簡易ネットワーク管理プロトコル

SNMP の概要

Simple Network Management Protocol（SNMP）は、「IP ネットワーク上のデバイスを管理するためのインターネット標準プロトコル」です。

SNMP とは何ですか?

簡易ネットワーク管理プロトコル（SNMP）は、アプリケーション層プロトコルであり、マネージャとエージェントとの通信に使用されます。SNMP システムは、SNMP マネージャ、SNMP エージェント、および管理情報ベース（MIB）で構成されます。

SNMP マネージャは、Cisco Prime Infrastructure などのネットワーク管理システム（NMS）に統合できます。エージェントと MIB はネットワークデバイス上に存在します。デバイスに SNMP を設定するには、マネージャとエージェントの間の関係を定義します。

SNMP エージェントは MIB 変数を格納し、SNMP マネージャはこの変数の値を要求または変更できます。マネージャはエージェントから値を取得したり、エージェントに値を格納したりできます。エージェントは、デバイスパラメータやネットワーク データの保存場所である MIB から値を収集します。また、エージェントはマネージャのデータ取得またはデータ設定の要求に応答できます。

エージェントは非送信請求トラップをマネージャに送信できます。トラップは、ネットワーク上のある状態を SNMP マネージャに通知するメッセージです。トラップは不正なユーザー認証、再起動、リンク ステータス（アップまたはダウン）、MAC アドレス追跡、TCP 接続の終了、ネイバーとの接続の切断などの重要なイベントの発生を意味する場合があります。

SNMP バージョン

SNMP バージョンには、ネットワークデバイスを管理するためのさまざまな機能があります。このソフトウェアリリースは、SNMPv1、SNMPv2C、および SNMPv3 をサポートします。

- SNMPv1：RFC 1157 に規定された完全インターネット標準の簡易ネットワーク管理プロトコルです。
- SNMPv2C：SNMPv2Classic のパーティベースの管理およびセキュリティフレームワークを、コミュニティストリングベースのフレームワークに置き換えます。SNMPv2Classic の一括取得機能は保持され、エラー処理が改善されます。

  SNMPv2C には次が含まれます。

  - SNMPv2：RFC 1902 ～ 1907 に規定された SNMP バージョン 2（ドラフト版インターネット標準）
  - SNMPv2C：RFC 1901 に規定された SNMPv2 のコミュニティ ストリング ベースの管理フレームワーク（試験版インターネットプロトコル）

  SNMPv1 と SNMPv2C は、ともにコミュニティベースのセキュリティモデルを使用します。Management Information Base（MIB）にアクセスできるマネージャのコミュニティは、IP アドレスアクセスコントロールリストとパスワードによって定義されます。

  SNMPv2C には、テーブルや大量の情報を取得し、必要な往復回数を削減する一括取得機能が含まれています。また、SNMPv1 では単一のエラーコードで報告されるさまざまなエラー状態を区別する拡張エラーコードを提供し、エラー処理が改善されています。

- SNMPv3：SNMPv3（SNMP のバージョン 3）は、RFC 2273 ～ 2275 に規定されている相互運用可能な標準ベースのプロトコルです。SNMPv3 は、ネットワーク経由のパケットの認証と暗号化によって、デバイスへのセキュア アクセスを実現します。これには、次のセキュリティ機能が含まれています。

  - メッセージの完全性：パケットが伝送中に改ざんされないようにします。
  - 認証：メッセージが有効な送信元からのものであることを確認します。
  - 暗号化：パッケージの内容をミキシングし、許可されていない送信元に内容が読まれることを防止します。

（注） 暗号化を選択するには、priv キーワードを入力します。

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、ユーザとユーザが属しているグループ用に設定された認証方式です。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。セキュリティレベルとセキュリティモデルの組み合わせにより、SNMP パケットを扱うときに使用するセキュリティ方式が決まります。使用可能なセキュリティ モデルは、SNMPv1、SNMPv2C、および SNMPv3 です。

次の表では、この特性を識別し、セキュリティ モデルとセキュリティ レベルの異なる組み合わせを比較します。

表 *1*：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	Level	認証	暗号化	結果
SNMPv1	noAuthNoPriv	コミュニティ ストリング	未対応	コミュニティ ストリングの照合を使用して認証します。
SNMPv2C	noAuthNoPriv	コミュニティ ストリング	未対応	コミュニティ ストリングの照合を使用して認証します。
SNMPv3	noAuthNoPriv	ユーザー名	未対応	ユーザ名の照合を使用して認証します。
SNMPv3	authNoPriv	Message Digest 5（MD5）または Secure Hash Algorithm（SHA）	未対応	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。

モデル	Level	認証	暗号化	結果
SNMPv3	authPriv	MD5 または SHA	データ暗号規格（DES）または Advanced Encryption Standard（AES）	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。 次の暗号化アルゴリズムで、User-based Security Model（USM）を指定できます。 • CBC-DES（DES-56）規格に基づく認証に加えた DES 56 ビット暗号化 • 3DES 168 ビット暗号化 • AES 128 ビット暗号化、192 ビット暗号化、または 256 ビット暗号化

