリシーは上書き可能です。

手順の概要

1. **configure terminal**
2. （任意） **show event manager policy-state***system-policy*
3. **event manager applet***applet-name***override***system-policy*
4. （任意） **description***policy-description*
5. **event***event-statement*
6. **action***number action-statement*
7. （任意） **show event manager policy-state***name*
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**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`
		（任意）
		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。システムポリシー名を突き止めるには、**show event manager system-policy** コマンドを使用します。システム ポリシーについては、Embedded Event Manager システム イベントおよびコンフィギュレーション例, （385 ページ）を参照してください。

ステップ 3	**event manager applet***applet-name***override***system-policy*
	
	例：
	`switch(config)# 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 29 の英数字を使用できます。*system-policy* は、既存のシステム 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： `description "Overrides link flap policy."`	（任意） ポリシーの説明になるストリングを設定します。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 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	**action***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。 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	**show event manager policy-state***name* 例： `switch(config-applet)# show event manager policy-state ethport`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

メモリのしきい値の設定

イベントを発生させるメモリしきい値を設定し、オペレーティング システムがメモリを割り当てられない場合にプロセスを終了させるかどうかを設定できます。

はじめる前に

管理者の権限でログインしていることを確認します。

手順の概要

1. **configure terminal**
2. **system memory-thresholds minor***minor***severe***severe***critical***critical***
3. （任意） **system memory-thresholds threshold critical no-process-kill**
4. （任意） **show running-config | include "system memory"**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 memory-thresholds minor***minor***severe***severe***critical***critical*** 例： `switch(config)# system memory-thresholds minor 60 severe 70 critical 80`	EEM メモリ イベントを生成するシステム メモリしきい値を設定します。デフォルト値は次のとおりです。 • Minor-85 • Severe-90 • Critical-95 これらのメモリのしきい値を超えた場合、システムは次の syslog を生成します。 • 2013 May 7 17:06:30 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR • 2013 May 7 17:06:30 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE

	コマンドまたはアクション	目的
		• 2013 May 7 17:06:30 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL • 2013 May 7 17:06:35 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR ALERT RECOVERED • 2013 May 7 17:06:35 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE ALERT RECOVERED • 2013 May 7 17:06:35 switch %$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL ALERT RECOVERED
ステップ 3	**system memory-thresholds threshold critical no-process-kill** 例： `switch(config)# system memory-thresholds threshold critical no-process-kill`	（任意） メモリを割り当てることができない場合にプロセスを停止しないようにシステムを設定します。デフォルト値では、最もメモリを消費するプロセスから終了できます。
ステップ 4	**show running-config | include "system memory"** 例： `switch(config-applet)# show running-config | include "system memory"`	（任意） システム メモリ設定に関する情報を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

EEM パブリッシャとしての syslog の設定

スイッチからの syslog メッセージをモニタできます。

（注） syslog メッセージをモニタする検索文字列の最大数は 10 です。

はじめる前に

EEM は、Syslog による登録に使用可能である必要があります。

Syslog デーモンが設定され、実行される必要があります。

手順の概要

1. **configure terminal**
2. **event manager applet***applet-name*
3. **event syslog** [**tag***tag*] {**occurs***number* | **period***seconds* | **pattern***msg-text* | **priority***priority*}
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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-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。

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`	syslog メッセージを監視し、ポリシーの検索文字列に基づいてポリシーを呼び出します。 • **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。 • **occurs***number* のキーワードと引数のペアは、発生回数を指定します。指定できる範囲は 1 ～ 65000 です。 • **period***seconds* のキーワードと引数のペアは、イベントの発生間隔を指定します。範囲は 1 ～ 4294967295 です。 • **pattern***msg-text* のキーワードと引数のペアは、一致する正規表現を指定します。パターンには、文字テキスト、環境変数、またはこの 2 つの組み合わせを含めることができます。文字列に空白が含まれる場合は引用符で囲みます。 • **priority***priority* のキーワードと引数のペアは、syslog メッセージのプライオリティを指定します。このキーワードを指定しないと、すべての Syslog メッセージのプライオリティ レベルが「情報レベル」に設定されます。

	コマンドまたはアクション	目的
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

EEM 設定の確認

EEM のコンフィギュレーション情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show 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。
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。

EEM のコンフィギュレーション例

モジュール 3 の中断のないアップグレード エラー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方法の例を示します。この例では、syslog メッセージも送信されます。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、CLI 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 CLI 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
```

（注） 追加の EEM の設定例については、Embedded Event Manager システム イベントおよびコンフィギュレーション例, （385 ページ）を参照してください。

第 16 章

オンボード障害ロギングの設定

この章では、Cisco NX-OS デバイスで Onboard Failure Logging（OBFL）機能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。

OBFL の概要

Cisco NX-OS には永続ストレージに障害データを記録する機能があるので、あとから記録されたデータを取得して表示し、分析できます。この OBFL 機能は、障害および環境情報をモジュールの不揮発性メモリに保管します。この情報は、障害モジュールの分析に役立ちます。

OBFL は次のタイプのデータを保存します。

- 最初の電源投入時刻
- モジュールのシャーシ スロット番号
- モジュールの初期温度
- ファームウェア、BIOS、FPGA、および ASIC のバージョン

- モジュールのシリアル番号
- クラッシュのスタック トレース
- CPU hog 情報
- メモリ リーク情報
- ソフトウェア エラー メッセージ
- ハードウェア例外ログ
- 環境履歴
- OBFL 固有の履歴情報
- ASIC 割り込みおよびエラー統計の履歴
- ASIC レジスタ ダンプ

OBFL のライセンス要件

製品	ライセンス要件
Cisco NX-OS	OBFL にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

OBFL の前提条件

network-admin ユーザ権限が必要です。

OBFL の注意事項と制約事項

OBFL に関する注意事項および制約事項は、次のとおりです。

- OBFL はデフォルトでイネーブルになっています。
- OBFL フラッシュがサポートする書き込みおよび消去の回数には制限があります。イネーブルにするロギング数が多いほど、この書き込みおよび消去回数に早く達してしまいます。

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

OBFL のデフォルト設定

次の表に、OBFL パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
OBFL	すべての機能がイネーブル

OBFL の設定

Cisco NX-OS デバイス上で OBFL 機能を設定できます。

はじめる前に

グローバル コンフィギュレーション モードになっていることを確認します。

手順の概要

1. **configure terminal**
2. **hw-module logging onboard**
3. **hw-module logging onboard counter-stats**
4. **hw-module logging onboard cpuhog**
5. **hw-module logging onboard environmental-history**
6. **hw-module logging onboard error-stats**
7. **hw-module logging onboard interrupt-stats**
8. **hw-module logging onboard module***slot*
9. **hw-module logging onboard obfl-logs**
10. （任意） **show logging onboard**
11. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**hw-module logging onboard**
	
	例：
	`switch(config)# hw-module logging onboard`
	`Module: 7 Enabling ... was successful.`
	`Module: 10 Enabling ... was successful.`
	`Module: 12 Enabling ... was successful.`
		すべての OBFL 機能をイネーブルにします。

ステップ 3	**hw-module logging onboard counter-stats**
	
	例：
	`switch(config)# hw-module logging onboard counter-stats`
	`Module: 7 Enabling counter-stats ... was successful.`
	`Module: 10 Enabling counter-stats ... was successful.`
	`Module: 12 Enabling counter-stats ... was successful.`
		OBFL カウンタ統計情報をイネーブルにします。

ステップ 4	**hw-module logging onboard cpuhog**
	
	例：
	`switch(config)# hw-module logging onboard cpuhog`
	`Module: 7 Enabling cpu-hog ... was successful.`
	`Module: 10 Enabling cpu-hog ... was successful.`
	`Module: 12 Enabling cpu-hog ... was successful.`
		OBFL CPU hog イベントをイネーブルにします。

ステップ 5	**hw-module logging onboard environmental-history**
	
	例：
	`switch(config)# hw-module logging onboard environmental-history`
	`Module: 7 Enabling environmental-history ... was successful.`
	`Module: 10 Enabling environmental-history ... was successful.`
	`Module: 12 Enabling environmental-history ... was successful.`
		OBFL 環境履歴をイネーブルにします。

ステップ 6	**hw-module logging onboard error-stats**
	
	例：
	`switch(config)# hw-module logging onboard error-stats`
	`Module: 7 Enabling error-stats ... was successful.`
	`Module: 10 Enabling error-stats ... was successful.`
	`Module: 12 Enabling error-stats ... was successful.`
		OBFL エラー統計をイネーブルにします。

ステップ 7	**hw-module logging onboard interrupt-stats**
	
	例：
	`switch(config)# hw-module logging onboard interrupt-stats`
	`Module: 7 Enabling interrupt-stats ... was successful.`
	`Module: 10 Enabling interrupt-stats ... was successful.`
	`Module: 12 Enabling interrupt-stats ... was successful.`
		OBFL 割り込み統計をイネーブルにします。

ステップ 8	**hw-module logging onboard module***slot*
	
	例：
	`switch(config)# hw-module logging onboard module 7`
	`Module: 7 Enabling ... was successful.`
		モジュールの OBFL 情報をイネーブルにします。

	コマンドまたはアクション	目的

ステップ 9	**hw-module logging onboard obfl-logs**
	
	例：
	`switch(config)# hw-module logging onboard obfl-logs`
	`Module: 7 Enabling obfl-log ... was successful.`
	`Module: 10 Enabling obfl-log ... was successful.`
	`Module: 12 Enabling obfl-log ... was successful.`
		ブート動作時間、デバイス バージョン、およびOBFL 履歴をイネーブルにします。

ステップ 10	**show logging onboard** 例： `switch(config)# show logging onboard`	（任意） OBFL に関する情報を表示します。
ステップ 11	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

OBFL コンフィギュレーションの確認

モジュールのフラッシュに保存されている OBFL 情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show logging onboard boot-uptime	ブートおよび動作時間の情報を表示します。
show logging onboard counter-stats	すべての ASIC カウンタについて、統計情報を表示します。
show logging onboard credit-loss	OBFL クレジット損失のログを表示します。
show logging onboard device-version	デバイス バージョン情報を表示します。
show logging onboard endtime	指定した終了時刻までのOBFL ログを表示します。
show logging onboard environmental-history	環境履歴を表示します。
show logging onboard error-stats	エラー統計情報を表示します。
show logging onboard exception-log	例外ログ情報を表示します。
show logging onboard interrupt-stats	割り込み統計情報を表示します。
show logging onboard module*slot*	指定したモジュールの OBFL 情報を表示します。

コマンド	目的
show logging onboard obfl-history	履歴情報を表示します。
show logging onboard obfl-logs	ログ情報を表示します。
show logging onboard stack-trace	カーネル スタック トレース情報を表示します。
show logging onboard starttime	指定した開始時刻からの OBFL ログを表示します。
show logging onboard status	OBFL ステータス情報を表示します。

OBFL の設定ステータスを表示するには、**show logging onboard status** コマンドを使用します。

```
switch# show logging onboard status
----------------------------
OBFL Status
----------------------------
Switch OBFL Log: Enabled

Module: 4 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled

Module: 22 OBFL Log: Enabled
cpu-hog Enabled
credit-loss Enabled
environmental-history Enabled
error-stats Enabled
exception-log Enabled
interrupt-stats Enabled
mem-leak Enabled
miscellaneous-error Enabled
obfl-log (boot-uptime/device-version/obfl-history) Enabled
register-log Enabled
request-timeout Enabled
stack-trace Enabled
system-health Enabled
timeout-drops Enabled
stack-trace Enabled
```

上記の各 **show** コマンド オプションの OBFL 情報を消去するには、**clear logging onboard** コマンドを使用します。

OBFL のコンフィギュレーション例

モジュール 2 で環境情報について OBFL をイネーブルにする例を示します。

