

Simple Network Management Protocol (SNMP; 簡易ネットワーク管理プロトコル) トラップについて

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[SNMP トラップの使用](#)

[Cisco IOS から送信されたトラップの例](#)

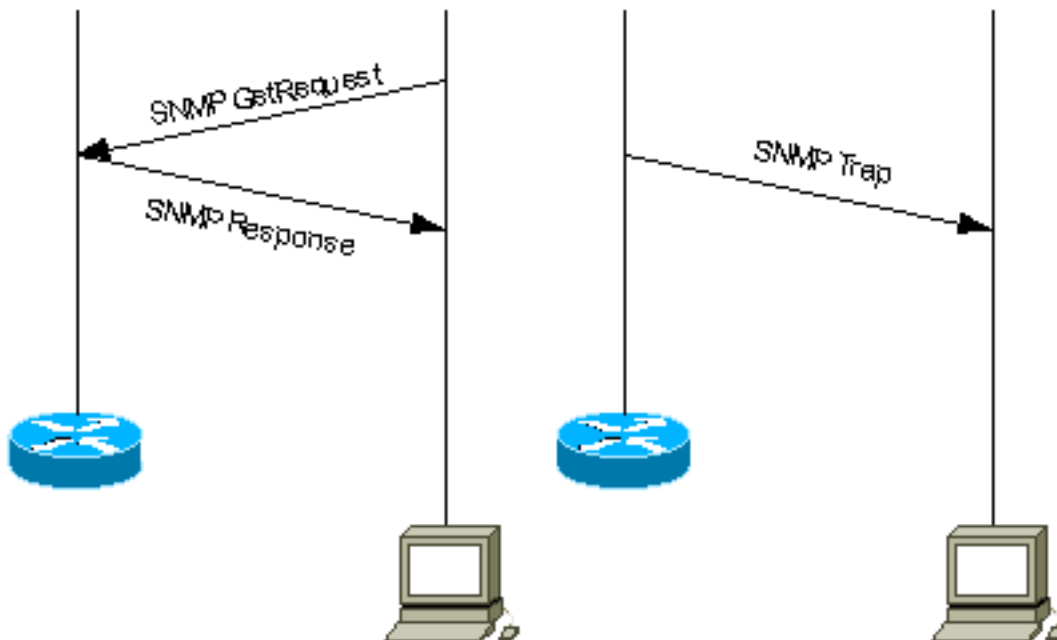
[関連情報](#)

概要

このドキュメントでは、SNMP トラップの概要について説明します。SNMP トラップがどのように使用されるか、また、データ ネットワークの管理においてどのような役割を果たすかについて説明します。

SNMP トラップを使用すると、エージェントは非送信請求 SNMP メッセージを使用して管理ステーションに重要なイベントを通知できます。

次の図において、左側の設定では、ネットワーク管理システムが情報をポーリングし、応答を取得します。右側の設定では、エージェントは非送信請求 (非同期) トラップをネットワーク管理システム (NMS) に送信します。



前提条件

要件

このドキュメントに特有の要件はありません。

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

表記法

ドキュメント表記の詳細は、『[シスコテクニカルティップスの表記法](#)』を参照してください。

SNMP トラップの使用

SNMP (Simple Network Management Protocol) v1 および SNMPv2c では、関連付けられた Management Information Base (MIB) とともに、トラップで指示される通知の使用が奨励されます。

トラップで指示される通知には、以下の狙いがあります。マネージャが多数のデバイスを管理する必要があり、各デバイスに多数のオブジェクトがある場合に、すべてのデバイスのすべてのオブジェクトに情報をポーリングまたは要求することは非現実的です。ソリューションは、送信要求を行わずに、管理対象デバイス上のエージェントごとにマネージャに通知することです。イベントのトラップと呼ばれるメッセージを送信することで、この処理を行います。

イベントの受信後、マネージャはイベントを表示し、イベントに基づくアクションの実行を選択できます。たとえば、マネージャは、エージェントを直接ポーリングするか、他の関連デバイスエージェントをポーリングしてイベントの詳細情報を取得できます。

トラップによる通知を使用すると、それほど重要ではない SNMP 要求を削減できるので、ネット

ワークとエージェントのリソースを大幅に節約できます。ただし、SNMP ポーリングを完全には排除できません。SNMP 要求は、検出とトポロジ変更が必要です。また、管理対象デバイス エージェントは、デバイスに致命的な停止が生じた場合にはトラップを送信できません。

