SNMP の設定

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

この章は、次の項で構成されています。

SNMP について

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

この項では、次のトピックについて取り上げます。

SNMP 機能の概要

SNMP フレームワークは 3 つの部分で構成されます。

- SNMP マネージャ：SNMP を使用してネットワーク デバイスの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：デバイスのデータを維持し、必要に応じてこれらのデータを管理システムに報告する、管理対象デバイス内のソフトウェア コンポーネント。Cisco NX-OS はエージェントと MIB をサポートします。SNMP エージェントをイネーブルにするには、マネージャとエージェントの関係を定義する必要があります。
- 管理情報ベース（MIB）：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。

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形式のセキュリティを使用します。

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

SNMP 通知

SNMP の重要な機能の 1 つは、SNMP エージェントから通知を生成できることです。これらの通知では、要求を SNMP マネージャから送信する必要はありません。通知によって、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、トラップまたはインフォームとして SNMP 通知を生成します。トラップは、エージェントからホスト レシーバ テーブルで指定された SNMP マネージャに送信される、非同期の非確認応答メッセージです（「VRF を使用する SNMP 通知レシーバの設定」（P.9-14）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。

トラップの信頼性はインフォームより低くなります。SNMP マネージャはトラップを受信しても確認応答（ACK）を送信しないからです。Cisco NX-OS では、トラップを受信したかどうかを判断できません。インフォーム要求を受信する SNMP マネージャは、SNMP 応答プロトコル データ ユニット（PDU）でメッセージの受信を確認します。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 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.9-11）を参照してください。

表 9-1 に、デフォルトでイネーブルになっている SNMP トラップを示します。

表 9-1　デフォルトでイネーブルになっている SNMP トラップ

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

表 9-1　デフォルトでイネーブルになっている SNMP トラップ （続き）

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

SNMPv3

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

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

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

この項では、次のトピックについて取り上げます。

- 「SNMPv1、SNMPv2、SNMPv3 のセキュリティ モデルおよびセキュリティ レベル」（P.9-4）
- 「ユーザベースのセキュリティ モデル」（P.9-4）

- 「CLI および SNMP のユーザ同期」（P.9-5）
- 「グループベースの SNMP アクセス」（P.9-6）

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

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

- noAuthNoPriv：認証または暗号化を実行しないセキュリティ レベル。
- authNoPriv：認証は実行するが、暗号化を実行しないセキュリティ レベル。
- authPriv：認証と暗号化両方を実行するセキュリティ レベル。

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

表 9-2 に、セキュリティ モデルとセキュリティ レベルの組み合わせの意味を示します。

表 9-2　SNMP セキュリティ モデルおよびセキュリティ レベル

モデル	レベル	認証	暗号化	結果
v1	noAuthNoPriv	コミュニティ ストリング	No	コミュニティ ストリングの照合を使用して認証します。
v2c	noAuthNoPriv	コミュニティ ストリング	No	コミュニティ ストリングの照合を使用して認証します。
v3	noAuthNoPriv	ユーザ名	No	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	No	Hash-Based Message Authentication Code（HMAC）メッセージ ダイジェスト 5（MD5）アルゴリズムまたは HMAC Secure Hash Algorithm（SHA）アルゴリズムに基づいて認証します。
v3	authPriv	HMAC-MD5 または HMAC-SHA	DES	HMAC-MD5 アルゴリズムまたは HMAC-SHA アルゴリズムに基づいて認証します。データ暗号規格（DES）の 56 ビット暗号化、および暗号ブロック連鎖（CBC）DES（DES-56）標準に基づいた認証を提供します。

ユーザベースのセキュリティ モデル

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

- メッセージの完全性：メッセージが不正な方法で変更または破壊されず、データ シーケンスが悪意なく起こり得る範囲を超えて変更されていないことを保証します。

- メッセージ発信元の認証：受信データを発信したユーザのアイデンティティが確認されたことを保証します。
- メッセージの機密性：情報が使用不可であること、または不正なユーザ、エンティティ、またはプロセスに開示されないことを保証します。