```
switch# configure terminal
switch(config)# hw-module logging onboard module 2 environmental-history
```

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
コンフィギュレーション ファイル	『*Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide*』

関連資料

第 17 章

SPAN の設定

この章では、Cisco NX-OS デバイス上のポート間のトラフィックを分析するようにイーサネット スイッチド ポート アナライザ（SPAN）を設定する方法について説明します。

この章の内容は、次のとおりです。

SPAN の概要

SPAN は、外付けアナライザが接続された宛先ポートに SPAN セッション トラフィックを送ることで、送信元ポート間のすべてのトラフィックを分析します。

ローカル デバイス上で、SPAN セッションでモニタする送信元と宛先を定義できます。

SPAN ソース

トラフィックを監視できる監視元インターフェイスのことを SPAN 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。SPAN 送信元には次のものが含まれます。

- イーサネット ポート(サブインターフェイスではない)
- ポート チャネル
- コントロール プレーン CPU への帯域内インターフェイス。

（注） SPAN 送信元としてスーパーバイザ インバンド インターフェイスを指定すると、デバイスはスーパーバイザ ハードウェアに入力方向に到達したすべてのパケットをモニタします。

- VLANs

（注） VLAN を SPAN 送信元として指定すると、VLAN 内でサポートされているすべてのインターフェイスが SPAN 送信元になります。

（注） VLAN ポートは、入力方向にのみ SPAN 送信元となることができます。

- Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）のサテライト ポートおよびホスト インターフェイス ポート チャネル

（注） これらのインターフェイスは、レイヤ 2 アクセス モードおよびレイヤ 2 トランク モードでサポートされます。これらはレイヤ 3 モードではサポートされず、レイヤ 3 サブインターフェイスはサポートされません。

（注） FEX ポートが SPAN 送信元としてサポートされるのは、入力方向についてはすべてのトラフィックに対してですが、出力方向については既知のレイヤ 2 ユニキャスト トラフィックだけです。

（注） 1 つの SPAN セッションに、上述の送信元を組み合わせて使用できます。

送信元ポートの特性

SPAN 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。
- スーパーバイザ帯域内インターフェイスを SPAN 送信元として使用する場合、次のパケットがモニタされます。

- スーパーバイザ ハードウェアに着信するすべてのパケット（入力）
- スーパーバイザ ハードウェアによって生成されるすべてのパケット（出力）

SPAN 宛先

SPAN 宛先とは、送信元ポートを監視するインターフェイスを指します。宛先ポートは SPAN 送信元からコピーされたトラフィックを受信します。SPAN 宛先には、次のものが含まれます。

- アクセス モードまたはトランク モードのイーサネット ポート
- アクセス モードまたはトランク モードのポート チャネル
- Cisco Nexus 9300 シリーズ スイッチのアップリンク ポート
- Cisco Nexus 9200 シリーズ スイッチの CPU（Cisco NX-OS リリース 7.0(3)I4(1) 以降）

（注） FEX ポートは SPAN 宛先ポートとしてサポートされません。

宛先ポートの特性

SPAN 宛先元ポートには、次の特性があります。

- 宛先ポートとして設定されたポートを送信元ポートとしても設定することはできません。
- 宛先ポートは、一度に 1 つの SPAN セッションだけで設定できます。
- 宛先ポートはスパニングツリー インスタンスに関与しません。SPAN 出力にはブリッジ プロトコル データ ユニット（BPDU）スパニングツリー プロトコル hello パケットが含まれます。

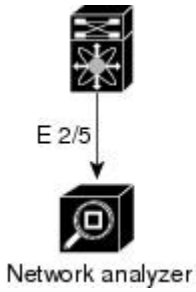
SPAN セッション

モニタする送信元と宛先を指定する SPAN セッションを作成できます。

サポートされる SPAN セッション数に関する情報については、『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。

この図では、SPAN の設定を示します。3 つのイーサネット ポート上のパケットが宛先ポートのイーサネット 2/5 にコピーされます。コピーされるのは、指定した方向のトラフィックだけです。

図 *4*：*SPAN* の設定



Source Port	Direction	Destination Ports
E 2/1	Rx	E 2/5
E 2/2	Rx, Tx	
E 2/3	Tx	

186383

ローカライズされた SPAN セッション

SPANセッションがローカライズされるのは、すべての送信元インターフェイスが同じライン カードにある場合です。セッションの宛先インターフェイスは、任意のライン カードとすることが可能です。

（注） VLAN 送信元の SPAN セッションはローカライズされません。

ACL TCAM リージョン

ハードウェアの ACL Ternary Content Addressable Memory（TCAM）リージョンのサイズを変更できます。SPAN セッションで使用される TCAM リージョンの詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』の「Configuring IP ACLs」の章を参照してください。

ハイ アベイラビリティ

SPAN 機能はステートレス リスタート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を適用します。ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*』を参照してください。

SPAN のライセンス要件

製品	ライセンス要件
Cisco NX-OS	SPAN にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

SPAN の前提条件

SPAN の前提条件は、次のとおりです。

- 各デバイス上で、まず所定の SPAN 設定をサポートするポートを設定する必要があります。詳細については、『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』を参照してください。

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。

- SPAN セッションの制限については、『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。
- ライン カードごとの SPAN セッションの数は、同じインターフェイスが複数セッションの双方向送信元として設定されている場合は、2 に減少します。
- 1 つのフォワーディング エンジン インスタンスで 4 つの SPAN セッションがサポートされます。Cisco Nexus 9300 シリーズ スイッチの場合は、最初の 3 つのセッションに双方向送信元が含まれていると、4 番目のセッションのハードウェア リソースは Rx 送信元専用になります。この制限は、SPAN 送信元のフォワーディング エンジン インスタンス マッピングに応じて、Cisco Nexus 9500 シリーズ スイッチにも適用される場合があります。
- SPAN は、管理ポートではサポートされません。
- Cisco Nexus 9300 シリーズ スイッチは、Tx SPAN を 40G アップリンク ポートでサポートしません。
- Cisco Nexus 9300 シリーズ スイッチ 40G アップリンク インターフェイスの SPAN コピーは、Rx 方向にスパンする際に dot1q 情報を取り逃がします。
- スイッチ インターフェイスのアクセス ポートの出力 SPAN コピーには、常に dot1q ヘッダーがあります。

- VLAN は、SPAN 送信元またはフィルタとして使用される場合、属することができるのは 1 つのセッションだけです。
- VLAN 送信元は Rx 方向でのみスパンします。
- スーパーバイザ生成の Stream Of Bytes Module Header（SOBMH）パケットには、インターフェイスから出力されるための情報がすべて含まれており、SPAN および ERSPAN を含めた、ハードウェア内部でのフォワーディング ルックアップはすべてバイパス可能です。パケットのブリッジ プロトコル データ ユニット（BPDU）クラスおよびレイヤ 3 インターフェイスでの CPU 生成のフレームは、SOBMH を使用して送信されます。
- ポート チャネルが SPAN の宛先として使用される場合、これらがロード バランシング用に使用するメンバーは 8 未満です。
- フィルタ アクセス グループの統計情報はサポートされません。
- SPAN セッションでの access-group フィルタは、vlan-accessmap として設定する必要があります。
- セッション フィルタリング機能（VLAN または ACL フィルタ）は、Rx 送信元でのみサポートされます。
- すべてのスパンのレプリケーションはハードウェアで行われます。スーパーバイザ CPU は関与しません。
- 1 つの SPAN セッションには 1 つの宛先ポートのみを設定できます。
- 宛先ポートは、一度に 1 つの SPAN セッションだけで設定できます。
- ポートをソース 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。
- SPAN は、N9K-X9408PC-CFP2 ライン カード ポートの宛先をサポートしません。
- SPAN 宛先ポートへの VLAN ACL リダイレクトはサポートされません。
- VLAN および ACL フィルタは、FEX ポートではサポートされません。
- 双方向 SPAN セッションで使用される送信元が同じ FEX から出されている場合、ハードウェア リソースは 2 つの SPAN セッションに限定されます。
- SPAN セッションに、送信方向または送信および受信方向でモニタ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 SPAN の宛先ポートに複製される可能性があります。ソース 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
- 不明ユニキャストでフラッディングされたパケットのルーティング後のフローは SPAN セッションに置かれますが、これはフローが転送されるポートをモニタしないよう SPAN セッションが設定されている場合であっても同様です。この制限は、ネットワーク フォワーディング エンジン（NFE）と NFE2 対応 EOR スイッチおよび SPAN セッションで TX ポートの送信元を持つものに適用されます。

- VLAN SPAN がモニタするのは、VLAN のレイヤ 2 ポートが受信するトラフィックだけです。
- VLAN は、入力方向のみの SPAN 送信元としてサポートされます。
- FEX ポートが SPAN 送信元としてサポートされるのは、入力方向についてはすべてのトラフィックに対してですが、出力方向については既知のレイヤ 2 ユニキャスト トラフィックだけです。
- SPAN 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。
- SPAN はレイヤ 3 モードでサポートされます。ただし、SPAN は、レイヤ 3 サブインターフェイスやレイヤ 3 ポート チャネル サブインターフェイスではサポートされません。
- Cisco NX-OS は、送信元インターフェイスがホスト インターフェイス ポート チャネルでないときは、リンク層検出プロトコル（LLDP）またはリンク集約制御プロトコル（LACP）パケットをスパンしません。
- SPAN セッションは、セッションの送信元がスーパーバイザのイーサネット インバンド インターフェイスの場合、ARP 要求および Open Shortest Path First（OSPF）プロトコル hello パケットのようなスーパーバイザに到達するブロードキャストまたはマルチキャスト MAC アドレスを持つパケットをキャプチャできません。これらのパケットをキャプチャするには、SPAN セッションの送信元として物理インターフェイスを使用する必要があります。
- Cisco NX-OS リリース 7.0(3)I4(1) 以降では、同じ送信元が複数のセッションに属することができます。
- Cisco NX-OS リリース 7.0(3)I4(1) 以降では、Cisco Nexus 9300 および 9500 シリーズ スイッチで同じ送信元の複数の ACL フィルタをサポートします。
- 次の注意事項と制約事項は、Cisco Nexus 9200 シリーズ スイッチに適用されます。
  - UDF ベース SPAN は、Cisco NX-OS リリース 7.0(3)I4(1) 以降でサポートされます。
  - Tx SPAN は、マルチキャスト、未知のマルチキャスト、およびブロードキャスト トラフィックではサポートされません。
  - SPAN 送信元と宛先が同じスライスにあり、スライスに転送インターフェイスがない場合、Rx SPAN はマルチキャストでサポートされません。これがサポートされるのは、スライスに転送インターフェイスがあるか、あるいは SPAN の送信元と宛先が異なるスライスにある場合です。
  - 同じスライスにある複数の出力ポートで出力 SPAN トラフィックの輻輳が生じると、これらの出力ポートではライン レートを得られません。
  - ACL フィルタを使用した、親インターフェイスでのサブインターフェイス トラフィックのスパンは、サポートされません。
  - CPU SPAN 送信元は、Rx 方向（CPU からの SPAN パケット）でのみ追加できます。
  - 複数の ACL フィルタは、同じ送信元ではサポートされません。
  - CPU への SPAN パケットはレート制限されており、インバンド パスでドロップされます。レート制限の変更は、**hardware rate-limiter span** コマンドで行えます。スーパーバ

イザの SPAN コピーの分析は、**ethanalyzer local interface inband mirror detail** コマンドで行えます。

SPAN のデフォルト設定

次の表に、SPAN パラメータのデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
SPAN セッション	シャット ステートで作成されます。

SPAN の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドと異なる場合があります。

SPAN セッションの設定

SPAN 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、SPAN セッションはシャット ステートで作成されます。

（注） 双方向性の従来のセッションでは、トラフィックの方向を指定せずにセッションを設定できます。

はじめる前に

アクセス モードまたはトランク モードで宛先ポートを設定する必要があります。詳細については、『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』を参照してください。

手順の概要

1. **configure terminal**
2. **interface***interface slot*/*port*
3. **switchport**
4. **switchport monitor**
5. （任意） ステップ 2 ～ 4 を繰り返して、追加の SPAN 宛先でモニタリングを設定します。
6. **no monitor session***session-number*
7. **monitor session***session-number* [**rx** | **tx**] [**shut**]
8. **description***description*
9. **source** {**interface***type* [**rx** | **tx** | **both**] | **vlan** {*number* | *range*} [**rx**]}
10. （任意） ステップ 9 を繰り返して、すべての SPAN 送信元を設定します。
11. （任意） **filter vlan** {*number* | *range*}
12. （任意） ステップ 11 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。
13. （任意） **filter access-group***acl-filter*
14. **destination interface***typeslot*/*port*
15. **no shut**
16. （任意） **show monitor session** {**all** | *session-number* | **range***session-range*} [**brief**]
17. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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***interface slot*/*port*
	
	例：
	`switch(config)# interface ethernet 2/5`
	`switch(config-if)#`
		選択したスロットおよびポート上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	**switchport** 例： `switch(config-if)# switchport`	選択したスロットおよびポートまたはポート範囲でスイッチポート パラメータを設定します。
ステップ 4	**switchport monitor** 例： `switch(config-if)# switchport monitor`	SPAN 宛先としてスイッチポート インターフェイスを設定します。

	コマンドまたはアクション	目的
ステップ 5	ステップ 2 ～ 4 を繰り返して、追加の SPAN宛先でモニタリングを設定します。	（任意） —
ステップ 6	**no monitor session***session-number* 例： `switch(config)# no monitor session 3`	指定した SPAN セッションのコンフィギュレーションを消去します。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 7	**monitor session***session-number* [**rx** | **tx**] [**shut**]
	
	例：
	`switch(config)# monitor session 3 rx`
	`switch(config-monitor)#`
	
	例：
	`switch(config)# monitor session 3 tx`
	`switch(config-monitor)#`
	
	例：
	`switch(config)# monitor session 3 shut`
	`switch(config-monitor)#`
		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。デフォルトでは、セッションが shut ステートで作成されます。このセッションは、ローカル SPAN セッションです。オプションの shut キーワードは、選択したセッションに対して shut ステートを指定します。

ステップ 8	**description***description* 例： `switch(config-monitor)# description my_span_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 9	**source** {**interface***type* [**rx** | **tx** | **both**] | **vlan** {*number* | *range*} [**rx**]}
	
	例：
	`switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	
	例：
	`switch(config-monitor)# source interface port-channel 2`
	
	例：
	`switch(config-monitor)# source interface sup-eth 0 both`
	
	例：
	`switch(config-monitor)# source vlan 3, 6-8 rx`
	
	例：
	`switch(config-monitor)# source interface ethernet 101/1/1-3`
		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。一定範囲のイーサネット ポート、ポート チャネル、インバンド インターフェイス、一定範囲の VLAN、または Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）上のサテライト ポートまたはホスト インターフェイス ポート チャネルを入力できます。
		
		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。
		
		コピー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。
		
		（注） 送信元 VLAN は、入力方向に限りサポートされます。送信元 FEX ポートがサポートされるのは、入力方向についてはすべてのトラフィックに対してですが、出力方向については既知のレイヤ 2 ユニキャスト トラフィックだけです。
		単一方向のセッションには、送信元の方向はセッションで指定された方向に一致する必要があります。

	コマンドまたはアクション	目的
ステップ 10	ステップ 9 を繰り返して、すべての SPAN 送信元を設定します。	（任意） —
ステップ 11	**filter vlan** {*number* | *range*} 例： `switch(config-monitor)# filter vlan 3-5, 7`	（任意） 設定された送信元から選択する VLAN を設定します。VLAN 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。 （注） SPAN 送信元として設定された FEX ポートは VLAN フィルタをサポートしません。
ステップ 12	ステップ 11 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。	（任意） —
ステップ 13	**filter access-group***acl-filter* 例： `switch(config-monitor)# filter access-group ACL1`	（任意） ACL を SPAN セッションにアソシエートします。

ステップ 14	**destination interface***typeslot/port*
	
	例：
	`switch(config-monitor)# destination interface ethernet 2/5`
	
	例：
	`switch(config-monitor)# destination interface sup-eth 0`
		コピーされたソース パケットの宛先を設定します。
		（注） SPAN 宛先ポートはアクセス ポートまたはトランク ポートを指定する必要があります。
		（注） 宛先ポートでモニタ モードをイネーブルにする必要があります。
		Cisco NX-OS リリース 7.0(3)I4(1) 以降では、Cisco Nexus 9200 シリーズ スイッチだけの SPAN 宛先として CPU を設定できます。そのためには、インターフェイス タイプとして **sup-eth 0** を入力します。

ステップ 15	**no shut** 例： `switch(config-monitor)# no shut`	SPAN 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 16	**show monitor session** {**all** | *session-number* | **range***session-range*} [**brief**] 例： `switch(config-monitor)# show monitor session 3`	（任意） SPAN 設定を表示します。
ステップ 17	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

UDF ベース SPAN の設定

デバイスの設定では、外部または内部パケット フィールド（ヘッダーまたはペイロード）のユーザ定義フィールド（UDF）と照合させ、一致したパケットを SPAN の宛先に送信させるようにできます。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。

はじめる前に

UDF ベース SPAN をイネーブルにするのに十分な空き領域を確保するために、**hardware access-list tcam region** コマンドを使用して適切な TCAM リージョン（racl、ifacl、または vacl）が設定されていることを確認します。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「Configuring ACL TCAM Region Sizes」の項を参照してください。

手順の概要

1. **configure terminal**
2. **udf***udf-name offset-base offset length*
3. **hardware access-list tcam region** {**racl** | **ifacl** | **vacl**} **qualify udf***udf-names*
4. **copy running-config startup-config**
5. **reload**
6. **ip access-list***span-acl*
7. 次のいずれかのコマンドを入力します。
   - **permit udf***udf-name value mask*
   - **permit ip***source destination***udf***udf-name value mask*
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**udf***udf-name offset-base offset length*
	
	例：
	`switch(config)# udf udf-x packet-start 12 1`
	`switch(config)# udf udf-y header outer l3 20 2`
		次のように UDF を定義します。
		• *udf-name*：UDF の名前を指定します。名前には最大 16 文字の英数字を入力できます。
		• *offset-base*：UDF オフセット ベースを次のように指定しますが、ここでの **header** はオフセットを考慮したパケット ヘッダーです。**packet-start** | **header** {**outer** | **inner** {**l3** | **l4**}}。

	コマンドまたはアクション	目的
		• *offset*：オフセット ベースからオフセットさせるバイト数を指定します。オフセット ベース（レイヤ 3/レイヤ 4 ヘッダー）からの先頭バイトに一致させるには、オフセットを 0 に設定します。 • *length*：オフセットからのバイト数を指定します。1 バイトまたは 2 バイトだけがサポートされます。追加のバイト数に一致させるには、複数の UDF の定義が必要です。 複数の UDF を定義できますが、必要な UDF のみを定義することを推奨します。
ステップ 3	**hardware access-list tcam region** {**racl** | **ifacl** | **vacl**} **qualify udf***udf-names* 例： `switch(config)# hardware access-list tcam region racl qualify udf udf-x udf-y`	次のいずれかの TCAM リージョンに UDF を付加します。 • **racl**：レイヤ 3 ポートに適用します。 • **ifacl**：レイヤ 2 ポートに適用します。 • **vacl**：送信元 VLAN に適用します。 UDF は TCAM リージョンに最大 8 個まで付加できます。 （注） UDF 修飾子が追加されると、TCAM リージョンはシングル幅からダブル幅になります。十分な空き領域があることを確認してください。そうでない場合、このコマンドは拒否されます。必要であれば、未使用リージョンでの TCAM スペースを削減してから、このコマンドを再入力することもできます。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「Configuring ACL TCAM Region Sizes」の項を参照してください。 （注） このコマンドの **no** 形式を使用すると、UDF の TCAM リージョンへの付加は解除され、リージョンはシングル幅に戻ります。
ステップ 4	**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。
ステップ 5	**reload** 例： `switch(config)# reload`	デバイスがリロードされます。 （注） UDF 設定は、**copy running-config startup-config** + **reload** を入力した後にのみ有効になります。

	コマンドまたはアクション	目的

ステップ 6	**ip access-list***span-acl*
	
	例：
	`switch(config)# ip access-list span-acl-udf-only`
	`switch(config-acl)#`
		IPv4 アクセス コントロール リスト（ACL）を作成し、IP アクセス リスト コンフィギュレーション モードを開始します。

ステップ 7	次のいずれかのコマンドを入力します。
	• **permit udf***udf-name value mask*
	• **permit ip***source destination***udf***udf-name value mask*
	
	例：
	`switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F`
	
	例：
	`switch(config-acl)# permit ip 10.0.0./24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F`
		ACL を UDF でのみ一致させるか（例 1）、あるいは外部パケット フィールドについて現在のアクセス コントロール エントリ（ACE）と併せて UDF で一致させるように設定します（例 2）。
		1 つの ACL には、UDF の有無にかかわらず、同時に複数の ACE を持たせることができます。各 ACE ごとに異なる UDF フィールドで一致させることもできれば、すべての ACE で同じリストの UDF で一致させることもできます。

ステップ 8	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

SPAN セッションのシャットダウンまたは再開

SPAN 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することができます。1 セッションをシャット ダウンしてハードウェア リソースを解放し、別のセッションをイネーブルにできます。デフォルトでは、SPAN セッションはシャット ステートで作成されます。

SPAN セッションを再開（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再開できます。すでにイネーブルになっていて、動作状況がダウンの SPAN 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。

SPAN セッションのシャット ステートおよびイネーブル ステートは、グローバルまたはモニタ コンフィギュレーション モードのどちらのコマンドでも設定できます。

手順の概要

1. **configure terminal**
2. [**no**] **monitor session** {*session-range* | **all**} **shut**
3. **monitor session***session-number*
4. [**no**] **shut**
5. （任意） **show monitor**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **monitor session** {*session-range*	**all**} **shut** 例： `switch(config)# monitor session 3 shut`	指定の SPAN 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。 コマンドの **no** 形式は、指定された SPAN セッションを再開（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） モニタ 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。

ステップ 3	**monitor session***session-number*
	
	例：
	`switch(config)# monitor session 3`
	`switch(config-monitor)#`
		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 4	[**no**] **shut** 例： `switch(config-monitor)# shut`	SPAN 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。 コマンドの **no** 形式は SPAN 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 5	**show monitor** 例： `switch(config-monitor)# show monitor`	（任意） SPAN セッションのステータスを表示します。

	コマンドまたはアクション	目的
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

SPAN の設定確認

SPAN 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 monitor session {**all** | *session-number* | **range***session-range*} [**brief**]	SPAN セッションの設定を表示します。

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。

手順の概要

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。
2. SPAN セッションを設定します。

手順の詳細

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。

手順の概要

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。
2. SPAN セッションを設定します。

手順の詳細

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 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
  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
ACL TCAM リージョン	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
FEX	『*Cisco Nexus 2000 Series NX-OS Fabric Extender Software Configuration Guide for Cisco Nexus 9000 Series Switches*』

関連資料

第 18 章

ERSPAN の設定

この章は、カプセル化リモート スイッチド ポート アナライザ（ERSPAN）を Cisco NX-OS デバイスの 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ように設定する方法について説明します。

この章の内容は、次のとおりです。

ERSPAN について

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て、ネットワーク内で複数のスイッチのリモート モニタリングを提供します。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 タイプ

Cisco Nexus 9300 シリーズ スイッチは ERSPAN タイプ II およびタイプ III をサポートし、Cisco Nexus 9500 シリーズ スイッチは ERSPAN のみをサポートします。

ERSPAN タイプ III は ERSPAN タイプ II のすべての特徴と機能をサポートし、以下の拡張機能が追加されています。

- ERSPAN タイプ III ヘッダーに、エッジ、集約、およびコア スイッチ間でパケット遅延を計算するために使用できる高精度時間プロトコル（PTP）タイムスタンプ情報（IEEE 1588 で定義）を表示。
- ERSPAN タイプ III ヘッダー フィールドを使用して潜在的なトラフィック ソースを識別。

（注） PTP の詳細については、「PTP の設定, （57 ページ）」を参照してください。

ERSPAN マーカー パケット

