Cisco NX-OS System Management コンフィギュレーション ガイド Release 4.0

Text Part Number: OL-15809-03-J

【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、米国シスコシステムズ発行ドキュメントの参考和訳です。米国サイト掲載ドキュメントとの差異が生じる場合があるため、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。

このマニュアルで使用している IP アドレスは、実際のアドレスを示すものではありません。マニュアル内の例、コマンド出力、および図は、説明のみを目的として使用されています。説明の中に実際のアドレスが使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco NX-OS System Management コンフィギュレーション ガイド Release 4.0

CONTENTS

新しい機能および変更された機能に関する情報

ここでは『*Cisco NX-OS System Management コンフィギュレーション ガイド Release 4.0*』の新しい機能および変更された機能について、リリース固有の情報を提供します。このマニュアルの最新バージョンは、シスコの Web サイトにあります。次の URL にアクセスしてください。
http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/configuration/guide/sm_nx-os_config.html

Cisco NX-OS Release 4.0 の詳細については、次の URL にアクセスし、シスコの Web サイトにある『*Cisco NX-OS Release Notes*』を参照してください。
http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/release/notes/401_nx-os_release_note.html

表 1 に、『*Cisco NX-OS System Management コンフィギュレーション ガイド Release 4.0*』の新しい機能および変更された機能の要約とともに、参照先を示します。

表 1　Release 4.0 の新しい機能および変更された機能

機能	説明	変更が行われたリリース	参照先
NTP	NTP プロトコルをディセーブルにするためのサポートを追加	4.0(3)	第 2 章「CDP および NTP の設定」
SNMP	SNMP プロトコルをディセーブルにするためのサポートを追加	4.0(3)	第 7 章「SNMP の設定」
SNMP — マルチインスタンス サポート	SNMP の拡張により、プロトコル インスタンス、VRF（Virtual Routing and Forwarding）を含め、複数の MIB コンテキストをサポート	4.0(2)	第 7 章「SNMP の設定」
新しい NetFlow **show** コマンド	NetFlow ハードウェア フローを表示するためのコマンドを追加	4.0(2)	第 13 章「NetFlow の設定」

はじめに

ここでは、『*Cisco NX-OS System Management コンフィギュレーション ガイド Release 4.0*』の対象読者、構成、および表記法について説明します。また、関連資料の入手方法についても説明します。

対象読者

このマニュアルは、NX-OS デバイスの設定およびメンテナンスを担当する、経験豊富なネットワーク管理者が対象です。

マニュアルの構成

このマニュアルは、次の章で構成されています。

タイトル	説明
第 1 章「概要」	このマニュアルで扱う機能の概要を説明します。
第 2 章「CDP および NTP の設定」	CDP（シスコ検出プロトコル）および NTP（ネットワーク タイム プロトコル）の設定方法について説明します。
第 3 章「システム メッセージ ロギングの設定」	システム メッセージ ロギングの設定方法について説明します。
第 4 章「Smart Call Home の設定」	Smart Call Home 機能を設定して、重要なシステム イベントを E メールで通知する方法について説明します。
第 5 章「ロールバックおよび Session Manager の設定」	ロールバック機能でコンフィギュレーション スナップショットを作成する方法、およびセッション マネージャを使用してバッチ モードでコマンドを適用する方法について説明します。
第 6 章「メンテナンス ジョブのスケジューリング」	バッチ コンフィギュレーション ジョブのスケジュール方法について説明します。
第 7 章「SNMP の設定」	SNMP を設定し、SNMP 通知をイネーブルにする方法について説明します。
第 8 章「RMON の設定」	RMON アラームおよびイベントを設定することによって、デバイスを監視する方法について説明します。
第 9 章「オンライン診断機能の設定」	オンライン診断機能を設定して、ソフトウェアおよびハードウェアを監視する方法について説明します。
第 10 章「Embedded Event Manager の設定」	組み込みイベント マネージャの設定方法について説明します。
第 11 章「OBFL の設定」	オンボード障害ロギングを設定して、永続ストレージに障害データを記録する方法について説明します。
第 12 章「SPAN の設定」	SPAN を設定して、ポートを出入りするトラフィックを監視する方法について説明します。
第 13 章「NetFlow の設定」	NetFlow を設定して、入力トラフィックおよび出力トラフィックの統計情報を収集する方法について説明します。
付録 A「Cisco NX-OS System Management Release 4.0 がサポートする IETF RFC」	サポート対象の IETF RFC を示します。

表記法

コマンドの説明では、次の表記法を使用しています。

表記	説明
太字	コマンドおよびキーワードは**太字**で示しています。
イタリック体	ユーザが値を指定する引数は、*イタリック体*で示しています。
[ ]	角カッコの中の要素は、省略可能です。
[x | y | z]	どれか 1 つを選択できる省略可能なキーワードは、角カッコで囲み、縦棒で区切って示しています。
ストリング	引用符を付けない一組の文字。ストリング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ストリングとみなされます。

出力例では、次の表記法を使用しています。

`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、かぎカッコ（< >）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

このマニュアルでは、次の表記法を使用しています。

（注） 「*注釈*」です。役立つ情報や、このマニュアル以外の参照資料などを紹介しています。

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

関連資料

Cisco NX-OS のマニュアルは、次の URL から入手できます。

http://www.cisco.com/en/US/products/ps9372/tsd_products_support_series_home.html

Cisco NX-OS のマニュアル セットには、次のマニュアルが含まれます。

リリース ノート

『*Cisco NX-OS Release Notes, Release 4.0*』

NX-OS コンフィギュレーション ガイド

『*Cisco NX-OS Getting Started with Virtual Device Contexts, Release 4.0*』

『*Cisco NX-OS Fundamentals Configuration Guide, Release 4.0*』

『*Cisco NX-OS Interfaces Configuration Guide, Release 4.0*』

『*Cisco NX-OS Layer 2 Switching Configuration Guide, Release 4.0*』

『*Cisco NX-OS Quality of Service Configuration Guide, Release 4.0*』

『*Cisco NX-OS Unicast Routing Configuration Guide, Release 4.0*』

『*Cisco NX-OS Multicast Routing Configuration Guide, Release 4.0*』

『*Cisco NX-OS Security Configuration Guide, Release 4.0*』

『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』

『*Cisco NX-OS Software Upgrade Guide, Release 4.0*』

『*Cisco NX-OS Licensing Guide, Release 4.0*』

『*Cisco NX-OS High Availability and Redundancy Guide, Release 4.0*』

『*Cisco NX-OS System Management Configuration Guide, Release 4.0*』

『*Cisco NX-OS XML Management Interface User Guide, Release 4.0*』

『*Cisco NX-OS System Messages Reference*』

『*Cisco NX-OS MIB Quick Reference*』

NX-OS コマンド リファレンス

『*Cisco NX-OS Command Reference Master Index, Release 4.0*』

『*Cisco NX-OS Fundamentals Command Reference, Release 4.0*』

『*Cisco NX-OS Interfaces Command Reference, Release 4.0*』

『*Cisco NX-OS Layer 2 Switching Command Reference, Release 4.0*』

『*Cisco NX-OS Quality of Service Command Reference, Release 4.0*』

『*Cisco NX-OS Unicast Routing Command Reference, Release 4.0*』

『*Cisco NX-OS Multicast Routing Command Reference, Release 4.0*』

『*Cisco NX-OS Security Command Reference, Release 4.0*』

『*Cisco NX-OS Virtual Device Context Command Reference, Release 4.0*』

『*Cisco NX-OS High Availability and Redundancy Command Reference, Release 4.0*』

『*Cisco NX-OS System Management Command Reference, Release 4.0*』

その他のソフトウェアのマニュアル

『*Cisco NX-OS Troubleshooting Guide, Release 4.0*』

マニュアルの入手方法、テクニカル サポート、およびセキュリティ ガイドライン

マニュアルの入手方法、テクニカル サポート、マニュアルに関するフィードバックの提供、セキュリティ ガイドライン、および推奨エイリアスや一般的なシスコのマニュアルについては、次の URL で、毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

シスコのテクニカル サポート

次の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。

http://www.cisco.com/en/US/support/index.html

以下を含むさまざまな作業にこの Web サイトが役立ちます。

- テクニカル サポートを受ける
- ソフトウェアをダウンロードする
- セキュリティの脆弱性を報告する、またはシスコ製品のセキュリティ問題に対する支援を受ける
- ツールおよびリソースへアクセスする
  - Product Alert の受信登録
  - Field Notice の受信登録
  - Bug Toolkit を使用した既知の問題の検索
- Networking Professionals（NetPro）コミュニティで、技術関連のディスカッションに参加する
- トレーニング リソースへアクセスする
- TAC Case Collection ツールを使用して、ハードウェアや設定、パフォーマンスに関する一般的な問題をインタラクティブに特定および解決する

Japan テクニカル サポート Web サイトでは、Technical Support Web サイト（http://www.cisco.com/techsupport）の、利用頻度の高いドキュメントを日本語で提供しています。

Japan テクニカル サポート Web サイトには、次の URL からアクセスしてください。

http://www.cisco.com/jp/go/tac

Service Request ツールの使用

Service Request ツールには、次の URL からアクセスできます。

http://www.cisco.com/techsupport/servicerequest

日本語版の Service Request ツールは次の URL からアクセスできます。

http://www.cisco.com/jp/go/tac/sr/

シスコの世界各国の連絡先一覧は、次の URL で参照できます。

http://www.cisco.com/warp/public/687/Directory/DirTAC.shtml

その他の情報の入手方法

シスコの製品、サービス、テクノロジー、ネットワーキング ソリューションに関する情報について、さまざまな資料をオンラインで入手できます。

- シスコの E メール ニュースレターなどの配信申し込みについては、Cisco Subscription Center にアクセスしてください。

  http://www.cisco.com/offer/subscribe

- 日本語の月刊 Email ニュースレター「Cisco Customer Bridge」については、下記にアクセスください。

  http://www.cisco.com/web/JP/news/cisco_news_letter/ccb/

- シスコ製品に関する変更やアップデートの情報を受信するには、Product Alert Tool にアクセスし、プロファイルを作成して情報の配信を希望する製品を選択してください。Product Alert Tool には、次の URL からアクセスできます。

  http://tools.cisco.com/Support/PAT/do/ViewMyProfiles.do?local=en

- 『*Cisco Product Quick Reference Guide*』はリファレンス ツールで、パートナーを通じて販売されている多くのシスコ製品に関する製品概要、主な機能、製品番号、および簡単な技術仕様が記載されています。『*Cisco Product Quick Reference Guide*』を発注するには、次の URL にアクセスしてください。

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

- ネットワークの運用面の信頼性を向上させることのできる最新の専門的サービス、高度なサービス、リモート サービスに関する情報については、Cisco Services Web サイトを参照してください。Cisco Services Web サイトには、次の URL からアクセスできます。

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

- Cisco Marketplace では、さまざまなシスコの書籍、参考資料、マニュアル、ロゴ入り商品を提供しています。Cisco Marketplace には、次の URL からアクセスできます。

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

- DVD に収録されたシスコの技術マニュアル（Cisco Product Documentation DVD）は、Product Documentation Store で発注できます。Product Documentation Store には、次の URL からアクセスできます。

  http://www.cisco.com/go/marketplace/docstore

- 日本語マニュアルの DVD は、マニュアルセンターから発注できます。マニュアルセンターには下記よりアクセスください。

  http://www.cisco.com/japanese/warp/public/3/jp/service/manual_j/manual_center/index.shtml

- Cisco Press では、ネットワーク、トレーニング、認定関連の出版物を発行しています。Cisco Press には、次の URL からアクセスできます。

  http://www.ciscopress.com

- 日本語のシスコプレスの情報は以下にアクセスください。

  http://www.seshop.com/se/ciscopress/default.asp

- 『*Internet Protocol Journal*』は、インターネットおよびイントラネットの設計、開発、運用を担当するエンジニア向けに、シスコが発行する季刊誌です。『*Internet Protocol Journal*』には、次の URL からアクセスできます。

  http://www.cisco.com/ipj

- 『*What's New in Cisco Product Documentation*』は、シスコ製品の最新マニュアル リリースに関する情報を提供するオンライン資料です。毎月更新されるこの資料は、製品カテゴリ別にまとめられているため、目的の製品マニュアルを見つけることができます。

  http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

- シスコの Web サイトの各国語版へは、次の URL からアクセスしてください。

  http://www.cisco.com/public/countries_languages.shtml

CHAPTER 1

概要

この章では、デバイスの監視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。

ここでは、次の内容を説明します。

Cisco NX-OS デバイス設定方式

デバイスは、直接ネットワーク設定方式、すなわち DCNM（データセンター ネットワーク管理）サーバが提供する Web サービスを使用して設定できます。

図 1-1 に、ネットワーク ユーザが使用できるデバイス設定方式を示します。

図 1-1　Cisco NX-OS デバイスの設定方式

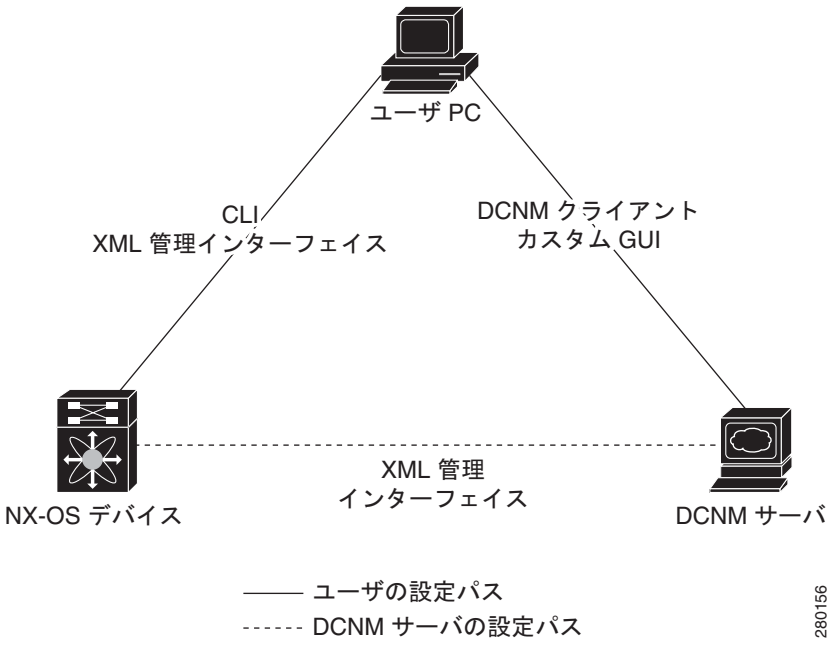


表 1-1 に、設定方式と詳細が記載されている資料を示します。

表 1-1　設定方式および参考資料

設定方式	マニュアル
SSH[1]、Telnet セッションまたはコンソール ポートから CLI	『*Cisco NX-OS Fundamentals Configuration Guide*』
XML 管理インターフェイス	『*Cisco NX-OS XML Management Interface User Guide*』
DCNM クライアント	『*Cisco DCNM Fundamentals Configuration Guide*』
ユーザ定義の GUI	『*Cisco DCNM Web Services API Programmer Guide*』

1. SSH（セキュア シェル）

ここでは、次の内容について説明します。

- CLI または XML 管理インターフェイスによる設定（p.1-3）
- DCNM または カスタム GUI による設定（p.1-3）

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 CLI（コマンドライン インターフェイス）または XML 管理インターフェイスを使用すると、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 ポートから CLI — SSH セッション、Telnet セッション、またはコンソール ポートを使用することによって、デバイスを設定できます。SSH では、デバイスへのセキュアな接続が提供されます。CLI コマンド リファレンスは、機能別に編成されています。詳細については、『*Cisco 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することによっ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。

DCNM または カスタム GUI による設定

次のように DCNM クライアントを使用することによって、または独自の GUI から Cisco NX-OS デバイスを設定できます。

- DCNM クライアント — DCNM クライアントを使用することによってデバイスを設定できます。DCNM クライアントはユーザのローカル PC 上で動作し、DCNM サーバの Web サービスを使用します。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。DCNM クライアントの詳細については、『*Cisco DCNM Fundamentals Configuration Guide*』を参照してください。
- カスタム GUI — 独自の GUI を作成すると、DCNM サーバ上の DCNM Web サービス API（アプリケーション プログラム インターフェイス）を使用してデバイスを設定できます。SOAP プロトコルを使用して、DCNM サーバと XML ベースの設定メッセージを交換します。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。カスタム GUI 作成の詳細については、『*Cisco DCNN Web Services API Programmer Guide*』を参照してください。

システム メッセージ

システム メッセージ ロギングを使用すると、システム プロセスが生成するメッセージの宛先を制御し、重大度に基づいてメッセージ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。

システム メッセージ設定の詳細については、第 3 章「システム メッセージ ロギングの設定」を参照してください。

Call Home

Call Home は重要なシステム イベントを E メールで通知します。Cisco NX-OS は豊富なメッセージ フォーマットを提供するので、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。この機能を使用すると、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、NOC（ネットワーク オペレーティング センター）に E メールを送信したり、Cisco Smart Call Home サービスを使用して TAC でケースを自動作成したりできます。

Call Home 設定の詳細については、第 4 章「Smart Call Home の設定」を参照してください。

ロールバックおよび Session Manager

ロールバック機能を使用すると、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、ロールバックによっ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、設定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。

ロールバックおよび Session Manager の設定については、第 5 章「ロールバックおよび Session Manager の設定」を参照してください。

コマンド スケジューラ

コマンド スケジューラを使用すると、ジョブ（一連の CLI コマンド）または複数のジョブをその後の指定した時刻にスケジューリングできます。Cisco NX-OS は将来の指定された時刻にジョブを 1 回だけ実行するか、または定期的な間隔で実行します。

この機能を使用すると、QoS（Quality Of Service）ポリシーの変更、データのバックアップ、コンフィギュレーションの保存などのジョブをスケジューリングできます。コマンド スケジューラの詳細については、第 6 章「メンテナンス ジョブのスケジューリング」を参照してください。

SNMP

SNMP（簡易ネットワーク管理プロトコル）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 はネットワーク デバイスの監視や管理に使用される、標準化されたフレームワークと共通言語を提供します。

SNMP の設定については、第 7 章「SNMP の設定」を参照してください。

RMON

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、IETF（インターネット技術特別調査委員会）の標準モニタリング仕様です。Cisco NX-OS は Cisco NX-OS デバイスを監視できるように、RMON アラーム、イベント、およびログをサポートします。

RMON の設定については、第 8 章「RMON の設定」を参照してください。

オンライン診断

Cisco GOLD（Generic Online Diagnostics）では、複数のシスコ プラットフォームにまたがるオペレーションを診断するため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。1 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提供し、診断テストの結果に応じて適切な対策を実行します。

オンライン診断機能の設定については、第 9 章「オンライン診断機能の設定」を参照してください。

EEM

EEM（Embedded Event Manager）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。

EEM の設定については、第 10 章「Embedded Event Manager の設定」を参照してください。

OBFL

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 OBFL（On-Board Failure Logging）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、障害モジュールの事後分析に役立ちます。OBFL の設定については、第 11 章「OBFL の設定」を参照してください。

SPAN

Ethernet SPAN（スイッチド ポート アナライザ）を設定すると、デバイスの入出力トラフィックを監視できます。これら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。

SPAN の設定については、第 12 章「SPAN の設定」を参照してください。

NetFlow

NetFlow は入力 IP パケットと出力 IP パケットの両方について、パケット フローを識別し、各パケット フローに基づいて統計情報を提供します。NetFlow のためにパケットやネットワーキング デバイスの変更が必要になることはありません。

NetFlow の設定については、第 13 章「NetFlow の設定」を参照してください。

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。各機能の詳細について、『*Cisco NX-OS Troubleshooting Guide, Release 4.0*』を参照してください。

CHAPTER 2

CDP および NTP の設定

この章では、デバイス上で CDP（シスコ検出プロトコル）および NTP（ネットワーク タイム プロトコル）を設定する方法について説明します。

ここでは、次の内容を説明します。

CDP および NTP の概要

ここでは、次の内容について説明します。

CDP の概要

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、メディアにもプロトコルにも依存しないプロトコルです。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 レイヤ上に限定されます。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最低 1 つは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 情報をディスモジュール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 TLV フィールドがアドバタイズされます。

- Device ID
- Address
- Port ID
- Capabilities
- Version
- Platform
- Native VLAN
- Full/Half Duplex
- MTU
- SysName
- SysObjectID
- Management Address
- Physical Location

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、次の URL にアクセスして、『*Cisco NX-OS Layer 2 Switching Configuration Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/layer2/configuration/guide/l2_nx-os_book.html

NTP の概要

NTP は、分散している一連のタイム サーバおよびクライアント間で、計時を同期させます。この同期によって、複数のネットワーク デバイスからシステム ログおよびその他の時刻特定イベントを受信したときに、イベントを相互に関連付けることができます。

NTP ではトランスポート プロトコルとして、UDP（ユーザ データグラム プロトコル）を使用します。すべての NTP 通信で UTC（協定世界時）規格を使用します。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、アトミック クロックなど、信頼できる時刻源から時刻を受信します。NTP はこの時刻をネットワークを介して配布します。NTP はきわめて効率的です。毎分 1 パケットだけで、2 台のマシンが相互に 1 ミリ秒以内で同期します。

NTP では層（stratum）を使用して、ネットワーク デバイスが正規の時刻源から NTP ホップ数にしてどれだけ離れているかを表します。Stratum 1 タイム サーバは、正規の時刻源（アトミック クロックなど）が直接接続されています。Stratum 2 の NTP サーバは、Stratum 1 NTP サーバから NTP を使用して時刻を受信し、それによって正規の時刻源に接続します。

NTP は正確な時刻を維持している可能性のあるネットワーク デバイスへの同期を回避します。また、順番通りに同期しないシステムには、同期しません。NTP は複数のネットワーク デバイスから伝えられた時刻を比較し、時刻が他と大きく異なっているネットワーク デバイスには、下位の層であっても同期しません。

Cisco NX-OS は、Stratum 1 サーバとしては動作できません。したがって、ラジオ クロックまたはアトミック クロックには接続できません。インターネット上で利用できる、パブリックな NTP サーバに由来するタイム サービスをネットワークに使用することを推奨します。

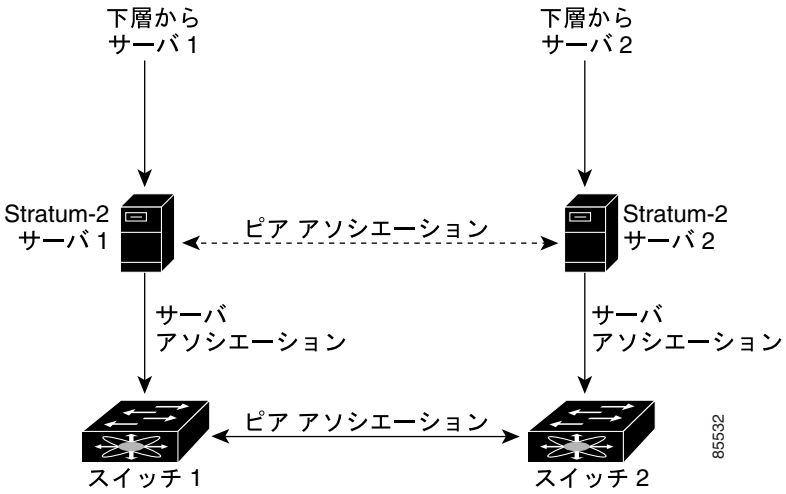
ネットワークがインターネットから切り離されている場合、Cisco NX-OS ではネットワーク デバイスが実際には他の方法で時刻を決定している場合でも、NTP によって同期しているものとして動作するように、ネットワーク デバイスを設定できます。その後、NTP を使用して、そのネットワーク デバイスに他のネットワーク デバイスを同期させることができます。

NTP ピア

NTP を使用すると、2 つのネットワーキング デバイス間にピア関係を設定できます。ピアはそのままで時刻を提供することも、または NTP サーバに接続することもできます。ローカル デバイスとリモート ピアの両方がそれぞれ異なる NTP サーバに接続すると、NTP サービスの信頼性が高くなります。ローカル デバイスはピアから得た時刻を使用することによって、接続先の NTP サーバが故障した場合でも、正確な時刻を維持できます。

図 2-1 に、2 台の NTP Stratum 2 サーバおよび 2 台のスイッチからなるネットワークを示します。

図 2-1　NTP ピアおよびサーバのアソシエーション



この構成では、スイッチ 1 およびスイッチ 2 が NTP サーバです。スイッチ 1 は Stratum-2 サーバ 1 を使用し、スイッチ 2 は Stratum-2 サーバ 2 を使用します。Stratum-2 サーバ 1 が故障した場合、スイッチ 1 はスイッチ 2 とのピア アソシエーションによって正確な時刻を維持します。

ハイ アベイラビリティ

Cisco NX-OS は、CDP および NTP のステートレス 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、Cisco NX-OS は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、次の URL にアクセスして、『*Cisco NX-OS High Availability and Redundancy Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/high_availability/configuration/guide

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。

仮想化サポート

Cisco NX-OS は、VDC（Virtual Device Context; 仮想デバイス コンテキスト）ごとに 1 インスタンスずつ、複数の CDP インスタンスをサポートしますが、NTP インスタンスに関してサポートするのは、プラットフォーム全体で 1 つだけです。NTP はデフォルト VDC で設定する必要があります。デフォルトでは、特に別の VDC を設定しない限り、Cisco NX-OS によりデフォルト VDC が使用されます。VDC の詳細については、次の URL にアクセスして、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

CDP および NTP 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
NX-OS	CDP および NTP 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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、次の URL にアクセスして、『*Cisco NX-OS Licensing Guide*』を参照してください。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/licensing/configuration/guide/nx-os_licensing.html

CDP および NTP の前提条件

CDP および NTP を使用する前提条件は、次のとおりです。

- NTP を設定する場合は、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 はデフォルト VDC で設定する必要があります。
- デフォルト VDC 以外の VDC で NTP を設定することはできません。

VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始する必要があります。次の URL にアクセスし、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

設定時の注意事項および制約事項

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。

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。

- 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（すなわち、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。