SNMP マネージャ機能

SNMP マネージャは、MIB 情報を使用して、この表に示すようにさまざまな動作を実行します。

表 *2: SNMP* の動作

動作	説明
get-request	特定の変数から値を取得します。
get-next-request	テーブル内の変数から値を取得します。この操作では、SNMP マネージャは正確な変数名を把握する必要はありません。テーブル内から必要な変数を見つけるために、シーケンシャル検索が実行されます。
get-bulk-request	テーブルの複数の行など、通常はサイズの小さい多数のデータ ブロックに分割して送信する必要がある巨大なデータ ブロックを取得します。 （注） このコマンドを使用できるのは、SNMPv2 以上に限られます。

動作	説明
get-response	NMS から送信される get-request、get-next-request、および set-request に対して応答します。
set-request	特定の変数に値を格納します。
trap	SNMP エージェントから SNMP マネージャに送られる、イベントの発生を伝える非送信請求メッセージです。

（注） パフォーマンスに関連する問題を回避するために、SNMP マネージャで ciscoFlashFileDate MIB オブジェクトをクエリから除外することを推奨します。これは、ciscoFlashFileDate オブジェクトが MIB で公開されていても、製品ではサポートされていないためです。

SNMP エージェント機能

SNMP エージェントは、1 つ以上の SNMP マネージャから要求を受信できます。すべての要求に、NMS の IP アドレス、NMS がエージェントをポーリングした回数、およびポーリングのタイムスタンプが含まれます。この情報は、IPv4 サーバーと IPv6 サーバーの両方で追跡できます。

SNMP エージェントは、次のようにして SNMP マネージャ要求に応答します。

- MIB 変数を取得する：SNMP エージェントは、NMS の要求に応じて要求された MIB 変数の値を取得し、その値で NMS に応答します。
- MIB 変数を設定する：SNMP エージェントは、NMS からのメッセージに応じて MIB 変数の値を要求された値に変更します。

show snmp stats hosts コマンドを使用して、キュー内の SNMP マネージャ要求のリストを表示します。**clear snmp stats hosts** コマンドを使用してキューをクリアします。

エージェントで重要なイベントが発生したことを NMS に通知するために、SNMP エージェントは非送信請求トラップ メッセージも送信します。トラップ条件の例には、ポートまたはモジュールがアップまたはダウン状態になった場合、スパニングツリートポロジが変更された場合、認証に失敗した場合などがあります。

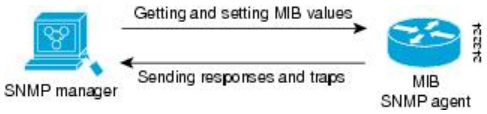
SNMP MIB 変数アクセス

Cisco Prime Infrastructure 3.1 ソフトウェアは NMS の例です。ソフトウェアは、デバイス MIB 変数を使用してデバイス変数を設定し、ネットワーク上のデバイスをポーリングして特定の情報を取得します。ポーリング結果は、グラフ形式で表示されます。この結果を解析して、イン

ターネットワーキング関連の問題のトラブルシューティング、ネットワーク パフォーマンスの改善、デバイス設定の確認、トラフィック負荷のモニターを行うことができます。

SNMP エージェントは MIB からデータを収集します。エージェントは SNMP マネージャに対し、トラップ（特定イベントの通知）を送信でき、SNMPマネージャはトラップを受信して処理します。トラップは、ネットワーク上で発生した不正なユーザー認証、再起動、リンクステータス（アップまたはダウン）、MACアドレストラッキングなどの状況をSNMPマネージャに通知します。SNMP エージェントはさらに、SNMP マネージャから、**get-request**、**get-next-request**、**set-request** 形式で送信される MIB 関連のクエリに応答します。

図 *1: SNMP* ネットワーク



SNMP フラッシュ MIB

Cisco フラッシュ MIB を使用すると、シスコ製のデバイスからフラッシュファイルデータを照会できます。フラッシュ MIB では、フラッシュファイルシステムからすべてのファイルを取得するようになりました。

フラッシュ MIB ウォークを実行するには、**snmp mib flash cache** コマンドを使用する必要があります。このコマンドは、すべてのファイルをローカルフラッシュ MIB キャッシュにプリフェッチします。

SNMP 通知

SNMPを使用すると、特定のイベントが発生した場合に、デバイスからSNMPマネージャに通知を送信できます。SNMP通知は、トラップまたは情報要求として送信できます。コマンド構文では、トラップまたは情報を選択するオプションがコマンドにない限り、キーワード traps はトラップ、情報、またはその両方を表します。**snmp-server host** コマンドを使用して、トラップまたは情報として SNMP 通知を送信するかどうかを指定します。