ERSPAN タイプ III ヘッダーでは、ハードウェア的に生成される 32 ビットのタイムスタンプが伝送されます。このタイムスタンプ フィールドは、定期的にラップされます。スイッチが 1 ns 粒度に設定されている場合、このフィールドは 4.29 秒ごとにラップされます。このようなラップ時間の存在は、タイムスタンプの真の値の取得を困難にしています。

ERSPAN タイムスタンプの実際の値を復元するには、定期的なマーカー パケットの設定において、オリジナルの UTC タイムスタンプ情報を伝えて ERSPAN タイムスタンプに参照できるようにすることができます。マーカー パケットは 1 秒間隔で送信されます。これにより宛先サイトは、参照パケットのタイムスタンプとパケット オーダーの違いをチェックすることで、32 ビットのラップを取得できます。

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 ソースと呼びます。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(サブインターフェイスではない)
- ポート チャネル
- コントロール プレーン CPU への帯域内インターフェイス。
- VLANs

（注） VLAN が ERSPAN 送信元として指定されている場合は、VLAN 内でサポートされているすべてのインターフェイスが ERSPAN 送信元になります。

（注） VLAN ポートは、入力方向にのみ ERSPAN 送信元となることができます。

- Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）のサテライト ポートおよびホスト インターフェイス ポート チャネル

（注） これらのインターフェイスは、レイヤ 2 アクセス モードおよびレイヤ 2 トランク モードでサポートされます。これらはレイヤ 3 モードではサポートされず、レイヤ 3 サブインターフェイスはサポートされません。

（注） FEX ポートが ERSPAN 送信元としてサポートされるのは、入力方向についてはすべてのトラフィックに対してですが、出力方向については既知のレイヤ 2 ユニキャスト トラフィックだけです。

（注） 1 つの ERSPAN セッションに、上述の送信元を組み合わせて使用できます。

ERSPAN セッション

モニタする送信元を指定する ERSPAN セッションを作成できます。

ローカライズされた ERSPAN セッション

ERSPAN セッションがローカライズされるのは、すべての送信元インターフェイスが同じラインカードにある場合です。

（注） VLAN 送信元の ERSPAN セッションはローカライズされません。

ハイ アベイラビリティ

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を適用します。

ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*』を参照してください。

ERSPAN のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	ERSPAN にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

ERSPAN の前提条件

ERSPAN には、次の前提条件があります。

- 各デバイス上で、まず所定の ERSPAN 設定をサポートするポートを設定する必要があります。詳細については、『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』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約事項は次のとおりです。

- ERSPAN セッションの制限については、『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』を参照してください。
- ライン カードごとの ERSPAN セッションの数は、同じインターフェイスが複数セッションの双方向送信元として設定されている場合は、2 に減少します。
- 1 つのフォワーディング エンジン インスタンスで 4 つの ERSPAN セッションがサポートされます。Cisco Nexus 9300 シリーズ スイッチの場合は、最初の 3 つのセッションに双方向送信元が含まれていると、4 番目のセッションのハードウェア リソースは Rx 送信元専用になります。この制限は、ERSPAN 送信元のフォワーディング エンジン インスタンス マッピングに応じて、Cisco Nexus 9500 シリーズ スイッチにも適用される場合があります。
- ERSPAN 送信元セッションだけがサポートされています。宛先セッションはサポートされません。
- Cisco Nexus 9300 シリーズ スイッチは、Tx ERSPAN を 40G アップリンク ポートでサポートしません。

- Cisco Nexus 9300 シリーズ スイッチ 40G アップリンク インターフェイスの ERSPAN コピーは、Rx 方向にスパンする際に dot1q 情報を取り逃がします。
- Cisco Nexus 9500 シリーズ スイッチの ERSPAN セッションは、スパンされたコピーの ERSPANv2 または ERSPANv3 ヘッダーをサポートしません。Cisco Nexus 9300 シリーズ スイッチは ERSPANv2 または ERSPANv3 ヘッダーをサポートしますが、それは 40G アップリンク SPAN を宛先とするセッションに限られます。
- VLAN が、ERSPAN 送信元またはフィルタとして使用される場合、属することができるセッションは 1 つだけです。
- 不明ユニキャストでフラッディングされたパケットのルーティング後のフローは ERSPAN セッションに置かれますが、これはフローが転送されるポートをモニタしないよう ERSPAN セッションが設定されている場合であっても同様です。この制限は、ネットワーク フォワーディング エンジン（NFE）、NFE2 対応 EOR スイッチ、TX ポートの送信元を持つ ERSPAN セッションに適用されます。
- VLAN 送信元は Rx 方向でのみスパンします。
- スーパーバイザの生成する Stream Of Bytes Module Header（SOBMH）パケットには、インターフェイスから出力される情報がすべて含まれており、SPAN および ERSPAN を含めた、ハードウェア内部でのフォワーディング ルックアップをすべてバイパス可能です。パケットのブリッジ プロトコル データ ユニット（BPDU）クラスおよびレイヤ 3 インターフェイスでの CPU 生成のフレームは、SOBMH を使用して送信されます。
- フィルタ アクセス グループの統計情報はサポートされません。
- ERSPAN セッションでの access-group フィルタは、vlan-accessmap として設定する必要があります。
- セッション フィルタリング機能（VLAN または ACL フィルタ）は、Rx 送信元でのみサポートされます。
- すべての ERSPAN レプリケーションはハードウェアで行われます。スーパーバイザ CPU は関与しません。
- スーパーバイザによって生成されたコントロール プレーン パケットは、ERSPAN カプセル化または ERSPAN アクセス コントロール リスト（ACL）によるフィルタ処理をすることはできません。
- ERSPAN および ERSPAN ACL セッションが宛先ルータで同時に終了されるのは、Cisco Nexus 9300 シリーズ スイッチのアップリンク ポート経由で ERSPAN 宛先 IP アドレスが解決された場合のみです。
- VLAN および ACL フィルタは、FEX ポートではサポートされません。
- 双方向 ERSPAN セッションで使用される送信元が同じ FEX から出されている場合、ハードウェア リソースは 2 つの ERSPAN セッションに限定されます。
- ERSPAN は、管理ポートではサポートされません。
- ERSPAN は、レイヤ 3 ポート チャネルのサブインターフェイスの宛先をサポートしません。

- ERSPAN は、N9K-X9408PC-CFP2 ライン カード ポートの宛先をサポートしません。
- VLAN ERSPAN がモニタするのは、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。
- VLAN は、入力方向のみの ERSPAN 送信元としてサポートされます。
- FEX ポートが ERSPAN 送信元としてサポートされるのは、入力方向についてはすべてのトラフィックに対してですが、出力方向については既知のレイヤ 2 ユニキャスト トラフィックだけです。
- vPC で ERSPAN をイネーブルにし、ERSPAN パケットが vPC を介して宛先にルーティングされる必要がある場合は、vPC ピア リンクを通過するパケットはキャプチャできません。
- ERSPAN タイプ III セッションのタイムスタンプの粒度は、CLI からは設定できません。この値は 100 ピコ秒であり、PTP を基に実行されます。
- ERSPAN はデフォルトおよび非デフォルトの VRF で動作しますが、ERSPAN マーカー パケットはデフォルト VRF でのみ動作します。
- プライオリティ フロー制御（PFC）ERSPAN には、次の制約事項があります。
  - 物理または port-channel インターフェイスの Rx 方向でのみサポートされています。VLAN インターフェイスの Rx 方向、または Tx 方向ではサポートされていません。
  - Cisco Nexus 9300 シリーズ アップリンク ポートではサポートされていません。
  - また、フィルタとは共存できません。
- Cisco NX-OS リリース 7.0(3)I4(1) 以降では、同じ送信元が複数のセッションに属することができます。
- Cisco NX-OS リリース 7.0(3)I4(1) 以降では、Cisco Nexus 9300 および 9500 シリーズ スイッチで同じ送信元の複数の ACL フィルタをサポートします。
- 次の注意事項と制約事項は、Cisco Nexus 9200 シリーズ スイッチに適用されます。
  - ERSPAN ACL の **set-erspan-gre-proto** および **set-erspan-dscp** アクションは、Cisco NX-OS リリース 7.0(3)I4(1) 以降でサポートされます。
  - UDF ベース ERSPAN は、Cisco NX-OS リリース 7.0(3)I4(1) 以降でサポートされます。
  - ERSPAN によるフォワード ドロップは、Cisco NX-OS リリース 7.0(3)I4(1) 以降でサポートされます。
  - Tx ERSPAN は、マルチキャスト、未知のマルチキャスト、およびブロードキャスト トラフィックではサポートされません。
  - ERSPAN 送信元と宛先が同じスライスにあり、スライスに転送インターフェイスがない場合、Rx ERSPAN はマルチキャストでサポートされません。これがサポートされるのは、スライスに転送インターフェイスがあるか、あるいは ERSPAN の送信元と宛先が異なるスライスにある場合です。

- 同じスライスにある複数の出力ポートで出力ERSPANトラフィックの輻輳が生じると、これらの出力ポートではライン レートを得られません。
- ACL フィルタを使用した、親インターフェイスでのサブインターフェイス トラフィックのスパンは、サポートされません。
- CPU ERSPAN 送信元は、Rx 方向（CPU からの ERSPAN パケット）でのみ追加できます。
- 複数の ACL フィルタは、同じ送信元ではサポートされません。

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 *15*：デフォルトの *ERSPAN* パラメータ

パラメータ（Parameters）	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 マーカー パケット間隔	100 マイクロ秒
ERSPAN タイプ III セッションのタイムスタンプ粒度	100 ピコ秒

ERSPAN の設定

（注） この機能の Cisco NX-OS コマンドは、Cisco IOS のコマンドとは異なる場合があるので注意してください。

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。

（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。

手順の概要

1. **configure terminal**
2. **monitor erspan origin ip-address***ip-address***global**
3. **no monitor session** {*session-number* | **all**}
4. **monitor session** {*session-number* | **all**} **type erspan-source** [**shut**]
5. （任意） [**no**] **header-type 3**
6. **description***description*
7. **source** {**interface***type* [**rx** | **tx** | **both**] [**allow-pfc**] | **vlan** {*number* | *range*} [**rx**]} [**forward-drops rx** [**priority-low**]]
8. （任意） ステップ 7 を繰り返して、すべての ERSPAN 送信元を設定します。
9. （任意） **filter vlan** {*number* | *range*}
10. （任意） ステップ 9 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。
11. （任意） **filter access-group***acl-filter*
12. **destination ip***ip-address*
13. **erspan-id***erspan-id*
14. **vrf***vrf-name*
15. （任意） **ip ttl***ttl-number*
16. （任意） **ip dscp***dscp-number*
17. （任意） [**no**] **marker-packet***microseconds*
18. **no shut**
19. **exit**
20. （任意） **show monitor session** {**all** | *session-number* | **range***session-range*} [**brief**]
21. （任意） **show running-config monitor**
22. （任意） **show startup-config monitor**
23. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**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。

	コマンドまたはアクション	目的
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** [**shut**]
	
	例：
	`switch(config)# monitor session 3 type erspan-source`
	`switch(config-erspan-src)#`
		ERSPAN タイプ II 送信元セッションを設定します。デフォルトでは、セッションは双方向です。オプションの shut キーワードは、選択したセッションに対して shut ステートを指定します。

ステップ 5	[**no**] **header-type 3** 例： `switch(config-erspan-src)# header-type 3`	（任意） ERSPAN 送信元セッションをタイプ II からタイプ III に変更します。 （注） ERSPAN 送信元セッションをタイプ III からタイプ II に変更するには、このコマンドの **no** 形式を使用します。
ステップ 6	**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 7	**source** {**interface***type* [**rx** | **tx** | **both**] [**allow-pfc**] | **vlan** {*number* | *range*} [**rx**]} [**forward-drops rx** [**priority-low**]]
	
	例：
	`switch(config-erspan-src)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	
	例：
	`switch(config-erspan-src)# source interface ethernet 2/1 rx allow-pfc`
	
	例：
	`switch(config-erspan-src)# source interface port-channel 2`
	
	例：
	`switch(config-erspan-src)# source interface sup-eth 0 both`
	
	例：
	`switch(config-erspan-src)# source vlan 3, 6-8 rx`
		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。一定範囲のイーサネット ポート、ポート チャネル、インバンド インターフェイス、一定範囲の VLAN、または Cisco Nexus 2000 シリーズ ファブリック エクステンダ（FEX）上のサテライト ポートまたはホスト インターフェイス ポート チャネルを入力できます。
		
		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。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。
		
		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。このオプションでは、PFC フレームのスパニングを Rx 方向のみサポートします。

	コマンドまたはアクション	目的

	例：
	`switch(config-erspan-src)# source interface ethernet 101/1/1-3`
	
	例：
	`switch(config-erspan-src)# source forward-drops rx`
		allow-pfc に **tx** または **both** オプションを設定すると、エラー メッセージが表示されます。
		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、VLAN 送信元で設定されているその他の ERSPAN セッションよりも優先度が低くなります。
		単一方向のセッションには、送信元の方向はセッションで指定された方向に一致する必要があります。
		（注） 送信元 VLAN は、入力方向に限りサポートされます。送信元 FEX ポートがサポートされるのは、入力方向についてはすべてのトラフィックに対してですが、出力方向については既知のレイヤ 2 ユニキャスト トラフィックだけです。

ステップ 8	ステップ 7 を繰り返して、すべての ERSPAN 送信元を設定します。	（任意） —
ステップ 9	**filter vlan** {*number* | *range*} 例： `switch(config-erspan-src)# filter vlan 3-5, 7`	（任意） 設定された送信元から選択する VLAN を設定します。VLAN 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。VLAN の範囲については、『*Cisco Nexus 9000 Series NX-OS Layer 2 Switching Configuration Guide*』を参照してください。 （注） ERSPAN 送信元として設定された FEX ポートは VLAN フィルタをサポートしません。
ステップ 10	ステップ 9 を繰り返して、すべての送信元 VLAN のフィルタリングを設定します。	（任意） —
ステップ 11	**filter access-group***acl-filter* 例： `switch(config-erspan-src)# filter access-group ACL1`	（任意） ACL を ERSPAN セッションにアソシエートします（標準の ACL 設定プロセスを使用して ACL を作成できます。詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください）。

	コマンドまたはアクション	目的
		（注） 入力パイプラインでフォワード パケット ドロップをスパンするよう ERSPAN を設定した場合、このコマンドは使用できません。
ステップ 12	**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 13	**erspan-id***erspan-id* 例： `switch(config-erspan-src)# erspan-id 5`	ERSPAN 送信元セッションの ERSPAN ID を設定します。ERSPAN の範囲は 1 ～ 1023 です。
ステップ 14	**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）インスタンスを設定します。VRF 名には最大 32 文字の英数字を使用できます。大文字と小文字は区別されます。
ステップ 15	**ip ttl***ttl-number* 例： `switch(config-erspan-src)# ip ttl 25`	（任意） 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 16	**ip dscp***dscp-number* 例： `switch(config-erspan-src)# ip dscp 42`	（任意） 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 17	[**no**] **marker-packet***microseconds* 例： `switch(config-erspan-src)# marker-packet 100`	（任意） ERSPAN タイムスタンプの実際の値を復元するため、セッションでの ERSPAN マーカー パケットをイネーブルにします。この間隔の範囲には、100 ～ 1000 マイクロ秒を設定できます。このコマンドの **no** 形式を使用すると、セッションでのマーカー パケットをディセーブルにします。
ステップ 18	**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。

ステップ 19	**exit**
	
	例：
	`switch(config-erspan-src)# exit`
	`switch(config)#`
		モニタ コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 20	**show monitor session** {**all** | *session-number* | **range***session-range*} [**brief**] 例： `switch(config)# show monitor session 3`	（任意） ERSPAN セッション設定を表示します。
ステップ 21	**show running-config monitor** 例： `switch(config)# show running-config monitor`	（任意） 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 22	**show startup-config monitor** 例： `switch(config)# show startup-config monitor`	（任意） 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 23	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。1 セッションをシャット ダウンしてハードウェア リソースを解放し、別のセッションをイネーブルにできます。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。

手順の概要

1. **configure terminal**
2. **monitor session** {*session-range* | **all**} **shut**
3. **no monitor session** {*session-range* | **all**} **shut**
4. **monitor session***session-number***type erspan-source**
5. **shut**
6. **no shut**
7. **exit**
8. （任意） **show monitor session all**
9. （任意） **show running-config monitor**
10. （任意） **show startup-config monitor**
11. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**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。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 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。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。 モニタ 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。

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。

	コマンドまたはアクション	目的
ステップ 5	**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 6	**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 7	**exit**
	
	例：
	`switch(config-erspan-src)# exit`
	`switch(config)#`
		モニタ コンフィギュレーション モードを終了します。

ステップ 8	**show monitor session all** 例： `switch(config)# show monitor session all`	（任意） ERSPAN セッションのステータスを表示します。
ステップ 9	**show running-config monitor** 例： `switch(config)# show running-config monitor`	（任意） 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	**show startup-config monitor** 例： `switch(config)# show startup-config monitor`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 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。

はじめる前に

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。

手順の概要

1. **configure terminal**
2. **ip access-list***acl-name*
3. [*sequence-number*] {**permit** | **deny**} *protocolsourcedestination* [**set-erspan-dscp***dscp-value*] [**set-erspan-gre-proto***protocol-value*]
4. （任意） **show ip access-lists***name*
5. （任意） **show monitor session** {**all** | *session-number* | **range***session-range*} [**brief**]
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**} *protocolsourcedestination* [**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 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 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 セッションを使用します。ERSPAN ACL ごとに、これらのアク

	コマンドまたはアクション	目的
		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	**show ip access-lists***name* 例： `switch(config-acl)# show ip access-lists erpsan-acl`	（任意） ERSPAN ACL の設定を表示します。
ステップ 5	**show monitor session** {**all** | *session-number* | **range***session-range*} [**brief**] 例： `switch(config-acl)# show monitor session 1`	（任意） ERSPAN セッション設定を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch(config-acl)# copy running-config startup-config`	（任意） 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。

UDF ベース ERSPAN の設定

デバイスの設定では、外部または内部パケット フィールド（ヘッダーまたはペイロード）のユーザ定義フィールド（UDF）と照合させ、一致したパケットを ERSPAN 宛先に送信させるようにできます。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。

はじめる前に

UDF ベース ERSPAN をイネーブルにするのに十分な空き領域を確保するために、**hardware access-list tcam region** コマンドを使用して適切な TCAM リージョン（racl、ifacl、または vacl）が設定され

ていることを確認します。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「Configuring ACL TCAM Region Sizes」の項を参照してください。

手順の概要

1. **configure terminal**
2. **udf***udf-name offset-base offset length*
3. **hardware access-list tcam region** {**racl** | **ifacl** | **vacl**} **qualify udf***udf-names*
4. **copy running-config startup-config**
5. **reload**
6. **ip access-list***erspan-acl*
7. 次のいずれかのコマンドを入力します。
   - **permit udf***udf-name value mask*
   - **permit ip***source destination***udf***udf-name value mask*
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**udf***udf-name offset-base offset length*
	
	例：
	`switch(config)# udf udf-x packet-start 12 1`
	`switch(config)# udf udf-y header outer l3 20 2`
		次のように UDF を定義します。
		• *udf-name*：UDF の名前を指定します。名前には最大 16 文字の英数字を入力できます。
		• *offset-base*：UDF オフセット ベースを次のように指定しますが、ここでの **header** はオフセットを考慮したパケット ヘッダーです。**packet-start** | **header** {**outer** | **inner** {**l3** | **l4**}}。
		• *offset*：オフセット ベースからオフセットさせるバイト数を指定します。オフセット ベース（レイヤ 3/レイヤ 4 ヘッダー）からの先頭バイトに一致させるには、オフセットを 0 に設定します。
		• *length*：オフセットからのバイト数を指定します。1 バイトまたは 2 バイトだけがサポートされます。追加のバイト数に一致させるには、複数の UDF の定義が必要です。
		
		複数の UDF を定義できますが、必要な UDF のみを定義することを推奨します。

	コマンドまたはアクション	目的
ステップ 3	**hardware access-list tcam region** {**racl** | **ifacl** | **vacl**} **qualify udf***udf-names* 例： `switch(config)# hardware access-list tcam region racl qualify udf udf-x udf-y`	次のいずれかの TCAM リージョンに UDF を付加します。 • **racl**：レイヤ 3 ポートに適用します。 • **ifacl**：レイヤ 2 ポートに適用します。 • **vacl**：送信元 VLAN に適用します。 UDF は TCAM リージョンに最大 8 個まで付加できます。 （注） UDF 修飾子が追加されると、TCAM リージョンはシングル幅からダブル幅になります。十分な空き領域があることを確認してください。そうでない場合、このコマンドは拒否されます。必要であれば、未使用リージョンでの TCAM スペースを削減してから、このコマンドを再入力することもできます。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「Configuring ACL TCAM Region Sizes」の項を参照してください。 （注） このコマンドの **no** 形式を使用すると、UDF の TCAM リージョンへの付加は解除され、リージョンはシングル幅に戻ります。
ステップ 4	**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。
ステップ 5	**reload** 例： `switch(config)# reload`	デバイスがリロードされます。 （注） UDF 設定は、**copy running-config startup-config** + **reload** を入力した後にのみ有効になります。

ステップ 6	**ip access-list***erspan-acl*
	
	例：
	`switch(config)# ip access-list erspan-acl-udf-only`
	`switch(config-acl)#`
		IPv4 アクセス コントロール リスト（ACL）を作成し、IP アクセス リスト コンフィギュレーション モードを開始します。

ステップ 7	次のいずれかのコマンドを入力します。 • **permit udf***udf-name value mask* • **permit ip***source destination* **udf***udf-name value mask*	ACL を UDF でのみ一致させるか（例 1）、あるいは外部パケット フィールドについて現在のアクセス コントロール エントリ（ACE）と併せて UDF で一致させるように設定します（例 2）。 1 つの ACL には、UDF の有無にかかわらず、同時に複数の ACE を持たせることができます。各 ACE ごとに異なる UDF フィールドで一致させることもできれば、すべての ACE で同じリストの UDF で一致させることもできます。

	コマンドまたはアクション	目的

	例：
	`switch(config-acl)# permit udf udf-x 0x40 0xF0 udf-y 0x1001 0xF00F`
	
	例：
	`switch(config-acl)# permit ip 10.0.0./24 any udf udf-x 0x02 0x0F udf-y 0x1001 0xF00F`	

ステップ 8	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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 ip access-lists*name*	ERSPAN ACL の設定を表示します。

コマンド	目的
show monitor session {**all** | *session-number* | **range***session-range*} [**brief**]	ERSPAN セッション設定を表示します。 出力には、ERSPANパケットの送信に使用される出力インターフェイスが含まれます。出力内容は、使用された出力インターフェイスのタイプによって異なります。 • 物理レイヤ3インターフェイス：インターフェイス名を表示します。 • SVIインターフェイス：ルートを学習したメンバーインターフェイスを表示します。 • レイヤ 3 ポート チャネル：port-channel インターフェイス名を表示します。 • レイヤ 3 サブインターフェイス：親インターフェイス名を表示します。 • ECMP パス：いずれかの等コスト マルチパス（ECMP）メンバーインターフェイスの名前を表示します。ルートが ECMP の場合でも、表示されているインターフェイスだけがトラフィックのミラーリングに使用されます。 • インターフェイスの PFC：インターフェイスのプライオリティ フロー制御（PFC）のステータスが表示されます。
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

単一方向 ERSPAN セッションの設定例

次に、単一方向 ERSPAN セッションを設定する例を示します。