CDP および NTP の設定

ここでは、次の内容について説明します。

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

CDP 機能のイネーブルまたはディセーブル

CDP はデフォルトで、デバイス上でイネーブルになります。デバイス上で CDP をディセーブルにしたあとで、再び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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***feature cdp***
3. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**feature cdp** **例：** `switch(config)# feature cdp`	デバイス全体で CDP 機能をイネーブルにします。CDP はデフォルトでイネーブルです。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

CDP 機能をディセーブルにし、関連付けられた設定をすべて削除するには、***no feature cdp*** コマンドを使用します。

コマンド	目的
no feature cdp **例：** `switch(config)# no feature cdp`	デバイス全体で CDP をディセーブルにして、関連付けられたすべての設定を削除します。

CDP 機能をイネーブルにする例を示します。

```
switch# config t
switch(config)# feature cdp
```

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。

操作の前に

CDP がイネーブルになっていることを確認します（「CDP 機能のイネーブルまたはディセーブル」[p.2-6] を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **interface** *interface-type slot/port*
3. **cdp enable**
4. ***show*** **cdp interface** *interface-type slot/port*
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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-type 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。

	コマンド	目的
ステップ 3	**cdp enable** **例：** `switch(config-if)# cdp enable`	このインターフェイスで CDP をイネーブルにします。CDP はデフォルトでイネーブルです。
ステップ 4	***show* cdp interface** *interface-type 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-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

イーサネット 1/2 で CDP をディセーブルにする例を示します。

```
switch# config t
switch(config)# interface ethernet 1/2
switch(config-if)# no cdp enable
switch(config-if)# copy running-config startup-config
```

ポート チャネル 2 で CDP をイネーブルにする例を示します。

```
switch# config t
switch(config)# interface port-channel 2
switch(config-if)# cdp enable
switch(config-if)# copy running-config startup-config
```

CDP オプション パラメータの設定

CDP を変更するには、グローバル コンフィギュレーション モードで次のオプション コマンドを使用します。

コマンド	目的
cdp advertise {***v1*** | ***v2***} **例：** `switch(config)# cdp advertise v1`	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。
cdp format device-id {***mac-address*** | ***other*** | ***serial-number***} **例：** `switch(config)# cdp format device-id mac-address`	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 • other — シャーシのシリアル番号 • serial-number — シャーシのシリアル番号 /OUI（組織固有識別子） デフォルトは other です。
cdp holdtime *seconds* **例：** `switch(config)# cdp holdtime 150`	CDP ネイバー情報をディスモジュールするまでに保持する時間を設定します。値の範囲は 10 ～ 255 秒です。デフォルト値は 180 秒です。
cdp timer *seconds* **例：** `switch(config)# cdp timer 50`	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。値の範囲は 5 ～ 254 秒です。デフォルト値は 60 秒です。

NTP プロトコルのイネーブルまたはディセーブル

NTP はデフォルトで、デバイス上でイネーブルになります。デバイス上で NTP をディセーブルにしたあとで、再びイネーブルにできます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***ntp enable***
3. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**ntp enable** **例：** `switch(config)# ntp enable`	デバイス全体で NTP プロトコルをイネーブルまたはディセーブルにします。NTP はデフォルトでイネーブルです。
ステップ 3	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

NTP プロトコルをディセーブルにするには、***no ntp enable*** コマンドを使用します。

コマンド	目的
no ntp enable **例：** `switch(config)# no ntp enable`	デバイス上で NTP プロトコルをディセーブルにします。

NTP プロトコルをディセーブルにする例を示します。

```
switch# config t
switch(config)# no ntp enable
```

NTP サーバおよびピアの設定

NTP を設定するには、IPv4 アドレス、IPv6 アドレス、または DNS（ドメイン ネーム サーバ）名を使用します。

操作の前に

デフォルトの VDC を使用していることを確認します（または、**switchback** コマンドを使用します）。

手順概要

1. **config t**
2. **ntp server** {*ip-address* | *ipv6-address* | *dns-name*}
3. **ntp peer** {*ip-address* | *ipv6-address* | *dns-name*}
4. ***show* ntp peers**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**ntp server** {*ip-address* | *ipv6-address* | *dns-name*} **例：** `switch(config)# ntp server 192.0.2.10`	サーバとのアソシエーションを作成します。
ステップ 3	**ntp peer** {*ip-address* | *ipv6-address* | *dns-name*} `switch(config)# ntp peer 2001:0db8::4101`	ピアとのアソシエーションを作成します。複数のピア アソシエーションを指定できます。
ステップ 4	**show ntp peers** **例：** `switch(config)# show ntp peers`	（任意）設定済みの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-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

NTP サーバおよびピアを設定する例を示します。

```
switch# config t
switch(config)# ntp server 192.0.2.10
switch(config# ntp peer 2001:0db8::4101
```

CDP および NTP の設定確認

CDP 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* 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** *interface-type slot/port*	CDP インターフェイスのステータスを表示します。
***show* cdp neighbors** {**device-id** | **interface** *interface-type slot/port*} [**detail**]	CDP ネイバーのステータスを表示します。**device-id** キーワードがサポートされるのは、Cisco NX-OS Release 4.0(2) 以降です。
***show* cdp traffic interface** *interface-type slot/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。

NTP 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* 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 statistics {**io** | **local** | **memory** | **peer** {*ip-address* | *dns-name*}	NTP 統計を表示します。
show ntp status	NTP 配信ステータスを表示します。
show ntp timestamp status	タイムスタンプ チェックがイネーブルかどうかを表示します。

NTP セッションを削除するには、***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。

CDP および NTP の設定例

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。

```
config t
feature cdp
cdp timer 50
cdp holdtime 100
```

NTP サーバの設定例を示します。

```
config t
ntp server 192.0.2.10
```

デフォルト設定

表 2-1 に、CDP および NTP パラメータのデフォルト設定を示します。

表 2-1　デフォルトの CDP および NTP パラメータ

パラメータ	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	バージョン 2
CDP device ID	シリアル番号
CDP timer	60 秒
CDP hold timer	180 秒
NTP	ディセーブル

その他の関連資料

CDP および NTP の実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.2-13）
- MIB（p.2-13）

関連資料

関連項目	マニュアル名
CDP および NTP の CLI コマンド	『*Cisco NX-OS System Management Command Reference, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html
VDC および VRF	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

MIB

MIB	MIB のリンク
• CISCO-CDP-MIB • CISCO-NTP-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

CDP および NTP 機能の履歴

表 2-2 に、この機能のリリース履歴を示します。

表 2-2　CDP および NTP 機能の履歴

機能名	リリース	機能情報
NTP プロトコル	4.0(3)	NTP プロトコルをディセーブルにする機能を追加

CHAPTER 3

システム メッセージ ロギングの設定

この章では、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。

この章では、次の内容について説明します。

システム メッセージ ロギングの概要

システム メッセージ ロギングを使用すると、システム プロセスが生成するメッセージの宛先を制御し、重大度に基づいてメッセージ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。

デバイスはデフォルトで、端末セッションにメッセージを出力します。端末セッションへのロギングの設定については、「端末セッションへのシステム メッセージ ロギングの設定」（p.3-4）を参照してください。

デバイスはデフォルトで、システム メッセージをログ ファイルに記録します。ファイルへのロギングの設定については、「ファイルへのシステム メッセージ ロギングの設定」（p.3-6）を参照してください。

表 3-1 で、システム メッセージに使用する重大度について説明します。重大度を設定すると、そのレベルとそれより下位レベルのメッセージが出力されます。

表 3-1　システム メッセージの重大度

レベル	説明
0 — 緊急事態	システムは使用不能
1 — アラート	即時対処が必要
2 — クリティカル	クリティカル条件
3 — エラー	エラー条件
4 — 警告	警告条件
5 — 通知	正常だが重要な条件
6 — 情報	情報目的のみのメッセージ
7 — デバッグ	デバッグ時限定の表示

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 へのロギングを設定することはできません。

メッセージを生成したファシリティとメッセージの重大度に基づいて、記録するシステム メッセージを設定できます。ファシリティについては、『*Cisco NX-OS System Management Command Reference, Release 4.0*』を参照してください。モジュールおよび重大度に基づく重大度の設定については、「記録するモジュールおよびファシリティ メッセージの設定」（p.3-7）を参照してください。

ここでは、次の内容について説明します。

- syslog サーバ（p.3-3）
- 仮想化サポート（p.3-3）

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ように設定されたリモート システム上で動作します。最大 3 つの syslog サーバを設定できます。syslog サーバの設定については、「syslog サーバの設定」（p.3-9）を参照してください。

（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。

仮想化サポート

VDC（Virtual Device Context; 仮想デバイス コンテキスト）は、一連のシステム リソースに対応する論理表現です。システム メッセージ ロギングが適用されるのは、コマンドが入力された VDC に限られます。

VDC の設定については、次の URL にアクセスして、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

システム メッセージ ロギ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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、次の URL にアクセスして、『*Cisco NX-OS Licensing Guide, Release 4.0*』を参照してください。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/licensing/configuration/guide/nx-os_licensing.html

注意事項および制約事項

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。

システム メッセージ ロギングの設定

ここでは、次の内容について説明します。

（注）　Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

端末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、端末セッションでのロギングがイネーブルです。

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **terminal monitor**
2. **config t**
3. **logging console** [*severity-level*]

   no logging console
4. **show logging console**
5. **logging monitor** [*severity-level*]

   no logging monitor
6. **show logging monitor**
7. **copy running-config startup-config**

詳細な手順

	コマンド	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	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 3	**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。重大度は表 3-1 に示したとおり、0 ～ 7 の範囲で指定できます。重大度を指定しなかった場合は、デフォルトの 2 が使用されます。
	no logging console [*severity-level*] **例：** `switch(config)# no logging console`	デバイスがコンソールにメッセージを記録できないようにします。
ステップ 4	**show logging console** **例：** `switch(config)# show logging console`	（任意）コンソール ロギングの設定を表示します。
ステップ 5	**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。この設定は、Telnet および SSH セッションに適用されます。重大度は表 3-1 に示したとおり、0 ～ 7 の範囲で指定できます。重大度を指定しなかった場合は、デフォルトの 2 が使用されます。
	no logging monitor [*severity-level*] **例：** `switch(config)# no logging monitor`	Telnet および SSH セッションへのメッセージ ロギングをディセーブルにします。
ステップ 6	**show logging monitor** **例：** `switch(config)# show logging monitor`	（任意）モニタ ロギングの設定を表示します。
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。

ファイルへのシステム メッセージ ロギングの設定

システム メッセージをファイルに記録するよう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。

ログ ファイルの表示および消去については、「ログ ファイルの表示および消去」（p.3-11）を参照してください。

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **logging logfile** *logfile-name severity-level* [**size** *bytes*]

   no logging logfile [*logfile-name severity-level* [**size** *bytes*]]
3. **show logging info**
4. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**logging logfile** *logfile-name severity-level* [**size** *bytes*] **例：** `switch(config)# logging logfile my_log 6`	システム メッセージを保管するログ ファイルの名前および記録する最小重大度を設定します。任意で最大ファイル サイズを指定できます。デフォルトの重大度は 5、ファイル サイズは 10485760 です。重大度は表 3-1 のとおりです。ファイル サイズの範囲は 4096 ～ 10485760 バイトです。
	no logging logfile [*logfile-name severity-level* [**size** *bytes*]] **例：** `switch(config)# no logging logfile`	ログ ファイルへのロギングをディセーブルにします。
ステップ 3	**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 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。

記録するモジュールおよびファシリティ メッセージの設定

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 ユニットを設定できます。

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **logging module** [*severity-level*]
   no logging module
3. **show logging module**
4. **logging level** *facility severity-level*
   no logging level [*facility severity-level*]
5. **show logging level** [*facility*]
6. **logging timestamp** {**microseconds** | **milliseconds** | **seconds**}
   no logging timestamp {**microseconds** | **milliseconds** | **seconds**}
7. **show logging timestamp**
8. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**logging module** [*severity-level*] **例：** `switch(config)# logging module 3`	指定された重大度以上のモジュール ログ メッセージをイネーブルにします。重大度は表 3-1 に示したとおり、0 ～ 7 の範囲で指定できます。重大度を指定しなかった場合は、デフォルトの 5 が使用されます。
	no logging module [*severity-level*] **例：** `switch(config)# no logging module`	モジュール ログ メッセージをディセーブルにします。
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。

	コマンド	目的
ステップ 4	**logging level** *facility severity-level* **例：** `switch(config)# logging level aaa 2`	指定されたファシリティからの、指定された重大度以上のメッセージ ロギングをイネーブルにします。ファシリティについては、『*Cisco NX-OS System Management Command Reference, Release 4.0*』を参照してください。重大度は表 3-1 に示したとおり、0 ～ 7 の範囲で指定できます。すべてのファシリティに同じ重大度を適用する場合は、facility に **all** を使用します。デフォルトについては、**show logging level** コマンドを参照してください。
	no logging level [*facility severity-level*] **例：** `switch(config)# no logging level aaa 3`	指定されたファシリティのロギング重大度をデフォルトの重大度にリセットします。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）ファシリティ別に、ロギング レベルの設定およびシステム 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	**logging timestamp** {**microseconds** | **milliseconds** | **seconds**} **例：** `switch(config)# logging timestamp milliseconds`	ロギング タイムスタンプ ユニットを設定します。デフォルトのユニットは秒です。
	no logging timestamp {**microseconds** | **milliseconds** | **seconds**} **例：** `switch(config)# no logging timestamp milliseconds`	ロギング タイムスタンプ ユニットをデフォルトの秒にリセットします。
ステップ 7	**show logging timestamp** **例：** `switch(config)# show logging timestamp`	（任意）設定されているロギング タイムスタンプ ユニットを表示します。
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。

syslog サーバの設定

システム メッセージの記録先であるリモート システムを参照する syslog サーバを 3 つまで設定できます。

（注） 管理 VRF（Virtual Routing and Forwarding）インスタンスを使用するものとして、syslog サーバを設定することを推奨します。VRF の詳細については『*Cisco NX-OS Unicast Routing Configuration Guide, Release 4.0*』を参照してください。次の URL にアクセスしてください。
http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/unicast/configuration/guide/l3_nxos-book.html

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **logging server** *host* [*severity-level* [**use_vrf** *vrf-name*]]

   no logging server *host*
3. **show logging server**
4. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**logging server** *host* [*severity-level* [**use-vrf** *vrf-name*]]
	
	例 1：
	`switch(config)# logging server 192.0.2.253`
	
	例 2：
	`switch(config)# logging server 192.0.254.254 5 use_vrf red`
		指定のホスト名または IPv4/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。重大度は表 3-1 に示したとおり、0 ～ 7 の範囲で指定できます。デフォルトの発信ファシリティは local7 です。
		
		例 1 では、ファシリティ local 7 のすべてのメッセージを転送します。
		
		例 2 では、VRF red で重大度が 5 以下のメッセージを転送します。

	no logging server *host* **例：** `switch(config)# no logging server host`	指定されたホストに対応するロギング サーバを削除します。
ステップ 3	**show logging server** **例：** `switch(config)# show logging server`	（任意）syslog サーバの設定を表示します。

	コマンド	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。

/etc/syslog.conf ファイルに次の行を追加すると、UNIX または Linux システム上で syslog サーバを設定できます。

```
facility.level <five tab characters> action
```

表 3-2 で、ユーザが設定できる syslog フィールドについて説明できます。

表 3-2　syslog.conf の syslog フィールド

フィールド	説明
facility	メッセージの作成元。auth、authpriv、cron、daemon、kern、lpr、mail、mark、news、syslog、user、local0 ～ local7、またはすべてを表すアスタリスク（*）。これらのファシリティ指定によって、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、コンフィギュレーションを確認してください。
level	メッセージを記録する最小の重大度。debug、info、notice、warning、err、crit、alert、emerg、またはすべてを表すアスタリスク（*）を指定できます。ファシリティをディセーブルにする場合は、none を使用します。
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。

UNIX または Linux システム上で syslog サーバを設定する場合、手順は次のとおりです。

ステップ 1　/etc/syslog.conf ファイルに次の行を追加することによって、ファシリティ local7 のデバッグ メッセージをファイル /var/log/myfile.log に記録します。

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

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

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

ステップ 3　コマンド入力後に myfile.log を調べ、システム メッセージ ロギング デーモンが新しい設定変更を読み取ったかどうかを確認します。

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

ログ ファイルの表示および消去

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。

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **show logging last** *number-lines*
2. **show logging logfile** [**start-time** *yyyy mmm 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**

詳細な手順

	コマンド	目的
ステップ 1	**show logging last** *number-lines* **例：** `switch# show logging last 40`	ログ ファイルの最終行番号を表示します。最終行番号として 1 ～ 9999 を指定できます。
ステップ 2	**show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*] **例：** `switch# show logging logfile start-time 2007 nov 1 15:10:0`	ログ ファイルの中で、タイムスタンプが入力した範囲内にあるメッセージを表示します。終了時刻を入力しなかった場合は、現在時が使用されます。月のフィールドには 3 文字、年および日付のフィールドには数字を入力します。
ステップ 3	**show logging nvram** [**last** *number-lines*] **例：** `switch# show logging nvram last 10`	NVRAM 内のメッセージを表示します。表示行数を制限するには、表示する最終行番号を入力します。最終行番号として 1 ～ 100 を指定できます。
ステップ 4	**clear logging logfile** **例：** `switch# clear logging logfile`	ログ ファイルの内容を消去します。
ステップ 5	**clear logging nvram** **例：** `switch# clear logging nvram`	NVRAM に記録されているメッセージを消去します。

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 ロギングの設定を表示します。
show logging info	ロギングの設定を表示します。
show logging last *number-lines*	ログ ファイルの最終行番号を表示します。
show logging level [*facility*]	ファシリティ ロギングの重大度の設定を表示します。
show logging logfile [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	ログ ファイル内のメッセージを表示します。
show logging module	モジュール ロギングの設定を表示します。
show logging monitor	モニタ ロギングの設定を表示します。
show logging nvram [**last** *number-lines*]	NVRAM ログのメッセージを表示します。
show logging server	syslog サーバの設定を表示します。
show logging session	ロギング セッション ステータスを表示します。
show logging status	ロギング ステータスを表示します。
show logging timestamp	設定されているロギング タイムスタンプ ユニットの設定を表示します。

これらのコマンド出力のフィールドの詳細については、次の URL にアクセスし、『*Cisco NX-OS System Management Command Reference, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html

システム メッセージ ロギングの設定例

システム メッセージ ロギングの設定例を示します。

```
config t
  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 local3
  copy running-config startup-config
```

デフォルト設定

表 3-3 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 3-3　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

パラメータ	デフォルト
コンソール ロギング	重大度 2 でイネーブル
モニタ ロギング	重大度 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。重大度については『*Cisco NX-OS System Management Command Reference, Release 4.0*』を参照
タイムスタンプ ユニット	秒
syslog サーバ ロギング	ディセーブル

その他の関連資料

システム メッセージ ロギングの実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.3-13）
- 規格（p.3-13）

関連資料

関連項目	マニュアル名
システム メッセージの CLI コマンド	『*Cisco NX-OS System Management Command Reference*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html
システム メッセージ	『*Cisco NX-OS System Messages Reference*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/system_messages/reference/sl_nxos_book.html

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 4

Smart Call Home の設定

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

ここでは、次の内容を説明します。

Call Home の概要

Call Home は重要なシステム イベントを E メールで通知します。Cisco NX-OS は豊富なメッセージ フォーマットを提供するので、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。この機能を使用すると、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、NOC（ネットワーク オペレーティング センター）に E メールを送信したり、Cisco Smart Call Home サービスを使用して TAC でケースを自動作成したりできます。

ここでは、次の内容について説明します。

Call Home の概要

Call Home を使用すると、デバイスで重要なイベントが発生したときに、外部エンティティに通知できます。Call Home は*宛先プロファイル*（「宛先プロファイル」[p.4-3] を参照）で設定した複数の受信先にアラートを配信します。

Call Home にはスイッチ上で定義済みの固定アラート セットが組み込まれています（「イベント トリガー」[p.4-23] を参照）。Cisco NX-OS はこれらのアラートをアラート グループに分け、アラート グループ内のアラートが発生したときに実行する CLI コマンドを割り当てます。Cisco NX-OS は、送信する Call Home メッセージにコマンド出力を組み込みます。アラートの一覧およびアラート開始時に送信される定義済み CLI コマンド セットについては、「Call Home のアラート グループ」（p.4-3）を参照してください。

Call Home 機能がもたらす利点は、次のとおりです。

- 関連 CLI コマンドの自動実行およびコマンド出力の添付
- 下記のとおり、複数のメッセージ フォーマット オプション
  - Short Text — ポケットベルまたは印刷レポート向き。
  - Full Text — 目視に適した完全なフォーマットのメッセージ情報。
  - XML — XML（Extensible Markup Language）および AML（Adaptive Messaging Language）XSD（XML Schema Definition）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（http://www.cisco.com/）で公開されています。XML フォーマットを使用すると、TAC とのコミュニケーションが可能です。
- 同時に使用する複数のメッセージ宛先。宛先プロファイルごとに、最大 50 の E 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、次の情報を指定します。

- 1 つまたは複数のアラート グループ — アラートが発生した場合に、特定の Call Home メッセージを開始するアラート グループ。
- 1 つまたは複数の E 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 グループが生成した Call Home メッセージの受信先リスト。
- メッセージ フォーマット — 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）。
- メッセージの重大度 — Cisco NX-OS が宛先プロファイルに指定されたすべての E メール アドレスに対して Call Home メッセージを生成する前に、アラートが満たしていなければならない Call Home の重大度。Call Home の重大度の詳細については、「Call Home のメッセージ レベル」（p.4-6）を参照してください。アラートの 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。

毎日、毎週、または毎月の形で、定期的にメッセージを送信するインベントリ アラート グループを使用することによって、定期的にインベントリ アップデート メッセージが送信されるように、宛先プロファイルを設定することもできます。

Cisco NX-OS は、次に示す定義済み宛先プロファイルを使用します。

- Cisco TAC-1 — XML メッセージ フォーマットで 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 が設定済みです。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 テキストのメッセージ フォーマットをサポートします。
- short-text-destination — ショート テキストのメッセージ フォーマットをサポートします。

メッセージ フォーマットの詳細については、「メッセージ フォーマット」（p.4-24）を参照してください。

Call Home のアラート グループ

アラート グループは、すべての Cisco NX-OS スイッチでサポートされる、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の宛先プロファイルに送信する、一組の Call Home アラートを選択できます。Cisco NX-OS が宛先プロファイルに指定されている E メールの宛先に Call Home アラートを送信するのは、その Call Home アラートがその宛先プロファイルに関連付けられたアラート グループのいずれかに属していて、なおかつ Call Home のメッセージ重大度が宛先プロファイルで設定されているメッセージ重大度と同じかそれより上の場合だけです（「Call Home のメッセージ レベル」[p.4-6] を参照）。

表 4-1 に、サポートされるアラート グループとともに、アラート グループに対して生成される Call Home メッセージに組み込まれるデフォルトの CLI コマンド出力を示します。

表 4-1　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 を宛先とする他のアラート グループからのすべてのクリティカル アラート	アラートが発生したアラート グループに基づいてコマンドが実行されます。
Configuration	コンフィギュレーション関連の定期的イベント	**show module** **show running-configuration vdc-all all** **show startup-configuration vdc-all** **show vdc current** **show vdc membership** **show version**
Diagnostic	診断機能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 platform** **show tech-support sysmgr** **show vdc current** **show vdc membership**
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 platform** **show tech-support sysmgr** **show vdc current** **show vdc membership**
Environmental	電源、ファン、および温度アラームなどの環境感知コンポーネントに関連するイベント	**show environment** **show logging last 200** **show module** **show vdc current** **show vdc membership** **show version**
Inventory	装置のコールド ブートのたびに、または FRU（現場交換可能ユニット）の着脱時に提示されるインベントリ ステータス。このアラートは非クリティカル イベントとみなされ、情報はステータスおよび資格目的で使用されます。	**show inventory** **show module** **show system uptime** **show sprom all** **show vdc current** **show vdc membership** **show version**
License	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 license usage** **show logging last 200** **show tech-support ethpm** **show vdc current** **show vdc membership**

表 4-1　アラート グループおよび実行されるコマンド（続き）

アラート グループ	説明	実行されるコマンド
Linemodule hardware	標準またはインテリジェント スイッチング モジュール関連のイベント	**show diagnostic result module all detail** **show diagnostic result module** *number* **detail** **show module** **show tech-support gold** **show tech-support platform** **show tech-support sysmgr** **show vdc current** **show vdc membership**
Supervisor hardware	スーパーバイザ モジュール関連のイベント	**show diagnostic result module all detail** **show diagnostic result module** *number* **detail** **show module** **show tech-support gold** **show tech-support platform** **show tech-support sysmgr** **show vdc current** **show vdc membership**
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** **show tech-support ethpm** **show vdc current** **show vdc membership**
System	装置の動作に重要なソフトウェア システムの障害によって生成されるイベント	**show diagnostic result module all detail** **show diagnostic result module** *number* **detail** **show module** **show tech-support gold** **show tech-support platform** **show tech-support sysmgr** **show vdc current** **show vdc membership**
Test	ユーザが生成するテスト メッセージ	**show module** **show vdc current** **show vdc membership** **show version**

Call Home は、syslog 重大度を syslog ポート グループ メッセージに対応する Call Home の重大度にマッピングします（「Call Home のメッセージ レベル」[p.4-6] を参照）。

定義済みのアラート グループをカスタマイズすると、特定のイベント発生時に他の CLI **show** コマンドを実行し、その **show** コマンドの出力を Call Home メッセージで送信できます。

show コマンドを追加できるのは、フル テキストおよび XML の宛先プロファイルだけです。ショート テキストの宛先プロファイルでは、使用できるテキストが 128 バイトだけなので、**show** コマンドの追加はサポートされません。

Call Home のメッセージ レベル

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。宛先プロファイル（定義済みおよびユーザ定義）ごとに Call Home メッセージ レベルのしきい値と関連付けることができます。Cisco NX-OS は宛先プロファイルに対して、このしきい値に満たない Call Home メッセージは生成しません。Call Home のメッセージ レベル範囲は 0（最低の緊急度）～ 9（最高の緊急度）です。デフォルトは 0（Cisco NX-OS はすべてのメッセージを送信）です。

syslog アラート グループに送信される Call Home メッセージは、syslog 重大度が Call Home メッセージ レベルにマッピングされています。