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

Cisco NX-OS 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、Advanced Encryption Standard（AES）を使用し、RFC 3826 に準拠します。

priv オプションで、SNMP セキュリティ暗号化方式として、DES または 128 ビット AES 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 キー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。ローカライズド キーを使用する場合は、最大 130 文字を指定できます。

（注） 外部 AAA（認証、許可、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 のユーザ同期

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

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

Cisco NX-OS は、次のようにユーザ設定を同期化します。

- **snmp-server user** コマンドで指定された認証パスフレーズが CLI ユーザのパスワードになります
- **username** コマンドで指定されたパスワードが SNMP ユーザの認証およびプライバシー パスフレーズになります。
- SNMP または CLI を使用してユーザを作成または削除すると、SNMP と CLI の両方でユーザが作成または削除されます。
- ユーザとロールの対応関係の変更は、SNMP と CLI で同期化されます。
- CLI から行ったロール変更（削除または変更）は、SNMP と同期します。

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

Cisco NX-OS はデフォルトで、同期したユーザ設定を 60 分間維持します。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.9-26）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 アクセス権は、グループ別に編成されます。SNMP 内の各グループは、CLI を使用する場合のロールに似ています。各グループは読み取りアクセス権または読み取りと書き込みアクセス権を指定して定義します。

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

SNMP および EEM

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

EEM の詳細については、第 12 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 インスタンスや仮想ルーティングおよびフォワーディング（VRF）インスタンスなどの論理ネットワーク エンティティの複数のインスタンスをサポートできます。大部分の既存 MIB は、これら複数の論理ネットワーク エンティティを識別できません。たとえば、元々の OSPF-MIB ではデバイス上のプロトコル インスタンスが 1 つであることが前提になりますが、現在はデバイス上で複数の OSPF インスタンスを設定できます。

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

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

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

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

SNMP 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 SNMPv2c 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.9-24）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。

仮想化のサポート

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

SNMP は複数の MIB モジュール インスタンスをサポートし、それらを論理ネットワーク エンティティにマッピングします。詳細については、「マルチインスタンス サポート」（P.9-6）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 レシーバに接続するように SNMP を設定できます。通知が発生した VRF に基づいて、SNMP ホスト レシーバへの通知をフィルタリングするように SNMP を設定することもできます。詳細については、「VRF を使用する SNMP 通知レシーバの設定」（P.9-14）を参照してください。

SNMP のライセンス要件

製品	ライセンス要件
Cisco NX-OS	SNMP にはライセンスは不要です。ライセンス パッケージに含まれていない機能は nx-os イメージにバンドルされており、無料で提供されます。Cisco NX-OS のライセンス スキームの詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

注意事項と制約事項

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。

- アクセス コントロール リスト（ACL）は、スイッチに設定されたローカル SNMPv3 ユーザのみに適用できます。ACL は、認証、許可、アカウンティング（AAA）サーバに保存されるリモート SNMPv3 ユーザに適用できません。
- Cisco NX-OS は一部の SNMP MIB 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。

  ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

デフォルト設定値

表 9-3 に、SNMP パラメータのデフォルト設定を示します。

表 9-3　デフォルト SNMP パラメータ

パラメータ	デフォルト
ライセンス通知	イネーブル

SNMP の設定

この項では、次のトピックについて取り上げます。

- 「SNMP ユーザの設定」（P.9-8）
- 「SNMP メッセージ暗号化の適用」（P.9-9）
- 「SNMPv3 ユーザに対する複数のロールの割り当て」（P.9-10）
- 「SNMP コミュニティの作成」（P.9-10）
- 「SNMP 要求のフィルタリング」（P.9-11）
- 「SNMP 通知レシーバーの設定」（P.9-11）
- 「SNMP 通知用の発信元 インターフェイスの設定」（P.9-12）
- 「通知ターゲット ユーザの設定」（P.9-13）
- 「VRF を使用する SNMP 通知レシーバの設定」（P.9-14）
- 「帯域内ポートを使用してトラップを送信するための SNMP 設定」（P.9-15）
- 「SNMP 通知のイネーブル化」（P.9-17）
- 「インターフェイスに関する linkUp/linkDown 通知のディセーブル化」（P.9-22）
- 「インターフェイスの SNMP ifIndex の表示」（P.9-22）
- 「TCP による SNMP のワンタイム認証のイネーブル化」（P.9-22）
- 「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」（P.9-23）
- 「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.9-24）
- 「SNMP のディセーブル化」（P.9-25）
- 「AAA 同期時間の変更」（P.9-26）