SNMPv1 トラップは RFC 1157 で定義されており、次のフィールドが有効です。

- *Enterprise* : トラップを生成する管理対象オブジェクトのタイプを識別します。
- *Agent address* : トラップを生成する管理対象オブジェクトのアドレスを示します。
- *Generic trap type* : 標準トラップ タイプの番号の 1 つを示しています。
- *Specific trap code* : 固有のトラップ コードの番号の 1 つを示しています。
- *Time stamp* : 最後のネットワーク再初期化からトラップの生成までの経過時間を示します。
- *Variable bindings* : PDU を含むトラップのデータ フィールド。個々の変数バインドでは、特定の MIB オブジェクト インスタンスとその現在の値が関連付けられます。

標準的な汎用トラップは、coldStart、warmStart、linkDown、linkUp、authenticationFailure、および egpNeighborLoss です。標準 SNMPv1 トラップでは、*Enterprise* フィールドには、トラップを送信するデバイスの [sysObjectID](#) 値が入っています。ベンダー固有のトラップでは、*Generic trap type* フィールドは *enterpriseSpecific(6)* に設定されています。シスコ固有のトラップは、通常とは異なる方法で実装されています。トラップの *Enterprise* フィールドに [sysObjectID](#) を固定して、*Specific trap code* ですべての Cisco デバイスでサポートされているすべての固有トラップを識別する代わりに、Cisco ではさまざまなトラップの *Enterprise* フィールドと *Specific trap code* フィールドを使用してトラップの識別を実装しています。[実際の値は、SNMP オブジェクトナビゲータで参照できます。](#) さらに Cisco では、標準トラップの再定義を [CISCO-GENERAL-TRAPS MIB](#) で行い、さらにバインド変数を追加しています。これらのトラップでは、*Generic trap type* は変更せず、*enterpriseSpecific(6)* に設定していません。

SNMPv2c トラップは NOTIFICATION として定義され、SNMPv1 とは形式が異なります。次のパラメータがあります。

- *sysUpTime* : これは、SNMPv1 トラップの Time スタンプと同じです。
- [snmpTrapOID](#) – トラップ識別フィールド。汎用トラップでは、この値は RFC 1907 で定義されています。ベンダー固有トラップでは、*snmpTrapOID* は基本的に *SNMPv1* の *Enterprise* パラメータに、2 つのサブ識別子である「0」と *SNMPv1 Specific trap code* パラメータとを結合した値となります。
- *VarBindList* : 変数バインドのリストです。

エージェントから送信されたトラップが管理システムで理解されるには、オブジェクト識別子 (OID) で定義されている内容が、管理システムで把握されている必要があります。そのためには、そのトラップの MIB をロードしておく必要があります。こうすることで、正しい OID 情報が提供され、ネットワーク管理システムに送信されたトラップが理解されます。

[特定の MIB で Cisco のデバイスによりサポートされるトラップについては](#)、Cisco SNMP オブジェクトナビゲータを参照してください。ここでは、特定の MIB で使用可能なトラップがリストされています。このいずれかのトラップを受信するには、リストされている MIB が、使用している Cisco IOS® ソフトウェア リリースでサポートされている必要があります。使用している Cisco デバイスでサポートされている MIB を確認するには、www.cisco.com/go/mibs をアクセスしてください。該当の MIB を、ネットワーク管理システムにロードする必要があります。通常この作業は、コンパイルと呼ばれます。使用している NMS プラットフォームでの MIB のコンパイルについては、使用しているネットワーク管理システム (HP OpenView、NetView など) のユーザガイドを参照してください。また、『[SNMP : MIB に関する FAQ](#)』および『[MIB コンパイラと MIB のロード](#)』も参照してください。