（注） Call Home がメッセージ テキストの syslog メッセージ レベルを変更することはありません。Call Home のログで syslog メッセージがどのように示されるかについては、『*Cisco NX-OS System Messages Guide*』を参照してください。

表 4-2 に、各 Call Home メッセージ レベルのキーワードと syslog ポート アラート グループの対応する syslog レベルを示します。

表 4-2　重大度と syslog レベルのマッピング

Call Home のレベル	キーワード	syslog のレベル	説明
9	**Catastrophic**	該当なし	ネットワーク全体の重大な障害
8	**Disaster**	該当なし	ネットワークに重大な影響
7	**Fatal**	緊急（0）	システム使用不可
6	**Critical**	アラート（1）	即時対応が必要であることを示すクリティカル条件
5	**Major**	クリティカル（2）	メジャー条件
4	**Minor**	エラー（3）	マイナー条件
3	**Warning**	警告（4）	警告条件
2	**Notification**	通知（5）	基本的な通知および情報メッセージ通常、単独での重要性は薄い。
1	**Normal**	情報（6）	正常な状態に戻ったことを伝える正常なイベント
0	**Debugging**	デバッグ（7）	デバッグ メッセージ

Smart Call Home の利用方法

シスコシステムズと直接サービス契約を結んでいる場合は、デバイスを Smart Call Home サービスに登録できます。Smart Call Home は、デバイスから送られた Call Home メッセージを分析し、背景説明と推奨処置を提供することによって、システムの問題を迅速に解決できるようにします。既知の問題として特定できた場合、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 が提供する機能は、次のとおりです。

- 継続的なデバイス ヘルス モニタリングおよびリアルタイム診断アラート。
- デバイスから送られた Call Home メッセージの分析。必要に応じて Automatic Service Request が作成され、詳細な診断情報を含め、適切な TAC チームにルーティングされて、問題解決の高速化が実現します。
- デバイスから直接、またはダウンロード可能な TG（トランスポート ゲートウェイ）集約ポイントを通じて行われるセキュア メッセージ トランスポート。TG 集約ポイントを使用できるのは、複数のデバイスにサポートが必要な場合、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です。
- あらゆる Call Home デバイスの Call Home メッセージおよび推奨事項、コンポーネント情報、設定情報への Web アクセス。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。

- 使用 switch の SMARTnet 契約番号
- E メール アドレス
- Cisco.com の ID

Smart Call Home の詳細については、次の URL にアクセスして Smart Call Home のページを参照してください。

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

ハイ アベイラビリティ

Cisco NX-OS は、Call Home 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あとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

Cisco NX-OS は、VDC（Virtual Device Context; 仮想デバイス コンテキスト）ごとに Call Home インスタンスを 1 つずつサポートします。Smart Call Home では、最初に登録された VDC のコンタクト情報を物理デバイス上のすべての VDC の管理者コンタクトとして使用します。たとえば、Smart Call Home でデフォルト VDC のコンタクト情報が使用されるようにするには、その VDC を使用して登録する必要があります。この情報は次の URL から、Smart Call Home の Web サイトでアップデートできます。

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

Smart Call Home は他のすべての VDC のコンタクトを、物理デバイスのすべての Call Home データを参照できるが、管理者として動作することはできないユーザとして登録します。すべての登録ユーザおよび登録管理者は、物理デバイス上のすべての VDC からすべての Call Home 通知を受け取ります。

デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照してください。

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

Call Home 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
NX-OS	Call Home の使用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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

Call Home の前提条件

Call Home の前提条件は、次のとおりです。

- E メール サーバを設定する必要があります。
- Call Home をイネーブルにする前に、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。受信メッセージの発行元を特定するために、この手順が必要です。
- スイッチから E メール サーバに IP で接続できなければなりません。
- Smart Call Home を使用する場合は、設定するデバイスに有効なサービス契約が必要です。
- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。このライセンスは Call Home ではなく、VDC のためだけに必要です。
- VDC を設定する場合は、デフォルトの VDC からデバイスを登録する必要があります。

設定時の注意事項および制約事項

Call Home 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。

- IP 接続機能がない場合、または宛先プロファイルに対する VRF のインターフェイスが停止している場合、Cisco NX-OS は Call Home メッセージを送信できません。
- あらゆる SMTP サーバで動作します。

Call Home の設定

ここでは、次の内容について説明します。

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

Call Home 設定時の注意事項

Call Home を設定する手順は、次のとおりです。

ステップ 1　コンタクト情報を割り当てます。

ステップ 2　宛先プロファイルを設定します。

ステップ 3　各プロファイルに 1 つまたは複数のアラート グループを関連付けます。

ステップ 4　（任意）アラーと グループに他の **show** コマンドを追加します。

ステップ 5　トランスポート オプションを設定します。

ステップ 6　Call Home をイネーブルにします。

ステップ 7　（任意）Call Home メッセージをテストします。

コンタクト情報の設定

Call Home 用に E メール、電話、住所情報を設定する必要があります。任意でコンタクト 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設定できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

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

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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)#`
		callhome 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 E メール アドレスを設定します。E メール アドレスのフォーマットで最大 255 文字の英数字を指定できます。 **（注）** 任意の有効な E メール アドレスを使用できます。スペースは使用できません。

	コマンド	目的
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番号のフォーマットで設定します。国際電話のフォーマットで最大 17 文字の英数字を指定できます。 **（注）** スペースは使用できません。必ず、番号の前に + のプレフィクスを使用します。
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。スペースを含め、最大 255 文字の英数字を指定できます。
ステップ 7	**contract-id** *contract-number* **例：** `switch(config-callhome)# contract-id Contract5678`	（任意）サービス契約に基づいて、このデバイスの契約番号を設定します。契約番号は任意のフォーマットで、最大 255 文字の英数字を使用して指定できます。
ステップ 8	**customer-id** *customer-number* **例：** `switch(config-callhome)# customer-id Customer123456`	（任意）サービス契約に基づいて、このデバイスのカスタマー番号を設定します。カスタマー番号は任意のフォーマットで、最大 255 文字の英数字を使用して指定できます。
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。サイト番号は任意のフォーマットで、最大 255 文字の英数字を使用して指定できます。
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。値の範囲は 0 ～ 7 で、0 が最高、7 が最低のプライオリティです。デフォルト値は 7 です。
ステップ 11	**show callhome** **例：** `switch(config-callhome)# show callhome`	（任意）Call Home 設定の要約を表示します。
ステップ 12	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

Call Home のコンタクト情報を設定する例を示します。

```
switch# config t
switch(config)# snmp-server contact personname@companyname.com
switch(config)# callhome
switch(config-callhome)# email-contact admin@Mycompany.com
switch(config-callhome)# phone-contact +1-800-123-4567
switch(config-callhome)# street-address 123 Anystreet st. Anytown,AnyWhere
```

宛先プロファイルの作成

ユーザ定義の宛先プロファイルを作成し、その新しい宛先プロファイル用のメッセージ フォーマットを設定できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***callhome***
3. **destination-profile** *name*
4. **destination-profile** *name* **format** {**XML** | **full-txt** | **short-txt**}
5. **show callhome destination-profile** [**profile** *name*]
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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)#`
		callhome 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。名前には最大 31 の英数字を使用できます。

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。名前には最大 31 の英数字を使用できます。

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。

ステップ 6	***copy running-config startup-config***
	
	例：
	`switch(config)# copy running-config`
	`startup-config`
		（任意）この設定変更を保存します。

Call Home の宛先プロファイルを作成する例を示します。

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

宛先プロファイルの変更

定義済みの宛先プロファイルまたはユーザ定義の宛先プロファイルでは、次のアトリビュートを変更できます。

- 宛先アドレス — アラートの送信先になる、トランスポート メカニズムに適した実際のアドレス
- メッセージ フォーマット — アラートの送信に使用するメッセージ 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に対応する、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 E メール アドレスに送信できる Call Home メッセージの長さ。

宛先プロファイルに対応するアラート グループの設定については、「アラート グループと宛先プロファイルの関連付け」（p.4-15）を参照してください。

（注）　Cisco TAC-1 宛先プロファイルは、変更も削除もできません。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
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**} **message-level** *number*
5. **destination profile** {*name* | **CiscoTAC-1** | **full-txt-destination** | **short-txt-destination**} **message-size** *number*
6. **show call-home destination-profile** [**profile** *name*]
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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)#`
		callhome 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宛先プロファイル用の E メール アドレスを設定します。 **ヒント**　1 つの宛先プロファイルで最大 50 の E 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**} **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に対応する 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、Call Home の重大度が同じかそれ以上のアラートだけです。値の範囲は 0 ～ 9 です。9 が最高の重大度です。
ステップ 5	**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。値の範囲は 0 ～ 5000000 です。デフォルト値は 2500000 です。
ステップ 6	**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 7	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

Call Home の宛先プロファイルを変更する例を示します。

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

アラート グループと宛先プロファイルの関連付け

1 つの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ることができます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***callhome***
3. **destination profile** *name* **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Linemodule-Hardware** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**}
4. **show callhome destination-profile** [**profile** *name*]
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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)#`
		callhome 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* **alert-group** {**All** | **Cisco-TAC** | **Configuration** | **Diagnostic** | **EEM** | **Environmental** | **Inventory** | **License** | **Linemodule-Hardware** | **Supervisor-Hardware** | **Syslog-group-port** | **System** | **Test**} **例：** `switch(config-callhome)# destination-profile Noc101 alert-group All`	この宛先プロファイルにアラート グループを関連付けます。すべてのアラート グループを宛先プロファイルに関連付ける場合は、**All** キーワードを使用します。
ステップ 4	**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。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

宛先プロファイル Noc101 にすべての アラート グループを関連付ける例を示します。

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

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）　Cisco TAC-1 宛先プロファイルにユーザ定義の CLI **show** コマンドを追加することはできません。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***callhome***
3. **alert-group {Configuration | Diagnostic | EEM | Environmental | Inventory | License | Linemodule-Hardware | Supervisor-Hardware | Syslog-group-port | System | Test} user-def-cmd** *show-cmd*
4. **show call-home user-def-cmds**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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)#	callhome 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 | Linemodule-Hardware | Supervisor-Hardware | Syslog-group-port | System | Test} user-def-cmd** *show-cmd* **例：** switch(config-callhome)# alert-group Configuration user-def-cmd “show ip routing”	このアラート グループに対して送信されるあらゆる Call Home メッセージに、**show** コマンド出力を追加します。**show** コマンドは二重引用符で囲む必要があります。使用できるのは、有効な **show** コマンドだけです。

	コマンド	目的
ステップ 4	**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ているユーザ定義のすべての **show** コマンドについて、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）この設定変更を保存します。

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

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

E メールの設定

Call Home を機能させるには、SMTP サーバ アドレスを設定する必要があります。送信元および返信先 E メール アドレスも設定できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***callhome***
3. **transport email smtp-server** *ip-address* [**port** *number*] [**use-vrf** *vrf-name*]
4. **transport email from** *email-address*
5. **transport email reply-to** *email-address*
6. **show callhome transport-email**
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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)#`
		callhome 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 smtp-server** *ip-address* [**port** *number*] [**use-vrf** *vrf-name*] **例：** `switch(config-callhome)# transport email smtp-server 192.0.2.1 use-vrf Red`	DNS（ドメイン ネーム サーバ）名、IPv4 アドレス、または IPv6 アドレスのいずれかで SMTP サーバを設定します。さらに、任意でポート番号を設定します。ポート範囲は 1 ～ 65535 であり、デフォルトのポート番号は 25 です。 さらに任意で、この SMTP サーバとの通信時に使用する VRF を設定します。
ステップ 4	**transport email from** *email-address* **例：** `switch(config-callhome)# transport email from person@company.com`	（任意）Call Home メッセージ用の email from（E メール送信元）フィールドを設定します。
ステップ 5	**transport email reply-to** *email-address* **例：** `switch(config-callhome)# transport email reply-to person@company.com`	（任意）Call Home メッセージ用の email reply-to（E メール返信先）フィールドを設定します。
ステップ 6	**show callhome transport-email** **例：** `switch(config-callhome)# show callhome transport-email`	（任意）Call Home の E メール設定に関する情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

Call Home メッセージの E メール オプションを設定する例を示します。

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

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であり、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、スイッチを設定できます。Cisco NX-OS は 2 種類の Call Home 通知を生成します。定期的コンフィギュレーション メッセージおよび定期的インベントリ メッセージです。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***callhome***
3. **periodic-inventory notification** [**interval** *days* | **timeofday** *time*]
4. **show callhome**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		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)#`
		callhome 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`	定期的インベントリ メッセージを設定します。インターバルの範囲は 1 ～ 30 日です。デフォルトは 7 で、ToD 値は HH:MM 形式です。
ステップ 4	**show callhome** **例：** `switch(config-callhome)# show callhome`	（任意）Call Home 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）この設定変更を保存します。

定期的インベントリメッセージが 20 日おきに生成されるように設定する例を示します。

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

重複メッセージ スロットリングのディセーブル化

同じイベントについて受け取る重複メッセージの数を制限できます。デフォルトでは、Cisco NX-OS は同じイベントについて受け取る重複メッセージの数を制限します。2 時間以内に送信された重複メッセージの数が 30 を超えると、Cisco NX-OS はそのアラート タイプについて、それ以上のメッセージをディセーブルにします。

重複メッセージ スロットリングをディセーブルにするには、Call Home 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
no duplicate-message throttle **例：** `switch(config-callhome)# no duplicate-message throttle`	Call Home の重複メッセージ スロットリングをディセーブルにします。デフォルトではイネーブルにされています。

Call Home のイネーブルまたはディセーブル

コンタクト情報を設定すると、Call Home 機能をイネーブルにできます。

Call Home をイネーブルにするには、Call Home 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
enable **例：** `switch(config-callhome)# enable`	Call Home をイネーブルにします。デフォルトではディセーブルです。

Call Home をディセーブルにするには、Call Home 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
no enable **例：** `switch(config-callhome)# no enable`	Call Home をディセーブルにします。デフォルトではディセーブルです。

Call Home 通信のテスト

Call Home 通信をテストするために、テスト メッセージを作成できます。

テスト用の Call Home メッセージを作成するには、任意のモードで次のコマンドを使用します。

コマンド	目的
callhome send [**configuration** | **diagnostic** | **inventory**] **例：** `switch(config-callhome)# callhome send diagnostic`	設定されているすべての宛先に、指定の Call Home テスト メッセージを送信します。
callhome test **例：** `switch(config-callhome)# callhome test`	設定されているすべての宛先に、テスト メッセージを送信します。

Call Home の設定確認

Call Home 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* callhome**	Call Home のステータスを表示します。
***show* callhome destination-profile** *name*	1 つまたは複数の Call Home 宛先プロファイルを表示します。
***show* callhome status**	Call Home のステータスを表示します。
***show* callhome transport-email**	Call Home の E メール設定を表示します。
***show* callhome user-def-cmds**	アラート グループに追加されている CLI コマンドを表示します。
show running-config callhome [**all**]	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 [**all**]	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**	Call Home に関するテクニカル サポート出力を表示します。

Call Home の設定例

Noc101 という宛先プロファイルを作成し、そのプロファイルに Cisco-TAC アラート グループを関連付けて、コンタクト情報および E メール情報を設定する例を示します。

```
config t
 snmp-server contact person@company.com
 callhome
  email-contact admin@Mycompany.com
  phone-contact +1-800-123-4567
  street-address 123 Anystreet st. Anytown,AnyWhere
  destination-profile Noc101
  destination-profile Noc101 full-text
  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 routing"
  transport email smtp-server 192.0.2.10 use-vrf Red
  enable
```

デフォルト設定

表 4-3 に、Call Home パラメータのデフォルト設定を示します。

表 4-3　デフォルトの Call Home パラメータ

パラメータ	デフォルト
フル テキスト フォーマットで送信するメッセージの宛先メッセージ サイズ	2500,000
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	2500,000
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
プロファイルとアラート グループの関連付け	full-text-destination および short-text-destination プロファイルではすべて。CiscoTAC-1 宛先プロファイルでは cisco-tac アラート グループ
フォーマット タイプ	XML
Call Home のメッセージ レベル	0（ゼロ）

その他の関連資料

Call Home の実装に関連する詳細情報については、次の項を参照してください。

イベント トリガー

表 4-4 に、イベント トリガーおよび対応する Call Home メッセージの重大度を示します。

表 4-4　イベント トリガー

アラート グループ	イベント名	説明	Call Home の重大度
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
Diagnostic	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	Call Home が生成した通常の診断アラート	2
Environmental および CISCO_TAC	FAN_FAILURE	冷却ファンの故障	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	温度センサによる温度が動作しきい値を超えたという表示	6
Inventory および CISCO_TAC	COLD_BOOT	スイッチの起動およびリセットによるコールド ブート シーケンス	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
License	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
Line module Hardware、Supervisor Hardware、および CISCO_TAC	BOOTFLASH_FAILURE	ブート コンパクト フラッシュ モジュールの故障	6
	EOBC_FAILURE	イーサネット帯域外チャネル通信の障害	6

表 4-4　イベント トリガー（続き）

アラート グループ	イベント名	説明	Call Home の重大度
Supervisor Hardware および CISCO_TAC	CMP_FAILURE	CMP モジュールの動作障害	5
	POWER_UP_DIAGNOSTICS_FAILURE	スーパーバイザの起動障害	7
	SUP_FAILURE	スーパーバイザ モジュールの動作障害	7
Syslog-group-port	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
System 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、すなわちサービスの停止	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
Test 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

Call Home がサポートするメッセージ フォーマットは、次のとおりです。

- ショート テキスト メッセージ フォーマット
- すべてのフル テキストおよび XML メッセージに共通するフィールド
- 対応型または予防型イベント メッセージに挿入されるフィールド
- インベントリ イベント メッセージに挿入されるフィールド
- ユーザが作成したテスト メッセージに挿入されるフィールド

表 4-5 で、すべてのメッセージ タイプに共通するショート テキスト フォーマット オプションについて説明します。

表 4-5　ショート テキスト メッセージ フォーマット

データ項目	説明
Device identification	設定されているデバイス名
Date/time stamp	トリガー イベントのタイムスタンプ
Error isolation message	トリガー イベントの英語による簡単な説明
Alarm urgency level	システム メッセージに適用されるようなエラー レベル

表 4-6 で、フル テキストまたは XML に共通のイベント メッセージ フォーマットについて説明します。

表 4-6　すべてのフル テキストおよび XML メッセージに共通するフィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
Time stamp	ISO の時刻表記で表した日付およびタイムスタンプ *YYYY-MM-DD HH:MM:SS GMT+HH:MM*	/aml/header/time
Message name	メッセージ名具体的なイベント名については表 4-4 を参照	/aml/header/name
Message type	reactive（対応型）、proactive（予防型）などのメッセージ タイプの名前	/aml/header/type
Message group	syslog など、アラート グループの名前	/aml/header/group
Severity level	メッセージの重大度（「Call Home のメッセージ レベル」[p.4-6] を参照）	/aml/header/level
Source ID	ルーティング製品タイプ。具体的には Catalyst 6500	/aml/header/source
Device ID	メッセージを生成したエンド デバイスの UDI（固有デバイス識別情報）。メッセージがデバイス固有ではない場合は、このフィールドを空にしておきます。フォーマットは *type@Sid@serial* です。 • *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 フィールドで特定された番号です。 例：WS-C6509@C@12345678	/aml/ header/deviceId
Customer ID	サポート サービスの契約情報またはその他の ID に任意で使用する、ユーザ側で設定可能なフィールド	/aml/ header/customerID
Contract ID	サポート サービスの契約情報またはその他の ID に任意で使用する、ユーザ側で設定可能なフィールド	/aml/ header /contractId
Site ID	シスコが指定したサイト ID またはその他、代替サポート サービスで意味のあるデータに使用する、ユーザ側で設定可能なフィールド	/aml/ header/siteId
Server ID	デバイスから発生するメッセージの場合、これはデバイスの UDI です。 フォーマットは *type@Sid@serial* です。 • *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 フィールドで特定された番号です。 例：WS-C6509@C@12345678	/aml/header/serverId
Message description	エラーを記述するショート テキスト	/aml/body/msgDesc

表 4-6　すべてのフル テキストおよび XML メッセージに共通するフィールド（続き）

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）
Device name	イベントが発生したノード（デバイスのホスト名）	/aml/body/sysName
Contact name	イベントが発生したノードに関連する問題の連絡となる担当者名	/aml/body/sysContact
Contact e-mail	この装置の連絡先として指定された担当者の E メール アドレス	/aml/body/sysContactEmail
Contact phone number	この装置の連絡先として指定された担当者の電話番号	/aml/body/sysContactPhoneNumber
Street address	この装置に関連する RMA 部品の出荷先住所を任意で指定するフィールド	/aml/body/sysStreetAddress
Model name	デバイスのモデル名（製品ファミリ名の一部としての特定モデル）	/aml/body/chassis/name
Serial number	装置のシャーシ シリアル番号	/aml/body/chassis/serialNo
Chassis part number	シャーシ上部のアセンブリ番号	/aml/body/chassis/partNo

ここで、特定のアラート グループ メッセージに固有のフィールドが挿入されます。

このアラート グループに対して複数の CLI を実行する場合、次のフィールドは反復可能です。		
Command output name	発行された CLI コマンドの正確な名前	/aml/attachments/attachment/name
Attachment type	特定のコマンド出力	/aml/attachments/attachment/type
MIME type	プレーン テキストまたは符号化タイプのどちらか。	/aml/attachments/attachment/mime
Command output text	自動的に実行されたコマンドの出力（「Call Home のアラート グループ」[p.4-3] を参照）	/aml/attachments/attachment/atdata

表 4-7 で、フル テキストまたは XML に対応する、対応型イベント メッセージ フォーマットについて説明します。

表 4-7　対応型または予防型イベント メッセージに挿入されるフィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
Chassis hardware version	シャーシのハードウェア バージョン	/aml/body/chassis/hwVersion
Supervisor module software version	上位レベルのソフトウェア バージョン	/aml/body/chassis/swVersion
Affected FRU name	イベント メッセージを生成している 関連 FRU 名	/aml/body/fru/name
Affected FRU serial number	関連 FRU のシリアル番号	/aml/body/fru/serialNo
Affected FRU part number	関連 FRU の部品番号	/aml/body/fru/partNo
FRU slot	イベント メッセージを生成している FRU のスロット番号	/aml/body/fru/slot
FRU hardware version	関連 FRU のハードウェア バージョン	/aml/body/fru/hwVersion
FRU software version	関連 FRU 上で動作しているソフトウェア バージョン（複数可）	/aml/body/fru/swVersion

表 4-8 で、フル テキストまたは XML に対応する、インベントリ イベント メッセージ フォーマットについて説明します。

表 4-8　インベントリ イベント メッセージに挿入されるフィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
Chassis hardware version	シャーシのハードウェア バージョン	/aml/body/chassis/hwVersion
Supervisor module software version	上位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 メッセージを生成している 関連 FRU 名	/aml/body/fru/name
FRU s/n	FRU のシリアル番号	/aml/body/fru/serialNo
FRU part number	FRU の部品番号	/aml/body/fru/partNo
FRU slot	FRU のスロット番号	/aml/body/fru/slot
FRU hardware version	FRU のハードウェア バージョン	/aml/body/fru/hwVersion
FRU software version	FRU 上で動作しているソフトウェア バージョン（複数可）	/aml/body/fru/swVersion