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

SNMP ユーザの設定

SNMP ユーザを設定できます。

手順の概要

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

手順の詳細

	コマンド	目的
ステップ 1	**configure terminal** **Example:** switch# configure terminal Enter configuration commands, one per line. End with CNTL/Z. switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp-server user** *name* [auth {**md5** | **sha**} *passphrase* [auto] [**p**riv [aes-128] *passphrase*] [engineID *id*] [localizedkey]] **Example:** switch(config)# snmp-server user Admin auth sha abcd1234 priv abcdefgh	認証およびプライバシー パラメータのある SNMP ユーザを設定します。パスフレーズには、最大 64 文字の英数字を使用できます。大文字と小文字は区別されます。**localizedkey** キーワードを使用する場合は、パスフレーズに英数字を 130 文字まで使用できます。大文字と小文字は区別されます。 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** **Example:** switch(config-callhome)# show snmp user	（任意）1 人または複数の SNMP ユーザに関する情報を表示します。
ステップ 4	**copy running-config startup-config** **Example:** switch(config)# copy running-config startup-config	（任意）リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

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

SNMP メッセージ暗号化の適用

着信要求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 要求に、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 **Example:** switch(config)# snmp-server user Admin enforcePriv	このユーザに対して SNMP メッセージ暗号化を適用します。

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 **Example:** switch(config)# snmp-server globalEnforcePriv	すべてのユーザに対して SNMP メッセージ暗号化を適用します。

SNMPv3 ユーザに対する複数のロールの割り当て

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 ロールに属するユーザだけです。

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* **Example:** switch(config)# snmp-server user Admin superuser	この SNMP ユーザと設定されたユーザ ロールをアソシエートします。

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** *group* | **ro** | **rw**} **Example:** 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 要求のフィルタリング

アクセス コントロール リスト（ACL）を SNMPv3 ユーザまたは SNMPv3 コミュニティに割り当てて、着信 SNMP 要求にフィルタを適用できます。割り当てた ACL により着信要求パケットが許可される場合、SNMP はその要求を処理します。ACL により要求が拒否される場合、SNMP はその要求を廃棄して、システム メッセージを送信します。

ACL は次のパラメータで作成します。

- 送信元 IP アドレス
- 宛先 IP アドレス
- 送信元ポート
- 宛先ポート
- プロトコル（UDP または TCP）

ACL を SNMPv3 ユーザまたは SNMPv3 コミュニティに割り当てて SNMP 要求をフィルタするには、グローバル コンフィギュレーション モードで次のコマンドを使用します。

コマンド	目的
snmp-server user *name* [**use-ipv4acl** *ipv4acl-name*] [**use-ipv6acl** *ipv6acl-name*] **Example:** `switch(config)# snmp-server community public use-ipv4acl myacl`	SNMPv3 ユーザに IPv4 ACL または IPv6 ACL を割り当てて SNMP 要求をフィルタします。 **（注）** AAA サーバは、SNMPv3 ユーザの作成をサポートする必要があります。
snmp-server community *name* [**use-ipv4acl** *ipv4acl-name*] [**use-ipv6acl** *ipv6acl-name*] **Example:** `switch(config)# snmp-server community public use-ipv4acl myacl`	SNMPv3 コミュニティに IPv4 ACL または IPv6 ACL を割り当てて SNMP 要求をフィルタします。

SNMP 通知レシーバーの設定

複数のホスト レシーバに対して 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*] **Example:** `switch(config)# snmp-server host 192.0.2.1 traps version 1 public`	SNMPv1 トラップのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。*community* には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。