Note SNMPv1 は informs をサポートしていません。

トラップとインフォーム

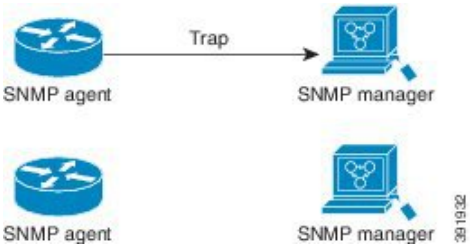
トラップは信頼性に欠けます。受信側はトラップを受信しても確認応答を送信しないので、トラップが受信されたかどうかが送信側にわからないからです。情報要求の場合、受信したSNMPマネージャは SNMP 応答プロトコル データ ユニット（PDU）でメッセージを確認します。送信側が応答を受信しなかった場合は、再び情報要求を送信できます。再送信できるので、情報の方がトラップより意図した宛先に届く可能性が高くなります。

情報の方がトラップより信頼性が高いのは、デバイスおよびネットワークのリソースを多く消費するという特性にも理由があります。送信と同時に廃棄されるトラップと異なり、情報要求は応答を受信するまで、または要求がタイムアウトになるまで、メモリ内に保持されます。トラップの送信は1回限りですが、情報は数回にわたって再送信つまり再試行が可能です。再送信の回数が増えるとトラフィックが増加し、ネットワークのオーバーヘッドが高くなる原因にもなります。したがって、トラップにするか情報にするかは、信頼性を取るかリソースを取るかという選択になります。SNMPマネージャですべての通知を受信することが重要な場合は、情報要求を使用してください。ネットワークまたはデバイスのメモリ上のトラフィックが問題になる場合で、なおかつ通知が不要な場合は、トラップを使用してください。

下図に、トラップとインフォームの違いを示します。

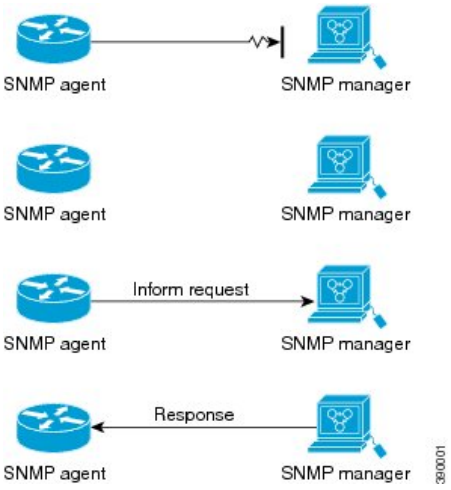
下図は、エージェントが SNMP マネージャへ正常にトラップを送信した場合を示します。マネージャはトラップを受信しても、確認応答を送信しません。エージェントには、トラップが宛先に到達したことを知る方法がありません。

Figure 2: SNMP マネージャに正常に送信されたトラップ



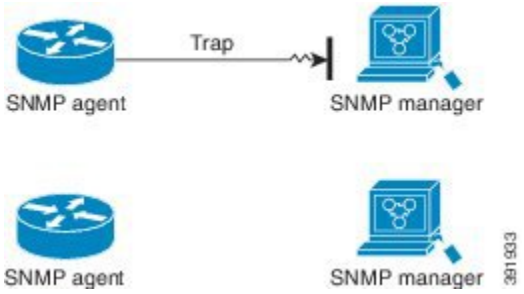
下図では、エージェントはマネージャへ正常にインフォームを送信しています。マネージャがインフォームを受信すると、応答がエージェントに送信されます。これにより、エージェントはインフォームが宛先に到達したことがわかります。この例では、上図に示すインタラクション内で 2 倍のトラフィックが生成されていることに注意してください。

Figure 3: SNMP マネージャに正常に送信されたインフォーム要求



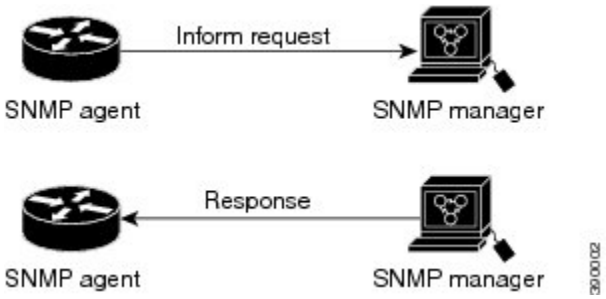
下図は、マネージャが受信しないトラップをマネージャに送信するエージェントを示します。エージェントには、トラップが宛先に到達しなかったことを知る方法がありません。トラップが再送信されないため、マネージャはトラップを受信しません。

Figure 4: SNMP マネージャに正常に送信されなかったトラップ



下図は、マネージャに到達しないインフォームをマネージャに送信するエージェントを示します。マネージャはインフォームを受信しなかったため、応答を送信しません。一定時間が経過すると、エージェントがインフォームを再送信します。マネージャは、2 番目の送信からインフォームを受信して応答します。この例では、上図に示すシナリオよりも多くのトラフィックが生成されますが、通知は SNMP マネージャに到達します。

Figure 5: SNMP マネージャに正常に送信されなかったインフォーム



Note SNMP プロセスが起動するたびに、予約ポート 161 および 162 が使用されます。これら 2 つの予約ポートに加えて、SNMP プロキシ フォワーダ アプリケーションを実行するためにダイナミックポートも開かれます。