表 4-9 で、フル テキストまたは XML に対応する、ユーザ作成テスト メッセージのフォーマットについて説明します。

表 4-9　ユーザが作成したテスト メッセージに挿入されるフィールド

データ項目（プレーン テキストおよび XML）	説明（プレーン テキストおよび XML）	XML タグ（XML のみ）
Process ID	固有のプロセス ID	/aml/body/process/id
Process state	プロセスの状態（running [実行中]、halted [停止] など）	/aml/body/process/processState
Process exception	例外または原因コード	/aml/body/process/exception

syslog アラート通知の例（フル テキスト フォーマット）

フル テキスト フォーマットを使用した syslog port アラート グループ通知の例を示します。

```
Severity Level:5
Series:Nexus7000
Switch Priority:0
Device Id:N7K-C7010@C@TXX12345678
Server Id:N7K-C7010@C@TXX12345678
Time of Event:2008-01-17 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 2008 Jan 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:N7K-C7010
Affected Chassis Serial Number:TXX12345678 Affected Chassis Hardware Version:0.405
Affected Chassis Software Version:4.0(1) Affected Chassis Part No:73-10900-04 end
chassis information:
start attachment
    name:show logging logfile | tail -n 200
    type:text
    data:
    2008 Jan 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile (messages)
cleared by user
    2008 Jan 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console
from /dev/ttyS0 /dev/ttyS0_console
    2008 Jan 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console
from /dev/ttyS0 /dev/ttyS0_console
    2008 Jan 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16:
Invalid argument: - sshd[14484]
    2008 Jan 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console
from /dev/ttyS0 /dev/ttyS0_console
    2008 Jan 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).
    2008 Jan 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console
from /dev/ttyS0 /dev/ttyS0_console
    2008 Jan 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 .
    2008 Jan 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 3504)
hasn't caught signal 9 (no core).
    2008 Jan 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.
    2008 Jan 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23210)
hasn't caught signal 9 (no core).
    2008 Jan 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.
    2008 Jan 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 23294)
hasn't caught signal 9 (no core).
    2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is
becoming active (pre-start phase).
    2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is
becoming active.
    2008 Jan 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send
failed - device_test
    2008 Jan 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR:  netstack [4336]
Unrecognized message from MRIB. Major type 1807
    2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 1
    2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 2
    2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 3
    2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 4
```

```
    2008 Jan 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2  -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10  -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2  -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0  -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0  -
ntpd[19045]
    2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET:  netstack [4336]  HA client
filter recovery failed (0)
    2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET:  netstack [4336]  HA client
filter recovery failed (0)
    2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
    2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19072]
    2008 Jan 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19073]
    2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
    2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19079]
    2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 1
    2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 2
    2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 3
    2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 4
    2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
    2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19105]
    2008 Jan 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
    2008 Jan 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
    2008 Jan 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
    2008 Jan 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
    2008 Jan 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
    2008 Jan 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
    2008 Jan 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
    2008 Jan 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 .
    2008 Jan 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 4820)
hasn't caught signal 9 (no core).
    2008 Jan 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 .
    2008 Jan 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24239)
hasn't caught signal 9 (no core).
    2008 Jan 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 .
    2008 Jan 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24401)
hasn't caught signal 9 (no core).
    2008 Jan 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
    2008 Jan 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 .
    2008 Jan 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service "eltm" (PID 24407)
hasn't caught signal 9 (no core).
    2008 Jan 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
```

```
    2008 Jan 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
    2008 Jan 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command
<more> return code <14>
    2008 Jan 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR:  netstack [4336]  (null)
    2008 Jan 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
    name:show vdc membership
    type:text
    data:

    vdc_id: 1 vdc_name: 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

    vdc_id: 2 vdc_name: dc3-aaa interfaces:

    vdc_id: 3 vdc_name: dc3-rbac interfaces:

    vdc_id: 4 vdc_name: dc3-call interfaces:



end attachment
start attachment
    name:show vdc current-vdc
    type:text
    data:
    Current vdc is 1 - dc3-test
end attachment
start attachment
    name:show license usage
    type:text
    data:
    Feature                      Ins  Lic   Status Expiry Date Comments
                                     Count
    --------------------------------------------------------------------------------
    LAN_ADVANCED_SERVICES_PKG     Yes   -   In use Never       -
    LAN_ENTERPRISE_SERVICES_PKG   Yes   -   Unused Never       -
    --------------------------------------------------------------------------------
end attachment
```

syslog アラート通知の例（XML フォーマット）

XML フォーマットを使用した syslog port アラート グループ通知の例を示します。


```
<?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>2008-01-17 16:31:33 GMT+0000</aml-block:CreationDate>
<aml-block:Builder> <aml-block:Name>DC3</aml-block:Name>
<aml-block:Version>4.0</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>2008-01-17 16:31:33 GMT+0000</ch:EventTime>
<ch:MessageDescription>SYSLOG_ALERT 2008 Jan 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>Nexus7000</ch:Series>
</ch:Event> <ch:CustomerData> <ch:UserData> <ch:Email>contact@example.com</ch:Email>
</ch:UserData>
<ch:ContractData>
<ch:DeviceId>N7K-C7010@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/4.0">
<rme:Model>N7K-C7010</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">
<![CDATA[2008 Jan 17 10:57:51 dc3-test %SYSLOG-1-SYSTEM_MSG : Logging logfile
(messages) cleared by user
2008 Jan 17 10:57:53 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 10:58:35 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 17 10:59:00 dc3-test %DAEMON-3-SYSTEM_MSG: error: setsockopt IP_TOS 16:
Invalid argument: - sshd[14484]
2008 Jan 17 10:59:05 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 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).
2008 Jan 17 16:28:03 dc3-test %VSHD-5-VSHD_SYSLOG_CONFIG_I: Configuring console from
/dev/ttyS0 /dev/ttyS0_console
2008 Jan 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 .
2008 Jan 17 16:28:44 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 3504)
hasn&apos;t caught signal 9 (no core).
2008 Jan 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 .
2008 Jan 17 16:29:08 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23210)
hasn&apos;t caught signal 9 (no core).
2008 Jan 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 .
2008 Jan 17 16:29:17 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 23294)
hasn&apos;t caught signal 9 (no core).
2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_PRE_START: This supervisor is
becoming active (pre-start phase).
2008 Jan 17 16:29:25 dc3-test %SYSMGR-2-HASWITCHOVER_START: This supervisor is
becoming active.
2008 Jan 17 16:29:26 dc3-test %USER-3-SYSTEM_MSG: crdcfg_get_srvinfo: mts_send failed
- device_test
2008 Jan 17 16:29:27 dc3-test %NETSTACK-3-IP_UNK_MSG_MAJOR:  netstack [4336]
Unrecognized message from MRIB. Major type 1807
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 1
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 2
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 3
2008 Jan 17 16:29:27 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is DOWN in vdc 4
2008 Jan 17 16:29:28 dc3-test %SYSMGR-2-SWITCHOVER_OVER: Switchover completed.
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2  -
ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 10  -
ntpd[19045]
```

```
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:ipv6 only defined -
ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:bindv6 only defined -
ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 2  -
ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0  -
ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %DAEMON-3-SYSTEM_MSG: ntp:socket family : 0  -
ntpd[19045]
2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET:  netstack [4336]  HA client
filter recovery failed (0)
2008 Jan 17 16:29:28 dc3-test %NETSTACK-3-CLIENT_GET:  netstack [4336]  HA client
filter recovery failed (0)
2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19072]
2008 Jan 17 16:29:29 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19072]
2008 Jan 17 16:29:31 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19073]
2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19079]
2008 Jan 17 16:29:32 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19079]
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 1
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 2
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 3
2008 Jan 17 16:29:34 dc3-test %IM-5-IM_INTF_STATE: mgmt0 is UP in vdc 4
2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: ssh disabled, removing -
dcos-xinetd[19105]
2008 Jan 17 16:29:34 dc3-test %DAEMON-3-SYSTEM_MSG: telnet disabled, removing -
dcos-xinetd[19105]
2008 Jan 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
2008 Jan 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
2008 Jan 17 16:29:38 dc3-test %CALLHOME-2-EVENT: SUP_FAILURE
2008 Jan 17 16:29:46 dc3-test vsh[19166]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:30:24 dc3-test vsh[23810]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:30:24 dc3-test vsh[23803]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:30:24 dc3-test vsh[23818]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 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 .
2008 Jan 17 16:30:47 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 4820)
hasn&apos;t caught signal 9 (no core).
2008 Jan 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 .
2008 Jan 17 16:31:02 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24239)
hasn&apos;t caught signal 9 (no core).
2008 Jan 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 .
2008 Jan 17 16:31:14 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24401)
hasn&apos;t caught signal 9 (no core).
2008 Jan 17 16:31:23 dc3-test %CALLHOME-2-EVENT: SW_CRASH alert for service: eltm
2008 Jan 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 .
2008 Jan 17 16:31:23 dc3-test %SYSMGR-2-SERVICE_CRASHED: Service \"eltm\" (PID 24407)
hasn&apos;t caught signal 9 (no core).
2008 Jan 17 16:31:24 dc3-test vsh[24532]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:31:24 dc3-test vsh[24548]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:31:24 dc3-test vsh[24535]: CLIC-3-FAILED_EXEC: Can not exec command
&lt;more&gt; return code &lt;14&gt;
2008 Jan 17 16:31:33 dc3-test %NETSTACK-3-INTERNAL_ERROR:  netstack [4336]  (null)
```

```
2008 Jan 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>show vdc
membership</aml-block:Name> <aml-block:Data encoding="plain"> <![CDATA[
vdc_id: 1 vdc_name: 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

vdc_id: 2 vdc_name: dc3-aaa interfaces:

vdc_id: 3 vdc_name: dc3-rbac interfaces:

vdc_id: 4 vdc_name: dc3-call interfaces:



]]>
</aml-block:Data>
</aml-block:Attachment>
<aml-block:Attachment type="inline">
<aml-block:Name>show vdc current-vdc</aml-block:Name> <aml-block:Data
encoding="plain"> <![CDATA[Current vdc is 1 - dc3-test ]]> </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">
<![CDATA[Feature                      Ins  Lic   Status Expiry Date Comments
                              Count
--------------------------------------------------------------------------------
LAN_ADVANCED_SERVICES_PKG     Yes   -   In use Never       -
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>
```

関連資料

関連項目	マニュアル名
Call Home CLI コマンド	『*Cisco NX-OS System Management Command Line Reference*』
VDC および VRF	『*Cisco NX-OS Virtual Device Contexts Configuration Guide*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

MIB

MIB	MIB のリンク
• CISCO-CALLHOME-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

■ その他の関連資料

CHAPTER 5

ロールバックおよび Session Manager の設定

この章では、Cisco NX-OS でロールバックおよび Session Manager 機能を設定する方法について説明します。

ここでは、次の内容を説明します。

ロールバックおよび Session Manager の概要

ここでは、次の内容について説明します。

ロールバックの概要

ロールバック機能を使用すると、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、ロールバックによってそのチェックポイント コンフィギュレーションを適用できます。

いつでも、現在の実行コンフィギュレーションのチェックポイント コピーを作成できます。Cisco NX-OS はこのチェックポイントを ASCII ファイルとして保存するので、将来、そのファイルを使用して、実行コンフィギュレーションをチェックポイント コンフィギュレーションにロールバックできます。複数のチェックポイントを作成すると、実行コンフィギュレーションのさまざまなバージョンを保存できます。

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までに、すで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。

Session Manager

Session Manager を使用すると、バッチ モードで設定変更を実行できます。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 — セッション マネージャ モードで実行するコマンドのリストを作成します。
- 検証 — コンフィギュレーションの基本的なセマンティクス検査を行います。Cisco NX-OS は、コンフィギュレーションのどこか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。
- 確認 — 既存のハードウェア / ソフトウェア 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を全体として確認します。Cisco NX-OS は、コンフィギュレーション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自動的に実行します。エラーが発生した場合、Cisco NX-OS は元のコンフィギュレーションに戻ります。
- 打ち切り — 実行しないでコンフィギュレーションの変更を破棄します。

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

ロールバック機能を使用すると、ソフトウェアをリロードしなくても、以前のチェックポイント コンフィギュレーションにロールバックできます。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 リスタートまたはシステム スイッチオーバー時に、無停止チェックポイントまたはロールバック操作を実行できます。

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。

仮想化サポート

Cisco NX-OS は、ユーザがログインした VDC（Virtual Device Context; 仮想デバイス コンテキスト）で、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。VDC ごとにさまざまなチェックポイント コピーを作成できます。ある VDC のチェックポイントを別の VDC に適用することはできません。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照してください。

チェックポイント ファイルから VDC を作成したり削除したりすることはできません。チェックポイントは特定の VDC から作成する必要があります。

ロールバックおよび Session Manager 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
NX-OS	ロールバックおよび 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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

ロールバックおよび Session Manager の前提条件

VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

ロールバック機能を使用するには、network-admin または vdc-admin のユーザ権限が必要です。Session Manager に関しては、あらゆるユーザがセッションを作成できます。ただし、セッション内のコマンドを確認できるのは、ユーザの権限で許可された場合だけです。

設定時の注意事項および制約事項

ロールバックに関する設定時の注意事項および制約事項は、次のとおりです。

- 1 つの VDC で作成できるチェックポイント コピーの最大数は 10 です。
- ある VDC のチェックポイント ファイルを別の VDC に適用することはできません。
- チェックポイント コンフィギュレーションと比較した場合に、実行コンフィギュレーションのグローバル コンフィギュレーション部分に変更がある場合、非デフォルト VDC のチェックポイント コンフィギュレーションは適用できません。
- チェックポイント ファイル名の長さは、最大 20 文字です。
- チェックポイント ファイル名を「auto」の単語で始めることはできません。
- チェックポイント ファイル名を「summary」または「summary」の省略形にすることはできません。
- 任意の 1 時点で、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実行できるのは、1 つの VDC で 1 ユーザだけです。
- チェックポイント ファイルはシステム リロード後も使用できます。**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。

Session Manager 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。

- Session Manager がサポートするのは、ACL（アクセス コントロール リスト）機能だけです。
- 1 つの VDC で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。
- アクティブ セッションの進行中に ISSU（インサービス ソフトウェア アップグレード）を実行することはできません。セッションをコミットして保存するか、または打ち切ってから ISSU を実行する必要があります。
- 設定できるコマンドは、1 つの VDC のすべてのセッションで最大 20K です。

ロールバックの設定

ここでは、次の内容について説明します。

- チェックポイントの作成（p.5-5）
- ロールバックの実装（p.5-6）

（注）　Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する 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

1 つの VDC で作成できるコンフィギュレーション チェックポイント コピーの最大数は 10 です。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. ***checkpoint*** [*name*]
2. **show checkpoint** [*name*]

詳細な手順

	コマンド	目的
ステップ 1	**checkpoint** [*name*] **例：** switch# checkpoint stable	実行コンフィギュレーションのチェックポイント コピーを作成します。名前には最大 79 の英数字を使用できます。名前を指定しなかった場合、Cisco NX-OS はチェックポイント名を ‘auto-<number> に設定します。number は 1 ～ 10 です。
ステップ 2	**show checkpoint** [*name*] **例：** switch# show checkpoint stable	（任意）チェックポイント ファイルの内容を表示します。

現在のコンフィギュレーションのチェックポイント コピーを作成する例を示します。

```
switch# checkpoint stable
```

チェックポイント ファイルを削除するには、次のコマンドを使用します。

コマンド	目的
no checkpoint *name* **例：** switch# no checkpoint stable	チェックポイント ファイルを削除します。

ロールバックの実装

保存したチェックポイント ファイルの 1 つにコンフィギュレーション ロールバックを実装できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **show diff rollback-patch** {**checkpoint** *name* | **running-config** | **startup-config**} {**checkpoint** *name* | **running-config** | **startup-config**}
2. **rollback running-config checkpoint** *name* [**atomic** | **best-effort** | **stop-at-first-failure**]

詳細な手順

	コマンド	目的
ステップ 1	**show diff rollback-patch** {**checkpoint** *name* | **running-config** | **startup-config**} {**checkpoint** *name* | **running-config** | **startup-config**} **例：** `switch# show diff rollback-patch checkpoint stable running-config`	コピー元ファイルとコピー先ファイル間の相違を表示します。名前は任意の英数字ストリングにできます。
ステップ 2	**rollback running-config checkpoint** *name* [**atomic** | **best-effort** | **stop-at-first-failure**] **例：** `switch# rollback running-config checkpoint stable`	設定したチェックポイント ファイルのロールバックを実装します。任意で次のロールバック 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。 デフォルトは best-effort です。

ロールバックを発生させる例を示します。

```
switch# rollback running-config checkpoint stable
```

Session Manager の設定

ここでは、次の内容について説明します。

（注）　Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

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

手順詳細

	コマンド	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にできます。

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。保管場所は bootflash:、slot0:、または volatile: にできます。

セッションでの ACL の設定

コンフィギュレーション セッション内で ACL を設定できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. ***configure session*** *name*
2. ACL コマンドを追加
3. **show configuration session** [*name*]

手順詳細

	コマンド	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にできます。

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 e 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`	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 6	**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。

セッションの確認

セッションを確認するには、セッション モードで次のコマンドを使用します。

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

セッションのコミット

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

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

セッションの保存

セッションを保存するには、セッション モードで次のコマンドを使用します。

コマンド	目的
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。保管場所は bootflash:、slot0:、または volatile: にできます。

セッションの廃棄

セッションを廃棄するには、セッション モードで次のコマンドを使用します。

コマンド	目的

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。

ロールバックおよび Session Manager の設定確認

ロールバックの設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show checkpoint *name*	チェックポイント ファイルの内容を表示します。
show checkpoint summary	現在の VDC に含まれるすべてのチェックポイント ファイルのリストを表示します。
show diff rollback-patch {**checkpoint** *name* | **running-config** | **startup-config**} {**checkpoint** *name* | **running-config** | **startup-config**}	2 つのコンフィギュレーション間の相違を表示します。

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。

Session Manager の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show 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 セッションの状況を表示します。
show configuration session summary	すべてのコンフィギュレーション セッションについて、要約を表示します。

ロールバックおよび Session Manager の設定例

チェックポイント ファイルを作成し、そのチェックポイントへのベストエフォート型ロールバックを実装する例を示します。

```
checkpoint stable
rollback running-config checkpoint stable
```

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

```
configure session name test2
 ip access-list acl2
  permit tcp any any

 interface Ethernet1/2
  ip access-group acl2 in
```

関連資料

コンフィギュレーション ファイルの詳細については、『*Cisco NX-OS Fundamentals Configuration Guide, Release 4.0*』を参照してください。

デフォルト設定

表 5-1 に、ロールバックおよび Session Manager パラメータのデフォルト設定を示します。

表 5-1　デフォルトのロールバック パラメータ

パラメータ	デフォルト
rollback type	best-effort

その他の関連資料

ロールバックの実装に関する詳細情報については、次の項を参照してください。

- 関連資料（p.5-11）
- 規格（p.5-11）

関連資料

関連項目	マニュアル名
ロールバックおよび Session Manager の CLI コマンド	『*Cisco NX-OS System Management Command Reference, Release 4.0*』
コンフィギュレーション ファイル	『*Cisco NX-OS Fundamentals Configuration Guide, Release 4.0*』
VDC	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

■ その他の関連資料

CHAPTER 6

メンテナンス ジョブのスケジューリング

この章では、デバイス上で Cisco NX-OS コマンド スケジューラを設定する方法について説明します。

ここでは、次の内容を説明します。

コマンド スケジューラに関する情報

Cisco NX-OS コマンド スケジューラを使用すると、ジョブ（一連の CLI コマンド）または複数のジョブをその後の指定した時刻にスケジューリングできます。ジョブは将来の指定された時刻にジョブを 1 回だけ実行することも、または定期的な間隔で実行することもできます。

この機能を使用すると、QoS（Quality Of Service）ポリシーの変更、データのバックアップ、コンフィギュレーションの保存などのジョブの実行をスケジューリングできます。

ここでは、次の内容について説明します。

コマンド スケジューラの概要

コマンド スケジューラは次の部分からなります。

- ジョブ — スケジュールで定義されたとおりに実行される一連の Cisco NX-OS CLI コマンド（EXEC モードおよびコンフィギュレーション モード）。
- スケジュール — 指定されたジョブの実行時期。1 つのスケジュールに複数のジョブを割り当てられます。

スケジュールは次のいずれかのモードで実行されます。

- 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インターバルで繰り返されます。次のタイプの定期的インターバルを設定できます。
  - Daily — Cisco NX-OS は 1 日 1 回、ジョブを実行します。
  - Weekly — Cisco NX-OS は 1 週間に 1 回、ジョブを実行します。
  - Monthly — Cisco NX-OS は毎月 1 回、ジョブを実行します。
  - Delta — Cisco XN-OS は指定時にジョブの実行を開始し、以後、ユーザが指定したインターバル（days:hours:minutes）でジョブを実行します。
- 1 回限定モード — Cisco NX-OS はユーザが指定した時期に 1 回だけジョブを実行します。

リモート ユーザ認証

コマンド スケジューラはスケジュール設定されたジョブを実行する前に、そのジョブを作成したユーザを認証します。Cisco NX-OS は、ローカル設定の認証および認証サーバを使用するリモート認証をサポートします。Cisco NX-OS は短時間だけ、リモート認証から得たユーザのクレデンシャルを保有します。しかし、この期間は、スケジューリングされたジョブをサポートできるほどの長さではありません。ジョブを作成するユーザには、認証パスワードをローカル設定する必要があります。これらのパスワードは、コマンド 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みなされません。

コマンド スケジューラはスケジューリングされたジョブを実行する前に、ローカル パスワードとリモート認証サーバから戻ったパスワードを照合して検証します。

実行ログ

コマンド スケジューラはログ ファイルを維持します。このログ ファイルには、実行したジョブの出力が含まれます。ジョブ出力がログ ファイルより大きい場合、このファイルに保管する出力が切り捨てられます。

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後またはソフトウェアのリロード後も使用可能です。

仮想化サポート

Cisco NX-OS は、ユーザがログインした VDC（Virtual Device Context; 仮想デバイス コンテキスト）で、スケジューラ ジョブを作成します。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

コマンド スケジューラのライセンス要件

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
NX-OS	コマンド スケジューラの使用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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide, Release 4.0*』を参照してください。

コマンド スケジューラの前提条件

コマンド スケジューラには、次の前提条件があります。

- 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場合は、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。
- スケジュールリングされたジョブを設定するには、network-admin または vdc-admin のユーザ権限が必要です。

設定時の注意事項および制約事項

コマンド 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。

- ジョブの実行時に次のいずれかの状況が発生すると、スケジューリングされたジョブは失敗します。
  - 機能のライセンスが期限切れになっていて、その機能に関連するコマンドが含まれたジョブがスケジューリングされている場合。
  - 機能がディセーブルになっているときに、その機能に関連するコマンドが含まれたジョブがスケジューリングされている場合。
  - スロットからモジュールを取り外したにもかかわらず、そのモジュールまたはスロットのインターフェイスに関連するコマンドがジョブに含まれていた場合。
- 時刻が設定されているかどうかを確認します。スケジューラではデフォルトの時刻が設定されません。スケジュールを作成し、ジョブをスケジュールに割り当てても、時刻を設定しなかった場合、そのスケジュールは開始されません。
- ジョブを定義するときに、インタラクティブまたは中断を伴うコマンド（**copy bootflash:** *file* **ftp:** *URI*、**write erase** など）がジョブの一部として指定されていないかどうかを確認します。ジョブはスケジューリングされた時刻に非インタラクティブ方式で実行されるからです。

コマンド スケジューラの設定

コマンド スケジューラを設定する手順は、次のとおりです。

ステップ 1　スケジューラをイネーブルにします（「コマンド スケジューラのイネーブル化」[p.6-5] を参照）。

ステップ 2　（任意）リモート ユーザ アクセスを許可します（「リモート ユーザ認証の設定」[p.6-6] を参照）。

ステップ 3　ジョブを定義します（「ジョブの定義」[p.6-7] を参照）。

ステップ 4　スケジュールを指定します（「スケジュールの指定」[p.6-9] を参照）。

ステップ 5　スケジュールド コンフィギュレーションを確認にします（「コマンド スケジューラの設定確認」[p.6-11] を参照）。

ここでは、次の内容について説明します。

コマンド スケジューラのイネーブル化

ジョブを設定してスケジューリングするには、コマンド スケジュール機能をイネーブルにしておく必要があります。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

詳細な手順

コマンド スケジューラをイネーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
feature scheduler **例：** `switch(config)# feature scheduler`	VDC でコマンド スケジューラをイネーブルにします。

VDC でコマンド スケジューラ機能をディセーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
no feature scheduler **例：** `switch(config)# no feature scheduler`	VDC でコマンド スケジューラ機能をディセーブルにします。

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、コマンド スケジューラを設定できます。

（注） AAA 認証では、コマンド スケジューラ ジョブを作成して設定する前に、リモート ユーザのクリアテキスト パスワードが要求されます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **scheduler aaa-authentication password** [**0** | **7**] *password*
3. **scheduler aaa-authentication username** *name* **password** [**0** | **7**] *password*
4. **show running-config | include “scheduler aaa-authentication”**
5. ***copy running-config startup-config***

詳細な手順

:

	コマンドまたはアクション	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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。
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）この設定変更を保存します。

（注） **show running-config** コマンドの出力において、スケジューラのリモート ユーザ パスワードは、必ず暗号形式で表示されます。コマンドに暗号オプション（**7**）があるのは、デバイスに ASCII コンフィギュレーションを適用できるようにするためです。

ジョブの定義

コマンド スケジューラを使用すると、ジョブを定義できます。ジョブ名を指定し、そのジョブで実行しなければならない CLI コマンド シーケンスを定義する必要があります。

注意 コマンド シーケンスの入力後は、コマンドの変更も削除もできません。変更するには、定義したジョブ名を明示的に削除してから、このプロセスをやり直す必要があります。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **scheduler job name** *string*
3. *command1*; [*command2*;*command2*;...]
4. **exit**
5. **show scheduler job** [**name**]
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 job name** *string*
	
	例：
	`switch(config)# scheduler job name bringup`
	`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。

ステップ 3	*command1*; [*command2*;*command2*;...]
	
