SNMP の設定

この章の内容は、次のとおりです。

- SNMP について，1 ページ
- SNMP のライセンス要件，6 ページ
- SNMP の注意事項および制約事項，6 ページ
- SNMP のデフォルト設定，6 ページ
- SNMP の設定，7 ページ
- SNMP のディセーブル化，21 ページ
- SNMP の設定の確認，21 ページ
- SNMP の機能の履歴，22 ページ

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 は、ネットワーク内のデバイスのモニタリングおよび管理に使用する標準フレームワークと共通言語を提供します。

SNMP 機能の概要

SNMP フレームワークは 3 つの部分で構成されます。

- SNMP マネージャ：SNMP を使用してネットワーク デバイスのアクティビティを制御し、モニタリングするシステム。
- SNMP エージェント：デバイスのデータを維持し、必要に応じてこれらのデータを管理システムに報告する、管理対象デバイス内のソフトウェア コンポーネント。 Cisco Nexus 5000

Series スイッチはエージェントおよび MIB をサポートします。 SNMP エージェントをイネーブルにするには、マネージャとエージェントの関係を定義する必要があります。

- MIB（Management Information Base; 管理情報ベース）：SNMP エージェントの管理対象オブジェクトの集まり

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

Cisco Nexus 5000 Series スイッチは、SNMPv1、SNMPv2c、および SNMPv3 をサポートします。SNMPv1 と SNMPv2c は、ともに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 は、RFC 3410（http://tools.ietf.org/html/rfc3410）、RFC 3411（http://tools.ietf.org/html/rfc3411）、RFC 3412（http://tools.ietf.org/html/rfc3412）、RFC 3413（http://tools.ietf.org/html/rfc3413）、RFC 3414（http://tools.ietf.org/html/rfc3414）、RFC 3415（http://tools.ietf.org/html/rfc3415）、RFC 3416（http://tools.ietf.org/html/rfc3416）、RFC 3417（http://tools.ietf.org/html/rfc3417）、RFC 3418（http://tools.ietf.org/html/rfc3418）、および RFC 3584（http://tools.ietf.org/html/rfc3584）で定義されています。

SNMP 通知

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

Cisco NX-OS は、トラップまたはインフォームとして SNMP 通知を生成します。 トラップは、エージェントからホスト レシーバ テーブルで指定された SNMP マネージャに送信される、非同期の非確認応答メッセージです。 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 Acknowledgment（ACK; 確認応答）を送信しないからです。 このため、トラップが受信されたかどうかをスイッチが判断できません。 インフォーム要求を受信する SNMP マネージャは、SNMP 応答 Protocol Data Unit（PDU; プロトコル データ ユニット）でメッセージの受信を確認します。Cisco Nexus 5000 Series スイッチが応答を受信しない場合、インフォーム要求を再度送信できます。

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

SNMPv3

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

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

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

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

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

- noAuthNoPriv：認証または暗号化を実行しないセキュリティ レベル。
- 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 *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	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。

モデル	レベル	認証	暗号化	結果
v3	authNoPriv	HMAC-MD5 または HMAC-SHA	No	Hash-Based Message Authentication Code（HMAC）Message Digest 5（MD5; メッセージ ダイジェスト 5）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 または HMAC-SHA アルゴリズムに基づいて認証します。データ暗号規格（DES）の 56 ビット暗号化、および暗号ブロック連鎖（CBC）DES（DES-56）標準に基づいて認証します。

ユーザベースのセキュリティ モデル

SNMPv3 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ていないことを保証します。また、データ シーケンスが、通常発生するものよりも高い頻度で変更されていないことを保証します。
- メッセージ発信元の認証：受信データを発信したユーザのアイデンティティが確認されたことを保証します。
- メッセージの機密性：情報が使用不可であること、または不正なユーザ、エンティティ、またはプロセスに開示されないことを保証します。

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

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

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

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

priv オプションで、SNMP セキュリティ暗号化方式として、DES または 128 ビット AES を選択できます。 **priv** オプションを **aes-128** トークンと併用すると、プライバシー パスワードは 128 ビット AES キーの生成に使用されます。AES のプライバシー パスワードは最小で 8 文字です。 パスフレーズをクリア テキストで指定する場合、最大 64 文字を指定できます。 ローカライズド キーを使用する場合は、最大 130 文字を指定できます。

（注） 外部の Authentication, Authorization, Accounting（AAA; 認証、許可、アカウンティング）サーバを使用して SNMPv3 を使う場合、外部 AAA サーバのユーザ設定でプライバシー プロトコルに AES を指定する必要があります。