SNMP ifIndex MIB オブジェクト値

SNMP エージェントの IF-MIB モジュールがリブート後すぐに起動されます。さまざまな物理インターフェイス ドライバが IF-MIB モジュールに登録を初期化し、ifIndex 番号を要求します。IF-MIB モジュールが先着順で使用可能な次の ifIndex 番号を割り当てます。1 つのリブートから他のリブートへのドライバの初期化順序のマイナーな違いが、同じ物理インターフェイスにリブートを行う以前のものとは別のインデックス番号を取得する可能性があるということです（インデックス持続が有効化されていない限り）。

SNMP ENTITY-MIB 識別子

ENTITY-MIBには、現場交換可能ユニット（FRU）、ファン、デバイスの電源装置などの物理エンティティを管理するための情報が含まれています。

各エンティティは、現在の MIB や他の MIB 内のエンティティに関する情報にアクセスする一意のインデックス番号（entPhysicalIndex）によって識別されます。エンティティの活性挿抜（OIR）により、新しいエンティティが挿入されたか、既存のエンティティが再挿入されたかに関係なく、エンティティには次に使用可能な entPhysicalIndex 番号が割り当てられます。

SNMP および Syslog Over IPv6

IPv4 と IPv6 の両方をサポートするには、IPv6 のネットワーク管理で IPv4 および IPv6 のトランスポートが必要になります。Syslog over IPv6 は、このトランスポートのアドレス データ タイプをサポートします。

Simple Network Management Protocol (SNMP) と syslog over IPv6 は、次の機能を提供します。

- IPv4 と IPv6 両方のサポート。
- SNMP に対する IPv6 トランスポート、および SNMP 変更による IPv6 ホストのトラップのサポート。
- IPv6 アドレス指定をサポートするための SNMP および syslog に関連する MIB。
- IPv6 ホストをトラップの受信者として設定。

Over IPv6 をサポートするため、SNMP は既存の IP トランスポート マッピングを変更して、IPv4 と IPv6 を同時にサポートします。次の SNMP 動作は、IPv6 トランスポート管理をサポートします。

- デフォルト設定のユーザー データグラム プロトコル（UDP）SNMP ソケットを開く。
- SR_IPV6_TRANSPORT と呼ばれる新しいトランスポートメカニズムを提供。
- IPv6 トランスポートによる SNMP 通知の送信。
- IPv6 トランスポートの SNMP 名のアクセスリストのサポート。
- IPv6 トランスポートを使用した SNMP プロキシ転送のサポート。
- SNMP マネージャ機能と IPv6 トランスポートの連動確認。

SNMP UDP ポート

SNMP プロセスはユーザーデータグラムプロトコル（UDP）ポート 161 および 162 を使用します。ポート 161 はデバイスのポーリングするために使用され、ポート 162 はエージェントから

サーバーに通知を送信するために使用されます。これらのポートは、いずれかの必須コマンドを設定しない限り、閉じたままになります。この設計により、必要な場合にのみポートが開くため、セキュリティが向上し、デバイスは不必要にポートをリッスンしなくなります。

SNMP のデフォルト設定

Table 3: 表 *3. SNMP* のデフォルト設定

機能	デフォルト設定
SNMP エージェント	ディセーブル[1]
SNMP トラップ レシーバ	未設定
SNMP トラップ	TCP 接続のトラップ（tty）以外は、イネーブルではありません。
SNMP バージョン	バージョン キーワードがない場合、デフォルトはバージョン 1 になります。
SNMPv3 認証	キーワードを入力しなかった場合、セキュリティ レベルはデフォルトで **noauth**（noAuthNoPriv）になります。
SNMP 通知タイプ	タイプが指定されていない場合、すべての通知が送信されます。

[1] これは、デバイスが起動し、スタートアップ コンフィギュレーションに **snmp-server** グローバル コンフィギュレーション コマンドが設定されていない場合のデフォルトです。

SNMP の制約事項

- SNMPv1 は informs をサポートしていません。
- SNMPv3 認証は、次のシナリオではサポートされません。
  - スイッチ優先順位の変更後にスタックリロードが発生した場合。
  - 低い MAC アドレスを持つデバイスがスタックに追加された場合、スタック内のすべてのスイッチの優先順位が同じであれば、そのデバイスがアクティブスイッチとして選択されます。
- SNMPv3 認証の失敗を回避するには、SNMPv3 ユーザーを設定する前に、デバイスで SNMP engineID を手動で設定します。この設定により、ユーザーは engineID に関連付けられているためデバイスを管理できます。
- SNMP ENTITY-MIB は、イーサネット管理ポートではサポートされていません。

SNMP の設定方法

ここでは、SNMP の設定方法について説明します。

SNMP 設定時の注意事項

デバイスでは、SNMP User Datagram Protocol（UDP）ポート 161 および 162 を開き、SNMP エージェントを有効にするために、次のいずれかのグローバル コンフィギュレーション コマンドを設定する必要があります。

snmp-server host、**snmp-server user**、**snmp-server community**、または **snmp-server manager**