```
switch# configure terminal
switch(config)# interface ethernet 14/30
```

```
switch(config-if)# no shut
switch(config-if)# exit
switch(config)# no monitor session 3
switch(config)# monitor session 3 rx
switch(config-erspan-src)# source interface ethernet 2/1-3 rx
switch(config-erspan-src)# erspan-id 1
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
```

マーカー パケットの設定例

次に、2 秒の間隔で ERSPAN マーカー パケットをイネーブルにする例を示します。

```
switch# configure terminal
switch(config)# monitor erspan origin ip-address 172.28.15.250 global
switch(config)# monitor session 1 type erspan-source
switch(config-erspan-src)# header-type 3
switch(config-erspan-src)# erspan-id 1
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)# source interface ethernet 1/15 both
switch(config-erspan-src)# marker-packet 100
switch(config-erspan-src)# no shut
switch(config-erspan-src)# show monitor session 1
session 1
---------------
type              : erspan-source
state             : up
granularity       : nanoseconds
erspan-id         : 1
vrf-name          : default
destination-ip    : 9.1.1.2
ip-ttl            : 16
ip-dscp           : 5
header-type       : 3
```

```
origin-ip          : 172.28.15.250 (global)
source intf        :
    rx             : Eth1/15
    tx             : Eth1/15
    both           : Eth1/15
source VLANs       :
    rx             :
marker-packet      : enabled
packet interval    : 100
packet sent        : 25
packet failed      : 0
egress-intf        :
```

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
ACL TCAM リージョン	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
FEX	『*Cisco Nexus 2000 Series NX-OS Fabric Extender Software Configuration Guide for Cisco Nexus 9000 Series Switches*』
高精度時間プロトコル（PTP）	『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』

第 19 章

LLDP の設定

この章では、ローカル ネットワーク上の他のデバイスを検出するために、Link Layer Discovery Protocol（LLDP）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。

LLDP について

Cisco Discovery Protocol（CDP）は、ネットワークに接続された他のシスコ デバイスを自動的に検出し学習することをネットワーク管理アプリケーションによって可能にするデバイス検出プロトコルです。

他社製デバイスのディスカバリを許可するために、スイッチは、IEEE 802.1ab 規格で定義されているベンダー ニュートラルなデバイス ディスカバリ プロトコルであるリンク層検出プロトコル（LLDP）もサポートしています。LLDP を使用すると、ネットワーク デバイスはネットワーク デバイスに関する情報を、ネットワーク上の他のデバイスにアドバタイズできます。このプロトコルはデータリンク層で動作するため、異なるネットワーク層プロトコルが稼働する 2 つのシステムで互いの情報を学習できます。

LLDP は、デバイスおよびそのインターフェイスの機能と現在のステータスに関する情報を送信する単一方向のプロトコルです。LLDP デバイスはこのプロトコルを使用して、他の LLDP デバイスからだけ情報を要求します。

LLDP は一連の属性をサポートしており、これを使用して他のデバイスを検出します。これらの属性には、タイプ、長さ、および値（TLV）の説明が含まれています。LLDP デバイスは TLV を使用して、ネットワーク上の他のデバイスと情報を送受信できます。設定情報、デバイスの機能、デバイス ID などの詳細情報は、このプロトコルを使用してアドバタイズできます。

LLDP は、デフォルトで次の TLV を通知します。

- DCBXP
- 管理アドレス
- ポート記述
- ポート VLAN
- システム機能
- システム記述
- システム名

DCBXP について

Data Center Bridging Exchange Protocol（DCBXP）は、LLDP を拡張したプロトコルです。これは、ピア間のノード パラメータのアナウンス、交換、およびネゴシエートに使用されます。DCBXP パラメータは、特定の DCBXP TLV にパッケージ化されます。この TLV は、受信した LLDP パケットに確認応答を提供するように設計されています。このように、DCBXP は負荷の軽い確認応答メカニズムを LLDP の上位に追加し、このためリンク レベル プロトコルからの要求応答セマンティックを必要とするすべてのアプリケーションが DCBXP を利用できるようになります。

DCBXP を使用してパラメータとピア ノードの交換およびネゴシエーションが必要な他のアプリケーションは次のとおりです。

- 優先度ベース フロー制御（PFC）：PFC は、イーサネットの既存のポーズ メカニズムを拡張するものです。これは、ユーザ プライオリティまたはサービス クラスに基づいてポーズをイネーブルにします。PFC を使用して 8 つの仮想リンクに分割された物理リンクは、他の仮想リンクのトラフィックに影響を与えることなく、単一の仮想リンクでポーズを使用できる機能を提供します。ユーザごとのプライオリティ単位でポーズをイネーブルにすることで、IP トラフィック用のパケット ドロップの輻輳管理を維持しながら、ドロップの無いサービスが必要なトラフィックに対し管理者がロスレス リンクを作成できます。
- イネーブル化転送選択（ETS）：ETS は、仮想リンクの最適帯域幅管理を可能にします。また、ETS は、優先度のグループ化とも呼ばれます。PFC の同じ優先度クラス内の処理の区別をイネーブルにします。ETS は帯域割り当て、低遅延、またはベストエフォートに基づいた順位付け処理を提供し、結果としてグループ単位のトラフィック クラスの割り当てを提供します。たとえば、同一クラス内では、トラフィックのイーサネット クラスが高いプライオリティの指定とベストエフォートがある可能性があります。ETS によって、同じ優先度クラスの中でトラフィックを区別でき、優先度グループを作成できます。

- アプリケーション プライオリティ設定 TLV：特定のプロトコルで使用される VLAN に関する情報を伝送します。

（注） Quality of Service（QoS）機能の詳細については、『*Cisco Nexus 9000 Series NX-OS Quality of Service Configuration Guide*』を参照してください。

DCBXP はデフォルトでイネーブルであり、提供された LLDP はイネーブルです。LLDP がイネーブルである場合、**[no] lldp tlv-select dcbxp** コマンドを使用して DCBXP をイネーブルまたはディセーブルにすることができます。LLDP の送信または受信がディセーブルになっているポートでは、DCBXP はディセーブルです。

ハイ アベイラビリティ

LLDP 機能はステートレス リスタート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を適用します。

ハイ アベイラビリティの詳細については、『*Cisco Nexus 9000 Series NX-OS High Availability and Redundancy Guide*』を参照してください。

仮想化のサポート

サポートされる LLDP のインスタンスは 1 個です。

LLDP のライセンス要件

製品	ライセンス要件
Cisco NX-OS	LLDP にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

LLDP に関する注意事項および制約事項

LLDP 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。

- インターフェイス上で LLDP をイネーブルまたはディセーブルにするには、事前にデバイス上で LLDP をイネーブルにしておく必要があります。
- LLDP は物理インターフェイスだけでサポートされています。
- LLDP は 1 つのポートにつき 1 つのデバイスを検出できます。

- Converged Network Adapter（CNA）を使用していない場合、LLDP は Linux サーバを検出できます。LLDP は他のタイプのサーバを検出できません。
- Cisco NX-OS リリース 7.0(3)I3(1) より以前では、DCBXP は LLDP ではサポートされません。
- DCBXP は次のプラットフォームでサポートされています。
  - Cisco Nexus 9332PQ、9372PX、9372PX-E、および 9396PX スイッチ
  - Cisco Nexus 9504 および 9508 スイッチで、X9432PQ、X9464PX、X9536PQ、X9564PX、および X9636PQ ライン カードを搭載したもの

（注） DCBXP は前面パネルの固定ポートでのみサポートされます。FEX ポートはサポートされません。

- DCBXP の非互換性のメッセージは、物理ループバック接続がデバイスにある場合に network QoS ポリシーを変更するときに表示されることがあります。非互換性があるのは短時間で、すぐに解消されます。

LLDP のデフォルト設定

この表は、LLDP のデフォルト設定を示します。

パラメータ（Parameters）	デフォルト
グローバル LLDP	ディセーブル
インターフェイス上の LLDP	イネーブル（LLDP がグローバルにイネーブルになった後）
LLDP 保持時間（ディセーブルになる前）	120 秒
LLDP 再初期化遅延	2 秒
LLDP タイマー（パケット更新頻度）	30 秒
LLDP TLV	イネーブル
LLDP 受信	イネーブル（LLDP がグローバルにイネーブルになった後）
LLDP 転送	イネーブル（LLDP がグローバルにイネーブルになった後）
DCBXP	イネーブル（提供された LLDP がイネーブルになります）

LLDP の設定

（注） この機能の Cisco NX-OS コマンドは、類似した機能の Cisco IOS コマンドと異なる場合があります。

LLDP のグローバルなイネーブルまたはディセーブル

デバイスで LLDP 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。デバイスで LLDP パケットの送信および受信を可能にするには、LLDP をグローバルにイネーブルにする必要があります。

手順の概要

1. **configure terminal**
2. [**no**] **feature lldp**
3. （任意） **show running-config lldp**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **feature lldp** 例： `switch(config)# feature lldp`	デバイス上で LLDP をイネーブルまたはディセーブルにします。LLDP はデフォルトでディセーブルです。
ステップ 3	**show running-config lldp** 例： `switch(config)# show running-config lldp`	（任意） LLDP のグローバル コンフィギュレーションを表示します。LLDP がイネーブルの場合、「feature lldp.」が表示されます。LLDP がディセーブルの場合、「Invalid command」エラーが表示されます。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

インターフェイス上での LLDP のイネーブルまたはディセーブル

LLDPをグローバルにイネーブルにすると、LLDPは、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す。ただし、LLDP パケットの送信だけ、または受信だけを実行するために、個々のインターフェイスでの LLDP のイネーブルまたはディセーブル、あるいはインターフェイスの選択的な設定を実行できます。

はじめる前に

デバイスで LLDP をグローバルにイネーブルにしていることを確認します。

手順の概要

1. **configure terminal**
2. **interface***interface slot*/*port*
3. [**no**] **lldp transmit**
4. [**no**] **lldp receive**
5. （任意） **show lldp interface***interface slot*/*port*
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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***interface slot*/*port*
	
	例：
	`switch(config)# interface ethernet 7/1`
	`switch(config-if)#`
		LLDP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 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**] **lldp transmit** 例： `switch(config-if)# lldp transmit`	インターフェイス上で LLDP パケットの送信をイネーブルまたはディセーブルにします。LLDPをグローバルにイネーブルにすると、LLDPは、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す。

	コマンドまたはアクション	目的
ステップ 4	[**no**] **lldp receive** 例： `switch(config-if)# lldp receive`	インターフェイス上で LLDP パケットの受信をイネーブルまたはディセーブルにします。LLDP をグローバルにイネーブルにすると、LLDP は、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す。
ステップ 5	**show lldp interface***interface slot*/*port* 例： `switch(config-if)# show lldp interface ethernet 7/1`	（任意） インターフェイス上の LLDP 設定を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

LLDP オプション パラメータの設定

LLDP の更新頻度、受信デバイスが情報を破棄するまでに保持している時間、および初期化の遅延時間を設定できます。TLV を選択して、LLDP パケットに含まれるようにすることもできます。

手順の概要

1. **configure terminal**
2. （任意） [**no**] **lldp holdtime***seconds*
3. （任意） [**no**] **lldp reinit***seconds*
4. （任意） [**no**] **lldp timer***seconds*
5. （任意） **show lldp timers**
6. （任意） [**no**] **lldp tlv-select***tlv*
7. （任意） **show lldp tlv-select**
8. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **lldp holdtime***seconds* 例： `switch(config)# lldp holdtime 200`	（任意） ユーザのデバイスから送信された情報が、受信側デバイスで廃棄されるまでに保持される時間を秒単位で指定します。 値の範囲は 10 ～ 255 秒で、デフォルト値は 120 秒です。
ステップ 3	[**no**] **lldp reinit***seconds* 例： `switch(config)# lldp reinit 5`	（任意） 任意のインターフェイス上でLLDPを初期化する際の遅延時間を秒単位で指定します。 指定できる範囲は 1 ～ 10 秒です。デフォルトは 2 秒です。
ステップ 4	[**no**] **lldp timer***seconds* 例： `switch(config)# lldp timer 50`	（任意） LLDP アップデートの送信頻度を秒単位で設定します。 値の範囲は 5 ～ 254 秒で、デフォルト値は 30 秒です。
ステップ 5	**show lldp timers** 例： `switch(config)# show lldp timers`	（任意） LLDP の保持時間、遅延時間、更新頻度の設定を表示します。
ステップ 6	[**no**] **lldp tlv-select***tlv* 例： `switch(config)# lldp tlv-select system-name`	（任意） LLDP パケットで送受信する TLV を指定します。使用できる TLV は、dcbxp、management-address、port-description、port-vlan、system-capabilities、system-description、および system-name です。使用できるすべての TLV はデフォルトでイネーブルになっています。
ステップ 7	**show lldp tlv-select** 例： `switch(config)# show lldp tlv-select`	（任意） LLDP TVL コンフィギュレーションを表示します。
ステップ 8	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

LLDP コンフィギュレーションの確認

LLDP 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 running-config lldp	LLDP のグローバル コンフィギュレーションを表示します。
show lldp all	LLDP DCBXP を表示し、すべてのインターフェイスの設定を送信および受信します。
show lldp interface*interfaceslot*/*port*	LLDP のインターフェイス コンフィギュレーションを表示します。
show lldp timers	LLDP の保持時間、遅延時間、更新頻度の設定を表示します。
show lldp tlv-select	LLDP TLV コンフィギュレーションを表示します。
show lldp dcbx interface*interfaceslot*/*port*	ローカルな DCBX 制御ステータスを表示します。
show lldp neighbors {**detail** | **interface***interfaceslot*/*port*}	LLDP ネイバーのデバイス ステータスを表示します。
show lldp traffic	LLDP カウンタ（デバイスによって送信および受信された LLDP パケットの数、破棄されたパケットの数、未確認 TLV の数など）を表示します。
show lldp traffic interface*interfaceslot*/*port*	インターフェイス上で送信および受信された LLDP パケットの数を表示します。

LLDP の統計を消去するには、**clear lldp counters** コマンドを使用します。

LLDP のコンフィギュレーション例

次に、1 つのデバイス上での LLDP のイネーブル化、一部のインターフェイス上での LLDP のディセーブル化、オプション パラメータ（保持時間、遅延時間、更新頻度など）の設定、およびいくつかの LLDP TLV のディセーブル化の例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# feature lldp
switch(config)# interface ethernet 7/9
switch(config-if)# no lldp transmit
switch(config-if)# no lldp receive
switch(config-if)# exit
switch(config)# interface ethernet 7/10
switch(config-if)# no lldp transmit
switch(config-if)# no lldp receive
switch(config-if)# exit
switch(config)# lldp holdtime 200
switch(config)# lldp reinit 5
switch(config)# lldp timer 50
```

```
switch(config)# no lldp tlv-select port-vlan
switch(config)# no lldp tlv-select system-name
```