	例：
	`switch(config-job)# config t;interface ethernet 2/1;no shutdown;show interface ethernet 2/1`
	`switch(config-job)# exit`
	`switch(config)#`
		指定のジョブに対応するアクション シーケンスを指定します。各コマンドをセミコロンで区切る必要があります。

ステップ 4	**exit**
	
	例：
	`switch(config-job)# exit`
	`switch(config)#`
		ジョブ コンフィギュレーション モードを終了し、ジョブを保存します。

ステップ 5	**show scheduler job** [*name*] **例：** `switch(config)# show scheduler job`	（任意）ジョブ情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

ジョブの削除

ジョブを削除するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
no scheduler job name *string* **例：** `switch(config)# no scheduler job name bringup`	定義済みのジョブとそのジョブで定義されているすべてのコマンドを削除します。

スケジュールの指定

ジョブの定義後、スケジュールを作成して、そのスケジュールにジョブを割り当てることができます。その後、実行する時刻を設定できます。必要に応じて、1 回だけ実行することも、定期的に実行することもできます。スケジュールに時刻を設定しなかった場合、スケジュールが実行されることはありません。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **scheduler schedule name** *string*
3. **job name** *strings*
4. **time daily** *time*
   time weekly [[*dow*:]*HH*:]*MM*
   time monthly [[*dow*:]*HH*:] *MM*
   time start {**now** | *start-time* | *delta-time*} [**repeat**]
5. **exit**
6. **show scheduler schedule** [**name**]
7. ***copy running-config startup-config***

詳細な手順

	コマンドまたはアクション	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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追加します。1 つのスケジュールに複数のジョブを追加できます。

	コマンドまたはアクション	目的
ステップ 4	**time daily** *time* **例：** `switch(config-schedule)# time daily 23:00`	指定のジョブを毎日、設定時刻に実行します。time 引数は HH:MM 形式で指定します。
	time weekly [[*dow:*]*HH*:]*MM* **例：** `switch(config-schedule)# time weekly Sun:23:00`	毎週実行することを指定します。dow 引数は、次のように指定できます。 • 曜日を表す整数。1 は日曜、2 は月曜、以下同様です。 • 曜日の省略形。たとえば、Sun は日曜日を表します。 引数全体の最大長は 10 です。
	time monthly [[*dm:*]*HH*:]*MM* **例：** `switch(config-schedule)# time monthly 28:23:00`	毎月実行することを指定します。dm 引数は日付を表す整数です。dm を 29、30、または 31 のいずれかに指定した場合は、各月の最終日にコマンドが自動的に実行されます。
	time start {**now** | *start-time* | *delta-time*} [**repeat**] **repeat** *interval* **例：** `switch(config-schedule)# time start now repeat 48:00`	ジョブの開始時刻および定期的に反復する間隔を指定します。 start-time の形式は [[[[yyyy:]mmm:]dd:]HH]:MM です。 delta-time では、スケジュールの設定後、ジョブの開始までの経過時間を指定します。delta-time の形式は +[[dd:]HH:]MM です。 反復間隔は [[dd:]HH:]MM の形式で指定します。各セクションは正の整数を表します。 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 5	**exit** **例：** `switch(config-schedule)# exit switch(config)#`	スケジュール コンフィギュレーション モードを終了し、スケジュールを保存します。
ステップ 6	**show scheduler schedule** [*name*] **例：** `switch(config)# show scheduler schedule`	（任意）スケジュール情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

time 引数の最上位フィールドは省略可能です。最上位フィールドを省略した場合、現在の時刻が値として使用されます。たとえば、現在の時刻が 2008 年 3 月 24 日 22 時 00 分の場合、コマンドは次のように実行されます。

- **time start 23:00 repeat 4:00:00** コマンドは開始時刻が 2008 年 3 月 24 日 23 時 00 分であることを意味します。
- **time daily 55** コマンドは、毎日 22 時 55 分という意味になります。
- **time weekly 23:00** コマンドは、毎週金曜日の 23 時 00 分を意味します。
- **time monthly 23:00** コマンドは、毎月 24 日の 23 時 00 分を表します。

（注） スケジュールに設定された時間間隔が割り当てられたジョブの実行に必要な時間より短い場合、その後のスケジュール実行は、最後のスケジュール反復の完了後、設定された時間間隔が経過して初めて実行されます。たとえば、スケジュールが 1 分間隔の実行になっていて、そのスケジュールに割り当てられたジョブの完了に 2 分かかるとします。最初のスケジュールが 22 時 00 分の場合、ジョブは 22 時 02 分に完了し、その後 1 分のインターバルを置いてから、次の実行が 22 時 03 分に発生し、22 時 05 分に完了します。

実行ログの設定

設定できる最大ログ ファイル サイズは 1024 KB です。デフォルトの実行ログ ファイル 大サイズは、16 KB です。

実行ログ ファイル サイズを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
***scheduler* logfile size** *value* **例：** `switch(config)# scheduler logfile size 1024`	ログ ファイルのサイズを設定します。範囲は 16 ～ 1024 KB です。デフォルトは 16 KB です。

このファイルを消去するには、**clear scheduler logfile** コマンドを使用します。

コマンド スケジューラの設定確認

コマンド スケジューラの設定情報を表示するには、次のコマンドを使用します。

コマンド	目的
show scheduler config	コマンド スケジューラの設定を表示します。
show scheduler job [**name** *string*]	設定されているジョブを表示します。
show scheduler logfile	スケジューラ実行ログ 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。

デフォルト設定

表 6-1 に、コマンド スケジューラ パラメータのデフォルト設定を示します。

表 6-1　コマンド スケジューラ パラメータのデフォルト設定

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

その他の関連資料

スケジュール ジョブの実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.6-12）
- 規格（p.6-12）

関連資料

関連項目	マニュアル名
コマンド スケジューラの CLI コマンド	『*Cisco NX-OS System Management Configuration Guide, Release 4.0*』
VDC	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 7

SNMP の設定

この章では、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。

ここでは、次の内容を説明します。

SNMP に関する情報

SNMP（簡易ネットワーク管理プロトコル）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 はネットワーク デバイスの監視や管理に使用される、標準化されたフレームワークと共通言語を提供します。

ここでは、次の内容について説明します。

SNMP 機能の概要

SNMP フレームワークは、3 つの部分からなります。

- SNMP マネージャ — SNMP を使用してネットワーク デバイスの動作を制御および監視するためのシステム。
- SNMP エージェント — 管理デバイス内部のソフトウェア コンポーネントで、デバイスに関するデータを維持し、必要に応じてこれらのデータを管理システムに伝えます。Cisco NX-OS 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間の関係を定義する必要があります。
- MIB（management information base; 管理情報ベース）— SNMP エージェント上の管理対象オブジェクトのコレクション。

SNMP は RFC 3411 ～ 3418 で定義されています。

（注） Cisco NX-OS は、SNMP セットをサポートしません。

Cisco NX-OS 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のセキュリティ形式を使用します。

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作成できるということです。これらの通知は、SNMP マネージャからの要求送信を必要としません。通知によって、不正なユーザ認証、再起動、接続の終了、ネイバー ルータとの接続切断、またはその他の重要イベントを示すことができます。

Cisco NX-OS はトラップまたは応答要求のどちらかとして 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 レシーバー 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（「VRF を使用する SNMP 通知レシーバーの設定」[p.7-12] を参照）。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。

トラップは応答要求より信頼性が低くなります。トラップの受信時に、SNMP マネージャが確認応答を送信しないので、トラップが受信されたかどうかを Cisco NX-OS が判断できないからです。応答要求を受信した場合、SNMP マネージャは SNMP 応答 PDU（プロトコル データ ユニット）を使用して、メッセージを確認します。応答がなかった場合、Cisco NX-OS はもう一度、応答要求を送信します。

複数のホスト レシーバーに通知を送信するように、Cisco NX-OS を設定できます。ホスト レシーバーの詳細については、「SNMP 通知レシーバーの設定」（p.7-11）を参照してください。

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 — 有効な送信元からのメッセージであることを判別します。
- 暗号化 — パケット内容のスクランブルによって、不正な送信元で判読できないようにします。

SNMPv3 は、セキュリティ モデルとセキュリティ レベルの両方を提供します。セキュリティ モデルは、ユーザおよびユーザに与えられている役割に合わせて設定される認証方式です。セキュリティ レベルは、セキュリティ モデル内で許可されるセキュリティ レベルです。セキュリティ モデルとセキュリティ レベルのコンビネーションによって、SNMP パケットを取り扱うときに使用するセキュリティ メカニズムが決まります。

ここでは、次の内容について説明します。

- SNMPv1、v2、v3 のセキュリティ モデルおよびセキュリティ レベル（p.7-3）
- ユーザベースのセキュリティ モデル（p.7-4）
- CLI および SNMP ユーザの同期（p.7-5）
- グループベースの SNMP アクセス（p.7-5）

SNMPv1、v2、v3 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 レベルによって、SNMP メッセージを開示から保護する必要があるか、メッセージの認証が必要かどうかが決定されます。セキュリティ モデル内に存在する各種セキュリティ レベルは、次のとおりです。

- noAuthNoPriv — 認証も暗号化も行わないセキュリティ レベル
- authNoPriv — 認証は行うが暗号化は行わないセキュリティ レベル
- authPriv — 認証と暗号化の両方を行うセキュリティ レベル

使用できるセキュリティ モデルは 3 種類あり、SNMPv1、SNMPv2c、および SNMPv3 です。セキュリティ レベルと組み合わされたセキュリティ モデルによって、SNMP メッセージの処理時に適用されるセキュリティ メカニズムが決まります。

表 7-1 に、セキュリティ モデルとセキュリティ レベルのコンビネーションが何を意味するかを示します。

表 7-1　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	動作
v1	noAuthNoPriv	コミュニティ ストリング	なし	コミュニティ ストリングの照合を使用して認証します。
v2c	noAuthNoPriv	コミュニティ ストリング	なし	コミュニティ ストリングの照合を使用して認証します。
v3	noAuthNoPriv	ユーザ名	なし	ユーザ名の照合を使用して認証します。
v3	authNoPriv	HMAC-MD5 または HMAC-SHA	なし	HMAC MD5 アルゴリズムまたは HMAC SHA に基づいて認証します。
v3	authPriv	HMAC-MD5 または HMAC-SHA	DES	HMAC-MD5 または HMAC-SHA アルゴリズムに基づいて認証します。CBC（暗号ブロック連鎖）DES（データ暗号規格）-56 規格に基づいた認証に加え、DES 56 ビット暗号化を行います。

ユーザベースのセキュリティ モデル

SNMPv3 User-Based Security Model（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。
- メッセージ起点認証 — ユーザのために受信したデータの起点として主張されたアイデンティティが確認されていることを保証します。
- メッセージの機密性 — 不正な個人、エンティティ、またはプロセスに対して、情報が使用可能になったり開示されたりしていないことを保証します。

SNMPv3 で管理操作が許可されるのは、設定ユーザおよび暗号化 SNMP メッセージによる場合だけです。

Cisco NX-OS では、SNMPv3 に対応する 2 種類の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、AES（高度暗号化規格）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に DES を使用するか、それとも 128 ビット AES 暗号化を使用するかを選択できます。**priv** オプションと **aes-128** トークンを組み合わせた場合は、このプライバシー パスワードが 128 ビットの AES 鍵を作成するためのものであることを意味します。AES priv パスワードは、8 文字以上の長さにできます。パスフレーズをクリア テキストで指定する場合は、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。ローカライズした鍵を使用する場合は、130 文字まで指定できます。

（注） 外部 AAA（認証、認可、アカウンティング）サーバを使用する SNMPv3 動作の場合は、外部 AAA サーバ上のユーザ コンフィギュレーションで、プライバシー プロトコルとして AES を使用する必要があります。

CLI および SNMP ユーザの同期

SNMPv3 のユーザ管理は、Access Authentication and Accounting（AAA）サーバ レベルで集中させることができます。この集中ユーザ管理によって、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活用できます。ユーザ認証が確認されると、SNMP PDU がさらに処理されます。また、ユーザ グループ名の保管に AAA サーバも使用されます。SNMP で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 / 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を変更すると、SNMP と AAA の両方について、データベースの同期が図られます。

Cisco NX-OS では次のように、ユーザ設定を同期させます。

- **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削除すると、SNMP と CLI の両方でユーザが削除されます。
- ユーザとロール（役割）間のマッピング変更は、SNMP と CLI で同期します。
- 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 / パスワードをローカライズした鍵 / 暗号形式で設定した場合、Cisco NX-OS はパスワードを同期させません。

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。このデフォルト値の変更方法については、「AAA 同期時間の変更」（p.7-18）を参照してください。

グループベースの SNMP アクセス

（注） *グループ*が業界全体で使用されている標準 SNMP 用語なので、この SNMP の項では、ロールのことをグループと言います。

SNMP のアクセス権は、グループ別に編成されます。SNMP の各グループは、CLI でのロールと同様です。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。

自分のユーザ名を作成すると、エージェントとの通信を開始し、管理者に自分のロールを設定してもらい、そのロールに自分を追加してもらうことができます。

SNMP および EEM

EEM（Embedded Event Manager）機能は、SNMP MIB オブジェクトを含めてイベントを監視し、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。

EEM の詳細については、第 10 章「Embedded Event Manager の設定」を参照してください。

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、VRF など、論理ネットワーク エンティティのインスタンスを複数サポートできます。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 Release 4.0(2) 以降のリリースでは、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。

SNMPv2 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 コンテキストを論理ネットワーク エンティティにマッピングする手順は、次のとおりです。

ステップ 1　SNMPv3 コンテキストを作成します。

ステップ 2　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。

ステップ 3　SNMPv3 コンテキストを論理ネットワーク エンティティにマッピングします。

ステップ 4　任意で、SNMPv3 コンテキストを SNMPv2 コミュニティにマッピングします。

詳細については、「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」（p.7-16）を参照してください。

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あとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

Cisco NX-OS は、VDC（Virtual Device Context; 仮想デバイス コンテキスト）ごとに SNMP インスタンスを 1 つずつサポートします。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

Cisco NX-OS Release 4.0(2) 以降のリリースでは、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。詳細については、「マルチインスタンス サポート」（p.7-6）を参照してください。

SNMP も VRF を認識します。特定の VRF を使用して、SNMP 通知ホスト レシーバーに接続するように SNMP を設定できます。通知が発生した VRF に基づいて、SHMP ホスト レシーバー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。詳細については、「VRF を使用する SNMP 通知レシーバーの設定」（p.7-12）を参照してください。

SNMP 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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

SNMP の前提条件

SNMP の前提条件は、次のとおりです。

- VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

設定時の注意事項および制約事項

SNMP 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。

- Cisco NX-OS は一部の SNMP MIB について、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。
  http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

SNMP の設定

ここでは、次の内容について説明します。

（注） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

SNMP ユーザの設定

SNMP ユーザを設定できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

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

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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]] **例：** switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh	認証およびプライバシー パラメータを指定して、SNMP ユーザを設定します。パスフレーズには最大 64 の英数字を使用できます。大文字と小文字を区別します。**localizekey** 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-callhome)# show snmp user	（任意）1 つまたは複数の 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）この設定変更を保存します。

SNMP のコンタクトおよびロケーション情報を設定する例を示します。

```
switch# config t
switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh
```

SNMP メッセージ暗号化の強制

着信要求の認証または暗号化を求めるように、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。プライバシーを強化する場合、Cisco NX-OS は noAuthoNoPriv または authNoPriv の securityLevel パラメータを使用している SNMPv3 PDU に、authorizationError で応答します。

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

コマンド	目的
snmp-server user *name* enforcePriv **例：** switch(config)# snmp-server user Admin enforcePriv	このユーザに SNMP メッセージの暗号化を強制します。

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

コマンド	目的
snmp-server globalEnforcePriv **例：** `switch(config)# snmp-server globalEnforcePriv`	すべてのユーザに SNMP メッセージの暗号化を強制します。

複数のロールに SNMPv3 ユーザを割り当てる場合

SNMP ユーザの設定後、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 ロールに属しているユーザだけです。

SNMP ユーザにロールを割り当てるには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
snmp-server user *name group* **例：** `switch(config)# snmp-server user Admin superuser`	この SNMP ユーザを設定済みのユーザ ルールに関連付けます。

SNMP コミュニティの作成

SNMPv1 または SNMPv2c に対応する SNMP コミュニティを作成できます。

SNMP コミュニティ ストリングを作成するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
snmp-server community *name 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。

SNMP 通知レシーバーの設定

複数のホスト レシーバーに対して SNMP 通知を作成するように、Cisco NX-OS を設定できます。

SNMPv1 トラップのホスト レシーバーを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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 レシーバーを設定します。コミュニティには最大 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。

SNMPv2c トラップまたは応答要求のホスト レシーバーを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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 トラップまたは応答要求のホスト レシーバーを設定します。コミュニティには最大 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。

SNMPv3 トラップまたは応答要求のホスト レシーバーを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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`	SNMPv2c トラップまたは応答要求のホスト レシーバーを設定します。ユーザ名には最大 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。

（注） SNMP マネージャは SNMPv3 メッセージを認証して解読するために、Cisco NX-OS デバイスの SNMP engineID 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。

通知ターゲット ユーザの設定

通知ホスト レシーバーに SNMPv3 応答要求通知を送信するには、デバイス上で通知ターゲット 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 レシーバーへの SNMPv3 応答要求通知メッセージを暗号化します。

（注） 受信した INFORM PDU を認証して解読する場合、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 レシーバーに必要です。

通知ターゲット ユーザを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
snmp-server user *name* [auth {md5 | sha} *passphrase* [auto] [**p**riv [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 レシーバーの engineID を指定して、通知ターゲット ユーザを設定します。engineID の形式は、12 桁のコロンで区切った 10 進数字です。

VRF を使用する SNMP 通知レシーバーの設定

SNMP 通知レシーバーの 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。

設定された VRF を使用してホスト レシーバーに接続するように Cisco NX-OS を設定できます。

ホスト レシーバーへの通知の送信に使用する VRF を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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 レシーバーと通信するように SNMP を設定します。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。
no snmp-server host *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*] **例：** switch(config)# no snmp-server host 192.0.2.1 use-vrf Blue	設定済みホストの VRF 到達可能性情報を削除し、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTarget VrfTable からエントリを削除します。 ホスト設定は削除しません。

通知が発生した VRF に基づいて、通知をフィルタリングするように Cisco NX-OS を設定できます。

設定された VRF に基づいて通知をフィルタリングするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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 レシーバーへの通知をフィルタリングします。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。
no snmp-server host *ip-address* **filter_vrf** *vrf_name* **例：** `switch(config)# no snmp-server host 192.0.2.1 filter_vrf Red`	設定済みホストの VRF フィルタ情報を削除し、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTarget VrfTable からエントリを削除します。 このコマンドによってホスト設定は削除されません。

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。通知の名前を指定しなかった場合、Cisco NX-OS はすべての通知をイネーブルにします。

表 7-2 に、Cisco NX-OS に関する通知をイネーブルにする、CLI コマンドを示します。

（注） **snmp-server enable traps** CLI コマンド を使用すると、設定されている通知ホスト サーバに応じて、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。

表 7-2　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**
CISCO-STP-BRIGE-MIB	**snmp-server enable traps bridge**
CISCO-CALLHOME-MIB	**snmp-server enable traps callhome**
EIGRP4-MIB	**snmp-server enable traps eigrp**
ENITY-MIB、 CISCO-ENTITY-FRU-CONTROL-MIB、 CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity fru**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license**
IF-MIB	**snmp-server enable traps link**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-STPX-MIB	**snmp-server enable traps stpx**

ライセンス通知は、デフォルトでイネーブルです。その他の通知はすべて、デフォルトでディセーブルです。

指定した通知をイネーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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。
snmp-server enable traps bridge [**newroot** | **topologychange**] **例：** `switch(config)# snmp-server enable traps bridge newroot`	STP ブリッジ SNMP 通知をイネーブルにします。
snmp-server enable traps callhome **例：** `switch(config)# snmp-server enable traps callhome`	CISCO-CALLHOME-MIB SNMP 通知をイネーブルにします。
snmp-server enable traps eigrp **例：** `switch(config)# snmp-server enable traps eigrp`	EIGRPv4-MIB SNMP 通知をイネーブルにします。
snmp-server enable traps entity [**fru**] **例：** `switch(config)# snmp-server enable traps entity`	ENTITY-MIB SNMP 通知をイネーブルにします。
snmp-server enable traps license **例：** `switch(config)# snmp-server enable traps license`	ライセンス SNMP 通知をイネーブルにします。
snmp-server enable traps link **例：** `switch(config)# snmp-server enable traps link`	リンク SNMP 通知をイネーブルにします。
snmp-server enable traps port-security **例：** `switch(config)# snmp-server enable traps port-security`	ポート セキュリティ SNMP 通知をイネーブルにします。

コマンド	目的
snmp-server enable traps snmp [**authentication**] **例：** `switch(config)# snmp-server enable traps snmp`	SNMP エージェント通知をイネーブルにします。
snmp-server enable traps stpx [**inconsistency** | **loop-inconsistency** | **root-inconsistency**] **例：** `switch(config)# snmp-server enable traps stpx root-inconsistency`	STPX SNMP 通知をイネーブルにします。

インターフェイスに関する linkUp/linkDown 通知のディセーブル化

個々のインターフェイスに関する linkUp および linkDown 通知をディセーブルにできます。これにより、フラッピング インターフェイス（アップとダウン間の移行を繰り返しているインターフェイス）に関する通知を制限できます。

インターフェイスに関する linkUp/linkDown 通知をディセーブルにするには、インターフェイス 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
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されています。

TCP による SNMP のワンタイム認証のイネーブル化

TCP セッションでの 1 回限りの SNMP 認証をイネーブルにできます。

TCP による SNMP のワンタイム認証をイネーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
snmp-server tcp-session [**auth**] **例：** `switch(config)# snmp-server tcp-session`	TCP セッションでの 1 回限りの SNMP 認証をイネーブルにします。デフォルトはディセーブルです。

SNMP スイッチのコンタクト（連絡先）およびロケーション情報の指定

32 文字までの長さで（スペースを含まない）のスイッチ コンタクト情報を指定できます。さらに、スイッチ ロケーションを指定できます。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***snmp-server contact*** *name*
3. **snmp-server location** *name*
4. **show snmp**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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。
ステップ 5	***copy running-config startup-config*** **例：** switch(config)# copy running-config startup-config	（任意）この設定変更を保存します。

SNMP のコンタクトおよびロケーション情報を設定する例を示します。

```
switch# config t
switch(config)# snmp contact Admin
switch(config)# snmp location Lab-7
```

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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

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 インスタスの詳細については、『*Cisco NX-OS Unicast Routing Configuration Guide, Release 4.0*』または『*Cisco NX-OS Multicast Routing Configuration Guide, Release 4.0*』を参照してください。

手順概要

1. **config t**
2. **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]
3. **snmp-server mib community-map** *community-name* **context** ***context-name***
4. **show snmp context**
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 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。
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	（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）この設定変更を保存します。

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# config t
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# config t
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
```

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を削除するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
no snmp-server context *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*] **例：** `switch(config)# no snmp-server context public1`	SNMP コンテキストとプロトコルインスタンス、VRF、またはトポロジ間のマッピングを削除します。名前には最大 32 の英数字を使用できます。 **（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てはなりません。**instance**、**vrf**、または **topology** キーワードを使用すると、コンテキストとゼロ長ストリング間のマッピングが設定されます。

SNMP のディセーブル化

デバイス上で SNMP プロトコルをディセーブルにできます。

SNMP プロトコルをディセーブルにするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
no snmp-server protocol enable **例：** `switch(config)# no snmp-server protocol enable`	SNMP プロトコル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されています。

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。

AAA 同期時間を変更するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
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。デフォルト値は 3600 です。

SNMP の設定確認

SNMP 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 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。
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 session	SNMP セッションを表示します。
show snmp trap	SNMP 通知がイネーブルなのかディセーブルなのかを表示します。
show snmp user	SNMPv3 ユーザを表示します。

SNMP の設定例

Blue VRF を使用してある通知ホスト レシーバーに Cisco linkUp/linkDown 通知を送信するように 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。

```
config t
 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
```

デフォルト設定

表 7-3 に、SNMP パラメータのデフォルト設定を示します。

表 7-3　デフォルトの SNMP パラメータ

パラメータ	デフォルト
license notifications	イネーブル

その他の関連資料

SNMP の実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.7-20）
- 規格（p.7-20）
- MIB（p.7-20）

関連資料

関連項目	マニュアル名
SNMP CLI コマンド	『*Cisco NX-OS System Management Configuration Guide, Release 4.0*』。URL は次のとおりです。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/configuration/guide/sm_nx-os_config.html
VDC および VRF	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』。URL は次のとおりです。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html
MIB	http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

MIB

MIB	MIB のリンク
• SNMP-COMMUNITY-MIB • SNMP-FRAMEWORK-MIB • SNMP-NOTIFICATION-MIB • SNMP-TARGET-MIB • SNMPv2-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

SNMP 機能の履歴

表 7-4 に、この機能のリリース履歴を示します。

表 7-4　SNMP 機能の履歴

機能名	リリース	機能情報
SNMP AAA 同期	4.0(3)	同期したユーザ設定のタイムアウトを変更する機能を追加
SNMP プロトコル	4.0(3)	SNMP プロトコルをディセーブルにする機能を追加

CHAPTER 8

RMON の設定

この章では、デバイス上で RMON 機能を設定する方法について説明します。

ここでは、次の内容を説明します。

RMON の概要

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、SNMP（簡易ネットワーク管理プロトコル）IETF（インターネット技術特別調査委員会）の標準モニタリング仕様です。Cisco NX-OS は Cisco NX-OS デバイスを監視できるように、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、指定されたインターバルで特定の MIB（management information base; 管理情報ベース）オブジェクトを監視し、指定されたしきい値になるとアラームを発生させ、次のしきい値でアラームをリセットします。RMON イベントでアラームを使用すると、RMON アラームの発生時に、ログ エントリまたは SNMP 通知を生成できます。

RMON はデフォルトでディセーブルであり、Cisco NX-OS ではイベントもアラームも設定されません。RMON アラームおよびイベントを設定するには、CLI または SNMP 互換ネットワーク管理ステーションを使用します。

ここでは、次の内容について説明します。

- RMON アラーム（p.8-2）
- RMON イベント（p.8-3）
- ハイ アベイラビリティ（p.8-3）
- 仮想化サポート（p.8-3）

RMON アラーム

SNMP INTEGER タイプとして解決される MIB オブジェクトであれば、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は、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 インターバル — MIB オブジェクトのサンプル値を収集するために Cisco NX-OS が使用するインターバル。
- 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 サンプルでは、2 つの連続するサンプルを使用し、その差を計算します。
- 上限しきい値 — Cisco NX-OS が上限アラームを発生させるか、下限アラームをリセットする値。
- 下限しきい値 — Cisco NX-OS が下限アラームを発生させるか、上限アラームをリセットする値。
- イベント — アラーム（上限または下限）発生時に Cisco NX-OS が開始するアクション。

（注） 64 ビット整数 MIB オブジェクトにアラームを設定するには、hcalarms オプションを使用します。

たとえば、エラー カウンタ MIB オブジェクトにデルタ タイプの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ると、SNMP 通知を送信するイベントを発生させ、上限アラーム イベントを記録できます。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らないかぎり、再び発生することはありません。

（注） 下限しきい値は上限しきい値未満でなければなりません。

RMON イベント

RMON アラームごとに特定のイベントを関連付けることができます。RMON がサポートするイベント タイプは、次のとおりです。

- SNMP notification — 関連付けられたアラームの発生時に、SNMP risingAlarm または fallingAlarm 通知を送信します。
- Log — 関連付けられたアラームの発生時に、RMON ログ テーブルにエントリを追加します。
- Both — 関連付けられたアラームの発生時に、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。

下限アラームと上限アラームとで異なるイベントを指定できます。

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あとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

Cisco NX-OS は、VDC（Virtual Device Context; 仮想デバイス コンテキスト）ごとに RMON インスタンスを 1 つずつサポートします。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。次の URL にアクセスして、『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/interfaces/configuration/guide/if_nxos_book.html

RMON は VRF（Virtual Routing and Forwarding）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

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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

RMON の前提条件

RMON の前提条件は、次のとおりです。

VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始する必要があります。次の URL にアクセスし、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/interfaces/configuration/guide/if_nxos_book.html

設定時の注意事項および制約事項

RMON 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。

- SNMP notification 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ーを設定する必要があります。
- RMON アラームを設定できるのは、整数として解決される MIB オブジェクトに限られます。

RMON の設定

ここでは、次の内容について説明します。

- RMON アラームの設定（p.8-4）
- RMON イベントの設定（p.8-5）

（注）　Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

RMON アラームの設定

RMON アラームは、整数ベースのあらゆる SNMP MIB オブジェクトに設定できます。

任意で次のパラメータを指定できます。

- 上限または下限しきい値が指定限度を超えた場合に発生させるイベントの数。
- アラームのオーナー。

操作の前に

SNMP ユーザを設定し、SNMP 通知をイネーブルにしてあることを確認します（「SNMP の設定」[p.7-8] を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***rmon alarm*** *index mib-object sample-interval* {**absolute** | **delta**} **rising-threshold** *value* [***event-index***] f**alling-threshold** *value* [***event-index***] [**owner** ***name***]

   または

   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*]
3. **show rmon** [**alarms** | **hcalarms**]
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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 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。オーナー名には任意の英数字を使用できます。
	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 大容量アラームを作成します。値の範囲は –2147483647 ～ 2147483647 です。オーナー名には任意の英数字を使用できます。 ストレージ タイプの範囲は 1 ～ 5 です。
ステップ 3	**show rmon** {**alarms** | **hcalarms**} **例：** switch(config)# show rmon alarms	（任意）RMON アラームまたは大容量アラーム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）この設定変更を保存します。

RMON イベントの設定

RMON アラームと関連付ける 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使用できます。

操作の前に

SNMP ユーザを設定し、SNMP 通知をイネーブルにしてあることを確認します（「SNMP の設定」[p.7-8] を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

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

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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**] [**owner** *name*] **例：** switch(config)# rmon event 1 trap	RMON イベントを設定します。説明のストリングおよびオーナー名には、任意の英数字を使用できます。
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）この設定変更を保存します。

RMON の設定確認

RMON 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 **rmon alarms**	RMON アラーム情報を表示します。
show rmon events	RMON イベント情報を表示します。
show rmon hcalarms	RMON hcalarm 情報を表示します。
show rmon logs	RMON ログ情報を表示します。

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例を示します。