SNMP グループを設定するときには、次の注意事項に従ってください。

- SNMP グループを設定するときは、通知ビューを指定しません。snmp-server host グローバル コンフィギュレーション コマンドがユーザーの通知ビューを自動生成し、そのユーザーを対応するグループに追加します。グループの通知ビューを変更すると、そのグループに対応付けられたすべてのユーザが影響を受けます。
- リモートユーザを設定する場合は、ユーザが存在するデバイスのリモート SNMP エージェントに対応する IP アドレスまたはポート番号を指定します。
- 特定のエージェントのリモートユーザーを設定する前に、**snmp-serverengineID** グローバル コンフィギュレーション コマンドをオプションとともに使用して、SNMP エンジン ID を設定してください。リモートエージェントの SNMP エンジン ID およびユーザーパスワードを使用して認証およびプライバシーダイジェストが算出されます。先にリモートエンジン ID を設定しておかないと、コンフィギュレーション コマンドがエラーになります。
- SNMP 情報を設定するときには、プロキシ要求または情報の送信先となるリモートエージェントの SNMP エンジン ID を SNMP データベースに設定しておきます。
- ローカルユーザがリモートホストと関連付けられていない場合、デバイスは **auth**（authNoPriv）および **priv**（authPriv）の認証レベルの情報を送信しません。
- SNMP エンジン ID を変更する場合は注意が必要です。（コマンドラインで入力された）ユーザーのパスワードは、パスワードおよびローカルエンジン ID に基づいて、MD5 または SHA セキュリティダイジェストに変換されます。コマンドラインのパスワードは、RFC 2274 の規定に従って廃棄されます。エンジン ID の値を変更すると、SNMPv3 ユーザーのセキュリティダイジェストが無効になります。その後、snmp-server user username グローバル コンフィギュレーション コマンドを使用して SNMP ユーザーを再設定する必要があります。エンジン ID を変更した場合は、同様の制限によってコミュニティ ストリングも再設定する必要があります。
- snmp-server host コマンドをデフォルトの UDP ポート（162）で設定すると、show running-config コマンドの出力に UDP ポート値が表示されません。snmp-server host {host-addr} community-string udp-port value コマンドを使用してデフォルト以外の UDP ポート値を指定すると、UDP ポート番号がコマンド出力に表示されます。デフォルトの UDP ポート 162

を使用しても使用しなくても snmp-server host コマンドを設定できます。ただし、両方を同時に設定することはできません。

正しい例を次に示します。

```
Device(config)# snmp-server host 10.10.10.10 community udp-port 163
Device(config)# snmp-server host 10.10.10.10 community
Device(config)# snmp-server host 10.10.10.10 community udp-port 163
Device(config)# snmp-server host 10.10.10.10 community udp-port 162
```

次の例は正しくありません。

```
Device(config)# snmp-server host 10.10.10.10 community udp-port 163
Device(config)# snmp-server host 10.10.10.10 community
Device(config)# snmp-server host 10.10.10.10 community udp-port 162
Device(config)# snmp-server host 10.10.10.10 community udp-port 163
Device(config)# snmp-server host 10.10.10.10 community udp-port 162
Device(config)# snmp-server host 10.10.10.10 community
```

SNMP グループおよびユーザの設定

はじめる前に

デバイスのローカルまたはリモート SNMP サーバエンジンを表す識別名（エンジン ID）を指定できます。SNMP ユーザーを SNMP ビューにマッピングする、SNMP サーバーグループを設定し、新規ユーザーを SNMP グループに追加します。

このセクションでは、デバイスで SNMP グループとユーザーを設定する方法について説明します。

手順

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： Device> enable	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： Device# **configure terminal**	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3：	**snmp-server engineID** {**local** engineid-string | **remote** ip-address [**udp-port** port-number] engineid-string} 例： Device(config)# **snmp-server engineID local 1234**	SNMP のローカル コピーまたはリモート コピーに名前を設定します。 • *engineid-string* は、SNMP のコピー名を指定する 24 文字の ID ストリングです。後続ゼロが含まれる場合は、24 文字のエンジン ID すべてを指定する必要はありません。指定するのは、エンジン ID のうちゼロのみが続く箇所を除いた部分だけです。手順例では、123400000000000000000000 のエンジン ID を設定します。 • **remote** を指定した場合、SNMP のリモートコピーが置かれているデバイスの ip-address を指定し、任意でリモートデバイスのユーザー データグラム プロトコル（UDP）ポートを指定します。デフォルトは 162 です。