第 20 章

sFlow の設定

この章では、Cisco NX-OS デバイスで sFlow 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。

sFlow について

サンプリングされた Flow（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できます。sFlow では、トラフィックをモニタ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、サンプル データを中央のデータ コレクタに転送します。

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。

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。

sFlow のライセンス要件

製品	ライセンス要件
Cisco NX-OS	sFlow にライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

sFlow の前提条件

sFlow には、次の前提条件があります。

- Cisco Nexus 9332PQ、9372PX、9372TX、93120TX スイッチ、および N9K-M6PQ または N9K-M12PQ 汎用拡張モジュール（GEM）搭載の Cisco Nexus 9396PX、9396TX、93128TX スイッチについては、sFlow データ ソースとして設定するすべてのアップリンク ポート用の sFlow および SPAN ACL TCAM リージョン サイズを設定する必要があります。そのためには、**hardware access-list tcam region sflow** および **hardware access-list tcam region span** コマンドを使用します。詳細については、『Configuring ACL TCAM Region Sizes』を参照してください。

（注） デフォルトでは、sflow リージョン サイズはゼロであり、SPAN リージョン サイズはゼロではありません。ポートを sFlow のデータ ソースとして設定するには、sflow リージョンを 256 に設定し、スパン リージョンに十分なエントリを割り当てる必要があります。

sFlow の注意事項および制約事項

sFlow には、次の注意事項と制限事項があります。

- 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。
- スイッチは 1 つの sFlow コレクタだけをサポートします。
- sFlow とネットワーク アドレス変換（NAT）は、同じポートではサポートされません。
- マルチキャスト パケットの入力 sFlow サンプルについて、出力ポートのレポートは、出力ポート数が確定した複数ポート（Cisco Nexus 9300 シリーズ スイッチの場合）、あるいはポート数が不明な複数ポート（Cisco Nexus 9500 シリーズ スイッチの場合）として報告されます。
- IPv6 sFlow はサポートされません。
- サブインターフェイスは、sFlow でサポートされません。
- sFlow は、Cisco Nexus 9300 と 9500 シリーズ スイッチおよび Cisco Nexus 3164Q、31128PQ、3232C、および 3264Q スイッチでサポートされます。Cisco Nexus 9200 シリーズ スイッチ上ではサポートされません。

sFlow のデフォルト設定

次の表に、sFlow パラメータのデフォルト設定を示します。

表 *16*：デフォルトの *sFlow* パラメータ

パラメータ（Parameters）	デフォルト
sFlow のサンプリング レート	4096
sFlow のサンプリング サイズ	128

パラメータ（Parameters）	デフォルト
sFlow カウンタのポーリング間隔	20
sFlow の最大データグラム サイズ	1400
sFlow のコレクタ IP アドレス	0.0.0.0
sFlow のコレクタ ポート	6343
sFlow エージェントの IP アドレス	0.0.0.0

sFlow の設定

sFlow 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。

手順の概要

1. **configure terminal**
2. [**no**] **feature sflow**
3. （任意） **show feature**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **feature sflow** 例： `switch(config)# feature sflow`	sFlow をイネーブルまたはディセーブルにします。

	コマンドまたはアクション	目的
ステップ 3	**show feature** 例： `switch(config)# show feature`	（任意） 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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

サンプリング レートの設定

sFlow のサンプリング レートを設定できます。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow sampling-rate***sampling-rate*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow sampling-rate***sampling-rate* 例： `switch(config)# sflow sampling-rate 50000`	パケットの sFlow のサンプリング レートを設定します。 *sampling-rate* には 4096 ～ 1000000000 の整数を指定できます。

	コマンドまたはアクション	目的
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow コンフィギュレーションを表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow max-sampled-size***sampling-size*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow max-sampled-size***sampling-size* 例： `switch(config)# sflow max-sampled-size 200`	sFlow の最大サンプリング サイズを設定します。 *sampling-size* の範囲は 64 ～ 256 バイトです。
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow 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 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow counter-poll-interval***poll-interval*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow counter-poll-interval***poll-interval* 例： `switch(config)# sflow counter-poll-interval 100`	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。
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow 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 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow max-datagram-size***datagram-size*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow max-datagram-size***datagram-size* 例： `switch(config)# sflow max-datagram-size 2000`	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。
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow 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 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

sFlow コレクタ アドレスの設定

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ことができます。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow collector-ip***ip-address***vrf***vrf* [**source***ip-address*]
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow collector-ip***ip-address***vrf***vrf* [**source***ip-address*] 例： `switch(config)# sflow collector-ip 192.0.2.5 vrf management`	sFlow コレクタの IPv4 アドレスを設定します。IP アドレスを 0.0.0.0 に設定した場合、すべてのサンプリングはディセーブルになります。 *vrf* には、次のいずれかを指定できます。 • ユーザ定義の 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。

	コマンドまたはアクション	目的
		• **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。 **source***ip-address* オプションを指定すると、送信される sFlow データグラムで送信元 IP アドレスが IP パケットの送信元アドレスとして使用されるようになります。送信元IPアドレスはローカルスイッチインターフェイスの1つで事前に設定されている必要があり、そうでない場合はエラーメッセージが表示されます。このオプションの設定後に、送信元IPアドレスのインターフェイスが変更または削除されると、sFlow データグラムは送信されなくなり、イベント履歴エラーおよび syslog エラーが記録されます。**source***ip-address* オプションが未設定の場合、Cisco NX-OS は送信される sFlow データグラムに対して、IP パケットの送信元アドレスを自動的に選択します。
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow コンフィギュレーションを表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

sFlow コレクタ 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。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow collector-port***collector-port*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow collector-port***collector-port* 例： `switch(config)# sflow collector-port 7000`	sFlow コレクタの UDP ポートを設定します。 *collector-port* の範囲は 0 ～ 65535 です。
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow コンフィギュレーションを表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

sFlow エージェント アドレスの設定

sFlow エージェントの IPv4 アドレスを設定できます。

はじめる前に

sFlow がイネーブルになっていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow agent-ip***ip-address*
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow agent-ip***ip-address* 例： `switch(config)# sflow agent-ip 192.0.2.3`	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 アドレスは 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 アドレスが、コレクタへの sFlow データグラム送信用の送信元 IP アドレスである必要はありません。
ステップ 3	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow コンフィギュレーションを表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

sFlow サンプリング データ ソースの設定

sFlow サンプラーのデータ ソースは、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 チャネルとして指定できます。

はじめる前に

sFlow 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。

Cisco Nexus 9332PQ、9372PX、9372TX、93120TX スイッチ、および N9K-M6PQ または N9K-M12PQ 汎用拡張モジュール（GEM）搭載の Cisco Nexus 9396PX、9396TX、93128TX スイッチについて、これらのデバイスで sFlow データ ソースとして設定されているすべてのアップリンク ポート用の sFlow および SPAN ACL TCAM リージョン サイズが設定されていることを確認します。

手順の概要

1. **configure terminal**
2. [**no**] **sflow data-source interface** [**ethernet***slot*/*port*[-*port*] | **port-channel***channel-number*]
3. （任意） **show sflow**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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**] **sflow data-source interface** [**ethernet***slot*/*port*[-*port*]	**port-channel***channel-number*] 例： `switch(config)# sflow data-source interface ethernet 1/5-12`	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	**show sflow** 例： `switch(config)# show sflow`	（任意） sFlow コンフィギュレーションを表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

sFlow 設定の確認

sFlow 設定を表示させるには、次のコマンドを使用します。

表 *17*：*sFlow Show* コマンド

コマンド	目的
show sflow	sFlow エージェント コンフィギュレーションおよびsFlow サンプラーのすべてのデータ ソースを表示します。

コマンド	目的
show process	sFlowプロセスが実行中であるかを確認します。
show running-config sflow [**all**]	現在の sFlow 実行コンフィギュレーションを表示します。

sFlow 統計情報のモニタリングとクリア

sFlow 統計情報を表示するには、**show sflow statistics** コマンドを使用します。

sFlow 統計情報をクリアするには、次のコマンドを使用します。

コマンド	説明
clear sflow statistics	**show sflow statistics** コマンドで示される sFlow 統計情報の大部分をクリアします。
clear counters interface all	**show sflow statistics** コマンドで示される Total Packets フィールドをクリアします。
clear hardware rate-limiter sflow	**show sflow statistics** コマンドで示される Total Samples フィールドをクリアします。

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
```

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
ACL TCAM リージョン	IP ACL の設定

関連資料

第 21 章

TAPアグリゲーションおよびMPLSストリッピングの設定

この章では、Cisco NX-OS デバイスで TAP アグリゲーションおよび MPLS ストリッピングを設定する方法について説明します。

この章の内容は、次のとおりです。

TAP アグリゲーションについて

ネットワーク TAP

さまざまなメソッドを使用して、パケットをモニタできます。その1つの方法は、物理ハードウェアの Test Access Point（TAP）を使用するものです。

ネットワーク TAP 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 TAP 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 TAP 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 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 TAP 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 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。

TAP 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 TAP は、分散 TAP 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 TAP 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。

TAP アグリゲーション

TAP アグリゲーションは、データセンターでのモニタリングやトラブルシューティング作業を支援する代替ソリューションです。これは 1 つのデバイスを指定して、複数の Test Access Point（TAP）の集約（アグリゲーション）ができるようにし、複数のモニタリング システムに接続させることで機能します。TAP 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 デバイスをリンクします。

TAP アグリゲーション スイッチ ソリューションでは、Cisco Nexus 9000 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 TAP を使用して、この TAP 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。TAP アグリゲーション スイッチは、ネットワーク ファブリック内のイベントをモニタ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。

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よう TAP アグリゲーション スイッチを設定できます。トラフィックを複数のインターフェイスにリダイレクトするため、マルチキャスト グループがスイッチ内部で作成され、リダイレクト リストに含まれるインターフェイスがメンバー ポートとして追加されます。リダイレクト アクションの付けられたアクセス コントロール リスト（ACL）ポリシーがインターフェイスに適用されると、ACL ルール

に一致するトラフィックは、作成された内部のマルチキャスト グループにリダイレクトされます。

TAP アグリゲーションのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	TAP アグリゲーションにライセンスは不要です。ライセンス パッケージに含まれていない機能はnx-osイメージにバンドルされており、無料で提供されます。Cisco 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。

TAP アグリゲーションの注意事項と制約事項

TAP 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。

- TAP アグリゲーションは、スイッチ ポートおよび入力方向でのみサポートされます。
- TAP アグリゲーションは、100G ポートでサポートされます。
- レイヤ2インターフェイスだけが、TAPアグリゲーション ポリシーをサポートします。レイヤ 3 インターフェイスはポリシーを適用できますが、そのポリシーは機能しなくなります。
- TAP アグリゲーションが IPv6 をサポートするのは、EtherType がリダイレクト アクションで MAC ACL を使用する場合です。
- リダイレクト ポートは、送信元（TAP）ポートと同じVLANに属している必要があります。
- 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。
- TAPアグリゲーション ポリシー用のインターフェイスの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-channel50** と入力します。

MPLS ストリッピングについて

Cisco Nexus 9000 シリーズ スイッチの入力ポートは、さまざまなマルチプロトコル ラベル スイッチング（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 アクセス コントロール リスト（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

一部の MPLS ラベルは、レイヤ 2 ヘッダーとレイヤ 3 ヘッダーの間に追加されます。これらのラベルにとって、ヘッダーとデータは標準のバイト オフセット位置にありません。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 ストリッピングのライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	MPLS ストリッピングにライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

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 ストリッピングをイネーブルにする前に、すべてのレイヤ 3 および vPC 機能をディセーブルにしておきます。
- スタティック MPLS、MPLS セグメント ルーティング、および MPLS ストリッピングを同時にイネーブルにすることはできません。
- MPLS ストリッピングに関係する入力インターフェイスのみで、TAP アグリゲーションがイネーブル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 TAP 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 TAP 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。
- ポート チャネルのロード バランシングは、MPLS ストリッピング パケットでサポートされます。
- 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 ストリッピング中、着信 VLAN は維持されます。

TAP アグリゲーションの設定

TAP アグリゲーション ポリシーの設定

IPv4 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。

はじめる前に

hardware access-list tcam region {ifacl | mac-ifacl} コマンドを使用して、IPv4 ポート ACL または MAC ポート ACL 用の ACL TCAM のリージョン サイズを設定する必要があります。詳細については『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』の「Configuring ACL TCAM Region Sizes」を参照してください。

（注） デフォルトでは、ifacl および mac-ifacl のリージョン サイズはどちらもゼロです。TAP アグリゲーションをサポートするには、ifacl または mac-ifacl リージョンに十分なエントリを割り当てる必要があります。

手順の概要

1. **configure terminal**
2. 次のいずれかのコマンドを入力します。
   - **ip access-list***access-list-name*
   - **mac access-list***access-list-name*
3. （任意） **statistics per-entry**
4. [**no**] **permit***protocol source destination***redirect***interfaces*
5. （任意） 次のいずれかのコマンドを入力します。
   - **show ip access-lists** [*access-list-name*]
   - **show mac access-lists** [*access-list-name*]
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	次のいずれかのコマンドを入力します。
	• **ip access-list***access-list-name*
	• **mac access-list***access-list-name*
	
	例：
	`switch(config)# ip access-list test`
	`switch(config-acl)#`
	
	`switch(config)# mac access-list mactap1`
	`switch(config-mac-acl)#`
		IPv4 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。

ステップ 3	**statistics per-entry** 例： `switch(config-acl)# statistics per-entry`	（任意） 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。

	コマンドまたはアクション	目的
ステップ 4	[**no**] **permit***protocol source destination***redirect***interfaces* 例： `switch(config-acl)# permit ip any any redirect ethernet1/8`	条件ごとのトラフィックのリダイレクトを許可する、IP または MAC ACL のルールを作成します。このコマンドの **no** 形式は、ポリシーから許可ルールを削除します。 （注） TAP アグリゲーション ポリシーのインターフェイスを入力する際には、省略しないでください。インターフェイスの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。

ステップ 5	次のいずれかのコマンドを入力します。
	• **show ip access-lists** [*access-list-name*]
	• **show mac access-lists** [*access-list-name*]
	
	例：
	`switch(config-acl)# show ip access-lists test`
	
	`switch(config-mac-acl)# show mac access-lists mactap1`
		（任意）
		すべての IPv4 または MAC ACL、あるいは特定の IPv4 または MAC ACL を表示します。

ステップ 6	**copy running-config startup-config** 例： `switch(config-acl)# copy running-config startup-config`	（任意） 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。

TAP 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

TAP アグリゲーションを設定した ACL は、レイヤ 2 インターフェイスに適用できます。

手順の概要

1. **configure terminal**
2. **interface***type slot/port*
3. **switchport**
4. 次のいずれかのコマンドを入力します。
   - [**no**] **ip port access-group***access-list-name***in**
   - [**no**] **mac port access-group***access-list-name***in**
5. （任意） **mode tap-aggregation**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**switchport** 例： `switch(config-if)# switchport`	レイヤ 3 インターフェイスをレイヤ 2 インターフェイスに変更します。 （注） 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てください。

ステップ 4	次のいずれかのコマンドを入力します。
	• [**no**] **ip port access-group***access-list-name***in**
	• [**no**] **mac port access-group***access-list-name***in**
	
	例：
	`switch(config-if)# ip port access-group test in`
	
	`switch(config-if)# mac port access-group test in`
		TAP アグリゲーションを設定した IPv4 または MAC ACL をインターフェイスに適用します。このコマンドの **no** 形式を使用すると、インターフェイスから ACL を削除します。

	コマンドまたはアクション	目的
ステップ 5	**mode tap-aggregation** 例： `switch(config-if)# mode tap-aggregation`	（任意） TAP 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許可します。 （注） このコマンドは、MPLS ストリッピングを使用するため Cisco NX-OS リリース 7.0(3)I2(1) で導入されました。このコマンドは、MPLS ストリッピングを入力レイヤ 2 インターフェイスおよびポート チャネルで機能させるために必要です。レイヤ 2 ポート チャネルに関しては、このコマンドが必要とされるのは親インターフェイスだけであり、チャネル グループ メンバー インターフェイスでは必要とされません。
ステップ 6	**copy running-config startup-config** 例： `switch(config-if)# copy running-config startup-config`	（任意） 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。

TAP アグリゲーションの設定の確認

TAP アグリゲーションの設定情報を表示するには、次の作業のいずれかを行います。

コマンド	目的
show ip access-lists [*access-list-name*]	すべての IPv4 ACL または特定の IPv4 ACL を表示します。
show mac access-lists [*access-list-name*]	すべての MAC ACL または特定の MAC ACL を表示します。

TAP アグリゲーションの設定例

次に、TAP アグリゲーション ポリシーを IPv4 ACL に設定する例を示します。