SNMPv2c トラップまたは応答要求のホスト レシーバを設定するには、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*] **Example:** `switch(config)# snmp-server host 192.0.2.1 informs version 2c public`	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。*community* には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。

SNMPv3 トラップまたは応答要求のホスト レシーバを設定するには、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*] **Example:** `switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS`	SNMPv3 トラップまたは応答要求のホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。*username* には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。

（注） SNMP マネージャは SNMPv3 メッセージを認証して解読するために、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。

SNMP 通知用の発信元 インターフェイスの設定

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

次のように発信元インターフェイスを設定できます。

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

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

発信元インターフェイスでホスト レシーバを設定するには、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* **source-interface** *if-type if-number* [**udp_port** *number*] **Example:** `switch(config)# snmp-server host 192.0.2.1 source-interface ethernet 2/1`	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。**?** サポートされているインターフェイス タイプを特定します。UDP ポート番号の範囲は 0 ～ 65535 です。 このコンフィギュレーションは、グローバル発信元インターフェイスのコンフィギュレーションよりも優先されます。

すべての SNMP 通知を送信するよう発信元インターフェイスを設定するには、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 source-interface {**traps** | **informs**} *if-type if-number* **Example:** `switch(config)# snmp-server source-interface traps ethernet 2/1`	SNMPv2c トラップまたは応答要求を送信するよう発信元インターフェイスを設定します。**?** サポートされているインターフェイス タイプを特定します。

設定されている発信元インターフェイスの情報を表示するには、**show snmp source-interface** コマンドを使用します。

通知ターゲット ユーザの設定

通知ホスト レシーバに 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） 受信した応答要求 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*] **Example:** switch(config)# snmp-server user NMS auth sha abcd1234 priv abcdefgh engineID 00:00:00:63:00:01:00:10:20:15:10:03	通知ホスト レシーバのエンジン ID を指定して、通知ターゲット ユーザを設定します。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 通知レシーバの設定

SNMP 通知レシーバの VRF 到達可能性およびフィルタリング オプションを設定すると、SNMP によって CISCO-SNMP-TARGET-EXT-MIB の cExtSnmpTargetVrfTable にエントリが追加されます。

（注）　VRF 到達可能性またはフィルタリング オプションを設定する前に、ホストを設定する必要があります。

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

ホスト レシーバ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*] **Example:** switch(config)# snmp-server host 192.0.2.1 use-vrf Blue	特定の VRF を使用してホスト レシーバと通信するように SNMP を設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。
no snmp-server host *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*] **Example:** 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 ExtSnmpTargetVrfTable からエントリを削除します。 *ip-address* は IPv4 または IPv6 アドレスを使用できます。 このコマンドによってホスト設定は削除されません。

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*] **Example:** `switch(config)# snmp-server host 192.0.2.1 filter-vrf Red`	設定された VRF に基づいて、通知ホスト レシーバへの通知をフィルタリングします。*ip-address* は IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。 このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。
no snmp-server host *ip-address* **filter-vrf** *vrf_name* **Example:** `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 ExtSnmpTargetVrfTable からエントリを削除します。 *ip-address* は IPv4 または IPv6 アドレスを使用できます。このコマンドによってホスト設定は削除されません。