	コマンドまたはアクション	目的
ステップ 4	**snmp-server group** *group-name* { **v1** | **v2c** | **v3** { **auth** | **noauth** | **priv**} } [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] 例： Device(config)# **snmp-server group public v2c access lmnop**	リモート デバイス上で新しい SNMP グループを設定します。 *group-name* には、グループの名前を指定します。 次のいずれかのセキュリティ モデルを指定します。 • **v1** は、最も安全性の低いセキュリティ モデルです。 • **v2c** は、2 番目に安全性の低いセキュリティ モデルです。標準の 2 倍の幅で情報および整数を伝送できます。 • **v3**、最も安全な場合には、次の認証レベルの 1 つを選択する必要があります。 **auth**：MD5 および SHA によるパケット認証が可能です。 **noauth**：noAuthNoPriv というセキュリティ レベルをイネーブルにします。キーワードを指定しなかった場合、これがデフォルトです。 **priv**：データ暗号規格（DES）によるパケット暗号化をイネーブルにします（privacy とも呼ばれます）。 （任意）**read** *readview* とともに、エージェントの内容を表示できるビュー名を表す文字列（64 文字以内）を入力します。 （任意）**write** *writeview* とともに、データを入力し、エージェントの内容を表示できるビュー名を表す文字列（64 文字以内）を入力します。 （任意）**notify** *notifyview* とともに、通知、情報、またはトラップを指定するビュー名を表す文字列（64 文字以内）を入力します。 （任意）**access** *access-list* とともに、アクセスリスト名を表す文字列（64 文字以内）を入力します。

	コマンドまたはアクション	目的
ステップ 5	**snmp-server user** username group-name {**remote** host [**udp-port** port]} { **v1** [**access** access-list] | **v2c** [**access** access-list] | **v3** [**encrypted**] [**access** access-list] [**auth** {**md5** | **sha**} auth-password]} [priv {**des** | **3des** | **aes** {**128** | **192** | **256**}} priv-password] 例： Device(config)# **snmp-server user Pat public v2c**	

	コマンドまたはアクション	目的
		SNMP グループに対して新規ユーザを追加します。 *username* は、エージェントに接続するホスト上のユーザ名です。 *group-name* は、ユーザが関連付けられているグループの名前です。 **remote** を入力して、ユーザーが所属するリモート SNMP エンティティおよびそのエンティティのホスト名または IP アドレスを指定し、任意で UDP ポート番号を指定します。デフォルトは 162 です。 SNMP バージョン番号（**v1**、**v2c**、または **v3**）を入力します。**v3** を入力する場合は、次のオプションを追加します。 • **encrypted** は、パスワードを暗号化形式で表示することを指定します。このキーワードは、**v3** キーワードが指定されている場合のみ使用可能です。 • **auth** は認証レベル設定セッションです。HMAC-MD5-96（**md5**）または HMAC-SHA-96（**sha**）のどちらかを指定でき、パスワードストリング *auth-password*（64 文字以下）が必要です。 **v3** を入力すると、次のキーワードを使用して（64 文字以内）、プライベート（**priv**）暗号化アルゴリズムおよびパスワード文字列 *priv-password* を設定することもできます。 • **priv**：User-based Security Model（USM）を指定します。 • **des**：56 ビット DES アルゴリズムの使用を指定します。 • **3des**：168 ビット DES アルゴリズムの使用を指定します。 • **aes**：DES アルゴリズムの使用を指定します。128 ビット暗号化、192 ビット暗号化、または 256 ビット暗号化のいずれかを選択する必要があります。

	コマンドまたはアクション	目的
		（任意）**access** *access-list* とともに、アクセスリスト名を表す文字列（64 文字以内）を入力します。 （注） コンプライアンスシールドが無効になっている場合、**md5**、**des**、**3des** のアルゴリズムは SNMPv3 グループでサポートされません。**crypto engine compliance shield enable** コマンドを使用してコンプライアンスシールドを有効にし、デバイスを再起動して、**md5**、**des**、および **3des** のアルゴリズムを設定する必要があります。
ステップ 6	**end** 例： Device(config)# **end**	特権 EXEC モードに戻ります。
ステップ 7	**show running-config** 例： Device# **show running-config**	入力を確認します。
ステップ 8	**copy running-config startup-config** 例： Device# **copy running-config startup-config**	（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP UDP ポートの開閉

SNMP UDP ポートを開くには、ユーザー EXEC モードで次の手順を実行します。

Procedure

ステップ 1 **enable**

Example:

```
Device> enable
```

特権 EXEC モードを有効にします。

パスワードを入力します（要求された場合）。

ステップ 2 **configure terminal**

Example:

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

ステップ 3 **snmp-server** {**host** | **user** | **community** | **manager**}

Example:

```
Device(config)# snmp-server host
```

SNMP UDP ポート 161 および 162 を開きます。

オプション（**host**、**user**、**community**、**manager**）のいずれかを設定すると、両方のポートが開きます。

ポートを閉じるには、設定したすべてのオプションの **no** 形式を入力します。キーワードが 1 つでも設定されていると、ポートは開いたままになります。

キーワードを指定せずに **no snmp-server** コマンドを入力すると、SNMP UDP ポートだけでなく、SNMP プロセスもシャットダウンされます。

ステップ 4 **end**

Example:

```
Device(config)# end
```

特権 EXEC モードに戻ります。

ステップ 5 **show udp**

Example:

```
Device# show udp
```

SNMP UDP ポートを表示します。

必要なコマンドのいずれかが設定されている場合、ポート 161 および 162 は、リモートフィールドの下に値 listen を表示します。

ステップ 6 **copy running-config startup-config**

Example:

```
Device# copy running-config startup-config
```

（任意）コンフィギュレーション ファイルに設定を保存します。

エージェント コンタクトおよびロケーションの設定

SNMP エージェントのシステム接点およびロケーションを設定して、コンフィギュレーション ファイルからこれらの記述にアクセスできるようにするには、次の手順を実行します。