```
switch# configure terminal
switch(config)# ip access-list test
switch(config-acl)# 10 deny ip 100.1.1/24 any
switch(config-acl)# 20 permit tcp any eq www any redirect port-channel4
switch(config-acl)# 30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
switch(config-acl)# show ip access-lists test
IP access list test
        10 deny ip 100.1.1/24 any
```

```
        20 permit tcp any eq www any redirect port-channel4
        30 permit ip any any redirect
Ethernet1/1,Ethernet1/2,port-channel7,port-channel8,Ethernet1/12,Ethernet1/13
```

次に、TAP アグリゲーション ポリシーを MAC ACL に設定する例を示します。

```
switch# configure terminal
switch(config)# mac access-list mactap1
switch(config-mac-acl)# 10 permit any any 0x86dd redirect port-channel1
switch(config-mac-acl)# show mac access-lists mactap1
MAC access list mactap1
        10 permit any any 0x86dd redirect port-channel1
```

次に、TAP アグリゲーション ポリシーをレイヤ 2 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 ethernet 1/2
switch(config-if)# ip port access-group test in
switch(config-if)#
```

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。

はじめる前に

MPLS ストリッピングをイネーブルにする前に、すべてのレイヤ 3 および vPC 機能をディセーブルにしておきます。

mode tap-aggregation コマンドを使用して、TAP アグリゲーション ポリシーを設定した ACL をレイヤ 2 インターフェイスまたはポート チャネルにアタッチします。詳細については、TAP 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, （337 ページ）を参照してください。

手順の概要

1. **configure terminal**
2. **[no] mpls strip**
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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] mpls strip** 例： `switch(config)# 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。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

MPLS ラベルの追加と削除

デバイスは、TAP インターフェイスで不明なラベルのフレームを受信するたびにラベルを動的に学習できます。スタティック MPLS ラベルを追加または削除することもできます。

はじめる前に

TAP アグリゲーション ポリシーを設定し、ポリシーをインターフェイスにアタッチします。詳細については、『*Cisco Nexus 9000 Series NX-OS System Management Configuration Guide*』を参照してください。

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 TAP アグリゲーション ACL を設定する必要があります。

手順の概要

1. **configure terminal**
2. **mpls strip label***label*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**mpls strip label***label* 例： `switch(config)# mpls strip label 100`	指定したスタティック MPLS ラベルを追加します。ラベルの値は 20 ビットで、使用できる範囲は 1 ～ 1048575 です。

	コマンドまたはアクション	目的
		[**no**] **mpls strip label** {*label* | **all**} コマンドは、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。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

手順の概要

1. **configure terminal**
2. **mpls strip dest-mac***mac-address*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**mpls strip dest-mac***mac-address* 例： `switch(config)# mpls strip dest-mac 1.1.1`	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。 • E.E.E • EE-EE-EE-EE-EE-EE • EE:EE:EE:EE:EE:EE • EEEE.EEEE.EEEE

	コマンドまたはアクション	目的
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

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。

手順の概要

1. **configure terminal**
2. **mpls strip label-age***age*
3. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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	**mpls strip label-age***age* 例： `switch(config)# mpls strip label-age 300`	ダイナミック MPLS ラベルがエージ アウトする時間を秒単位で指定します。範囲は 61 ～ 31622400 です。
ステップ 3	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

MPLS ストリッピング設定の確認

MPLS ストリッピング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 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*：表示させるラベルです。 • **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。

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
```

MPLS ストリッピング カウンタおよびラベル エントリのクリア

MPLS ストリッピング カウンタとラベル エントリをクリアするには、次のタスクを実行します。

コマンド	目的
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。
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。

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 ストリッピングの設定例

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
```

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
IP ACL	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
MAC ACL	『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』
ポート チャネル シンメトリック ハッシュ	『*Cisco Nexus 9000 Series NX-OS Interfaces Configuration Guide*』
リモート モニタリング（RMON）	RMON の設定，（209 ページ）
スイッチド ポート アナライザ（SPAN）	SPAN の設定，（261 ページ）
トラブルシューティング	『*Cisco Nexus 9000 Series NX-OS Troubleshooting Guide*』

第 22 章

グレースフル挿入と削除の設定

この章では、Cisco Nexus 9000 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。

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）
- プロトコルに依存しないマルチキャスト（PIM）
- ルーティング情報プロトコル（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。

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。
シャットダウン	プロトコルまたは vPC ドメインをシャットダウンします。

コマンド	説明
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。

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

GIR のライセンス要件

製品	ライセンス要件
Cisco NX-OS	グレースフル挿入と削除（GIR）にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco 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。

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, （350 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, （352 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, （353 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, （356 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, （359 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, （353 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, （360 ページ） を参照）。

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。

手順の概要

1. **configure maintenance profile maintenance-mode**
2. **end**
3. **show maintenance profile maintenance-mode**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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, （348 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。

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)# ip pim isolate
switch(config-mm-profile)# vpc domain 10
switch(config-mm-profile-config-vpc-domain)# shutdown
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]
ip pim isolate
vpc domain 10
  shutdown
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。

手順の概要

1. **configure maintenance profile normal-mode**
2. **end**
3. **show maintenance profile normal-mode**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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, （348 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。

次に、通常モード 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)# vpc domain 10
```

```
switch(config-mm-profile-config-vpc-domain)# no shutdown
switch(config-mm-profile)# no ip pim isolate
switch(config-mm-profile)# 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
vpc domain 10
  no shutdown
no ip pim isolate
```

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。スナップショットを作成する場合、定義済みの一連の **show** コマンドが実行され、出力が保存されます。

手順の概要

1. **snapshot create***snapshot-name description*
2. **show snapshots**
3. **show snapshots compare***snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]

手順の詳細

	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。

	コマンドまたはアクション	目的

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。 **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。

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。

手順の概要

1. **snapshot section add***section "show-command" row-id element-key1* [*element-key2*]
2. **show snapshots sections**
3. **show snapshots compare***snapshot-name-1 snapshot-name-2* [**summary** | **ipv4routes** | **ipv6routes**]

手順の詳細

	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。

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。

はじめる前に

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, （350 ページ）を参照してください。

手順の概要

1. **configure terminal**
2. **system mode maintenance** [**dont-generate-profile** | **timeout***value* | **shutdown** | **on-reload reset-reason***reason*]
3. （任意） **show system mode**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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:`
	
	`ip pim isolate`
	`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 : ip pim isolate`
	`Applying : router bgp 65502`
	`Applying : isolate`
	`Applying : router ospf p1`
	`Applying : isolate`
	`Applying : router ospfv3 p1`
	`Applying : isolate`
		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、vPC ドメイン、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**

	コマンドまたはアクション	目的
	`Maintenance mode operation successful.`	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	**show system mode**
	例：
	`switch(config)# show system mode`
	`System Mode: Maintenance`
		（任意）
		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	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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。

次に、スイッチのすべてのプロトコル、vPC ドメイン、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:

   vpc domain 10
     shutdown
   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 : vpc domain 10
Applying :   shutdown
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。

はじめる前に

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, （350 ページ）を参照してください。

手順の概要

1. **configure terminal**
2. **no system mode maintenance** [**dont-generate-profile**]
3. （任意） **show system mode**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure 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 system mode maintenance** [**dont-generate-profile**]
	
	例：
	`switch(config)# no system mode maintenance dont-generate-profile`
	`Following configuration will be applied:`
		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

	コマンドまたはアクション	目的
	no ip pim isolate 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 : no ip pim isolate 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。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	**show system mode** 例： switch(config)# show system mode System Mode: Normal	（任意） 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。

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,（356 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。

コマンド	目的
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。
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 23 章

ソフトウェア メンテナンス アップグレードの実行

この章では、Cisco NX-OS デバイスでソフトウェア 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 9000 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。

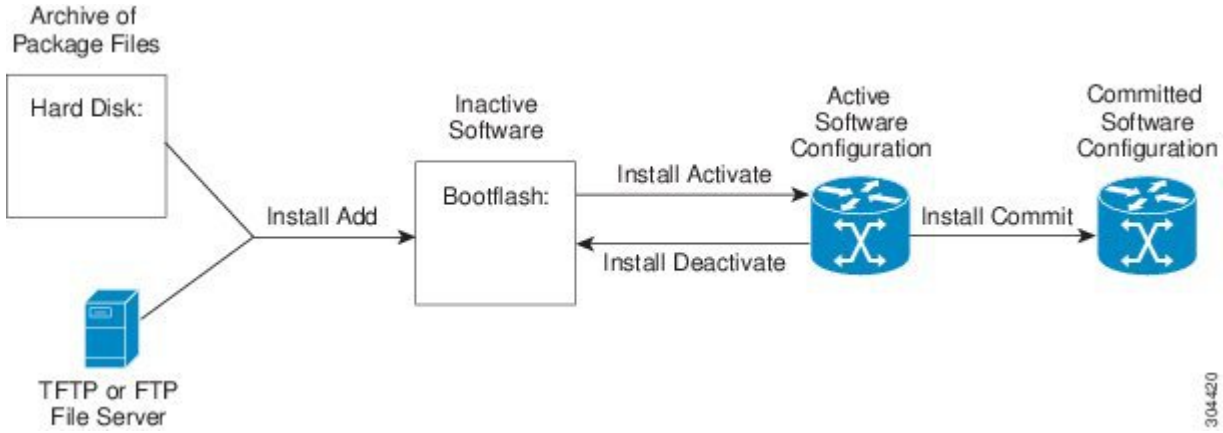
パッケージ管理

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. （任意）必要に応じて、パッケージを非アクティブ化して削除します。

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 *5*：*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



パッケージのアクティブ化と非アクティブ化の影響

SMUパッケージのアクティブ化または非アクティブ化は、システムにすぐさま影響を与える可能性があります。システムは次のように影響を受ける場合があります。

- 新しいプロセスが開始する場合があります。
- 実行しているプロセスが停止または再起動する場合があります。

- ライン カードのすべてのプロセスが再起動する場合があります。ライン カードのプロセスの再起動は、ソフト リセットと同等です。
- ライン カードがリロードする場合があります。
- ライン カードのプロセスは影響を受けない場合があります。

（注） 必要に応じて、改訂されたコンフィギュレーションおよびコンフィギュレーションの再適用によって起こる問題に対処する必要があります。

ヒント パッケージをアクティブ化する際に**test**オプションを使用すると、稼働中のシステムに影響を与えることなく、コマンドの効果をテストすることができます。アクティブ化プロセスが完了したら、**show install log** コマンドを入力してプロセスの結果を表示します。

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 NX-OS 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 NX-OS 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。

はじめる前に

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。

手順の概要

1. **show install active**
2. **show module**
3. **show clock**

手順の詳細

	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。

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
```

Cisco.com からの SMU パッケージ ファイルのダウンロード

SMU パッケージ ファイルをダウンロードするには、次の手順に従ってください。

手順の概要

1. Cisco.com にログインします。
2. 次の URL から Download Software ページに移動します。 http://software.cisco.com/download/navigator.html
3. [Select a Product] リストから、[**Switches**] > [**Data Center Switches**] > [**Cisco Nexus 9000 Series Switches**] > [*model*] を選択します。
4. デバイスに適した SMU ファイルを選択し、[**Download**] をクリックします。

手順の詳細

ステップ 1 Cisco.com にログインします。

ステップ 2 次の URL から Download Software ページに移動します。 http://software.cisco.com/download/navigator.html

ステップ 3 [Select a Product] リストから、[**Switches**] > [**Data Center Switches**] > [**Cisco Nexus 9000 Series Switches**] > [*model*] を選択します。

ステップ 4 デバイスに適した SMU ファイルを選択し、[**Download**] をクリックします。

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。詳細については、『*Cisco Nexus 9000 Series NX-OS Security Configuration Guide*』を参照してください。

（注） お使いのネットワーク サーバの場所と可用性については、システム管理者に問い合わせてください。

ファイル転送プロトコルを使用してサーバからデバイスに SMU パッケージ ファイルをコピーするには、次の表のコマンドを使用します。

表 *18*：*SMU* パッケージ ファイルをデバイスにコピーするためのコマンド

コマンド	目的
copy tftp://*hostname-or-ipaddress*/*directory-path*/*filename***bootflash:** `switch# copy tftp://10.1.1.1/images/ n9000-dk9.6.1.2.I2.1.CSCab00001.bin bootflash:`	TFTP サーバから bootflash: にパッケージ ファイルをコピーします。 • *hostname-or-ipaddress*：ネットワーク ファイル サーバのホスト名または IP アドレス。 • *directory-path*：追加されるパッケージ ファイルに導くネットワーク ファイルのサーバ パス。 • *filename*：追加するパッケージ ファイルの名前。

コマンド	目的
copy ftp:*//username:password@hostname-or-ipaddress/directory-path/filename***bootflash:** `switch# copy ftp://john:secret@10.1.1.1/images/ n9000-dk9.6.1.2.I2.1.CSCab00001.bin bootflash:`	FTP サーバから bootflash: にパッケージ ファイルをコピーします。 • *username*：パッケージ ファイルを保存するディレクトリへのアクセス権を持つユーザのユーザ名。 • *password*：パッケージ ファイルを保存するディレクトリへのアクセス権を持つユーザのユーザ名に関連付けられたパスワード。パスワードを指定しないと、ネットワーキング デバイスは、anonymous FTP を受け入れます。 • *hostname-or-ipaddress*：ネットワーク ファイル サーバのホスト名または IP アドレス。 • *directory-path*：追加されるパッケージ ファイルに導くネットワーク ファイルのサーバパス。指定されるディレクトリは、ユーザのホームディレクトリの下のディレクトリである必要があります。この例では、ダウンロードされたファイルはユーザ「john」のホーム ディレクトリ内の「images」というサブディレクトリにあります。 （注） FTP サービスの場合、*directory-path* は *username* ホーム ディレクトリの相対パスです。ディレクトリの絶対パスを指定するには、サーバ アドレスの後ろに「/」を追加する必要があります。 • *filename*：追加するパッケージ ファイルの名前。

コマンド	目的
copy sftp://*hostname-or-ipaddress*/*directory-path*/*filename***bootflash:** `switch# copy sftp://10.1.1.1/images/n9000-dk9.6.1.2.I2.1.CSCab00001.bin bootflash:`	SFTP サーバから bootflash: にパッケージ ファイルをコピーします。 • *hostname-or-ipaddress*：ネットワーク ファイル サーバのホスト名または IP アドレス。 • *directory-path*：追加されるパッケージ ファイルに導くネットワーク ファイルのサーバ パス。 • *filename*：追加するパッケージ ファイルの名前。

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） 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。

はじめる前に

追加するすべてのパッケージがローカル ストレージ デバイスまたはネットワーク ファイル サーバにあることを確認します。

パッケージのアクティブ化の前提条件をすべて満たしていることを確認します。

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，（368 ページ） に記載されている手順を完了します。

手順の概要

1. コンソール ポートに接続して、ログインします。
2. （任意） **dir bootflash:**
3. **install add***filename* [**activate**]
4. （任意） **show install inactive**
5. **install activate***filename* [**test**]
6. すべてのパッケージがアクティブ化されるまで手順 5 を繰り返します。
7. （任意） **show install active**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	コンソール ポートに接続して、ログインします。	コンソール ポートに CLI 管理セッションを確立します。
ステップ 2	**dir bootflash:**	（任意） 追加可能なパッケージ ファイルを表示します。 （注） このプロシージャを使用して追加およびアクティブ化できるのは SMU パッケージ ファイルだけです。
ステップ 3	**install add***filename* [**activate**] 例： `switch# install add bootflash: n9000-dk9.6.1.2.I2.1.CSCab00001.bin`	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* • **usb1:***filename* • **usb2:***filename* CSCur02700 SMU パッケージを除くすべての SMU パッケージで、正常に追加された後に自動的にパッケージをアクティブにするには、オプションの **activate** キーワードを使用できます。 （注） CSCur02700 SMU パッケージについては、パッケージをアクティブにするのに手順 5 の **install activate** コマンドを使用します。パッケージの実行に失敗してリブートが必要となる可能性があるため、オプションの **activate** キーワードを付けた **install add** コマンドは使用しないでください。

	コマンドまたはアクション	目的
		SMUパッケージの複数バージョンが、実行コンフィギュレーションに影響を与えずにストレージデバイスに追加できます。しかし、ライン カードに対してアクティブ化できるのは、1 つのバージョンのパッケージだけです。
ステップ 4	**show install inactive** 例： `switch# show install inactive`	（任意） 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 5	**install activate***filename* [**test**]
	
	例：
	`switch# install activate n9000-dk9.6.1.2.I2.1.CSCab00001.bin`
	
	例：
	`switch# install activate n9000-dk9.6.1.2.I2.1.CSCab00001.bin test`
	`WARNING! No changes will occur due to 'test' option`
	`WARNING! The following is the predicted output for this commad`
	`Install operation 1 completed successfully at Thu Jan 9 01:27:56 2014`
	
	例：
	`switch# install activate n9000-dk9.6.1.2.I2.1.CSCab00001.bin`
	`Install operation 18 !!WARNING!! This patch will get activated only after a reload of the switch. at Sun Mar 9 00:42:12 2014`
		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。
		ヒント パッケージをアクティブ化する際に **test** キーワードを使用すると、稼働中のシステムに影響を与えることなく、コマンドの効果をテストすることができます。アクティブ化プロセスが終了したら、**show install log** コマンドを入力してプロセスの結果を表示します。

ステップ 6	すべてのパッケージがアクティブ化されるまで手順 5 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 7	**show install active** 例： `switch# show install active`	（任意） 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。

（注） 起動時に、デバイスはコミットされたパッケージ セットをロードします。現在のアクティブなパッケージがコミットされる前にシステムがリロードされると、以前にコミットされたパッケージ セットが使用されます。

はじめる前に

パッケージ セットをコミットする前に、デバイスが正常に動作し、想定どおりにパットを転送していることを検証します。

パッケージの追加とアクティブ化, （371 ページ） に記載されている手順を完了します。

手順の概要

1. **install commit***filename*
2. **reload module***standby-sup-slot*
3. （任意） **show install committed**