帯域内ポートを使用してトラップを送信するための SNMP 設定

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

手順の概要

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

手順の詳細

	コマンド	目的
ステップ 1	**configure terminal** **Example:** switch# configure terminal Enter configuration commands, one per line. End with CNTL/Z. switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	**snmp-server source-interface traps** *if-type if-number* **Example:** switch(config)# snmp-server source-interface traps ethernet 1/2	SNMP トラップを送信するための発信元インターフェイスをグローバルに設定します。**?** サポートされているインターフェイス タイプを特定します。 グローバル レベルまたはホスト レベルで発信元インターフェイスを設定できます。発信元インターフェイスをグローバルに設定すると、新しいホスト コンフィギュレーションはグローバルなコンフィギュレーションを使用してトラップを送信します。 **（注）** 発信元インターフェイスをホスト レベルで設定するには、**snmp-server host** *ip-address* **source-interface** *if-type if-number* コマンドを使用します。
ステップ 3	**show snmp source-interface** **Example:** switch(config)# show snmp source-interface	（任意）設定した発信元インターフェイスの情報を表示します。
ステップ 4	**snmp-server host** *ip-address* **use-vrf** *vrf_name* [**udp_port** *number*] **Example:** switch(config)# snmp-server host 171.71.48.164 use_vrf default	特定の VRF を使用してホスト レシーバと通信するように SNMP を設定します。*ip-address* は IPv4 または IPv6 アドレスを使用できます。VRF 名には最大 255 の英数字を使用できます。UDP ポート番号の範囲は 0 ～ 65535 です。このコマンドによって、CISCO-SNMP-TARGET-EXT-MB の ExtSnmpTargetVrfTable にエントリが追加されます。 **（注）** デフォルトでは、SNMP は管理 VRF を使用してトラップを送信します。管理 VRF を使用しない場合は、このコマンドを使用して対象の VRF を指定する必要があります。
ステップ 5	**show snmp host** **Example:** switch(config)# show snmp host	（任意）設定した SNMP ホストの情報を表示します。
ステップ 6	**copy running-config startup-config** **Example:** switch(config)# copy running-config startup-config	（任意）リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

次に、グローバルに設定した帯域内ポートを使用してトラップを送信するよう SNMP を設定する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server source-interface traps ethernet 1/2
switch(config)# show snmp source-interface
-----------------------------------------------------------------------
Notification                     source-interface
-----------------------------------------------------------------------
trap                            Ethernet1/2

inform                                -
-----------------------------------------------------------------------

switch(config)# snmp-server host 171.71.48.164 use_vrf default
switch(config)# show snmp host
-----------------------------------------------------------------------
Host                            Port Version  Level  Type   SecName
-----------------------------------------------------------------------
171.71.48.164                    162  v2c      noauth trap   public