```
config t
 rmon alarm 20 1.3.6.1.2.1.2.2.1.14 2900 delta rising-threshold 1500 1
falling-threshold 0 owner test
 rmon event 1 trap
```

関連資料

次の関連項目を参照してください。

- SNMP の設定（p.7-1）

デフォルト設定

表 8-1 に、RMON パラメータのデフォルト設定を示します。

表 8-1　デフォルトの RMON パラメータ

パラメータ	デフォルト
アラーム	未設定
イベント	未設定

その他の関連資料

RMON の実装に関する詳細情報については、次の項を参照してください。

- 関連資料（p.8-8）
- 規格（p.8-8）
- MIB（p.8-8）

関連資料

関連項目	マニュアル名
RMON CLI コマンド	『*Cisco NX-OS System Management Command Reference, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html
VDC および VRF	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/interfaces/configuration/guide/if_nxos_book.html

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

MIB

MIB	MIB のリンク
• RMON-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

CHAPTER 9

オンライン診断機能の設定

この章では、デバイス上で Generic Online Diagnostics（GOLD）機能を設定する方法について説明します。

ここでは、次の内容を説明します。

（注） この章で扱うコマンドの詳細な構文および使用方法については、『*Cisco NX-OS System Management Command Reference, Release 4.0*』を参照してください。

オンライン診断機能に関する情報

オンライン診断機能を使用すると、システムをたえず監視することによって、ハードウェアおよび内部データ パスが設計通りに動作しているかどうかを確認できます。この機能によって、障害を迅速に分離できます。

ここでは、次の内容について説明します。

オンライン診断機能の概要

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。破壊モードのループバック テストといった中断を伴うオンライン診断テストおよび ASIC レジスタ検査などの中断を伴わないオンライン診断テストは、起動時、ライン モジュールの活性挿抜（online insertion and removal: 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ます。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。

起動診断

起動診断では、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、デバイスがトラフィックの転送にそのモジュールを使用しないうちに、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、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。表 9-1 で、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 9-1　起動診断

テスト ID	診断テスト	説明
1	ManagementPortLoopback	中断を伴うテスト、非オンデマンド型テスト モジュールの管理ポートでループバックをテスト
2	EOBCPortLoopback	中断を伴うテスト、非オンデマンド型テスト イーサネット帯域外
4	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
5	CryptoDevice	中断を伴わないテスト モジュールにおける Cisco Trusted Security（CTS）デバイスの初期化を検査

（注） モジュールでは、テスト ID 1 を使用し、非破壊モードの起動テストとして EOBCPortLoopback テストが実行されます。

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

起動診断テストをバイパスするように Cisco NX-OS 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。「起動診断レベルの設定」（p.9-7）を参照してください。

ランタイム診断

ランタイム診断は HM（ヘルス モニタリング）診断もといいます。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。表 9-2 で、ヘルス モニタリング診断テストについて説明し、スーパーバイザ用のテスト ID を示します。

表 9-2　非破壊モードのスーパーバイザ用ヘルス モニタリング診断

テスト ID	診断テスト	デフォルトのインターバル	デフォルト設定	説明
3	ASICRegisterCheck	20 秒	アクティブ	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。
6	NVRAM	30 秒	アクティブ	スーパーバイザの NVRAM ブロックの健全性を確認します。
7	RealTimeClock	5 分	アクティブ	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。
8	PrimaryBootROM	30 分	アクティブ	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。
9	SecondaryBootROM	30 分	アクティブ	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。
10	CompactFlash	30 分	アクティブ	内蔵コンパクト フラッシュ デバイスにアクセスできるかどうかを確認します。
11	ExternalCompactFlash	30 分	アクティブ	外部コンパクト フラッシュ デバイスにアクセスできるかどうかを確認します。
12	PwrMgmtBus	30 秒	アクティブ	スタンバイの電源管理制御バスを確認します。
13	SpineControlBus	30 秒	アクティブ	スタンバイ スパイン モジュール制御バスの使用可能性を確認します。
14	SystemMgmtBus	30 秒	アクティブ	スタンバイ システム管理バスの使用可能性を確認します。

表 9-3 で、モジュールのヘルス モニタリング診断テストについて説明します。

表 9-3　非破壊モードのモジュール用ヘルス モニタリング診断

テスト ID	診断テスト	デフォルトのインターバル	デフォルト設定	説明
2	ASICRegisterCheck	1 分	アクティブ	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。
3	PrimaryBootROM	30 分	アクティブ	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。
4	SecondaryBootROM	30 分	アクティブ	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。
5	PortLoopback	15 分	非アクティブ	スーパーバイザ モジュールからモジュール上の ADMIN DOWN ステートの物理ポートへのパケット パスをテストします。
6	RewriteEngineLoopback	10 分	アクティブ	Rewrite Engine ASIC デバイスまでのすべてのポートについて、非破壊モードのループバック テストを実行します。

オンデマンド診断

オンデマンド テストによって、障害を局所化して解決を図ることができます。オンデマンド診断テストが必要になるのは、通常、次の状況のいずれかの場合です。

- 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。詳細については、「オンデマンド診断テストの開始または中止」（p.9-9）を参照してください。

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。詳細については、「診断テストのアクティブ化」（p.9-8）を参照してください。

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、ハイ アベイラビリティ ソフトウェア コンポーネントに伝え、スイッチオーバーが決定されるようにします。

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あとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

Cisco NX-OS は、VDC（Virtual Device Context; 仮想デバイス コンテキスト）ごとにオンライン診断インスタンスを 1 つずつサポートします。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

オンライン診断機能は VRF（Virtual Routing and Forwarding）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

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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

オンライン診断機能の前提条件

オンライン診断機能の前提条件は、次のとおりです。

- VDC を設定する場合は、Advanced services ライセンスをインストールしてから、設定する VDC にアクセスします。詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

注意事項および制約事項

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。

オンライン診断機能の設定

ここでは、次の内容について説明します。

- 起動診断レベルの設定（p.9-7）
- 診断テストのアクティブ化（p.9-8）
- オンデマンド診断テストの開始または中止（p.9-9）

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する 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。

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） 起動時のオンライン診断レベルはを **complete** に設定することを推奨します。起動時オンライン診断テストのバイパスは推奨できません。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config *t***
2. **diagnostic bootup level [*complete* | *bypass*]**
3. **show diagnostic bootup level**
4. **copy running-config startup-config**

手順詳細

	コマンド	目的

ステップ 1	**`config t`**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**`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 — 起動診断テストをまったく実行しません。

	コマンド	目的
ステップ 3	**show diagnostic bootup level** **例：** `switch(config)# show diagnostic bootup level`	（任意）デバイスに現在設定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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config** *t*
2. **diagnostic monitor interval module** *slot* **test** [*test-id* | *name* | **all**] **hour** *hour* **min** *minutes* **second** *sec*
3. **diagnostic monitor module** *slot* **test** [*test-id* | *name* | **all**]
4. **show diagnostic content module** *slot*

手順詳細

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**diagnostic monitor interval module** *slot* **test** [*test-id* | *name* | **all**] **hour** *hour* **min** *minutes* **second** *sec* **例：** `switch(config)# diagnostic monitor interval module 6 test 3 hour 1 min 0 sec 0`	（任意）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 • name — 最大 32 の英数字を使用できます。大文字と小文字を区別します。 • hour — 範囲は 0 ～ 23 時間 • minute — 範囲は 0 ～ 59 分 • second — 範囲は 0 ～ 59 秒

	コマンド	目的
ステップ 3	**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。大文字と小文字を区別します。
ステップ 4	**show diagnostic content module** *slot* **例：** `switch(config)# show diagnostic content module 6`	(任意) 診断テストおよび対応するアトリビュートの情報を表示します。

診断テストを非アクティブとして設定する場合

診断テストを非アクティブとして設定できます。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。

診断テストを非アクティブに設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
no diagnostic monitor module *slot* **test** [*test-id* | *name* | **all**] **例：** `switch(config)# no diagnostic monitor interval module 6 test 3`	指定されたテストを非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。大文字と小文字を区別します。

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したり中止したりできます。任意で、このテストを繰り返す回数を変更したり、テストが失敗した場合のアクションを変更したりできます。

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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **diagnostic ondemand iteration** *number*
2. **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. **show diagnostic content module** *slot*

手順詳細

	コマンド	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。有効値の範囲は 1 ～ 999 であり、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 テストが実行した場合のアクションを設定します。*num-fails* の範囲は 1 ～ 999 であり、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。テスト名には、大文字と小文字を区別した英数字を 32 文字まで使用できます。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。テスト名には、大文字と小文字を区別した英数字を 32 文字まで使用できます。
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。

診断結果の消去

診断テスト結果を消去できます。

診断テスト結果を消去するには、任意のモードで次のコマンドを使用します。

コマンド	目的
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

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。

診断テスト結果のシミュレーションを行うには、任意のモードで次のコマンドを使用します。

コマンド	目的
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。

シミュレーションした診断テスト結果を消去するには、任意のモードで次のコマンドを使用します。

コマンド	目的
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* 範囲は 1 ～ 14 です。

オンライン診断の設定確認

オンライン診断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* diagnostic bootup level**	起動診断に関する情報を表示します。
***show* diagnostic content module** *slot*	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 ondemand setting**	オンデマンド診断に関する情報を表示します。
***show* diagnostic results 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。

オンライン診断テストの設定例

モジュール 6 ですべてのオンデマンド テストを開始する例を示します。

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

モジュール 6 でテスト 2 をアクティブにして、テスト インターバルを設定する例を示します。

```
conf t
 diagnostic monitor module 6 test 2
 diagnostic monitor interval module 6 test 2 hour 3 min 30 sec 0
```

デフォルト設定

表 9-4 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。

表 9-4　デフォルトのオンライン診断パラメータ

パラメータ	デフォルト
起動診断レベル	complete
中断を伴わないテスト	active

その他の関連資料

オンライン診断の実装に関する詳細情報については、次の項を参照してください。

- 関連資料（p.9-12）
- 規格（p.9-12）

関連資料

関連項目	マニュアル名
オンライン診断 CLI コマンド	『*Cisco NX-OS System Management Command Reference, Release 4.0*』
VDC および VRF	『*Cisco NX-OS Virtual Device Context Command Reference, Release 4.0*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 10

Embedded Event Manager の設定

この章では、EEM（Embedded Event Manager）を設定して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。

ここでは、次の内容を説明します。

EEM 情報

EEM はデバイス上で発生するイベントを監視し、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。

ここでは、次の内容について説明します。

EEM の概要

EEM は次の 3 種類の主要コンポーネントからなります。

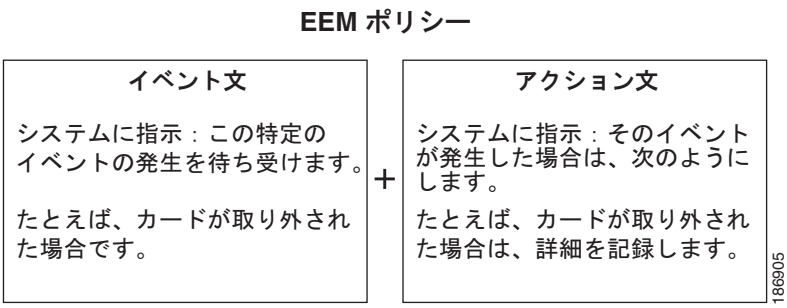
- イベント文 — 別の Cisco NX-OS コンポーネントから監視し、アクション、回避策、または通知が必要になる可能性のあるイベント。
- アクション文 — E メールの送信、インターフェイスのディセーブル化など、イベントから回復するために EEM が実行できるアクション。
- ポリシー — イベントのトラブルシューティングまたはイベントからの回復を目的とした 1 つまたは複数のアクションとペアになったイベント。

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、求めるイベントとともに、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。

図 10-1 に、EEM ポリシーの基本的な 2 種類の文を示します。

図 10-1　EEM ポリシー文



EEM ポリシーを設定するには、CLI または VSH スクリプトを使用します。

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（複数可）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）、トリガー 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 アクションを EEM が開始したあとで、ユーザ ポリシーで指定したアクションが行われます。ユーザ ポリシーを設定する場合には、「CLI によるユーザ ポリシーの定義」（p.10-7）を参照してください。

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別するには、**show event manager system-policy** コマンドを使用します。

上書きポリシーを設定する場合には、「ポリシーの上書き」（p.10-13）を参照してください。

（注） **show running-config eem** コマンドを使用して、各ポリシーの設定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すべてのイベントが上書きされます。

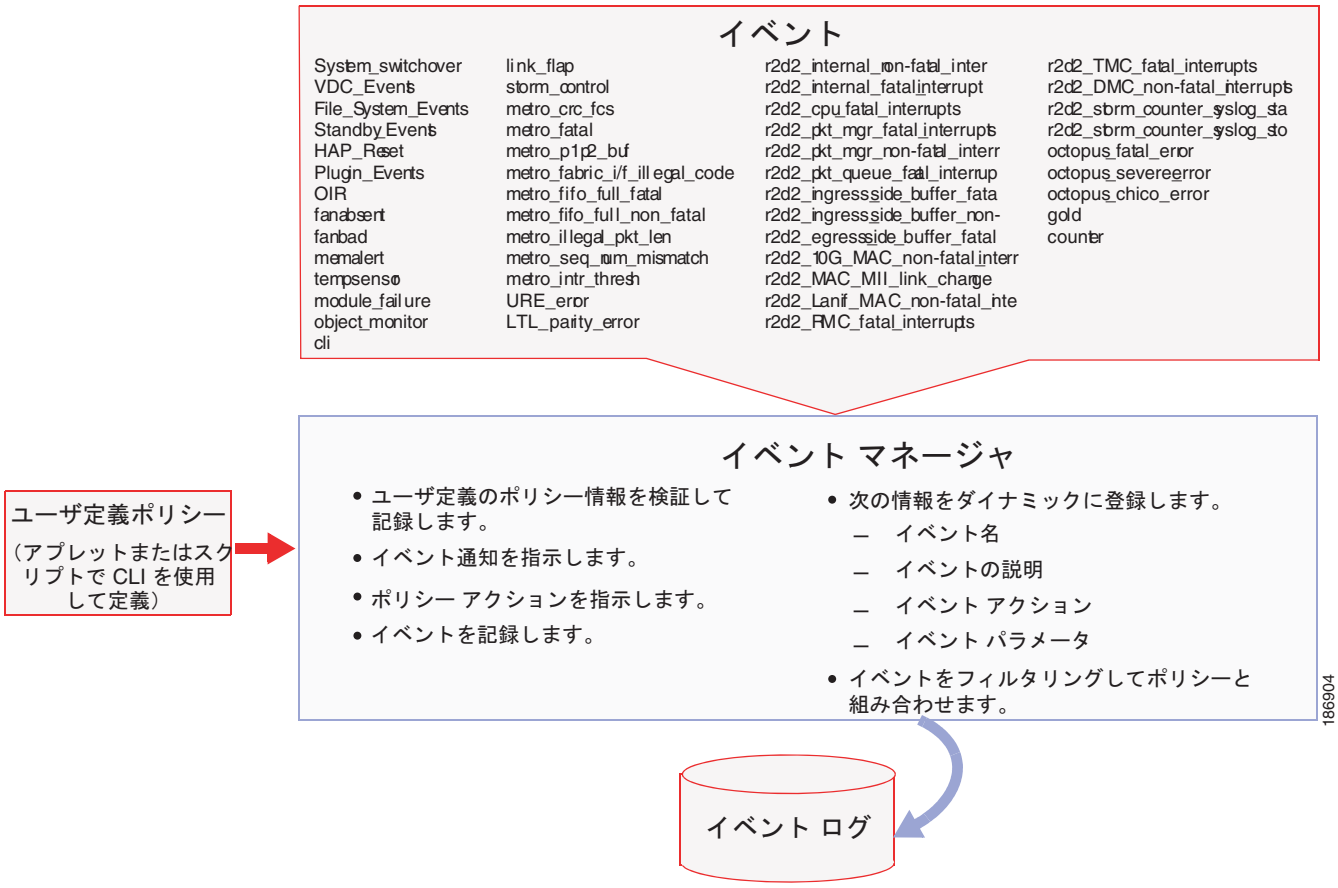
イベント文

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。

図 10-2 に、EEM が処理するイベントを示します。

図 10-2　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。設定できるイベント文は、1 つのポリシーに 1 つだけです。

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。

アクション文

アクション文では、ポリシーによって開始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） 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、VHS 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。スクリプト ポリシーの作成後、そのポリしイーをデバイスにコピーしてアクティブにします。スクリプト ポリシーを設定する場合には、「VSH スクリプトによるポリシーの定義」（p.10-12）を参照してください。

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 E 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。

例 10-1 に、「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。

例 10-1　アクション文

```
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、例 10-2 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。

例 10-2　環境変数を使用するアクション文

```
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。環境変数の詳細については、「環境変数の定義」（p.10-15）を参照してください

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あとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

ログインした VDC（Virtual Device Context; 仮想デバイス コンテキスト）で、EEM を設定します。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。モジュールベースのイベントに対応するポリシーを設定する場合は、この VDC を使用する必要があります。

すべての VDC ですべてのアクションまたはイベントを確認できるわけではありません。ポリシーを設定するには、network-admin または vdc-admin の権限が必要です。

VDC の詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

EEM 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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide, Release 4.0*』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 または vdc-admin のユーザ権限が必要です。

設定時の注意事項および制約事項

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。

- ユーザ ポリシーまたは上書きポリシー内のアクション ステートメントが、相互に否定したり、関連付けられたシステム ポリシーに悪影響を与えるようなことがないようにする必要があります。
- 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すべてのイベントが上書きされます。

EEM の設定

ここでは、次の内容について説明します。

CLI によるユーザ ポリシーの定義

CLI を使用してユーザ ポリシーを定義できます。

ここでは、次の内容について説明します。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

管理者の権限でログインしていることを確認します。

手順概要

1. **config t**
2. **event manager applet** *applet-name*
3. **description** *policy-description*
4. **event** *event-statement*
5. **action** *number action-statement*
   （アクション文が複数ある場合は、ステップ 5 を繰り返す）
6. **show event manager policy** *name*
7. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 32 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。ストリングには最大 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。「イベント文の設定」（p.10-9）を参照してください。
ステップ 5	**action** *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。「アクション文の設定」（p.10-11）を参照してください。 アクション文が複数の場合は、ステップ 5 を繰り返します。
ステップ 6	***show event manager policy*** *name* **例：** `switch(config-applet)# show event manager policy monitorShutdown`	(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 7	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	(任意) この設定変更を保存します。

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 cli match *expression* [**count** repeats | **time** *seconds*] **例：** switch(config-applet)# event cli match "shutdown"	正規表現と一致する CLI コマンドが入力された場合に、イベントを発生させます。*repeats* の範囲は 0 ～ 4294967295 です。time の範囲は 0 ～ 4294967295 秒です。
event counter 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	カウンタが開始のしきい値を超えた場合にイベントを発生させます（大なり、小なりなど、開始演算子に基づく）。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。*counter name* 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。*entry* および *exit* の値の範囲は 0 ～ 2147483647 です。
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 1 ～ 4 です。*seconds* の範囲は 0 ～ 4294967295 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 1 ～ 4 です。*seconds* の範囲は 0 ～ 4294967295 です。
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* は設定済み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 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。
event module-failure 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。障害タイプについては、『*Cisco NX-OS System Management Command Reference, Release 4.0*』を参照してください。 *slot* の範囲は 1 ～ 10 です。*repeats* の範囲は 1 ～ 4294967295 です。*seconds* の範囲は 1 ～ 4294967295 です。

コマンド	目的
event oir {**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。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 — 範囲は 1 ～ 4 • モジュール番号 — 範囲は 1 ～ 10 • 電源モジュール番号 — 範囲は 1 ～ 3
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 ～ 4294967295 です。*seconds* の範囲は 1 ～ 4294967295 です。
event poweroverbudget [**time** *seconds*] **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 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 が開始のしきい値を超えた場合にイベントを発生させます（大なり、小なりなど、開始演算子に基づく）。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。*entry* および *exit* の値の範囲は 0 ～ 18446744073709551615 です。time および interval 範囲は 0 ～ 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 temperature [**module** *slot*] [*sensor number*] **threshold** {**any** | **major** | **minor**} **例：** `switch(config-applet)# event temperature module 2 threshold any`	温度センサ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。*slot* の範囲は 1 ～ 10 です。sensor の範囲は 1 ～ 18 です。
event track *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。*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 32 の英数字を使用できます。*val* は 0 ～ 2147483647 の整数または置換パラメータにできます。
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。
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。 *slot* の範囲は 1 ～ 10 または置換パラメータです。*xbar-number* の範囲は 1 ～ 4 または置換パラメータです。 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。 *slot* の範囲は 1 ～ 10 または置換パラメータです。

コマンド	目的
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*] **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。 *slot* の範囲は 1 ～ 10 または置換パラメータです。
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* は最大 16 桁の任意の数にできます。*number2* の範囲は 0 ～ 9 です。 *data* 引数は、最大 80 桁の任意の数にできます。*string* には最大 80 文字の英数字を使用できます。
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* は最大 16 桁の任意の数にできます。*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。

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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

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。

詳細な手順

ステップ 1　テキスト エディタで、ポリシーを定義する CLI 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　ファイルを次のシステム ディレクトリにコピーします。

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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

管理者の権限でログインしていることを確認します。

手順概要

1. **config t**
2. **event manager policy** *policy-script*
3. **show event manager policy** *name*
4. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 32 の英数字を使用できます。
ステップ 3	***show event manager policy*** *name* **例：** `switch(config-applet)# show event manager policy moduleScript`	（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 4	***copy running-config startup-config*** **例：** `switch(config)# copy running-config startup-config`	（任意）この設定変更を保存します。

ポリシーの上書き

システム ポリシーは上書き可能です。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

管理者の権限でログインしていることを確認します。

上書きするポリシーのイベントおよびデフォルト アクションがわかっていることを確認します。

手順概要

1. **config t**
2. **show event manager policy-state** *system-policy*
3. **event manager applet** *applet-name* **override** *system-policy*
4. **description** *policy-description*
5. **event** *event-statement*
6. **action** *number action-statement*
   (Repeat Step 6 for multiple action statements.)
7. **show event manager policy-state** *name*
8. **copy running-config startup-config**

詳細な手順

	コマンド	目的

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

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 32 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：** `switch(config-applet)# 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。ストリングには最大 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。「イベント文の設定」（p.10-9）を参照してください。
ステップ 6	**action** *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。「アクション文の設定」（p.10-11）を参照してください。 アクション文が複数の場合は、ステップ 6 を繰り返します。

	コマンド	目的
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）この設定変更を保存します。

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。

操作の前に

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. ***event manager environment*** *variable-name variable-value*
3. **show event manager environment**
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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 32 の英数字を使用できます。*variable-value* には最大 32 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*** **例：** `switch(config-applet)# show event manager environment`	（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）この設定変更を保存します。

EEM の設定確認

EEM 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 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 [*policy-name*] [**inactive**]	設定したポリシーに関する情報を表示します。
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 の設定例

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
```

__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
```

デフォルト設定

表 10-1 に、EEM パラメータのデフォルト設定を示します。

表 10-1　デフォルトの EEM パラメータ

パラメータ	デフォルト
システム ポリシー	active

その他の関連資料

EEM の実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.10-17）
- 規格（p.10-17）

関連資料

関連項目	マニュアル名
EEM CLI コマンド	『*Cisco NX-OS System Management Configuration Guide, Release 4.0*』。URL は次のとおりです。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/configuration/guide/sm_nx-os_config.html
VDC	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』。URL は次のとおりです。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

■ その他の関連資料

CHAPTER 11

OBFL の設定

この章では、Cisco NX-OS で OBFL（Onboard Failure Logging）機能を設定する方法について説明します。

ここでは、次の内容を説明します。

OBFL 情報

ここでは、次の内容について説明します。

- OBFL の概要（p.11-2）
- 仮想化サポート（p.11-2）

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 が保管するデータは、次のとおりです。

- 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 は Cisco NX-OS クラッシュが発生した場合に、カーネル トレースを保管します。

仮想化サポート

OBFL 情報を設定したり表示したりするには、デフォルトの VDC（Virtual Device Context; 仮想デバイス コンテキスト）を使用する必要があります。VDC の詳細については、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

OBFL 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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide, Release 4.0*』を参照してください。

OBFL の前提条件

VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始する必要があります（『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照）。

network-admin ユーザ権限で、デフォルト VDC にログインする必要があります。

設定時の注意事項および制約事項

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です。
- 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） Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

OBFL の設定

OBFL を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
hw-module logging onboard	すべての OBFL 機能をイネーブルにします。
hw-module logging onboard environmental-history	OBFL 環境履歴をイネーブルにします。
hw-module logging onboard error-stats	OBFL エラー統計をイネーブルにします。
hw-module logging onboard interrupt-stats	OBFL 割り込み統計をイネーブルにします。
hw-module logging onboard module *slot*	モジュールの OBFL 情報をイネーブルにします。
hw-module logging onboard obfl-log	ブート時間、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。
no hw-module logging onboard	すべての OBFL 機能をディセーブルにします。

OBFL の設定確認

OBFL の設定状況を表示するには、**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:  2 OBFL Log:                                    Enabled
    cpu-hog                                                 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
    stack-trace                                             Enabled
    system-health                                           Enabled

    Module:  6 OBFL Log:                                    Enabled
    cpu-hog                                                 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
    stack-trace                                             Enabled
    system-health                                           Enabled
    temp Error                                              Enabled
```