手順の詳細

	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 n9000-dk9.6.1.2.I2.1.CSCab00001.bin`	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	**reload module***standby-sup-slot* 例： `switch# reload module 2`	インストールされている場合、スタンバイ スーパーバイザ モジュールをリロードします。 （注） SMUパッケージの適用先がデュアル スーパーバイザ システムであり、Cisco NX-OS リリース 6.1(2)I2(3) または Cisco NX-OS 6.1(2)I2(2b) より以前のソフトウェア リリースでデバイスを動作させる場合は、スタンバイ スーパーバイザ モジュールのリロードが必要です。

	コマンドまたはアクション	目的
ステップ 3	**show install committed** 例： `switch# show install committed`	（任意） コミットされたパッケージを表示します。

次に、デバイス上でアクティブなSMUパッケージをコミットして、次にコミットされたパッケージを確認する例を示します。

```
switch# install commit n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 2 completed successfully at Thu Jan 9 01:20:46 2014

switch# show install committed
Committed Packages:
n9000-dk9.6.1.2.I2.1.CSCab00001.bin
```

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。

Cisco NX-OS ソフトウェアでは、選択されたパッケージ セットを前に保存されたパッケージ セットにロールバックする柔軟性も提供されます。以前のパッケージセットの方が現在アクティブなパッケージセットよりも適切であることがわかった場合は、**install deactivate** および **install commit** コマンドを使用して、以前アクティブだったパッケージ セットを再びアクティブにできます。

はじめる前に

別のアクティブなパッケージに必要なパッケージを非アクティブ化することはできません。パッケージを非アクティブ化しようとすると、システムがそのパッケージが他のアクティブなパッケージによって必要とされていないかを自動的にチェックします。非アクティブ化を実行するのは、すべての互換性が確認できた場合だけです。

デバイスの実行中のソフトウェアまたはコミットされたソフトウェアの一部であるパッケージは削除できません。

手順の概要

1. コンソール ポートに接続して、ログインします。
2. **reload module***standby-sup-slot*
3. **install deactivate***filename*
4. （任意） **show install inactive**
5. （任意） **install commit**
6. （任意） **install remove** {*filename* | **inactive**}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	コンソールポートに接続して、ログインします。	コンソール ポートに CLI 管理セッションを確立します。
ステップ 2	**reload module***standby-sup-slot* 例： `switch# reload module 2`	インストールされている場合、スタンバイ スーパーバイザ モジュールをリロードします。 （注） デュアル スーパーバイザ システムで SMU パッケージを非アクティブ化し、Cisco NX-OS リリース 6.1(2)I2(3) または Cisco NX-OS 6.1(2)I2(2b) より以前のソフトウェア リリースでデバイスを動作させる場合は、スタンバイ スーパーバイザ モジュールのリロードが必要です。
ステップ 3	**install deactivate***filename* 例： `switch# install deactivate n9000-dk9.6.1.2.I2.1.CSCab00001.bin`	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	**show install inactive** 例： `switch# show install inactive`	（任意） 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 5	**install commit** 例： `switch# install commit`	（任意） 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 6	**install remove** {*filename* | **inactive**}
	
	例：
	`switch# install remove n9000-dk9.6.1.2.I2.1.CSCab00001.bin`
	`Proceed with removing n9000-dk9.6.1.2.I2.1.CSCab00001.bin? (y/n)? [n] y`
	
	例：
	`switch# install remove inactive`
	`Proceed with removing? (y/n)? [n] y`
		（任意）
		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。

	コマンドまたはアクション	目的
		• 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。

次に、パッケージを非アクティブ化し、変更内容をコミットし、デバイスから非アクティブなパッケージを削除する例を示します。

```
switch# install deactivate n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 3 completed successfully at Thu Jan 9 01:20:36 2014

switch# show install inactive
Inactive Packages:
n9000-dk9.6.1.2.I2.1.CSCab00001.bin

switch# install commit
Install operation 4 completed successfully at Thu Jan 9 01:20:46 2014

switch# install remove n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Proceed with removing n9000-dk9.6.1.2.I2.1.CSCab00001.bin? (y/n)? [n] y
Install operation 5 completed successfully at Thu Jan 9 01:20:57 2014
```

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
Thu Jan 9 01:26:09 2014
Install operation 1 by user 'admin' at Thu Jan 9 01:19:19 2014
Install add bootflash:n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 1 completed successfully at Thu Jan 9 01:19:24 2014
----------------------------------------
Install operation 2 by user 'admin' at Thu Jan 9 01:19:29 2014
Install activate n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 2 completed successfully at Thu Jan 9 01:19:45 2014
----------------------------------------
Install operation 3 by user 'admin' at Thu Jan 9 01:20:05 2014
Install commit n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 3 completed successfully at Thu Jan 9 01:20:08 2014
----------------------------------------
Install operation 4 by user 'admin' at Thu Jan 9 01:20:21 2014
Install deactivate n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 4 completed successfully at Thu Jan 9 01:20:36 2014
```

```
----------------------------------------
Install operation 5 by user 'admin' at Thu Jan 9 01:20:43 2014
Install commit n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 5 completed successfully at Thu Jan 9 01:20:46 2014
----------------------------------------
Install operation 6 by user 'admin' at Thu Jan 9 01:20:55 2014
Install remove n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 6 completed successfully at Thu Jan 9 01:20:57 2014
----------------------------------------
Install operation 7 by user 'admin' at Thu Jan 9 01:21:07 2014
Install remove
Install operation 7 completed successfully at Thu Jan 9 01:21:10 201
```

次に、ノードやプロセスへの影響を含む追加情報を表示する例を示します。

```
switch# show install log detail
Thu Jan 9 01:24:03 2014
Install operation 1 by user 'admin' at Thu Jan 9 01:19:19 2014
Installer started downloading the package: /n9000-dk9.6.1.2.I2.1.CSCab00001.bin
via bootflash
Install add bootflash:n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Copying file at Thu Jan 9 01:19:20 2014
Download success, 238545 bytes received
Verifying package
Checking MD5 at Thu Jan 9 01:19:21 2014
MD5 checksum OK
Checking HW platform at Thu Jan 9 01:19:22 2014
Checking SW platform at Thu Jan 9 01:19:23 2014
Package verified successfully
Sending patch file to plugin manager at Thu Jan 9 01:19:23 2014
The following package is now available to be activated: n9000-dk9.6.1.2.I2.1.CSC
ab00001.bin
Install operation 1 completed successfully at Thu Jan 9 01:19:24 2014
----------------------------------------
Install operation 2 by user 'admin' at Thu Jan 9 01:19:29 2014
Install activate n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install activate action started
The software will be activated with process restart
2 processes affected
sysinfo (modified)
vman (modified)
Install operation 2 completed successfully at Thu Jan 9 01:19:45 2014
----------------------------------------
Install operation 3 by user 'admin' at Thu Jan 9 01:20:05 2014
Install commit n9000-dk9.6.1.2.I2.1.CSCab00001.bin
MD5 checksum OK for patch: n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 3 completed successfully at Thu Jan 9 01:20:08 2014
----------------------------------------
Install operation 4 by user 'admin' at Thu Jan 9 01:20:21 2014
Install deactivate n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install deactivate action started
The software will be deactivated with process restart
2 processes affected
sysinfo (modified)
vman (modified)
Install operation 4 completed successfully at Thu Jan 9 01:20:36 2014
----------------------------------------
Install operation 5 by user 'admin' at Thu Jan 9 01:20:43 2014
Install commit n9000-dk9.6.1.2.I2.1.CSCab00001.bin
MD5 checksum OK for patch: n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 5 completed successfully at Thu Jan 9 01:20:46 2014
----------------------------------------
Install operation 6 by user 'admin' at Thu Jan 9 01:20:55 2014
Install remove n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install operation 6 completed successfully at Thu Jan 9 01:20:57 2014
----------------------------------------
Install operation 7 by user 'admin' at Thu Jan 9 01:21:07 2014
Install remove
Install operation 7 completed successfully at Thu Jan 9 01:21:10 2014
```

次に、SMU パッケージが起動した後、スイッチがリロードされる前の出力の例を示します。

```
switch# show install log detail
Install operation 18 by user 'admin' at Sun Mar 9 00:42:10 2014
Install activate n9000-dk9.6.1.2.I2.1.CSCab00001.bin
Install activate action started
The software will be activated with system reload
Install operation 18 !!WARNING!! This patch will get activated only after
a reload of the switch. at Sun Mar 9 00:42:12 2014
```

Guest Shell Bash のソフトウェア メンテナンス アップグレードの実行

Guest Shell 中の Bash のソフトウェア メンテナンス アップグレードを実行することができます。

手順の概要

1. Guest Shell Bash 用の SMU パッケージ ファイルを Cisco.com からダウンロードします。
2. SMU パッケージ ファイルをスイッチの bootflash: にコピーします。
3. **guestshell**
4. **sudo rpm -Uvh** /bootflash/*filename*
5. **rpm -qa** | **grep bash**
6. **guestshell sync**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Guest Shell Bash 用の SMU パッケージ ファイルを Cisco.com からダウンロードします。	Cisco.com からパッケージファイルを取得します。手順については、Cisco.com からの SMU パッケージファイルのダウンロード, （367 ページ）を参照してください。
ステップ 2	SMU パッケージ ファイルをスイッチの bootflash: にコピーします。	パッケージ ファイルをデバイスにコピーします。手順については、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, （368 ページ）を参照してください。

ステップ 3	**guestshell**
	
	例：
	`switch# guestshell`
	`guestshell:~$`
		Guest Shell にアクセスします。

	コマンドまたはアクション	目的

ステップ 4	**sudo rpm -Uvh** /bootflash/*filename*
	
	例：
	`guestshell:~$ sudo rpm -Uvh /bootflash/bash-4.2-r8.x86_64.rpm`
	`Preparing...`
	`########################################## [100%]`
	
	`1:bash`
	`########################################## [100%]`
	`update-alternatives: Linking //bin/sh to /bin/bash`
		Guest Shell 中の既存の Bash シェル ファイルをアップグレードします。

ステップ 5	**rpm -qa | grep bash**
	
	例：
	`guestshell:~$ rpm -qa | grep bash`
	`bash-4.2-r8.x86_64`
		新バージョンの Bash ファイルが正しくインストールされたかを確認します。

ステップ 6	**guestshell sync**
	
	例：
	`switch# guestshell sync`
	`Access to the guest shell will be temporarily disabled while it synchronizes contents to standby.`
	`Are you sure you want to continue? (y/n) [n] y`
	`dt-n9k3-1# 2014 Oct 7 05:00:01 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-INSTALL_STATE: Deactivating virtual service 'guestshell+'`
	`dt-n9k3-1# 2014 Oct 7 05:00:06 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Successfully deactivated virtual service 'guestshell+'`
	`2014 Oct 7 05:00:12 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Successfully deactivated virtual service 'guestshell+' ; Starting sync to standby sup`
	`2014 Oct 7 05:00:32 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-MOVE_STATE: Successfully synced virtual service 'guestshell+' ;`
	`Activating`
	`2014 Oct 7 05:00:32 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Activating virtual service 'guestshell+'`
	`2014 Oct 7 05:00:56 dt-n9k3-1 %$ VDC-1 %$ %VMAN-2-ACTIVATION_STATE: Successfully activated virtual service 'guestshell+'`
		デュアル スーパーバイザ システムで、スイッチオーバーの実行前に Bash SMU バージョンの rootfs をスタンバイ スーパーバイザに同期させます。このコマンドを実行しない場合、スーパーバイザのスイッチオーバー後にこの手順を繰り返す必要があります。
		
		（注） Guest Shell の再起動または Guest Shell の disable+enable 後も、新しい Bash ファイルは維持されます。ただし、Guest Shell の destroy+enable 後に Guest Shell Bash SMU パッケージ ファイルを再インストールする必要があります。

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
ソフトウェア アップグレード	『*Cisco Nexus 9000 Series NX-OS Software Upgrade and Downgrade Guide*』

SMU 履歴

次の表に、SMU パッケージ ファイルのリリースの履歴を示します。

SMU パッケージ ファイル	リリース	説明
bash-4.2-r8.x86_64.rpm	6.1(2)I3(1)	Bash 脆弱性 CVE-2014-6277、CVE-2014-6278、CVE-2014-7186、および CVE-2014-7187 用 Guest Shell Bash SMU
n9000-dk9.6.1.2.I3.1.CSCur02700.bin	6.1(2)I3(1) およびすべての 6.1(2)I2(x) リリース	CSCur02700 用 Cisco NX-OS SMU（Bash 脆弱性 CVE-2014-6277、CVE-2014-6278、CVE-2014-7186、および CVE-2014-7187）
n9000-dk9.6.1.2.I2.1.CSCup81353.bin	6.1(2)I2(1)、6.1(2)I2(2)、6.1(2)I2(2a)、および 6.1(2)I2(3)	CSCup81353 用 Cisco NX-OS SMU

付 録 A

Cisco NX-OS システム管理でサポートされている IETF RFC

Cisco NX-OS でサポートされているシステム管理に関する IETF RFC は次のとおりです。

- Cisco NX-OS システム管理でサポートされている IETF RFC, 383 ページ

Cisco NX-OS システム管理でサポートされている IETF RFC

Cisco NX-OS でサポートされているシステム管理に関する IETF RFC は次のとおりです。

RFC	Title
RFC 2819	『*Remote Network Monitoring Management Information Base*』
RFC 3164	『*The BSD syslog Protocol*』
RFC 3176	『*InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks*』
RFC 3411 および RFC 3418	『*An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*』

Cisco NX-OS システム管理でサポートされている IETF RFC

付 録 B

Embedded Event Manager システム イベントおよびコンフィギュレーション例

この付録では、Embedded Event Manager（EEM）システム ポリシー、イベント、およびポリシーのコンフィギュレーション例について説明します。

この付録は、次の項で構成されています。

EEM システム ポリシー

次の表に、Embedded Event Manager（EEM）のシステム ポリシーを示します。

イベント	説明
__PortLoopback	CallHome を実行し、Syslog、OBFL、または例外ログにエラーを記録し、GOLD "PortLoopback" テストに 10 回連続で失敗した場合は、その後影響を受けたポートでの HM テストをディセーブルにします。
__RewriteEngineLoopback	CallHome を実行し、Syslog、OBFL、または例外ログにエラーを記録し、GOLD "RewriteEngine" テストに 10 回連続で失敗した場合は、その後影響を受けたポートでの HM テストをディセーブルにします。

イベント	説明
__asic_register_check	CallHome を実行し、エラーを記録し、GOLD "ASICRegisterCheck" テストに 20 回連続で失敗した場合は、その後その ASIC デバイスおよびインスタンスの HM テストをディセーブルにします。
__compact_flash	CallHome を実行し、エラーを記録し、GOLD "CompactFlash" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__crypto_device	CallHome を実行し、GOLD "CryptoDevice" テストに失敗するとエラーを記録します。
__eobc_port_loopback	CallHome を実行し、GOLD "EOBCPortLoopback" テストに失敗するとエラーを記録します。
__ethpm_debug_1	アクション：なし
__ethpm_debug_2	アクション：なし
__ethpm_debug_3	アクション：なし
__ethpm_debug_4	アクション：なし
__ethpm_link_flap	420 秒間隔でリンク フラップが 30 を超えています。アクション：エラー。ポートをディセーブルにします。
__external_compact_flash	CallHome を実行し、エラーを記録し、GOLD "ExternalCompactFlash" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__lcm_module_failure	2 度電源を切って入れ直し、電源を切ります。
__management_port_loopback	CallHome を実行し、GOLD "ManagementPortLoopback" テストに失敗するとエラーを記録します。
__nvram	CallHome を実行し、エラーを記録し、GOLD "NVRAM" テストに 20 回連続で失敗した場合は、その後 HM テストをディセーブルにします。

イベント	説明
__pfm_fanabsent_all_systemfan	両方のファン トレイ（f1 と f2）が 2 分間存在しない場合は、シャットダウンします。
__pfm_fanbad_all_systemfan	ファンで障害が発生した場合シスログに記録します。
__pfm_fanbad_any_singlefan	ファンで障害が発生した場合シスログに記録します。
__pfm_power_over_budget	不十分な電力超過バジェットに対するシスログ警告
__pfm_tempev_major	TempSensor メジャーしきい値アクション：シャットダウン
__pfm_tempev_minor	TempSensorマイナーしきい値アクション：シスログ
__primary_bootrom	CallHome を実行し、エラーを記録し、GOLD "PrimaryBootROM" テストに 20 回連続で失敗した場合は、その後 HMテストをディセーブルにします。
__pwr_mgmt_bus	CallHome を実行し、エラーを記録し、GOLD "PwrMgmtBus" テストに20回連続で失敗した場合は、モジュールまたはスパインカードの HM テストをディセーブルにします。
__real_time_clock	CallHome を実行し、エラーを記録し、GOLD "RealTimeClock" テストに 20 回連続で失敗した場合は、その後 HMテストをディセーブルにします。
__secondary_bootrom	CallHome を実行し、エラーを記録し、GOLD "SecondaryBootROM" テストに 20 回連続で失敗した場合は、その後 HMテストをディセーブルにします。
__spine_control_bus	CallHome を実行し、エラーを記録し、GOLD "SpineControlBus" テストに 20 回連続で失敗した場合は、そのモジュールまたはスパインカードの HM テストをディセーブルにします。

イベント	説明
__standby_fabric_loopback	CallHome を実行し、エラーを記録し、10 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__status_bus	CallHome を実行し、エラーを記録し、GOLD "StatusBus" テストに 5 回連続で失敗した場合は、その後 HM テストをディセーブルにします。
__system_mgmt_bus	CallHome を実行し、エラーを記録し、GOLD "SystemMgmtBus" テストに 20 回連続で失敗した場合は、そのファンまたは電源の HM テストをディセーブルにします。
__usb	CallHome を実行し、GOLD "USB" テストに失敗するとエラーを記録します。

EEM イベント

次の表は、デバイスで使用できる EEM イベントについて説明します。

EEM イベント	説明
アプリケーション	アプリケーション固有のイベントをパブリッシュします。
cli	ワイルドカードを使用したパターンを照合する CLI コマンドが入力されます。
counter	EEM カウンタが指定された値または範囲に達します。
fanabsent	システム ファン トレイがありません。
fanbad	システム ファンで障害が生成されます。
fib	ユニキャスト FIB のルートまたは TCAM の使用状況をモニタします。
Gold	GOLD テスト失敗条件がヒットします。

EEM イベント	説明
interface	インターフェイス カウンタがしきい値を超えます。
メモリ	使用可能なシステム メモリがしきい値を超えます。
module	指定したモジュールが、選択したステータスになります。
module-failure	モジュール障害が生成されます。
none	指定されたイベントがないポリシー イベントを実行します。
oir	活性挿抜が発生します。
policy-default	デフォルトのパラメータおよびしきい値が、上書きするシステム ポリシーのイベントに使用されます。
poweroverbudget	プラットフォーム ソフトウェアが電力バジェット条件を検出します。
snmp	SNMP オブジェクト ID（OID）の状態が変化します。
storm-control	プラットフォーム ソフトウェアがイーサネット パケット ストーム条件を検出します。
syslog	syslog メッセージを監視し、ポリシーの検索文字列に基づいてポリシーを呼び出します。
sysmgr	システム マネージャがイベントを生成します。
temperature	システムの温度レベルがしきい値を超えます。
timer	指定された時間に到達します。
track	トラッキング対象オブジェクトの状態が変化します。

EEM ポリシーのコンフィギュレーション例

CLI イベントのコンフィギュレーション例

インターフェイス シャットダウンのモニタリング

インターフェイスのシャットダウンをモニタする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorShutdown
switch(config-applet)#
switch(config-applet)# description "Monitors interface shutdown."
switch(config-applet)# event cli match "conf t; interface *; shutdown"
switch(config-applet)# action 1.0 cli show interface e 3/1
switch(config)# copy running-config startup-config
```