Use VRF: default

Source interface: Ethernet 1/2
-----------------------------------------------------------------------
```

SNMP 通知のイネーブル化

通知をイネーブルまたはディセーブルにできます。通知名を指定しないと、Cisco NX-OS は通知をすべてイネーブルにします。

表 9-4 には、Cisco NX-OS MIB 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** コマンドを使用すると、設定されている通知ホスト レシーバに応じて、トラップおよび応答要求の両方がイネーブルになります。

表 9-4　SNMP 通知のイネーブル化

MIB	関連コマンド
すべての通知	**snmp-server enable traps**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa** **snmp-server enable traps aaa server-state-change**
CISCO-BGP4-MIB	**snmp-server enable traps bgp**
CISCO-CALLHOME-MIB	**snmp-server enable traps callhome** **snmp-server enable traps callhome event-notify** **snmp-server enable traps callhome smtp-send-fail**
CISCO-CONFIG-MAN-MIB	**snmp-server enable traps config** **snmp-server enable traps config ccmCLIRunningConfigChanged**
CISCO-EIGRP-MIB	**snmp-server enable traps eigrp** [*tag*]

表 9-4　SNMP 通知のイネーブル化 （続き）

MIB	関連コマンド
ENTITY-MIB, CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity entity_fan_status_change** **snmp-server enable traps entity entity_mib_change** **snmp-server enable traps entity entity_module_inserted** **snmp-server enable traps entity entity_module_removed** **snmp-server enable traps entity entity_module_status_change** **snmp-server enable traps entity entity_power_out_change** **snmp-server enable traps entity entity_power_status_change** **snmp-server enable traps entity entity_unrecognised_module**
CISCO-FEATURE-CONTROL-MIB	**snmp-server enable traps feature-control** **snmp-server enable traps feature-control FeatureOpStatusChange**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license** **snmp-server enable traps license notify-license-expiry** **snmp-server enable traps license notify-license-expiry-warning** **snmp-server enable traps license notify-licensefile-missing** **snmp-server enable traps license notify-no-license-for-feature**
IF-MIB	**snmp-server enable traps link** **snmp-server enable traps link IETF-extended-linkDown** **snmp-server enable traps link IETF-extended-linkUp** **snmp-server enable traps link cisco-extended-linkDown** **snmp-server enable traps link cisco-extended-linkUp** **snmp-server enable traps link linkDown** **snmp-server enable traps link Up**
OSPF-MIB, OSPF-TRAP-MIB	**snmp-server enable traps ospf** [*tag*] **snmp-server enable traps ospf lsa** **snmp-server enable traps ospf rate-limit** *rate*
CISCO-RF-MIB	**snmp-server enable traps rf** **snmp-server enable traps rf redundancy_framework**
CISCO-RMON-MIB	**snmp-server enable traps rmon** **snmp-server enable traps rmon fallingAlarm** **snmp-server enable traps rmon hcFallingAlarm** **snmp-server enable traps rmon hcRisingAlarm** **snmp-server enable traps rmon risingAlarm**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-SYSTEM-EXT-MIB	**sysmgr** **sysmgr cseFailSwCoreNotifyExtended**
UPGRADE-MIB	**upgrade** **upgrade UpgradeJobStatusNotify** **upgrade UpgradeOpNotifyOnCompletion**

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 **Example:** 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**] **Example:** switch(config)# snmp-server enable traps aaa	AAA SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **server-state-change**：AAA サーバの状態変化通知をイネーブルにします。
snmp-server enable traps bgp **Example:** switch(config)# snmp-server enable traps bgp	ボーダー ゲートウェイ プロトコル（BGP）SNMP 通知をイネーブルにします。
snmp-server enable traps callhome [**event-notify**] [**smtp-send-fail**] **Example:** switch(config)# snmp-server enable traps callhome	Call Home 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **event-notify**：Call Home の外部イベント通知をイネーブルにします。 • **smtp-send-fail**：簡易メール転送プロトコル（SMTP）メッセージの送信失敗通知をイネーブルにします。
snmp-server enable traps config [**ccmCLIRunningConfigChanged**] **Example:** switch(config)# snmp-server enable traps config	コンフィギュレーションの変更に対して SNMP 通知をイネーブルにします。 • **ccmCLIRunningConfigChanged**：実行中または起動時のコンフィギュレーションで、コンフィギュレーションの変更に対して SNMP 通知をイネーブルにします。
snmp-server enable traps eigrp [*tag*] **Example:** switch(config)# snmp-server enable traps eigrp	CISCO-EIGRP-MIB SNMP 通知をイネーブルにします。

コマンド	目的