コマンドライン インターフェイス（CLI）および SNMP ユーザの同期

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

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

Cisco NX-OSは、次のようにユーザ設定を同期化します。

- **snmp-server user** コマンドで指定された **auth** パスフレーズは、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は、SNMP ユーザの **auth** および **priv** パスフレーズになります。
- SNMP または CLI を使用してユーザを作成または削除すると、SNMP と CLI の両方でユーザが作成または削除されます。
- ユーザとロールの対応関係の変更は、SNMP と CLI で同期化されます。
- 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、ルールなど）を同期させません。

グループベースの SNMP アクセス

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

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

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

SNMP のライセンス要件

この機能にはライセンスは不要です。 ライセンスパッケージに含まれていない機能はすべて Cisco NX-OS システム イメージにバンドルされており、追加費用は一切発生しません。 NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

SNMP の注意事項および制約事項

Cisco NX-OS は、イーサネット MIB への読み取り専用アクセスをサポートします。

サポートされる MIB の詳細については、次の URL を参照してください。

http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

SNMP のデフォルト設定

表 *2*： デフォルトの *SNMP* パラメータ

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

SNMP の設定

SNMP ユーザの設定

（注） Cisco NX-OS で SNMP ユーザを設定するために使用するコマンドは、Cisco IOS でユーザを設定するために使用されるものとは異なります。

手順の概要

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

手順の詳細

	コマンドまたはアクション	目的

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

ステップ 2	switch(config)# **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。大文字と小文字を区別します。 **localizedkey** キーワードを使用する場合は、パスフレーズに大文字と小文字を区別した英数字を 130 文字まで使用できます。 engineID の形式は、12 桁のコロンで区切った 10 進数字です。
ステップ 3	switch# **show snmp user** 例： `switch(config) # show snmp user`	（任意） 1 人または複数の SNMP ユーザに関する情報を表示します。

	コマンドまたはアクション	目的
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） 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、変更を永続的に保存します。

次の例は、SNMP ユーザを設定します。

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

SNMP メッセージ暗号化の適用

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

SNMP メッセージの暗号化を特定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
switch(config)# **snmp-server user** *name* **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
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 ユーザを作成した後で、そのユーザに複数のロールを割り当てることができます。

（注） 他のユーザにロールを割り当てることができるのは、network-admin ロールに属するユーザだけです。

コマンド	目的
switch(config)# **snmp-server user** *name group*	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 の SNMP コミュニティを作成できます。

グローバル コンフィギュレーション モードで SNMP コミュニティ ストリングを作成する手順は、次のとおりです。

コマンド	目的
switch(config)# **snmp-server community** *name group* {**ro** | **rw**}	SNMP コミュニティ ストリングを作成します。

SNMP 要求のフィルタリング

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

ACL は次のパラメータで作成します。

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

ACL は、UDP および TCP を介する IPv4 および IPv6 の両方に適用されます。 ACL を作成したら、ACL を SNMP コミュニティに割り当てます。

ヒント　ACL の作成の詳細については、使用している Cisco Nexus シリーズ ソフトウェアの『*NX-OS Security Configuration Guide*』を参照してください。 Nexus 5000 用の入手可能なセキュリティ設定ガイドラインは http://www.cisco.com/en/US/products/ps9670/products_installation_and_configuration_guides_list.html にあります。

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

コマンド	目的
switch(config)# **snmp-server community** *community name* **use-acl** *acl-name* **Example:** `switch(config)# snmp-server community public use-acl my_acl_for_public`	ACL を SNMP コミュニティに割り当てて SNMP 要求をフィルタします。

はじめる前に

SNMP コミュニティに割り当てる ACL を作成します。

ACL を SNMP コミュニティに割り当てます。

SNMP 通知レシーバの設定

複数のホスト レシーバに対して SNMP 通知を生成するよう Cisco NX-OSを設定できます。

グローバル コンフィギュレーション モードで SNMPv1 トラップのホスト レシーバを設定できます。

コマンド	目的
switch(config)# **snmp-server host** *ip-address* **traps version 1** *community* [**udp_port** *number*]	SNMPv1 トラップのホスト レシーバを設定します。 *ip-address* は IPv4 または IPv6 アドレスを使用できます。 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。

グローバル コンフィギュレーション モードで SNMPv2c トラップまたはインフォームのホスト レシーバを設定できます。

コマンド	目的
switch(config)# **snmp-server host** *ip-address* {**traps** | **informs**} **version 2c** *community* [**udp_port** *number*]	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。 *ip-address* は IPv4 または IPv6 アドレスを使用できます。 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。