（注） EEM ポリシーの一部として入力された **show** コマンドの出力は、「eem_archive_」というプレフィックスが付加されたテキスト ファイルとして logflash にアーカイブされます。アーカイブされている出力を表示するには、**show file logflash:eem_archive_***n* コマンドを使用します。

モジュール パワーダウンのモニタリング

モジュールのパワーダウンをモニタする例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorPoweroff
switch(config-applet)#
switch(config-applet)# description "Monitors module power down."
switch(config-applet)# event cli match "conf t; poweroff *"
switch(config-applet)# action 1.0 cli show module
switch(config)# copy running-config startup-config
```

ロールバックを開始するトリガーの追加

ロールバックを開始するトリガーを追加する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#
switch(config)# event manager applet rollbackTrigger
switch(config-applet)#
switch(config-applet)# description "Rollback trigger."
switch(config-applet)# event cli match "rollback *"
switch(config-applet)# action 1.0 cli copy running-config bootflash:last_config
switch(config)# copy running-config startup-config
```

メジャーしきい値を上書き（ディセーブル）するコンフィギュレーション例

メジャーしきい値に達したときにシャットダウンを防ぐ方法

メジャーしきい値に達したことによるシャットダウンを防ぐ例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

1 つの不良センサーをディセーブルにする方法

センサー 3 で障害が発生した場合（他のセンサーに影響なし）に、モジュール 2 でセンサー 3 だけをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

複数の不良センサーをディセーブルにする方法

モジュール 2 のセンサー 5、6、7 で障害が発生した場合（他のセンサーに影響なし）に、これらのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 5 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 6 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
```

```
switch(config)# end
```

モジュール全体の上書き（ディセーブル）

誤動作するモジュール 2 をディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

複数のモジュールおよびセンサーの上書き（ディセーブル）

誤動作するモジュール 2 のセンサー 3、4、7 とモジュール 3 のすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

1 つのセンサーをイネーブルにして、すべてのモジュールの残りのセンサーをすべてディセーブルにする方法

モジュール 9 のセンサー 4 を除くすべてのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
```

```
switch(config-applet)# end
```

複数のセンサーをイネーブルにして、すべてのモジュールの残りのセンサーをすべてディセーブルにする方法

モジュール 9 のセンサー 4、6、7 を除くすべてのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 6 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

1 つのモジュールのすべてのセンサーをイネーブルにして、残りのモジュールのすべてのセンサーをディセーブルにする方法

モジュール 9 のすべてのセンサーを除く残りのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

モジュールのセンサーを組み合わせてイネーブルにして、残りのモジュールのすべてのセンサーをディセーブルにする方法

モジュール 2 のセンサー 3、4、7 とモジュール 3 のすべてのセンサーを除くすべてのモジュールのすべてのセンサーをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
```

```
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet5 override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

ファントレイ取り外しのためのシャットダウンを上書き（ディセーブル）するコンフィギュレーション例

1つまたは複数のファントレイを取り外すためのシャットダウンの上書き（ディセーブル）

1つまたは複数（またはすべて）のファン トレイを取り外せるように、シャットダウンをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

指定したファントレイを取り外すためのシャットダウンの上書き（ディセーブル）

指定したファン トレイ（ファン トレイ 3）を取り外せるように、シャットダウンをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config) no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

指定した複数のファントレイを取り外すためのシャットダウンの上書き（ディセーブル）

指定した複数のファン トレイ（ファン トレイ 2、3、4）を取り外せるように、シャットダウンをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# end
```

デフォルト コンフィギュレーションに戻す例を示します。

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

1 つを除くすべてのファン トレイを取り外すためのシャットダウンの上書き（ディセーブル）

1台（ファントレイ2）を除くすべてのファントレイを取り外せるように、シャットダウンをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

ファントレイの指定したセットを除くファントレイを取り外すためのシャットダウンの上書き（ディセーブル）

指定したファン トレイのセット（ファン 2、3、4）を除くファン トレイを取り外せるように、シャットダウンをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2,3,4 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

ファントレイのセットから1台を除くすべてのファントレイを取り外すためのシャットダウンの上書き（ディセーブル）

指定したファン トレイのセット（ファン トレイ 2、3、4）の 1 台を除くすべてのファン トレイを取り外せるように、シャットダウンをディセーブルにする例を示します。

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

補足ポリシーを作成するコンフィギュレーション例

ファン トレイが存在しないイベントの補足ポリシーの作成

event fanabsent コマンドを使用して、補足ポリシーを作成する例を示します。

[**no**] **event fanabsent** [**fan***fan-tray-number*] **time***time-interval*

ファン トレイ 1 が 60 秒間存在しない場合に、デフォルトのポリシーに加えて、ポリシー myappletname とアクション 3 を実行する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event fanabsent fan 1 time 60
switch(config-applet)# action 3 cli "show env fan"
switch(config-applet)# end
```

温度しきい値イベントの補足ポリシーの作成

event temperature コマンドを使用して、補足ポリシーを作成する例を示します。

[**no**] **event temperature** [**mod***module-number*] [**sensor***sensor-number*] **threshold** {**major** | **minor** | **any**}

モジュール 2 のセンサー 3 で温度がマイナーしきい値を超えた場合に、デフォルトのポリシーに加えて、ポリシー myappletname とアクション 1 を実行する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event temperature module 2 sensor 3 threshold minor
switch(config-applet)# action 1 cli "show environ temperature"
switch(config-applet)# end
```

電力のバジェット超過ポリシーのコンフィギュレーション例

電力のバジェット超過ポリシーは、使用可能な電力がゼロ未満に低下し、前に起動されたモジュールを起動状態で維持できなくなった場合に開始します。デフォルトのアクションでは、ユーザに電力のバジェット超過が発生したことを通知する syslog を出力します。

利用可能な電力が赤（負）のゾーンから回復するまでモジュールの電源を落とす追加アクションをイネーブルにできます。

モジュールのシャットダウン

モジュールを指定しない場合、電力のバジェット超過シャットダウンはスロット 1 から始まり、電力が赤（負）のゾーンから回復するまでモジュールをシャットダウンします。空のスロットやスーパーバイザ、スタンバイ スーパーバイザ、スパイン、クロスバーを含むスロットは飛ばされます。

利用可能な電力がゼロ未満に低下した場合に、モジュール 1 からモジュールをシャットダウンする例を示します。

```
switch# configure terminal
switch(config)# event manager applet <myappletname4a> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 4 overbudgetshut
switch(config-applet)# end
```

指定された一連のモジュールのシャットダウン

電力のバジェット超過アクションによって、電力が赤（負）のゾーンから回復するまでシャットダウンされるモジュールのリストを指定できます。空のスロットやスーパーバイザ、スタンバイ スーパーバイザ、スパイン、クロスバーを含むスロットは飛ばされます。

利用可能な電力がゼロ未満に低下した場合に、指定されたモジュールのリスト（1、2、7、8）からモジュールをシャットダウンする例を示します。

```
switch# configure terminal
switch(config)# event manager applet <myappletname4b> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 5 overbudgetshut module 1,2,7,8
switch(config-applet)# end
```

シャットダウンするモジュールを選択するコンフィギュレーション例

デフォルトでシャットダウンに非上書きモジュールを選択するポリシーの使用

メジャーしきい値を超えた場合に、デフォルトで非上書きモジュールをシャットダウンするよう選択するポリシーを使用する例を示します。

```
switch# configure terminal
switch(config)# event manager applet my5a1 override __pfm_tempev_major
switch(config-applet)# end
```

```
switch# configure terminal
switch(config)# event manager applet my5a2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 4 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

シャットダウンに非上書きモジュールを選択するパラメータ置き換えの使用

メジャーしきい値を超えた場合に、パラメータの置き換えを使用してシャットダウンする非上書きモジュールを選択する例を示します。

```
switch# configure terminal
switch(config)# event manager applet my5b1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet my5b2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 8 threshold major
switch(config-applet)# action 6 forceshut module my_module_list reset "temperature-sensor
policy trigger"
switch(config-applet)# end
```

イベント マネージャ パラメータを作成するには、**event manager environment** コマンドを使用します。イベント マネージャ パラメータの値を表示するには、**show event manager environment all** コマンドを使用します。

活性挿抜イベントのコンフィギュレーション例

活性挿抜イベント（OIR）には、デフォルトのポリシーがありません。

event oir コマンドを使用して、OIR イベントを設定する例を示します。

event oir*device-typeevent-type* [*device-number*]

device-type は、**fan**、**module** または **powersupply** です。

event-type は、**insert**、**remove**、または **anyoir**（装着または取り外し）です。

オプションの *device-number* では 1 台のデバイスを指定します。省略すると、すべてのデバイスが選択されます。

装着イベントを設定する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module insert
switch(config-applet)# action 1 syslog priority critical msg "OIR insert event: A Module
is inserted"
```

取り外しイベントを設定する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module remove
switch(config-applet)# action 1 syslog priority critical msg "OIR remove event: A Module
is removed"
```

ユーザ syslog を生成するコンフィギュレーション例

action syslog コマンドを使用して、ユーザ syslog を生成する例を示します。

```
switch# configure terminal
switch(config)# event manager applet myoir
switch(config-applet)# event oir module remove
switch(config-applet)# action 1 syslog priority critical msg "Module is removed"
```

このイベントが発生すると、次の syslog が生成されます。

```
switch(config)# 2013 May 20 00:08:27 p1b-57 %$ VDC-1 %$ %EEM_ACTION-2-CRIT: "Module is
removed"
```

Syslog メッセージをモニタする設定例

次に、スイッチからの Syslog メッセージをモニタする例を示します。

```
switch(config)# event manager applet a1
switch(config-applet)# event syslog occurs 6 period 4294967 pattern "authentication failed"
```

このイベントがトリガーされると、ポリシーで定義されているアクションが実行されます。

SNMP 通知のコンフィギュレーション例

SNMP OID のポーリングによる EEM イベントの生成

スイッチの CPU 使用率を問い合わせるには、SNMP オブジェクト ID（OID）CISCO-SYSTEM-EXT-MIB::cseSysCPUUtilization が使用されます。

```
cseSysCPUUtilization OBJECT-TYPE
SYNTAX Gauge32 (0..100 )
UNITS "%"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The average utilization of CPU on the active supervisor."
::= { ciscoSysInfoGroup 1 }
```

10 秒間隔でポーリングされ、しきい値が 95 % の SNMP OID を使用する例を示します。

```
switch# configure terminal
switch(config)# event manager applet test_policy
switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.305.1.1.1.0 get-type exact entry-op
gt entry-val 95 exit-op lt exit-val 90 poll-interval 10
```

イベント ポリシーのイベントへの応答で SNMP 通知を送信

このタイプのコンフィギュレーションを使用して、重大なイベント トリガーで SNMP 通知を生成できます。

イベント マネージャのアプレット コンフィギュレーション モードからイベントに対して SNMP 通知を送信する例を示します。

```
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "CPU Hogging
 at switch1"
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "Port Failure
 eth9/1"
```

このコンフィギュレーションでは、スイッチから SNMP ホストに SNMP 通知（トラップ）を行います。SNMP ペイロードには、ユーザ定義フィールド intdata1、intdata2、および strdata の値が含まれます。

ポート トラッキングのコンフィギュレーション例

1 つのポートの状態を別のポートの状態と一致させるように設定する例を示します（ポート トラッキング）。

イーサネット インターフェイス 1/2 によるイーサネット インターフェイス 3/23 のポート トラッキングを設定するには、次のステップに従います。

手順の概要

1. イーサネット インターフェイス 3/23 のステータスを追跡するオブジェクトを作成します。
2. トラッキング オブジェクトがシャットダウンされたらイーサネット インターフェイス 1/2 をシャットダウンする EEM イベントを設定します。
3. イーサネット インターフェイス 3/23 が起動したらイーサネット インターフェイス 1/2 を起動する EEM イベントを設定します。

手順の詳細

ステップ 1 イーサネット インターフェイス 3/23 のステータスを追跡するオブジェクトを作成します。

例：

```
switch# configure terminal
switch(config)# track 1 interface ethernet 3/23
switch(config-track)# end
```

ステップ 2 トラッキング オブジェクトがシャットダウンされたらイーサネット インターフェイス 1/2 をシャットダウンする EEM イベントを設定します。

例：

```
switch(config)# event manager applet track_3_23_down
switch(config-applet)# event track 1 state down
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down shutting down port eth1/2
due to eth3/23 being down
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli shut
```

```
switch(config-applet)# end
```

ステップ 3 イーサネット インターフェイス 3/23 が起動したらイーサネット インターフェイス 1/2 を起動する EEM イベントを設定します。

例：

```
switch# configure terminal
switch(config)# event manager applet track_3_23_up
switch(config-applet)# event track 1 state up
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down bringing up port eth1/2 due
 to eth3/23 being up
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli no shut
switch(config-applet)# end
```

EEM によって EEM ポリシーを登録する設定例

次に、EEM によって EEM ポリシーを登録する例を示します。

基本的なスイッチ設定：

```
event manager applet vpc_check_peer_at_startup
event track 101 state up
action 1.0 cli copy bootflash:eem/user_script_policies/load_schedules running-config

feature scheduler

!!## 2 x dummy loopbacks are required ##!!
interface loopback 101
interface loopback 102

track 1 list boolean or
object 13
object 12
object 102
track 2 list boolean and
object 13
object 12
track 12 interface Ethernet 2/24 line-protocol
track 13 interface port-channel 3000 line-protocol
track 101 interface loopback 101 line-protocol
track 102 interface loopback 102 line-protocol
```

（注） この例では、ポート チャネル 3000 が vPC ピア リンクで、イーサネット 2/24 が vPC キープアライブ リンクです。

ブートフラッシュに次のファイルをコピーする必要があります。

- スーパーバイザのブートフラッシュに作成する必要がある、/eem/user_script_policies と呼ばれるディレクトリ。
- 次の 5 つのファイルを上記のディレクトリに作成してロードする必要があります。
  - load_schedules

- remove_vpc_if_peer_failed
- clean_up
- unload_schedules
- restore_vpc

load_schedules ファイルの設定

```
feature scheduler

configure terminal
scheduler job name vpc_check
configure terminal
event manager policy remove_vpc_if_peer_failed
end

configure terminal
scheduler job name clean_up
configure terminal
event manager policy clean_up
end

configure terminal
scheduler job name trigger
configure terminal
interface loopback 102
shutdown
no shutdown
end

configure terminal
scheduler schedule name load_vpc_check
time start +00:00:04
job name vpc_check

scheduler schedule name trigger_vpc_check
time start +00:00:05
job name trigger

scheduler schedule name load_clean_up
time start +00:00:08
job name clean_up


scheduler schedule name trigger_clean_up
time start +00:00:10
job name trigger
```

remove_vpc_if_peer_failed ファイルの設定

```
event manager applet remove_vpc_if_peer_failed
event track 1 state down
action 1.0 cli show run vpc > bootflash://sup-active/eem/user_script_policies/vpc_saved.cfg
action 2.0 cli show run vpc > bootflash://sup-standby/eem/user_script_policies/vpc_saved.cfg
action 3.0 cli configure terminal
action 4.0 cli no feature vpc
action 5.0 syslog msg severity alert "##### WARNING!!!! PEER SWITCH FAILED TO COME ONLINE.
 VPC CONFIG REMOVED #####"
action 6.0 cli event manager policy restore_vpc
action 7.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 8.0 cli no event manager applet remove_vpc_if_peer_failed
action 9.0 cli end
```

clean_up ファイルの設定

```
event manager applet clean_up
event track 102 state up
action 1.0 cli configure terminal
```

```
action 2.0 cli no event manager applet remove_vpc_if_peer_failed
action 3.0 cli copy bootflash:eem/user_script_policies/unload_schedules running
action 4.0 cli no event manager applet clean_up
action 5.0 end
```

unload_schedules ファイルの設定

```
no scheduler schedule name load_vpc_check
no scheduler schedule name trigger_vpc_check
no scheduler schedule name load_clean_up
no scheduler schedule name trigger_clean_up
no scheduler job name vpc_check
no scheduler job name trigger
no scheduler job name clean_up
```

restore_vpc ファイルの設定

```
event manager applet restore_vpc
event track 2 state up
action 1.0 cli copy bootflash:eem/user_script_policies/vpc_saved.cfg running-config
action 2.0 syslog msg severity alert "##### VPC PEER DETECTED. VPC CONFIG RESTORED #####"
action 3.0 cli configure terminal
action 4.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 5.0 cli no event manager applet restore_vpc
action 6.0 cli end
```

付 録 C

Cisco NX-OS システム管理の設定制限事項

設定の制限は、『*Cisco Nexus 9000 Series NX-OS Verified Scalability Guide*』に記載されています。

索 引

A

C

D

E

F

G

H

I

J

L

M

N

P

R

S

T

U

V

あ

い

し