モジュールのフラッシュに保管されている OBFL 情報を表示するには、次のコマンドを使用します。

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 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 kernel-trace	カーネル トレース情報を表示します。
show logging onboard module *slot*	指定したモジュールの OBFL 情報を表示します。
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。

（注）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 の設定例

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。

```
conf t
 hw-module logging onboard module 2 environmental-history
```

デフォルト設定

表 11-1 に、OBFL パラメータのデフォルト設定を示します。

表 11-1　デフォルトの OBFL パラメータ

パラメータ	デフォルト
OBFL	すべての機能がイネーブル

その他の関連資料

OBFL の実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.11-5）
- 規格（p.11-5）

関連資料

関連項目	マニュアル名
OBFL CLI コマンド	『*Cisco NX-OS System Management Configuration Guide, Release 4.0*』
コンフィギュレーション ファイル	『*Cisco NX-OS Fundamentals Configuration Guide, Release 4.0*』
VDC	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

その他の関連資料

CHAPTER 12

SPAN の設定

この章では、デバイス上でイーサネット SPAN（スイッチド ポート アナライザ）を設定する方法について説明します。

ここでは、次の内容を説明します。

SPAN の概要

イーサネット SPAN を設定すると、デバイスの入出力トラフィックを監視できます。これらの機能によって、送信元から宛先へのパケットをコピーできます。

SPAN セッションを作成し、ネットワーク トラフィックに使用する送信元および宛先を定義します。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。宛先ポートはすべての送信元からコピーされたトラフィックを受信します。

SPAN セッションはローカル デバイスに適用されます。

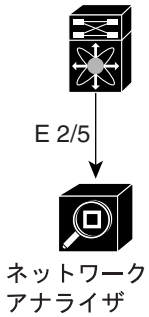
ここでは、次の内容について説明します。

SPAN セッション

最大 18 の SPAN セッションを作成し、ローカル デバイス上で送信元および宛先を定義できますが、同時に実行できるセッションは 2 つだけです。

図 12-1 に、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 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 12-1　SPAN の設定



送信元ポート	方向	宛先ポート
E 2/1	Rx	E 2/5
E 2/2	Rx, Tx	
E 2/3	Tx	

186283

SPAN セッションの送信元にはイーサネット ポート、VLAN、リモート SPAN（RSPAN）VLAN、およびコントロール プレーン CPU の帯域内インターフェイスが含まれます。

（注） NX-OS は RSPAN セッションの終了をサポートしません。

SPAN セッションの宛先には、アクセス モードまたはトランク モードのポートが含まれます。

（注） 同時に実行できる SPAN セッションは 2 つだけです。

SPAN セッションの設定については、「SPAN セッションの設定」（p.12-6）を参照してください。

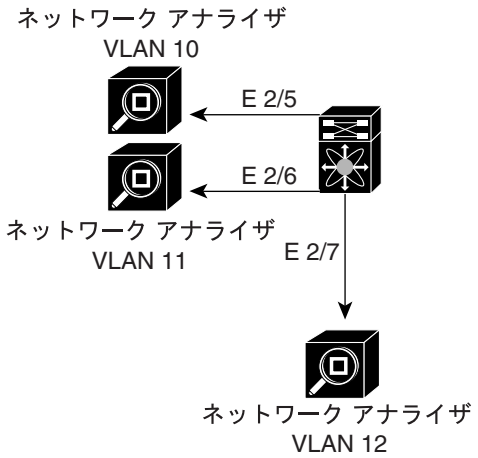
仮想 SPAN セッション

仮想 SPAN セッションを作成すると、複数の VLAN 送信元を監視し、複数の宛先ポートでの送信に関係する VLAN だけを選択できます。たとえば、トランク ポートで SPAN を設定し、さまざまな宛先ポートでさまざまな VLAN からのトラフィックを監視できます。

図 12-2 に、仮想 SPAN の設定を示します。仮想 SPAN セッションでは、3 つの VLAN から指定した 3 つの宛先ポートへトラフィックがコピーされます。各宛先ポートで許可する VLAN を選択することによって、そのポートでデバイスが送信するトラフィックを制限できます。図 12-2 では、デバイスは各宛先ポートへ、1 つの VLAN からのパケットを送信します。

（注） 仮想 SPAN セッションでは、パケットが宛先で必要かどうかに関係なく、すべての送信元パケットがすべての宛先にコピーされます。VLAN トラフィックのフィルタリングは、出力側の宛先ポートレベルで行われます。

図 12-2　仮想 SPAN の設定



送信元 VLAN	トラフィックの方向	宛先ポート
10	Rx	E 2/5 E 2/6 E 2/7
11	Rx, Tx	
12	Tx	

186284

Rx は受信
Tx は送信

仮想 SPAN セッションの設定については、「仮想 SPAN セッションの設定」（p.12-9）を参照してください。

マルチ SPAN セッション

最大 18 の SPAN セッションを定義できますが、同時に実行できる SPAN セッションは 2 つだけです。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断ち切ることができます。

SPAN セッションのシャットダウンについては、「SPAN セッションのシャットダウンまたは再開」（p.12-13）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 スイッチオーバーのあとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

VDC（Virtual Device Context; 仮想デバイス コンテキスト）は、一連のシステム リソースに対応する論理表現です。SPAN が適用されるのは、コマンドが入力された VDC だけです。

（注） 帯域内インターフェイスを監視できるのは、デフォルトの VDC からだけです。すべての VDC からの帯域内トラフィックが監視されます。

VDC の設定については、次の URL にアクセスして、『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

SPAN 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
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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、次の URL にアクセスして『*Cisco NX-OS Licensing Guide, Release 4.0*』を参照してください。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/licensing/configuration/guide/nx-os_licensing.html

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、所定の SPAN 設定をサポートするポートを設定します。詳細については、次の URL にアクセスして『*Cisco NX-OS Interfaces Configuration Guide, Release 4.0*』を参照してください。
http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/interfaces/configuration/guide/if_nxos_book.html
- **switchport monitor** コマンドを使用して、SPAN セッションを監視する宛先ポートを設定します。

注意事項および制約事項

SPAN 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。

- デバイス上で設定できる SPAN セッションは最大 18 です。
- デバイス上で同時に実行できる SPAN セッションは最大 2 つです。
- 特定の宛先ポートを設定できるのは、1 つの SPAN セッションに限られます。
- 1 つのポートを送信元ポートと宛先ポートの両方に設定することはできません。
- RSPAN セッションは終了できません。
- 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 BPDU（ブリッジ プロトコル データ ユニット）STP（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 セッションに複数の出力側送信元ポートが含まれている場合、これらのポートが受信するパケットは、そのポートで送信しない場合でも複製される可能性があります。送信元ポートでこの動作が生じる例の一部を示します。
  - フラッディングから生じたトラフィック
  - ブロードキャストおよびマルチキャスト トラフィック
- 入力と出力の両方が設定されている VLAN SPAN セッションでは、パケットが同じ VLAN 上でスイッチングされる場合に、宛先ポートから 2 つのパケット（入力側から 1 つ、出力側から 1 つ）が転送されます。
- VLAN SPAN が監視するのは、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。
- 帯域内インターフェイスを監視できるのは、デフォルトの VDC からだけです。すべての VDC からの帯域内トラフィックが監視されます。
- RSPAN VLAN を設定できるのは、SPAN セッションの送信元として使用する場合に限られます。

SPAN の設定

SPAN セッションを設定できるのは、ローカル デバイス上の送信元から宛先にパケットをコピーする場合だけです。

仮想 SPAN セッションを設定するには、複数の VLAN 送信元を選択してから、各宛先ポートで許可する VLAN を選択し、そのポート上でデバイスが送信するトラフィックを制限します。

ここでは、次の内容について説明します。

（注）Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

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 デバイス上の送信元から宛先にパケットをコピーする場合だけです。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。

送信元にはイーサネット ポート、ポート チャネル、スーパーバイザ帯域内インターフェイス、VLAN、および RSPAN VLAN を指定できます。SPAN 送信元ではプライベート VLAN（プライマリ、分離、およびコミュニティ）を指定できます。

SPAN 送信元としてスーパーバイザ帯域内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 ハードウェアに到達したすべてのパケット（入力）およびスーパーバイザ ハードウェアによって生成されたすべてのパケット（出力）を監視します。

宛先ポートには、アクセス モードまたはトランク モードのイーサネット ポートを指定できます。すべての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。

操作の前に

- 正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。
- 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します。詳細については、次の URL にアクセスして『*Cisco NX-OS Interfaces Configuration Guide, Release 4.0*』を参照してください。
http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/interfaces/configuration/guide/if_nxos_book.html

手順概要

1. **config t**
2. **interface ethernet** *slot/port*[*-port*]
3. **switchport monitor**
4. ステップ 2 および 3 を繰り返して、すべての SPAN 宛先でモニタリングを設定します。
5. **no monitor session** *session-number*
6. **monitor session** *session-number*
7. **description** *description*

8. **source** {**interface** *type* | **vlan** {*number* | *range*} [**rx** | **tx** | **both**]
9. ステップ 8 を繰り返して、すべての SPAN 送信元を設定します。
10. **filter vlan** {*number* | *range*}
11. ステップ 10 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。
12. **destination interface** *type* {*number* | *range*}
13. ステップ 12 を繰り返して、すべての SPAN 宛先ポートを設定します。
14. **no shut**
15. **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**]
16. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**interface ethernet** *slot/port[-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またはポート範囲で、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 monitor** **例：** `switch(config-if)# switchport monitor allowed vlan 3-5`	SPAN トラフィックを監視するスイッチポート インターフェイスを設定します。 **（注）** これは、トランク モードのスイッチポートとして設定済みのインターフェイスでなければなりません。
ステップ 4	（任意）ステップ 2 および 3 を繰り返して、すべての SPAN 宛先でモニタリングを設定します。	—
ステップ 5	**no monitor session** *session-number* **例：** `switch(config)# no monitor session 3`	指定した SPAN 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 6	**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。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	**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。

	コマンド	目的

ステップ 8	**source** {**interface** *type* | **vlan**} {*number* | *range*} [**rx** | **tx** | **both**]
	
	例 1：
	`switch(config-monitor)# source interface ethernet 2/1-3, ethernet 3/1 rx`
	
	例 2：
	`switch(config-monitor)# source interface port-channel 2`
	
	例 3：
	`switch(config-monitor)# source interface sup-eth 0 both`
	
	例 4：
	`switch(config-monitor)# source vlan 3, 6-8 tx`
		送信元およびパケットをコピーするトラフィックの方向を設定します。イーサネット ポート範囲、ポート チャネル、帯域内インターフェイス、または VLAN 範囲を入力できます。
		
		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。インターフェイス番号の値は 1 ～ 128 です。VLAN 番号の値は 1 ～ 3967 または 4048 ～ 4093 です。
		
		コピーするトラフィックの方向は、受信（rx）、送信（tx）、または両方（both）を設定できます。方向のデフォルトは both です。
		
		（注） 帯域内インターフェイスを監視できるのは、デフォルトの VDC からだけです。すべての VDC からの帯域内トラフィックが監視されます。

ステップ 9	（任意）ステップ 8 を繰り返して、すべての SPAN 送信元を設定します。	—
ステップ 10	**filter vlan** {*number* | *range*} **例：** `switch(config-monitor)# filter vlan 3-5, 7`	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 番号の値は 1 ～ 3967 または 4048 ～ 4093 です。
ステップ 11	（任意）ステップ 10 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。	—
ステップ 12	**destination interface** *type* {*number* | *range*} **例：** `switch(config-monitor)# destination interface ethernet 2/5, ethernet 3/7`	コピーする送信元パケットの宛先を設定します。宛先は 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。インターフェイス番号の値は 1 ～ 128 です。 **（注）** SPAN 宛先ポートは、アクセス ポートまたはトランク ポートのどちらかにする必要があります。**switchport monitor** コマンドを指定することによって、インターフェイス上でモニタ モードをイネーブルにする必要があります。
ステップ 13	（任意）ステップ 12 を繰り返して、すべての SPAN 宛先ポートを設定します。	—

	コマンド	目的
ステップ 14	**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。 **（注）** 同時に実行できる SPAN セッションは 2 つだけです。
ステップ 15	**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 16	**copy running-config startup-config** **例：** `switch(config-monitor)# 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。

仮想 SPAN セッションの設定

仮想 SPAN セッションを設定すると、送信元ポート、VLAN、および RSPAN VLAN からローカルデバイス上の宛先ポートへのパケットをコピーできます。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。

送信元には、ポート、VLAN または RSPAN VLAN を指定できます。

宛先ポートにはイーサネット ポートを指定できます。各宛先ポートで許可する VLAN を選択することによって、そのポートでデバイスが送信するトラフィックを制限できます。

操作の前に

- 正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。
- トランク モードで宛先ポートを設定します。詳細については、次の URL にアクセスして『*Cisco NX-OS Interfaces Configuration Guide, Release 4.0*』を参照してください。
http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/interfaces/configuration/guide/if_nxos_book.html
- **switchport monitor** コマンドを使用して、SPAN セッションを監視する宛先ポートを設定します。

手順概要

1. **config t**
2. **no monitor session** *session-number*
3. **monitor session** *session-number*
4. **source** {**interface** *type* | **vlan**} {*number* | *range*} [**rx** | **tx** | **both**]
5. ステップ 4 を繰り返して、すべての仮想 SPAN VLAN 送信元を設定します。
6. **destination interface** *type* {*number* | *range*}
7. ステップ 6 を繰り返して、すべての仮想 SPAN 宛先ポートを設定します。
8. **no shut**
9. **show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**]

10. **interface ethernet** *slot*/*port*[-*port*]
11. **switchport trunk allowed vlan** {{*number* | *range*}| **add** {*number* | *range*} | **except** {*number* | *range*} | **remove** {*number* | *range*} | **all** | **none**}
12. ステップ 10 および 11 を繰り返して、各宛先ポートで許可する VLAN を設定します。
13. **show interface ethernet** *slot*/*port*[-*port*] **trunk**
14. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**no monitor session** *session-number* **例：** `switch(config)# no monitor session 3`	指定した SPAN 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 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	**source** {**interface** *type* | **vlan**} {*number* | *range*} [**rx** | **tx** | **both**] **例：** `switch(config-monitor)# source vlan 3, 6-8 tx`	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。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。インターフェイス番号の値は 1 ～ 128 です。VLAN 番号の値は 1 ～ 3967 または 4048 ～ 4093 です。 コピーするトラフィックの方向は、受信（rx）、送信（tx）、または両方（both）を設定できます。方向のデフォルトは both です。
ステップ 5	（任意）ステップ 4 を繰り返して、すべての仮想 SPAN 送信元 VLAN を設定します。	—
ステップ 6	**destination interface** *type* {*number* | *range*} **例：** `switch(config-monitor)# destination interface ethernet 2/5, ethernet 3/7`	コピーする送信元パケットの宛先を設定します。インターフェイスは 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。インターフェイス番号の値は 1 ～ 128 です。 **（注）** 宛先ポートをトランク ポートとして設定します。詳細については、『*Cisco NX-OS Interfaces Configuration Guide, Release 4.0*』を参照してください。
ステップ 7	（任意）ステップ 6 を繰り返して、すべての仮想 SPAN 宛先ポートを設定します。	—

	コマンド	目的
ステップ 8	**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。 **（注）** 同時に実行できる SPAN セッションは 2 つだけです。
ステップ 9	**show monitor session** {**all** | *session-number* | **range** *session-range*} [**brief**] **例：** `switch(config-monitor)# show monitor session 3`	（任意）仮想 SPAN 設定を表示します。

ステップ 10	**interface ethernet** *slot/port*[*-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またはポート範囲で、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 11	**switchport trunk allowed vlan** {{*number* | *range*} | **add** {*number* | *range*} | **except** {*number* | *range*} | **remove** {*number* | *range*} | **all** | **none**} **例：** `switch(config-if)# switchport trunk allowed vlan 3-5`	インターフェイスで許可する VLAN の範囲を設定します。既存の VLAN に対して追加または削除する、指定した以外のすべての VLAN を選択する、すべての VLAN を選択する、またはすべての VLAN を選択しないでおくことができます。デフォルトでは、インターフェイス上ですべての VLAN が許可されます。 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 番号の値は 1 ～ 3967 または 4048 ～ 4093 です。
ステップ 12	（任意）ステップ 10 および 11 を繰り返して、各宛先ポートで許可する VLAN を設定します。	—
ステップ 13	**show interface ethernet** *slot/port*[*-port*] **trunk** **例：** `switch(config-if)# show interface ethernet 2/5 trunk`	（任意）選択したスロットおよびポートまたはポート範囲に対応するトランキング設定を表示します。
ステップ 14	**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。

RSPAN VLAN の設定

RSPAN を SPAN セッション送信元として指定できます。

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **vlan** *vlan*
3. **remote-span**
4. **exit**
5. **show vlan**
6. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**vlan** *vlan*
	例：
	`switch(config)# vlan 901`
	`switch(config-vlan)#`
		指定した VLAN の VLAN 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	**remote-span** **例：** `switch(config-vlan)# remote-span`	VLAN を RSPAN VLAN として設定します。

ステップ 4	**exit**
	例：
	`switch(config-vlan)# exit`
	`switch(config)#`
		VLAN コンフィギュレーション モードを終了します。

ステップ 5	**show vlan** **例：** `switch(config)# show vlan`	（任意）VLAN 設定を表示します。RSPAN VLAN が一覧表示されます。
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。

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断ち切ることができます。同時に実行できる SPAN セッションは 2 つだけなので、セッションの 1 つをシャットダウンしてハードウェア リソースを解放することによって、別のセッションが使用できるようになります。デフォルトでは、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、動作上ダウンの 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。

操作の前に

正しい VDC を使用していることを確認します（または **switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **monitor session** {*session-range* | **all**} **shut**
3. **no monitor session** {*session-range* | **all**} **shut**
4. **monitor session** *session-number*
5. **shut**
6. **no shut**
7. **show monitor**
8. **copy running-config startup-config**

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**monitor session** {*session-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。セッション範囲は 1 ～ 18 です。デフォルトでは、セッションはシャット ステートで作成されます。同時に実行できるセッションは 2 つだけです。

	コマンド	目的
ステップ 3	**no monitor session** {*session-range* | **all**} **shut** **例：** `switch(config)# no monitor session 3 shut`	指定の SPAN セッションを再開（イネーブルに）します。セッション範囲は 1 ～ 18 です。デフォルトでは、セッションはシャット ステートで作成されます。同時に実行できるセッションは 2 つだけです。 **（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 4	**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 5	**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。
ステップ 6	**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。 **（注）** 同時に実行できる SPAN セッションは 2 つだけです。
ステップ 7	**show monitor** **例：** `switch(config-monitor)# show monitor`	（任意）SPAN セッションの状況を表示します。
ステップ 8	**copy running-config startup-config** **例：** `switch(config-monitor)# 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 monitor session {**all** | *session-number* | **range** *session-range*} [**brief**]	SPAN セッションの設定を表示します。

各コマンド出力のフィールドの詳細については、次の URL にアクセスして、『*Cisco NX-OS System Management Command Reference*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html

SPAN の設定例

ここでは、次の内容について説明します。

- SPAN セッションの設定例（p.12-15）
- 仮想 SPAN セッションの設定例（p.12-16）
- SPAN セッションにおけるプライベート VLAN 送信元の設定例（p.12-17）

SPAN セッションの設定例

SPAN セッションを設定する手順は、次のとおりです。

ステップ 1　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し、SPAN モニタリングをイネーブルにします。

```
switch# config t
  switch(config)# interface ethernet 2/5
    switch(config-if)# switchport
    switch(config-if)# switchport mode trunk
    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。

```
  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 tx
    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
```

仮想 SPAN セッションの設定例

仮想 SPAN セッションを設定する手順は、次のとおりです。

ステップ 1　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し、SPAN モニタリングをイネーブルにします。

```
switch# config t
  switch(config)# interface ethernet 3/1
    switch(config-if)# switchport
    switch(config-if)# switchport mode trunk
    switch(config-if)# switchport trunk allowed vlan add 100-200
    switch(config-if)# switchport monitor
    switch(config-if)# no shut
    switch(config-if)# exit
  switch(config)# interface ethernet 3/2
    switch(config-if)# switchport
    switch(config-if)# switchport mode trunk
    switch(config-if)# switchport trunk allowed vlan add 201-300
    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。

```
  switch(config)# no monitor session 3
  switch(config)# monitor session 3
    switch(config-monitor)# source vlan 100-300
    switch(config-monitor)# destination interface ethernet 3/1-2
    switch(config-monitor)# no shut
    switch(config-monitor)# exit
  switch(config)# show monitor session 3
  switch(config)# copy running-config startup-config
```

SPAN セッションにおけるプライベート VLAN 送信元の設定例

プライベート VLAN 送信元が含まれる SPAN セッションを設定する手順は、次のとおりです。

ステップ 1　送信元 VLAN を設定します。

```
switch# config t
  switch(config)# vlan 100
    switch(config-vlan)# private-vlan primary
    switch(config-vlan)# exit
  switch(config)# interface ethernet 3/1
    switch(config-if)# switchport
    switch(config-if)# switchport access vlan 100
    switch(config-if)# no shut
    switch(config-if)# exit
  switch(config)# interface ethernet 3/2
    switch(config-if)# switchport
    switch(config-if)# switchport mode trunk
    switch(config-if)# switchport trunk native vlan 100
    switch(config-if)# no shut
    switch(config-if)# exit
  switch(config)#
```

ステップ 2　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し、SPAN モニタリングをイネーブルにします。

```
switch# config t
  switch(config)# interface ethernet 3/3
    switch(config-if)# switchport
    switch(config-if)# switchport mode trunk
    switch(config-if)# switchport trunk allowed vlan add 100-200
    switch(config-if)# switchport monitor
    switch(config-if)# no shut
    switch(config-if)# exit
  switch(config)#
```

ステップ 3　SPAN セッションを設定します。

```
  switch(config)# no monitor session 3
  switch(config)# monitor session 3
    switch(config-monitor)# source vlan 100
    switch(config-monitor)# destination interface ethernet 3/3
    switch(config-monitor)# no shut
    switch(config-monitor)# exit
  switch(config)# show monitor session 3
  switch(config)# copy running-config startup-config
```

その他の関連資料

SPAN の実装に関する詳細情報については、次の項を参照してください。

- 関連資料（p.12-18）
- 規格（p.12-18）

関連資料

関連項目	マニュアル名
VDC	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html
SPAN コマンド（コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト、使用上の注意事項、および例）	『*Cisco NX-OS System Management Command Reference*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

CHAPTER 13

NetFlow の設定

この章では、Cisco NX-OS の NetFlow 機能を設定する手順について説明します。

ここでは、次の内容を説明します。

NetFlow 情報

NetFlow は入力 IP パケットと出力 IP パケットの両方について、パケット フローを識別し、各パケット フローに基づいて統計情報を提供します。NetFlow のためにパケットやネットワーキング デバイスの変更が必要になることはありません。

ここでは、次の内容について説明します。

- NetFlow の概要（p.13-2）
- ハイ アベイラビリティ（p.13-4）
- 仮想化サポート（p.13-5）

NetFlow の概要

NetFlow ではフローを使用して、アカウンティング、ネットワーク モニタリング、およびネットワーク プランニングに関連する統計情報を提供します。フローは送信元インターフェイス（または VLAN）に届く単方向のパケット ストリームで、キーの値は同じです。キーは、パケット内のフィールドを識別する値です。フローを作成するには、フロー レコード マップを使用して、フロー固有のキーを定義します。Cisco NX-OS はフレクシブル NetFlow をサポートします。送信元および宛先 IP アドレスなど、共通のキーを使用することも、ユーザ独自のキーを定義することもできます。フロー レコード マップの詳細については、「フロー レコード マップ」（p.13-3）を参照してください。

1 つのフローとみなされるパケットでは、すべてのキー値が一致していなければなりません。フローには、設定したエクスポート レコード バージョンに基づいて、関係のある他のフィールドを集めることもあります。フローは NetFlow キャッシュに格納されます。

フロー用に NetFlow が収集したデータをエクスポートするには、エクスポート マップを使用し、このデータをリモート NetFlow コレクタにエクスポートします。Cisco NX-OS は次の状況で、NetFlow エクスポート UDP（ユーザ データグラム プロトコル）データグラムの一部としてフローをエクスポートします。

- あまりにも長期間にわたってフローが非アクティブまたはアクティブである。
- フロー キャッシュが満杯になった。
- カウンタの 1 つ（パケットまたはバイト）が最大値を超えた。
- ユーザがフローの強制的エクスポートを行った。

エクスポータ マップの詳細については、「エクスポータ マップ」（p.13-3）を参照してください

モニタ マップを使用してフローのために収集するデータのサイズを定義します。モニタ マップで、フロー レコード マップおよびエクスポータ マップを NetFlow キャッシュ情報と結合します。モニタ マップの詳細については、「モニタ マップ」（p.13-4）を参照してください

Cisco NX-OS は、フル モードまたはサンプル モードのどちらでも、NetFlow 統計情報を収集できます。フル モードの場合、Cisco NX-OS はインターフェイスまたはサブインターフェイス上のすべてのパケットを分析します。サンプル モードの場合は、サンプリング アルゴリズムおよび Cisco NX-OS にパケットを分析させるレートを設定します。サンプラ マップの詳細については、「サンプラ マップ」（p.13-4）を参照してください