グローバル コンフィギュレーション モードで SNMPv3 トラップまたはインフォームのホスト レシーバを設定できます。

コマンド	目的
switch(config)# **snmp-server host** *ip-address* {**traps** | **informs**} **version 3** {**auth** | **noauth** | **priv**} *username* [**udp_port** *number*]	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。 *ip-address* は IPv4 または IPv6 アドレスを使用できます。 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暗号解除するために、Cisco Nexus 5000 Series スイッチの SNMP engineID に基づくユーザ クレデンシャル（authKey/PrivKey）を認識する必要があります。

次に、SNMPv1 トラップのホスト レシーバを設定する例を示します。

```
switch(config)# snmp-server host 192.0.2.1 traps version 1 public
```

次に、SNMPv2 インフォームのホスト レシーバを設定する例を示します。

```
switch(config)# snmp-server host 192.0.2.1 informs version 2c public
```

次に、SNMPv3 インフォームのホスト レシーバを設定する例を示します。

```
switch(config)# snmp-server host 192.0.2.1 informs version 3 auth NMS
```

すべての SNMP 通知を送信するための送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。

（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。

すべての SNMP 通知を送信するよう送信元インターフェイスを設定するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. switch(config) # **snmp-server source-interface** {**traps** | **informs**} *if-type if-number*

手順の詳細

	コマンドまたはアクション	目的

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

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

次に、SNMPv2c トラップを送信するよう送信元インターフェイスを設定する例を示します。

```
switch# configure terminal
switch(config) # snmp-server source-interface traps ethernet 2/1
```

次の作業

設定した送信元インターフェイスの情報を表示するには、**show snmp source-interface** コマンドを入力します。

SNMP 通知のホスト レシーバの設定

（注） 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 通知を受信する、送信元インターフェイス上のホスト レシーバを設定するには、次の手順を実行します。

手順の概要

1. **configure terminal**
2. switch(config) # **snmp-server host** *ip-address* **source-interface** *if-type if-number* [**udp_port** *number*]

手順の詳細

	コマンドまたはアクション	目的

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

ステップ 2	switch(config) # **snmp-server host** *ip-address* **source-interface** *if-type if-number* [**udp_port** *number*] 例： `switch(config) # snmp-server host 192.0.2.1 source-interface traps ethernet 2/1`	SNMPv2c トラップまたはインフォームのホスト レシーバを設定します。 *ip-address* は IPv4 または IPv6 アドレスを使用できます。 **?** を使用して、 サポートされているインターフェイス タイプを特定します。

次に、すべての SNMP 通知を受信する、送信元インターフェイスを設定する例を示します。

```
switch# config t
switch(config) # snmp-server host 192.0.2.1 source-interface ethernet 2/1
```

次の作業

設定した送信元インターフェイスの情報を表示するには、**show snmp source-interface** コマンドを入力します。

インバンド アクセスのための SNMP の設定

次のものを使用して、インバンド アクセス用に SNMP を設定できます。

- コンテキストのない SNMP v2 の使用：コンテキストにマッピングされたコミュニティを使用できます。 この場合、SNMP クライアントはコンテキストについて認識する必要はありません。
- コンテキストのある SNMP v2 の使用：SNMP クライアントはコミュニティ、たとえば、<community>@<context> を指定して、コンテキストを指定する必要があります。
- SNMP v3 の使用：コンテキストを指定できます。

手順の概要

1. switch# **configuration terminal**
2. switch(config)# **snmp-server context** *context-name* **vrf** *vrf-name*
3. switch(config)# **snmp-server community** *community-name* **group** *group-name*
4. switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	switch# **configuration terminal**	グローバル コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# **snmp-server context** *context-name* **vrf** *vrf-name*	管理 VRF またはデフォルト VRF に SNMP コンテキストをマッピングします。 カスタム VRF はサポートされません。 名前には最大 32 の英数字を使用できます。
ステップ 3	switch(config)# **snmp-server community** *community-name* **group** *group-name*	SNMPv2c コミュニティと SNMP コンテキストにマッピングし、コミュニティが属するグループを識別します。 名前には最大 32 の英数字を使用できます。
ステップ 4	switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*	SNMPv2c コミュニティを SNMP コンテキストにマッピングします。 名前には最大 32 の英数字を使用できます。

次の SNMPv2 の例は、コンテキストに snmpdefault という名前のコミュニティをマッピングする方法を示しています。

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)# snmp-server community snmpdefault group network-admin
switch(config)# snmp-server mib community-map snmpdefault context def
switch(config)#
```

次の SNMPv2 の例は、マッピングされていないコミュニティ comm を設定し、インバンド アクセスする方法を示しています。

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
```

```
switch(config)# snmp-server community comm group network-admin
switch(config)#
```