snmp-server enable traps entity [**entity_fan_status_change**] [**entity_mib_change**] [**entity_module_inserted**] [**entity_module_removed**] [**entity_module_status_change**] [**entity_power_out_change**] [**entity_power_status_change**] [**entity_unrecognised_module**] **Example:** switch(config)# snmp-server enable traps entity	ENTITY-MIB SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **entity_fan_status_change**：エンティティ ファンの状態変化通知をイネーブルにします。 • **entity_mib_change**：エンティティ MIB 変更通知をイネーブルにします。 • **entity_module_inserted**：エンティティ モジュール挿入通知をイネーブルにします。 • **entity_module_removed**：エンティティ モジュール削除通知をイネーブルにします。 • **entity_module_status_change**：エンティティ モジュール ステータス変更通知をイネーブルにします。 • **entity_power_out_change**：エンティティの出力パワー変更通知をイネーブルにします。 • **entity_power_status_change**：エンティティのパワー ステータス変更通知をイネーブルにします。 • **entity_unrecognised_module**：エンティティの未確認モジュール通知をイネーブルにします。
snmp-server enable traps feature-control [**FeatureOpStatusChange**] **Example:** switch(config)# snmp-server enable traps feature-control	機能制御 SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **FeatureOpStatusChange**：機能操作の状態変化通知をイネーブルにします。
snmp-server enable traps license [**notify-license-expiry**] [**notify-license-expiry-warning**] [**notify-licensefile-missing**] [**notify-no-license-for-feature**] **Example:** 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。任意で、次の特定の通知をイネーブルにします。 • **notify-license-expiry**：ライセンス失効通知をイネーブルにします。 • **notify-license-expiry-warning**：ライセンス失効の警告通知をイネーブルにします。 • **notify-licensefile-missing**：ライセンス ファイル不明通知をイネーブルにします。 • **notify-no-license-for-feature**：no-license-installed-for-feature 通知をイネーブルにします。

コマンド	目的
snmp-server enable traps link [**IETF-extended-linkDown**] [**IETF-extended-linkUp**] [**cisco-extended-linkDown**] [**cisco-extended-linkUp**] [**linkDown**] [**linkUp**] **Example:** switch(config)# snmp-server enable traps link	IF-MIB リンク通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **IETF-extended-linkDown**：インターネット技術特別調査委員会（IETF）の拡張リンク ステート ダウン通知をイネーブルにします。 • **IETF-extended-linkUp**：IETF の拡張リンク ステート アップ通知をイネーブルにします。 • **cisco-extended-linkDown**：Cisco 拡張リンク ステート ダウン通知をイネーブルにします。 • **cisco-extended-linkUp**：Cisco 拡張リンク ステート アップ通知をイネーブルにします。 • **linkDown**：IETF リンク ステート ダウン通知をイネーブルにします。 • **linkUp**：IETF リンク ステート アップ通知をイネーブルにします。
snmp-server enable traps ospf [*tag*] [**lsa**] **Example:** switch(config)# snmp-server enable traps ospf	Open Shortest Path First（OSPF）通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **lsa**：OSPF リンク ステート アドバタイズメント（LSA）通知をイネーブルにします。
snmp-server enable traps rf [**redundancy-framework**] **Example:** switch(config)# snmp-server enable traps rf	冗長フレームワーク（RF）SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **redundancy-framework**：RF スーパーバイザ スイッチオーバー MIB 通知をイネーブルにします。
snmp-server enable traps rmon [**fallingAlarm**] [**hcFallingAlarm**] [**hcRisingAlarm**] [**risingAlarm**] **Example:** switch(config)# snmp-server enable traps rmon	リモート モニタリング（RMON）SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **fallingAlarm**：RMON 下限アラーム通知をイネーブルにします。 • **hcFallingAlarm**：RMON high-capacity 下限アラーム通知をイネーブルにします。 • **hcRisingAlarm**：RMON high-capacity 上限アラーム通知をイネーブルにします。 • **risingAlarm**：RMON 上限アラーム通知をイネーブルにします。
snmp-server enable traps snmp [**authentication**] **Example:** switch(config)# snmp-server enable traps snmp	一般的な SNMP 通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **authentication**：SNMP 認証通知をイネーブルにします。

コマンド	目的
snmp-server enable traps sysmgr [**cseFailSwCoreNotifyExtended**] **Example:** switch(config)# snmp-server enable traps sysmgr	ソフトウェア変更通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **cseFailSwCoreNotifyExtended**：ソフトウェア コア通知をイネーブルにします。
snmp-server enable traps upgrade [**UpgradeJobStatusNotify**] [**UpgradeOpNotifyOnCompletion**] **Example:** switch(config)# snmp-server enable traps upgrade	アップグレード通知をイネーブルにします。任意で、次の特定の通知をイネーブルにします。 • **UpgradeJobStatusNotify**：アップグレード ジョブ ステータス通知をイネーブルにします。 • **UpgradeOpNotifyOnCompletion**：アップグレード グローバル ステータス通知をイネーブルにします。

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

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

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 **Example:** switch(config-if)# no snmp trap link-status	インターフェイスの SNMP リンクステート トラップをディセーブルにします。このコマンドは、デフォルトでイネーブルになっています。

インターフェイスの SNMP ifIndex の表示

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

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

TCP による SNMP のワンタイム認証のイネーブル化

TCP セッションでの 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**] **Example:** `switch(config)# snmp-server tcp-session`	TCP セッション上で SNMP に対するワンタイム認証をイネーブルにします。デフォルトではディセーブルになっています。

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）デバイスのコンタクト情報とデバイスのロケーションを指定できます。

手順の概要

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

手順の詳細

	コマンド	目的