フロー レコード マップ

フロー レコード マップでは、フロー内のパケットを識別するために NetFlow に使用させるキーとともに、フローのために NetFlow に収集させる関連フィールドを定義します。キーと関連フィールドを任意の組み合わせで指定して、フロー レコード マップを定義できます。Cisco NX-OS は、レイヤ 2 およびレイヤ 3 パラメータを含め、幅広いキー セットをサポートします。フロー レコードでは、フロー単位で収集するカウンタのタイプも定義します。32 ビットまたは 64 ビットのパケット カウンタまたはバイト カウンタを設定できます。Cisco NX-OS では、フロー レコードの作成時に次の match フィールドを使用できます。

- match interface input
- match interface output
- match flow direction

詳細については、「フロー レコードの作成」（p.13-8）を参照してください。

エクスポータ マップ

エクスポータ マップでは、NetFlow エクスポート パケットに関して、ネットワーク レイヤおよびトランスポート レイヤの詳細を指定します。エクスポータ マップで設定できる情報は、次のとおりです。

- エクスポート宛先 IP アドレス
- 送信元インターフェイス
- UDP ポート番号（コレクタが NetFlow パケットを待機するところ）
- エクスポート フォーマット

（注） NetFlow エクスポート パケットでは、送信元インターフェイスに割り当てられた IP アドレスを使用します。送信元インターフェイスに IP アドレスが割り当てられていない場合、エクスポータはアクティブになりません。

Cisco NX-OS は、タイムアウトが発生するたびに、またはフローが終了したときに（TCP Fin または Rst を受信した場合など）、コレクタにデータをエクスポートします。次のタイマーを設定すると、フローを強制的にエクスポートできます。

- アクティブ タイムアウト — Cisco NX-OS はキャッシュからキャッシュ エントリを削除しません。
- 非アクティブ タイムアウト — Cisco NX-OS はキャッシュからキャッシュ エントリを削除します。

エクスポート フォーマット

Cisco NX-OS は、大部分のフローで Version 9 のエクスポート フォーマットをサポートします。Version 9 では、ハードウェア集約フロー レコードと非集約フロー レコードを同じエクスポート フレームで結合します。また、次の機能もサポートします。

- 即時フロー エージング — フローの作成時点で、Cisco NX-OS にフロー レコードを送信させます。Cisco NX-OS はデバイスのパフォーマンス低下を避けるために、サンプル モードでこの機能をサポートします。
- 適応型フレクシブル NetFlow — NetFlow 用に予約されたシステム リソースを使い果たしたときのデバイス動作を定義します。
- パケット チャンク フィールド — パケットの一部分をコピーしてコレクタに送信します。この機能を使用できるのは、サンプル モードの場合だけです。また、デバイスのパフォーマンスに悪影響を与える可能性があります。

(注) Cisco NX-OS は、最大 4 つのコレクタにエクスポートする場合のトランスポート プロトコルとして、UDP をサポートします。

Version 5 のエクスポート フォーマットを使用するように IPv4 を設定できますが、Version 5 エクスポート フォーマットで定義済みのキーおよびフィールドを使用するように、このフォーマットを使用するフローに制限を課す必要があります。

モニタ マップ

モニタ マップは、フロー レコード マップおよびフロー エクスポータ マップを参照します。モニタ マップはインターフェイスに適用します。

サンプラ マップ

サンプル モードを使用する場合は、サンプラ マップを使用してパケットのサンプリング レートを指定します。広帯域幅のインターフェイス上で、個々のあらゆるパケットに NetFlow 処理を適用すると、CPU 使用率が上昇する可能性があります。サンプラ マップ設定は通常、このような高速インターフェイスに使用します。次のいずれかのオプションでサンプルを設定できます。

- N のうちの M — たとえば、10,000 パケットごとに 100 パケットをサンプリングします。
- タイムベース — たとえば、100 ミリ秒ごとに 1 パケットをサンプリングします。

サンプラ マップをインターフェイスに関連付けなかった場合、NetFlow はそのインターフェイス上のフロー キーと一致するあらゆるパケットについて、統計情報を収集します。

ハイ アベイラビリティ

Cisco NX-OS は、NetFlow 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あとに、Cisco NX-OS は実行コンフィギュレーションを適用します。

仮想化サポート

Cisco NX-OS では、VDC（Virtual Device Context; 仮想デバイス コンテキスト）の中で NetFlow フローを定義します。デフォルトでは、Cisco NX-OS はデフォルトの VDC が使用されるようにします。また、このモードで定義したフローを使用できるのは、デフォルト VDC のインターフェイスに限られます。次の URL にアクセスして、『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照してください。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

NetFlow 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
NX-OS	NetFlow 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、Cisco NX-OS システム イメージにバンドルされて提供されます。追加料金は発生しません。NX-OS ライセンス方式の詳細については、次の URL にアクセスして、『*Cisco NX-OS Licensing Guide*』を参照してください。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/licensing/configuration/guide/nx-os_licensing.html

NetFlow の前提条件

NetFlow の前提条件は、次のとおりです。

- NetFlow はメモリおよび CPU リソースをよけいに消費するので、デバイス上で必要なリソースについて理解しておく必要があります。

VDC を設定する場合は、Advanced Services ライセンスをインストールし、所定の VDC を開始してください（次の ULR から『*Cisco NX-OS Virtual Device Context Configuration Guide*』を参照）。

http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html

設定時の注意事項および制約事項

NetFlow 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。

- 送信元インターフェイスを設定する必要があります。送信元インターフェイスを設定しなかった場合、エクスポータはディセーブル ステートのままです。
- フロー モニタ マップごとに、有効なレコード マップ名を設定する必要があります。

Cisco NX-OS では、マップ固有のサブモードで NetFlow マップの設定を行います。

グローバル コンフィギュレーション モードを使用して、フロー レコード マップを作成し、フロー レコード サブモードを開始します。

```
switch(config)# flow record Test
switch(config-flow-record)#
```

グローバル コンフィギュレーション モードを使用して、フロー エクスポータ マップを作成し、フロー エクスポータ サブモードを開始します。

```
switch(config)# flow exporter ExportTest
switch(config-flow-exporter)#
```

フロー エクスポータ サブモードから、エクスポータのバージョンを指定できます。Version 9 を指定する場合は、フロー エクスポータ バージョン サブモードを開始します。

```
switch(config)-flow-exporter# version 9
switch(config-flow-exporter-version-9)#
```

グローバル コンフィギュレーション モードを使用して、フロー モニタ マップを作成し、フロー モニタ サブモードを開始します。

```
switch(config)# flow monitor MonitorTest
switch(config-flow-monitor)#
```

ヒント　サブモードから ? を入力すると、そのサブモードで使用できるすべてのコマンドのリストが表示されます。

NetFlow の設定

NetFlow を設定する手順は、次のとおりです。

ステップ 1　NetFlow 機能をイネーブルにします（「NetFlow 機能のイネーブル化」[p.13-8] を参照）。

ステップ 2　フローにキーおよびフィールドを指定することによって、フロー レコードを定義します（「フロー レコードの作成」[p.13-8] を参照）。

ステップ 3　エクスポート フォーマット、プロトコル、宛先、およびその他のパラメータを指定することによって、任意でフロー エクスポータを定義します（「フロー エクスポータの作成」[p.13-11] を参照）。

ステップ 4　フロー レコードおよびフロー エクスポータに基づいて、フロー モニタを定義します（「フロー モニタの作成」[p.13-13] を参照）。

ステップ 5　送信元インターフェイス、サブインターフェイス、VLAN インターフェイス（「インターフェイスへのフローの適用」[p.13-15] を参照）、または VLAN（「VLAN 上でのブリッジ型 NetFlow の設定」[p.13-16] を参照）にフロー モニタを適用します。

ここでは、次の内容について説明します。

- NetFlow 機能のイネーブル化（p.13-8）
- フロー レコードの作成（p.13-8）
- フロー エクスポータの作成（p.13-11）
- フロー モニタの作成（p.13-13）
- サンプラの作成（p.13-14）
- インターフェイスへのフローの適用（p.13-15）
- VLAN 上でのブリッジ型 NetFlow の設定（p.13-16）
- NetFlow タイムアウトの設定（p.13-17）

（注）　Cisco IOS CLI の詳しい知識がある場合は、この機能で使用する Cisco NX-OS コマンドが、よく使用される Cisco IOS コマンドとは異なる可能性があることに注意してください。

NetFlow 機能のイネーブル化

フローを設定するには、先に NetFlow をグローバルでイネーブルにしておく必要があります。

NetFlow を設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
feature netflow **例：** `switch(config)# feature netflow`	NetFlow 機能をイネーブルにします。

NetFlow をディセーブルにして、すべてのフローを削除するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
no feature netflow **例：** `switch(config)# no feature netflow`	NetFlow 機能をディセーブルにします。デフォルトはディセーブルです。

フロー レコードの作成

フロー レコードを作成し、フローにアソシエート キーおよびフィールドを追加します。Cisco NX-OS では、フロー レコードの作成時に次の match フィールドを使用できます。

- match interface input
- match interface output
- match flow direction

操作の前に

NetFlow 機能がイネーブルになっていることを確認します（「NetFlow 機能のイネーブル化」[p.13-8]を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **flow record** *name*
3. **description** *string*
4. **match** *type*
5. **collect** *type*
6. **show flow record** [*name*]
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***flow record*** *name*
	
	例：
	`switch(config)# flow record Test`
	`switch(config-flow-record)#`
		フロー レコードを作成し、フロー レコード 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	***description*** *string* **例：** `switch(config-flow-record)# description Ipv4Flow`	（任意）最大 63 文字で、このフローの説明を指定します。
ステップ 4	**match** *type* **例：** `switch(config-flow-record)# match interface input`	一致キーを指定します。*type* 引数の詳細については、「match パラメータの指定」（p.13-9）を参照してください。
ステップ 5	**collect** *type* **例：** `switch(config-flow-record)# collect counter packets`	コレクション フィールドを指定します。*type* 引数の詳細については、「collect パラメータの指定」（p.13-10）を参照してください。
ステップ 6	**show flow record** [*name*] **例：** `switch(config-flow-exporter)# show flow record`	（任意）NetFlow のフロー レコード情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-flow-exporter)# copy running-config startup-config`	（任意）この設定変更を保存します。

match パラメータの指定

フロー レコード マップごとに、次の match パラメータを 1 つ以上設定する必要があります。

コマンド	目的
match flow direction **例：** `switch(config-flow-record)# match flow direction`	フローの方向をキーとして指定します。
match interface **例：** `switch(config-flow-record)# match interface`	インターフェイスの入力または出力アトリビュートをキーとして指定します。
match ip {**protocol** | **tos**} **例：** `switch(config-flow-record)# match ip protocol`	IP プロトコルまたは ToS フィールドをキーとして指定します。

コマンド	目的
match ipv4 {**destination** | **source**} **address** **例：** `switch(config-flow-record)# match ipv4 destination address`	IPv4 送信元または宛先アドレスをキーとして指定します。
match transport {**destination-port** | **source-port**} **例：** `switch(config-flow-record)# match transport destination-port`	トランスポート送信元または宛先ポートをキーとして指定します。

collect パラメータの指定

フロー レコード マップごとに、次の collect パラメータを 1 つ以上設定する必要があります。

コマンド	目的
collect counter {**bytes** | **packets**} [**long**] **例：** `switch(config-flow-record)# collect counter packets`	フローからパケットベースまたはバイト カウンタを収集します。任意で、64 ビット カウンタを使用することを指定できます。
collect flow {**direction** | **sampler id**} **例：** `switch(config-flow-record)# collect flow direction`	フローの方向またはフローに使用するサンプラ識別情報を収集します。
collect interface {**input** | **output**} **例：** `switch(config-flow-record)# collect interface input`	入力または出力インターフェイスのアトリビュートを収集します。
collect routing {**destination** | **source**} **as** [**peer**] **例：** `switch(config-flow-record)# collect routing destination as`	ローカル デバイスまたはピアの送信元または宛先 AS 番号を収集します。
collect routing forwarding-status **例：** `switch(config-flow-record)# collect routing forwarding-status`	パケットのフォワーディング ステータスを収集します。
collect routing next-hop address ipv4 [**bgp**] **例：** `switch(config-flow-record)# collect routing next-hop address ipv4`	ネクストホップ アドレスを収集します。

コマンド	目的
collect timestamp sys-uptime {**first** | **last**} **例：** `switch(config-flow-record)# collect timestamp sys-uptime last`	フローの先頭または最終パケットに関するシステム稼働時間を収集します。
collect transport tcp flags **例：** `switch(config-flow-record)# collect transport tcp flags`	フローのパケットに対応する TCP トランスポート レイヤ フラグを収集します。

フロー エクスポータの作成

フロー エクスポートを作成すると、フローのエクスポート パラメータを定義できます。

操作の前に

NetFlow 機能がイネーブルになっていることを確認します（「NetFlow 機能のイネーブル化」[p.13-8]を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **flow exporter** *name*
3. **destination** {*ipv4-address* | *ipv6-address*} [**use-vrf** *name*]
4. **source** *interface-type number*
5. **version** {**5** | **9**}
6. **show flow exporter** [*name*]
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***flow exporter*** *name*
	
	例：
	`switch(config)# flow exporter ExportTest`
	`switch(config-flow-exporter)#`
		フロー エクスポータ マップを作成し、フロー エクスポータ マップ 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** {*ipv4-address* | *ipv6-address*} [**use-vrf** *name*] **例：** `switch(config-flow-exporter)# destination 192.0.2.1`	このエクスポータ マップの宛先 IPv4 または IPv6 アドレスを設定します。任意で、NetFlow コレクタに到達するために使用する VRF を設定できます。

	コマンド	目的
ステップ 4	**source** *interface-type number* **例：** `switch(config-flow-exporter)# source ethernet 2/1`	設定された宛先で NetFlow コネクタに到達するために使用するインターフェイスを指定します。

ステップ 5	**version** {**5** | **9**}
	
	例：
	`switch(config-flow-exporter)# version 9`
	`switch(config-flow-exporter-version-9)#`
		NetFlow エクスポート バージョンを指定します。Version 9 でエクスポート バージョン コンフィギュレーション サブモードを開始します。

ステップ 6	**show flow exporter** [*name*] **例：** `switch(config-flow-exporter)# show flow exporter`	（任意）NetFlow のフロー エクスポータ マップ情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-flow-exporter)# copy running-config startup-config`	（任意）この設定変更を保存します。

任意で、フロー エクスポータに次のパラメータを設定できます。

コマンド	目的
description *string* **例：** `switch(config-flow-exporter)# description ExportV9`	最大 63 文字で、このフロー エクスポータ マップの説明を指定します。
dscp *value* **例：** `switch(config-flow-exporter)# dscp 0`	DSCP（DiffServ コードポイント）値を指定します。範囲は 0 ～ 63 です。
transport udp number **例：** `switch(config-flow-exporter)# transport udp 200`	NetFlow コレクタに到達するために使用する UDP ポートを指定します。範囲は 0 ～ 65535 です。

任意で、フロー エクスポータ バージョン コンフィギュレーション サブモードで次のパラメータを設定できます。

コマンド	目的
option {***exporter-stats*** | ***interface-table*** | ***sampler-table***} ***timeout*** *seconds* **例：** `switch(config-flow-exporter-version-9)# option exporter-stats timeout 1200`	エクスポータ再送信タイマーを設定します。値の範囲は 1 ～ 86400 秒です。
template data timeout *seconds* **例：** `switch(config-flow-exporter-version-9)# template data timeout 1200`	テンプレート データ再送信タイマーを設定します。値の範囲は 1 ～ 86400 秒です。

フロー モニタの作成

フロー モニタを作成して、フロー レコードおよびフロー エクスポータと関連付けることができます。

操作の前に

NetFlow 機能がイネーブルになっていることを確認します（「NetFlow 機能のイネーブル化」[p.13-8]を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **show flow monitor** [*name*]
7. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***flow monitor*** *name*
	
	例：
	`switch(config)# flow monitor MonitorTest`
	`switch(config-flow-monitor)#`
		フロー モニタ マップを作成し、フロー モニタ マップ 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	**description** *string* **例：** `switch(config-flow-monitor)# description Ipv4Monitor`	（任意）最大 63 文字の英数字で、フロー モニタ マップの説明を指定します。
ステップ 4	**exporter** *name* **例：** `switch(config-flow-monitor)# exporter Exportv9`	フロー エクスポータ マップとこのフロー モニタ マップを関連付けます。
ステップ 5	**record** *name* **例：** `switch(config-flow-monitor)# record IPv4Flow`	フロー レコード マップとこのフロー モニタ マップを関連付けます。

	コマンド	目的
ステップ 6	**show flow monitor** [*name*] **例：** `switch(config-flow-monitor)# show flow monitor`	（任意）NetFlow のフロー モニタ マップ情報を表示します。
ステップ 7	***copy running-config startup-config*** **例：** `switch(config-flow-monitor)# copy running-config startup-config`	（任意）この設定変更を保存します。

サンプラの作成

サンプラを作成すると、フローに関する NetFlow サンプリング レートを定義できます。

操作の前に

NetFlow 機能がイネーブルになっていることを確認します（「NetFlow 機能のイネーブル化」[p.13-8]を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **sampler** *name*
3. **description** *string*
4. **mode** *samples* **out-of** *packets*
5. **show sampler** [*name*]
6. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	***sampler*** *name*
	
	例：
	`switch(config)# sampler SampleTest`
	`switch(config-flow-sampler)#`
		サンプラ マップを作成し、サンプラ マップ 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	**description** *string* **例：** `switch(config-flow-sampler)# description Samples`	（任意）最大 63 文字の英数字で、サンプラ マップの説明を指定します。
ステップ 4	**mode** *samples* **out-of** *packets* **例：** `switch(config-flow-sampler)# mode 1 out-of 100`	受信パケット数あたりの取得サンプル数を定義します。サンプル数の範囲は 1 ～ 64 です。パケット数の範囲は 1 ～ 8192 パケットです。

	コマンド	目的
ステップ 5	**show sampler** [*name*] **例：** switch(config-flow-sampler)# show sampler	（任意）NetFlow のサンプラ マップ情報を表示します。
ステップ 6	***copy running-config startup-config*** **例：** switch(config-flow-sampler)# copy running-config startup-config	（任意）この設定変更を保存します。

インターフェイスへのフローの適用

フロー モニタ マップおよびオプションのサンプラ マップをインターフェイスに適用できます。

操作の前に

NetFlow 機能がイネーブルになっていることを確認します（「NetFlow 機能のイネーブル化」[p.13-8]を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **interface** *interface-type number*
3. **ip flow monitor** *name* {**input** | **output**} [**sampler** *name*]
4. **show flow interface** [*interface-type number*]
5. ***copy running-config startup-config***

詳細な手順

	コマンド	目的
ステップ 1	**config t** **例：** switch# config t switch(config)#	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-type number* **例：** switch(config)# interface ethernet 2/1 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。インターフェイス タイプはイーサネット、ポート チャネル、管理、VLAN インターフェイス、またはサブインターフェイスにできます。
ステップ 3	**ip flow monitor** *name* {**input** | **output**} [**sampler** *name*] **例：** switch(config-if)# ip flow monitor MonitorTest input	入力または出力パケットに対応するインターフェイスに、フロー モニタ マップおよびオプションのサンプラ マップを関連付けます。

	コマンド	目的
ステップ 4	**show flow interface** [*interface-type number*] **例：** `switch(config-if# show flow interface`	（任意）インターフェイスの NetFlow 情報を表示します。
ステップ 5	***copy running-config startup-config*** **例：** `switch(config-if)# copy running-config startup-config`	（任意）この設定変更を保存します。

VLAN 上でのブリッジ型 NetFlow の設定

フロー モニタ マップおよびオプションのサンプラ マップを VLAN に適用できます。

操作の前に

NetFlow 機能がイネーブルになっていることを確認します（「NetFlow 機能のイネーブル化」[p.13-8]を参照）。

正しい VDC を使用していることを確認します（または、**switchto vdc** コマンドを使用します）。

手順概要

1. **config t**
2. **vlan** *vlan-id*
3. **ip flow monitor** *name* {**input** | **output**} [**sampler** *name*]
4. ***copy running-config startup-config***

詳細な手順

	コマンド	目的

ステップ 1	**config t**
	
	例：
	`switch# config t`
	`switch(config)#`
		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	**vlan** *vlan-id*
	
	例：
	`switch(config)# vlan 30`
	`switch(config-vlan)#`
		VLAN コンフィギュレーション モードを開始します。vlan-id の範囲は 1 ～ 3967 または 4048 ～ 4093 です。

ステップ 3	**ip flow monitor** *name* {**input** | **output**} [**sampler** *name*] **例：** `switch(config-vlan)# ip flow monitor MonitorTest input`	入力または出力パケットに対応する VLAN に、フロー モニタ マップおよびオプションのサンプラ マップを関連付けます。
ステップ 4	***copy running-config startup-config*** **例：** `switch(config-vlan)# copy running-config startup-config`	（任意）この設定変更を保存します。

NetFlow タイムアウトの設定

任意で、すべてのフローに適用されるグローバルな NetFlow タイムアウトを設定できます。

NetFlow timeout パラメータを設定するには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
flow timeout active *seconds* **例：** `switch(config)# flow timeout active 90`	アクティブ タイムアウト値を設定します。有効値の範囲は 60 ～ 4092 秒です。
flow timeout aggressive threshold *percent* **例：** `switch(config)# flow timeout aggressive threshold 90`	アグレッシブ エージングが開始されるまでに Net Flow テーブルに必要なパーセンテージを設定します。範囲は 50 ～ 99% です。
flow timeout fast *seconds* ***threshold*** *packets* **例：** `switch(config)# flow timeout fast 40 threshold 1200`	高速タイムアウト値およびエージングが開始されるまでのフローのパケット数を設定します。高速タイムアウトの範囲は 32 ～ 512 秒です。パケットの範囲は 1 ～ 4000 です。
flow timeout inactive *seconds* **例：** `switch(config)# flow timeout inactive 900`	非アクティブ タイムアウト値を設定します。有効値の範囲は 15 ～ 4092 秒です。
flow timeout session **例：** `switch(config)# flow timeout session`	TCP セッション エージングをイネーブルにします。

NetFlow の設定確認

NetFlow の設定情報を確認するには、次のコマンドを使用します。

コマンド	目的
show **flow exporter** [*name*]	NetFlow のフロー エクスポータ マップ情報を表示します。
show **flow interface** [*interface-type number*]	NetFlow インターフェイスに関する情報を表示します。
show **flow monitor** [*name*] [**cache** [**detailed**]]	NetFlow のフロー モニタ マップ情報を表示します。
show **flow record** [*name*]	NetFlow のフロー レコード マップ情報を表示します。
show **flow timeout**	NetFlow タイムアウト情報を表示します。
show **hardware flow aging** [**vdc** *vdc_id*] [**detail**] [**module** *module*]	ハードウェアの NetFlow エージング フロー情報を表示します。
show **hardware flow entry address** *table-address* **type** {*ip* | *ipv6*} [**module** *module*]	ハードウェアの NetFlow テーブル エントリ情報を表示します。
show **hardware flow ip [interface** *type number* | **monitor** *monitor_name* | **profile** *profile-id* | **vdc** *vdc_id* | **vlan** *vlan_id*] [**detail**] [**module** *module*]	ハードウェアの NetFlow IPv4 フロー情報を表示します。
show **hardware flow sampler** [**all** | **count** | **index** *number* | **name** *sampler-name* | **vdc** *vdc_id*] [**detail**] [**module** *module*]	ハードウェアの NetFlow サンプラ情報を表示します。
show **hardware flow utilization** [**module** *module*]	ハードウェアの NetFlow テーブル使用率情報を表示します。
show **sampler** [*name*]	NetFlow のサンプラ マップ情報を表示します。

NetFlow の設定例

フローを作成してインターフェイスに適用する例を示します。

```
feature netflow
flow exporter ee
 version 9
flow record rr
 match ipv4 source address
 match ipv4 destination address
 collect counter bytes
 collect counter packets
flow monitor foo
 record rr
 exporter ee
interface Ethernet2/45
 ip flow monitor foo output
 ip address 10.20.1.1/24
 no shutdown
```

デフォルト設定

表 13-1 に、NetFlow パラメータのデフォルト設定を示します。

表 13-1　デフォルトの NetFlow パラメータ

パラメータ	デフォルト
アカウンティング キャッシュ サイズ	8 K
出力および入力キャッシュ サイズ	512 K

その他の関連資料

NetFlow の実装に関連する詳細情報については、次の項を参照してください。

- 関連資料（p.13-19）
- 規格（p.13-19）
- MIB（p.13-19）

関連資料

関連項目	マニュアル名
NetFlow CLI コマンド	『*Cisco NX-OS System Management Command Reference, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/system_management/command/reference/sm_cmd_ref.html
VDC および VRF	『*Cisco NX-OS Virtual Device Context Configuration Guide, Release 4.0*』。URL は次のとおり。 http://www.cisco.com/en/US/docs/switches/datacenter/sw/4_0/nx-os/virtual_device_context/configuration/guide/vdc_nx-os_book.html
Cisco NetFlow の概要	http://cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html

規格

規格	タイトル
この機能がサポートする新しい規格または変更された規格はありません。また、この機能で変更された既存規格のサポートはありません。	—

MIB

MIB	MIB のリンク
• CISCO-NETFLOW-MIB	MIB を見つけてダウンロードするには、次の URL を参照してください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

■ その他の関連資料

APPENDIX **A**

Cisco NX-OS System Management Release 4.0 がサポートする IETF RFC

この付録では、Cisco NX-OS Release 4.0 がシステム管理に関してサポートする IETF（インターネット技術特別調査委員会）RFC を示します。

RFC

RFC	タイトル
RFC2819	『*Remote Network Monitoring Management Information Base*』
RFC 3164	『*BSD syslog Protocol*』
RFC 3411 ～ 3418	SNMP RFC

INDEX

C

E

G

M

N

O

R

S

え

お

か

き

こ

さ

し

す

せ

て