次の SNMPv3 の例は、v3 ユーザ名とパスワードを使用する方法を示しています。

```
switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# snmp-server context def vrf default
switch(config)#
```

SNMP 通知のイネーブル化

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

（注） **snmp-server enable traps** CLI コマンドを使用すると、設定通知ホスト レシーバによっては、トラップとインフォームの両方をイネーブルにできます。

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

表 *3*： ***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**
BRIDGE-MIB	**snmp-server enable traps bridge newroot** **snmp-server enable traps bridge topologychange**
CISCO-AAA-SERVER-MIB	**snmp-server enable traps aaa**
ENITY-MIB、 CISCO-ENTITY-FRU-CONTROL-MIB、 CISCO-ENTITY-SENSOR-MIB	**snmp-server enable traps entity** **snmp-server enable traps entity fru**
CISCO-LICENSE-MGR-MIB	**snmp-server enable traps license**
IF-MIB	**snmp-server enable traps link**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
SNMPv2-MIB	**snmp-server enable traps snmp** **snmp-server enable traps snmp authentication**
CISCO-FCC-MIB	**snmp-server enable traps fcc**
CISCO-DM-MIB	**snmp-server enable traps fcdomain**
CISCO-NS-MIB	**snmp-server enable traps fcns**

MIB	関連コマンド
CISCO-FCS-MIB	**snmp-server enable traps fcs discovery-complete** **snmp-server enable traps fcs request-reject**
CISCO-FDMI-MIB	**snmp-server enable traps fdmi**
CISCO-FSPF-MIB	**snmp-server enable traps fspf**
CISCO-PSM-MIB	**snmp-server enable traps port-security**
CISCO-RSCN-MIB	**snmp-server enable traps rscn** **snmp-server enable traps rscn els** **snmp-server enable traps rscn ils**
CISCO-ZS-MIB	**snmp-server enable traps zone** **snmp-server enable traps zone default-zone-behavior-change** **snmp-server enable traps zone merge-failure** **snmp-server enable traps zone merge-success** **snmp-server enable traps zone request-reject** **snmp-server enable traps zone unsupp-mem**

（注） ライセンス通知は、デフォルトではイネーブルです。

グローバルコンフィギュレーションモードで指定の通知をイネーブルにするには、次の作業を行います。

コマンド	目的
switch(config)# **snmp-server enable traps**	すべての SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps aaa** [**server-state-change**]	AAA SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps entity** [**fru**]	ENTITY-MIB SNMP 通知をイネーブルにします。
switch(config)# **snmp-server enable traps license**	ライセンス SNMP 通知をイネーブルにします。
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
switch(config)# **snmp-server enable traps snmp** [**authentication**]	SNMP エージェント通知をイネーブルにします。

リンクの通知の設定

デバイスに対して、イネーブルにする linkUp/linkDown 通知を設定できます。 次のタイプの linkUp/linkDown 通知をイネーブルにできます。

- Cisco：Cisco NX-OS は、インターフェイスに対して ifLinkUpDownTrapEnable（IF-MIB で定義）がイネーブルの場合は、シスコ定義の通知（CISCO-IF-EXTENSION-MIB.my の cieLinkUp、cieLinkDown）だけを送信します。
- IETF：Cisco NX-OS は、インターフェイスに対して ifLinkUpDownTrapEnable（IF-MIB で定義）がイネーブルの場合は、定義されている変数バインドだけを IETF 定義の通知（IF-MIB の linkUp、linkDown）と一緒に送信します。
- IEFT extended：Cisco NX-OS は、インターフェイスに対して ifLinkUpDownTrapEnable（IF-MIB で定義）がイネーブルの場合は、IETF 定義の通知（IF-MIB の linkUp、linkDown）だけを送信します。 Cisco NX-OS は、IF-MIB に定義されている変数バインドに加え、シスコに固有の変数バインドも送信します。 これがデフォルトの設定です。
- IEFT Cisco：Cisco NX-OS は、インターフェイスに対して ifLinkUpDownTrapEnable（IF-MIB で定義）がイネーブルの場合は、IF-MIB に定義された通知（linkUp、linkDown）および CISCO-IF-EXTENSION-MIB.my に定義された通知（cieLinkUp、cieLinkDown）を送信します。 Cisco NX-OS は、linkUp および linkDown 通知に定義された変数バインドだけを送信します。
- IEFT extended Cisco：Cisco NX-OS は、インターフェイスに対して ifLinkUpDownTrapEnable（IF-MIB で定義）がイネーブルの場合は、IF-MIB に定義された通知（linkUp、linkDown）および CISCO-IF-EXTENSION-MIB.my に定義された通知（cieLinkUp、cieLinkDown）を送信します。 Cisco NX-OS は、linkUp および linkDown 通知の IF-MIB に定義されている変数バインドに加え、シスコ固有の変数バインドも送信します。