ステップ 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`Enter configuration commands, one per line.  End with CNTL/Z.`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**snmp-server contact** *name* **Example:** `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* **Example:** `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** **Example:** `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** **Example:** `switch(config)# copy running-config startup-config`	（任意）この設定の変更を保存します。

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# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
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。

はじめる前に

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

手順の概要

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

手順の詳細

	コマンド	目的

ステップ 1	**configure terminal**
	
	Example:
	`switch# configure terminal`
	`Enter configuration commands, one per line.  End with CNTL/Z.`
	`switch(config)#`
		グローバル コンフィギュレーション モードを開始します。

ステップ 2	**snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*] **Example:** `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* **Example:** `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。

	コマンド	目的
ステップ 4	**show snmp context** **Example:** switch(config)# show snmp context	（任意）1 つまたは複数の SNMP コンテキストに関する情報を表示します。
ステップ 5	**copy running-config startup-config** **Example:** switch(config)# copy running-config startup-config	（任意）リブートおよびリスタート時に実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーして、変更を継続的に保存します。

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

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

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

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

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*] **Example:** 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ないでください。**instance**、**vrf**、または **topology** キーワードを使用すると、コンテキストとゼロ長ストリング間のマッピングが設定されます。

SNMP のディセーブル化

デバイス上で SNMP をディセーブルにすることができます。

SNMP 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 **Example:** switch(config)# no snmp-server protocol enable	SNMP をディセーブルにします。このコマンドは、デフォルトでイネーブルになっています。

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

コマンド	目的
show interface snmp-ifindex	すべてのインターフェイスについて（IF-MIB から）SNMP の ifIndex 値を表示します。
show running-config snmp [**all**]	SNMP の実行コンフィギュレーションを表示します。
show snmp	SNMP ステータスを表示します。
show snmp community	SNMP コミュニティ ストリングを表示します。
show snmp context	SNMP コンテキスト マッピングを表示します。
show snmp engineID	SNMP engineID を表示します。
show snmp group	SNMP ロールを表示します。
show snmp host	設定した SNMP ホストの情報を表示します。
show snmp session	SNMP セッションを表示します。
show snmp source-interface	設定した発信元インターフェイスの情報を表示します。
show snmp trap	イネーブルまたはディセーブルである SNMP 通知を表示します。
show snmp user	SNMPv3 ユーザを表示します。

SNMP の設定例

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

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

次に、ホスト レベルで設定された帯域内ポートを使用してトラップを送信するよう、SNMP を設定する例を示します。

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server host 171.71.48.164 version 2c public
switch(config)# snmp-server host 171.71.48.164 source-interface ethernet 1/2
switch(config)# show snmp host
-----------------------------------------------------------------------
Host                            Port Version  Level  Type   SecName
-----------------------------------------------------------------------
171.71.48.164                    162  v2c      noauth trap   public

Source interface: Ethernet 1/2
-----------------------------------------------------------------------

switch(config)# snmp-server host 171.71.48.164 use_vrf default
switch(config)# show snmp host
-----------------------------------------------------------------------
Host                            Port Version  Level  Type   SecName
-----------------------------------------------------------------------
171.71.48.164                    162  v2c      noauth trap   public

Use VRF: default

Source interface: Ethernet 1/2
-----------------------------------------------------------------------
```

その他の関連資料

SNMP の実装に関する詳細情報については、次の各項を参照してください。

- 「関連資料」（P.9-28）
- 「RFC」（P.9-28）
- 「MIB」（P.9-28）

関連資料

関連項目	マニュアル タイトル
IP ACL および AAA	*『Cisco Nexus 9000 Series NX-OS Security Configuration Guide』*
MIB	*『Cisco Nexus 7000 Series and 9000 Series NX-OS MIB Quick Reference』*

RFC

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

MIB

MIB	MIB のリンク
SNMP に関連する MIB	サポートされている MIB を検索およびダウンロードするには、次の URL にアクセスしてください。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html