手順

ステップ 1 **enable**

例：

```
Device> enable
```

特権 EXEC モードを有効にします。

パスワードを入力します（要求された場合）。

ステップ 2 **configure terminal**

例：

configure terminal

例：

```
Device# configure terminal
```

グローバル設定モードを開始します。

ステップ 3 **snmp-server contact** *text*

例：

```
Device(config)# snmp-server contact Dial System Operator at beeper 21555
```

システムの連絡先文字列を設定します。

ステップ 4 **snmp-server location** *text*

例：

```
Device(config)# snmp-server location Building 3/Room 222
```

システムの場所を表す文字列を設定します。

ステップ 5 **end**

例：

```
Device(config)# end
```

特権 EXEC モードに戻ります。

ステップ 6 **show running-config**

例：

```
Device# show running-config
```

入力を確認します。

ステップ 7 **copy running-config startup-config**

例：

```
Device# copy running-config startup-config
```

（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP を通して使用する TFTP サーバの制限

SNMP を介したコンフィギュレーション ファイルの保存とロードに使用する TFTP サーバを、アクセス リストで指定されたサーバに限定するには、次の手順を実行します。

手順

ステップ 1 **enable**

例：

```
Device> enable
```

特権 EXEC モードを有効にします。

パスワードを入力します（要求された場合）。

ステップ 2 **configure terminal**

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

ステップ 3 **snmp-server tftp-server-list** *access-list-number*

例：

```
Device(config)# snmp-server tftp-server-list 44
```

SNMP を介したコンフィギュレーション ファイルのコピーに使用する TFTP サーバを、アクセス リストのサーバに限定します。

access-list-number には、1 ～ 99 および 1300 ～ 1999 の標準 IP アクセス リスト番号を入力します。

ステップ 4 **access-list** *access-list-number* { deny | permit} *source* [*source-wildcard*]

例：

```
Device(config)# access-list 44 permit 10.1.1.2
```

標準アクセス リストを作成し、コマンドを必要な回数だけ実行します。

- *access-list-number* には、ステップ 3 で指定したアクセス リスト番号を入力します。
- **deny** キーワードは、条件が一致した場合にアクセスを拒否します。**permit** キーワードは、条件が一致した場合にアクセスを許可します。
- *source* には、デバイスにアクセスできる TFTP サーバの IP アドレスを入力します。

- （任意）*source-wildcard* には、source に適用されるワイルドカード ビットをドット付き 10 進表記で入力します。無視するビット位置には 1 を設定します。

  アクセス リストの末尾には、すべてに対する暗黙の拒否ステートメントが常に存在します。

ステップ 5 end

例：

```
Device(config)# end
```

特権 EXEC モードに戻ります。

ステップ 6 show running-config

例：

```
Device# show running-config
```

入力を確認します。

ステップ 7 copy running-config startup-config

例：

```
Device# copy running-config startup-config
```

（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP エージェントのディセーブル化

SNMP エージェントをディセーブルにするには、次の手順を実行します。

Before you begin

SNMP エージェントをディセーブルにする前にイネーブルにする必要があります。デバイス上で入力した最初の **first snmp-server** グローバル コンフィギュレーション コマンドによって SNMP エージェントがイネーブルになります。

no snmp-server グローバル コンフィギュレーション コマンドは、デバイス上で実行している SNMP エージェントのすべてのバージョン（バージョン 1、バージョン 2C、バージョン 3）をディセーブルにして、SNMP プロセスをシャットダウンします。グローバル コンフィギュレーション モードで、**snmp-server host**、**snmp-server user**、**snmp-server community**、**nmp-server manager** のいずれかのコマンドを入力して、SNMP エージェントのすべてのバージョンを再度イネーブルにできます。特に SNMP をイネーブルにするために指定された Cisco IOS コマンドはありません。

手順

ステップ 1 enable

例：

```
Device> enable
```

特権 EXEC モードを有効にします。

パスワードを入力します（要求された場合）。

ステップ 2　configure terminal

例：

```
Device# configure terminal
```

グローバル コンフィギュレーション モードを開始します。

ステップ 3　no snmp-server

例：

```
Device(config)# no snmp-server
```

SNMP エージェント動作をディセーブルにします。

ステップ 4　end

例：

```
Device(config)# end
```

特権 EXEC モードに戻ります。

ステップ 5　show running-config

例：

```
Device# show running-config
```

入力を確認します。

ステップ 6　copy running-config startup-config

例：

```
Device# copy running-config startup-config
```

（任意）コンフィギュレーション ファイルに設定を保存します。

SNMP ステータスのモニタリング

不正なコミュニティ ストリング エントリ、エラー、要求変数の数など、SNMP の入出力統計情報を表示するには、**show snmp** 特権 EXEC コマンドを使用します。また、次の表にリストされたその他の特権 EXEC コマンドを使用して、SNMP 情報を表示することもできます。