手順の概要

1. **configure terminal**
2. **snmp-server enable traps link** [**cisco**] [**ietf** | **ietf-extended**]

手順の詳細

	コマンドまたはアクション	目的

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

ステップ 2	**snmp-server enable traps link** [**cisco**] [**ietf** | **ietf-extended**] 例： `switch(config)# snmp-server enable traps link cisco`	リンク SNMP 通知をイネーブルにします。

インターフェイスでのリンク通知のディセーブル化

個別のインターフェイスで linkUp および linkDown 通知をディセーブルにできます。 フラッピング インターフェイス（Up と Down の間を頻繁に切り替わるインターフェイス）で、この制限通知を使用できます。

手順の概要

1. switch# **configure terminal**
2. switch(config)# **interface** *type slot/port*
3. switch(config -if)# **no snmp trap link-status**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	switch# **configure terminal**	コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# **interface** *type slot/port*	変更するインターフェイスを指定します。
ステップ 3	switch(config -if)# **no snmp trap link-status**	インターフェイスの SNMP リンクステート トラップをディセーブルにします。 デフォルトでは、イネーブルです。

TCP での SNMP に対するワンタイム認証のイネーブル化

TCP セッション上で SNMP に対するワンタイム認証をイネーブルにできます。

コマンド	目的
switch(config)# **snmp-server tcp-session** [**auth**]	TCP セッション上で SNMP に対するワンタイム認証をイネーブルにします。デフォルトはディセーブルです。

SNMP スイッチの連絡先および場所の情報の割り当て

スイッチの連絡先情報（スペースを含めず、最大32文字まで）およびスイッチの場所を割り当てることができます。

手順の概要

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

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	switch# **configuration terminal**	コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# **snmp-server contact** *name*	sysContact（SNMP 担当者名）を設定します。
ステップ 3	switch(config)# **snmp-server location** *name*	sysLocation（SNMP ロケーション）を設定します。
ステップ 4	switch# **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	switch# **copy running-config startup-config**	（任意） この設定変更を保存します。

コンテキストとネットワーク エンティティ間のマッピング設定

プロトコル インスタンス、VRF などの論理ネットワーク エンティティに対する SNMP コンテキストのマッピングを設定できます。

手順の概要

1. switch# **configuration terminal**
2. switch(config)# **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]
3. switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*
4. （任意） switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	switch# **configuration terminal**	コンフィギュレーション モードを開始します。
ステップ 2	switch(config)# **snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	SNMP コンテキストをプロトコル インスタンス、VRF、またはトポロジにマッピングします。 名前には最大 32 の英数字を使用できます。
ステップ 3	switch(config)# **snmp-server mib community-map** *community-name* **context** *context-name*	SNMPv2c コミュニティを SNMP コンテキストにマッピングします。 名前には最大 32 の英数字を使用できます。
ステップ 4	switch(config)# **no snmp-server context** *context-name* [**instance** *instance-name*] [**vrf** *vrf-name*] [**topology** *topology-name*]	（任意） 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

手順の概要

1. **configure terminal**
2. switch(config) # **no snmp-server protocol enable**

手順の詳細

	コマンドまたはアクション	目的

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

ステップ 2	switch(config) # **no snmp-server protocol enable** 例： `no snmp-server protocol enable`	SNMP をディセーブルにします。 SNMP は、デフォルトでディセーブルになっています。

SNMP の設定の確認

SNMP 設定情報を表示するには、次の作業を行います。

コマンド	目的
switch# **show snmp**	SNMP のステータスを表示します。
switch# **show snmp community**	SNMP コミュニティ ストリングを表示します。
switch# **show snmp engineID**	SNMP engineID を表示します。
switch# **show snmp group**	SNMP ロールを表示します。
switch# **show snmp sessions**	SNMP セッションを表示します。
switch# **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。
switch# **show snmp user**	SNMPv3 ユーザを表示します。

SNMP の機能の履歴

表 *4* : ***SNMP*** の機能の履歴

機能名	リリース	情報
IPv6 のサポート	5.2(1)N1(1)	この機能が導入されました。