Table 4: SNMP 情報を表示するためのコマンド

コマンド	目的
show snmp	SNMP 統計情報を表示します。
show snmp engineID	デバイスに設定されているローカルSNMPエンジンおよびすべてのリモート エンジンに関する情報を表示します。
show snmp group	ネットワーク上の各 SNMP グループに関する情報を表示します。
show snmp pending	保留中の SNMP 要求の情報を表示します。
show snmp sessions	現在の SNMP セッションの情報を表示します。
show snmp user	SNMP ユーザ テーブルの各 SNMP ユーザ名に関する情報を表示します。 **Note** このコマンドは、**auth** | **noauth** | **priv** モードの SNMPv3 設定情報を表示する際に使用する必要があります。この情報は、**show running-config** の出力には表示されません。

SNMP の例

次に、SNMP の全バージョンをイネーブルにする例を示します。この設定では、任意の SNMP マネージャがコミュニティ ストリング *public* を使用して、読み取り専用権限ですべてのオブジェクトにアクセスできます。この設定では、デバイスはトラップを送信しません。

```
Device(config)# snmp-server community public
```

次に、任意の SNMP マネージャがコミュニティストリング *public* を使用して、読み取り専用権限ですべてのオブジェクトにアクセスする例を示します。デバイスはさらに、SNMPv1 を使用してホスト 192.180.1.111 および 192.180.1.33 に、SNMPv2C を使用してホスト 192.180.1.27 に VTP トラップを送信します。コミュニティ ストリング *public* は、トラップとともに送信されます。

```
Device(config)# snmp-server community public
Device(config)# snmp-server enable traps vtp
Device(config)# snmp-server host 192.180.1.27 version 2c public
Device(config)# snmp-server host 192.180.1.111 version 1 public
Device(config)# snmp-server host 192.180.1.33 public
```

次に、*comaccess* コミュニティ ストリングを使用するアクセス リスト 4 のメンバに、すべてのオブジェクトへの読み取り専用アクセスを許可する例を示します。その他の SNMP マネージャは、どのオブジェクトにもアクセスできません。SNMP 認証障害トラップは、SNMPv2C がコミュニティ ストリング *public* を使用してホスト *cisco.com* に送信します。

```
Device(config)# snmp-server community comaccess ro 4
Device(config)# snmp-server enable traps snmp authentication
Device(config)# snmp-server host cisco.com version 2c public
```

次に、エンティティ MIB トラップをホスト *cisco.com* に送信する例を示します。コミュニティストリングは制限されます。1 行目で、デバイスはすでにイネーブルになっているトラップ以外に、エンティティ MIB トラップを送信できるようになります。2 行目はこれらのトラップの宛先を指定し、ホスト *cisco.com* に対する以前の **snmp-server** ホストコマンドを無効にします。

```
Device(config)# snmp-server enable traps entity
Device(config)# snmp-server host cisco.com restricted entity
```

次に、コミュニティストリング public を使用して、すべてのトラップをホスト myhost.cisco.com に送信するようにデバイスをイネーブルにする例を示します。

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host myhost.cisco.com public
```

次に、ユーザとリモート ホストを関連付けて、ユーザがグローバル コンフィギュレーション モードの際に auth（authNoPriv）認証レベルで情報を送信する例を示します。

```
Device(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Device(config)# snmp-server group authgroup v3 auth
Device(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5
mypassword
Device(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Device(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Device(config)# snmp-server enable traps
Device(config)# snmp-server inform retries 0
```

次に、SNMP エージェントにポーリングされた SNMP マネージャのエントリを表示する例を示します。

```
Device# show snmp stats host
Request Count Last Timestamp Address
2 00:00:01 ago 3.3.3.3
1 1w2d ago 2.2.2.2
```

次の例は、コンプライアンスシールドが無効になっている場合に SNMPv3 グループで 3 つのアルゴリズム（**md5**、**des**、**3des**）のいずれかを設定したときにデバイスに表示されるメッセージを示しています。

```
Device(config)# snmp-server user md5user grp v3 auth md5 cisco1234 priv des
Sep 1 00:14:51.582 IST: %SNMP-6-AUTHPROTOCOLMD5: Authentication protocol md5 support
will be deprecated in future
Sep 1 00:14:51.582 IST: %SNMP-6-PRIVPROTOCOLDES: Privacy protocol des support will be
deprecated in future
Sep 1 00:14:51.645 IST: %SNMP-5-WARMSTART: SNMP agent on host Switch is undergoing a
warm start
```

次の例は、コンプライアンスシールドが有効になっている場合に SNMPv3 グループで 3 つのアルゴリズム（**md5**、**des**、**3des**）のいずれかを設定したときにデバイスに表示されるメッセージを示しています。以下に示すように、暗号化アルゴリズムは警告メッセージとともにサポートされています。

```
Device(config)# snmp-server user md5user grp v3 auth md5 cisco1234
weaker algorithm MD5, DES and 3DES is not allowed for snmp user
```

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。