さらに、ネットワーク管理システムへトラップを送信するようにデバイスを設定しなければ、ト

ラップは送信されません。トラップを送信する必要があることを、デバイスに正しく設定しておく必要があります。トラップの宛先は、通常 IP アドレスで定義しますが、ドメイン ネーム システム (DNS) サーバに問い合わせるようにデバイスが設定されている場合は、ホスト名を指定することもできます。Cisco IOS ソフトウェアの最近のバージョンでは、送信するトラップをデバイスの管理者が選択できるようになっています。シスコ デバイスを SNMP 用に設定する方法、およびトラップの送信方法についての情報は、関連デバイスの構成ガイドと、[基本ダイヤル NMS 実装ガイド](#)、『[サポート対象の Cisco IOS SNMP トラップとその設定方法](#)』、および『[Cisco Catalyst OS SNMP トラップのサポートおよび設定方法](#)』を参照してください。

注：マネージャは通常、UDPポート番号162でSNMP通知 (TRAPおよびINFORM) を受信します。

Cisco IOS から送信されたトラップの例

ここでは、Cisco IOS から送信され、`debug snmp packet` によって取得されるトラップの例をいくつか紹介します。

シスコにより再定義された SNMPv1 汎用トラップ：

```
Nov 21 07:44:17: %LINK-3-UPDOWN: Interface Loopback1, changed state to up
4d23h: SNMP: Queuing packet to 172.17.246.162
4d23h: SNMP: V1 Trap, ent products.45, addr 172.17.246.9, gentrap 3, spectrap 0
  ifEntry.1.23 = 23
  ifEntry.2.23 = Loopback1
  ifEntry.3.23 = 24
  lifEntry.20.23 = up
```

この出力は、[CISCO-GENERAL-TRAPS MIB](#) からの `linkUp` トラップを、シスコが 4 つのバインド変数によって再定義したものです。以下のフィールドから成ります。

- *Enterprise* = products.45(この例では、デバイスがトラップを送信する際のsysObjectID。これはc7507ルータです)
- *Generic trap type* = 3 (リンクアップ)
- *Specific trap code* = 0

シスコ固有の SNMPv1 トラップ：

```
4d23h: SNMP: Queuing packet to 172.17.246.162
4d23h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.17.246.9, gentrap 6, spectrap 1
  clogHistoryEntry.2.954 = LINK
  clogHistoryEntry.3.954 = 4
  clogHistoryEntry.4.954 = UPDOWN
  clogHistoryEntry.5.954 = Interface Loopback1, changed state to up
  clogHistoryEntry.6.954 = 43021184
```

この出力では、5 つのバインド変数を伴う [CISCO-SYSLOG-MIB](#) からの Cisco 固有の `clogMessageGenerated` トラップが示されています。以下のフィールドから成ります。

- *Enterprise* = `clogMessageGenerated` トラップの *Enterprise* 値
- *Generic trap type* = 6 (企業固有)
- *Specific trap code* = 1 (`clogMessageGenerated` 固有のトラップコード)

シスコ固有の SNMPv2c トラップ：

```
4d23h: SNMP: Queuing packet to 172.17.246.162
4d23h: SNMP: V2 Trap, reqid 2, errstat 0, erridx 0
sysUpTime.0 = 43053404
snmpTrapOID.0 =
clogHistoryEntry.2.958 = SYS
clogHistoryEntry.3.958 = 6
clogHistoryEntry.4.958 = CONFIG_I
clogHistoryEntry.5.958 = Configured from console by vty0 (10.10.10.10)
clogHistoryEntry.6.958 = 43053403
```

この出力は、次の3つのバインド変数を持つ[CISCO-CONFIG-MAN-MIB](#)からのCisco固有の[ciscoConfigManEvent SNMPv2c](#)通知を示しています。

- [ccmHistoryEventCommandSource](#)
- [ccmHistoryEventConfigSource](#)
- [ccmHistoryEventConfigDestination](#)

このトラップは、デバイスの設定に何らかの変更が加えられた場合に使用できます。最後の2つのコンポーネントの値は、`show` コマンドが実行されたかどうか、または設定が修正されたかどうかを示します。

```
6506E#term mon
6506E#debug snmp packet
SNMP packet debugging is on
```

```
6506E#sh run
Building configuration...
```

```
...
6506E#
19:24:18: SNMP: Queuing packet to 10.198.28.80
19:24:18: SNMP: V2 Trap, reqid 2, errstat 0, erridx 0
sysUpTime.0 = 6981747
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.100 = 1
!--- 1 -> commandLine. Executed via CLI. ccmHistoryEventEntry.4.100 = 3 !--- 3 -> running
ccmHistoryEventEntry.5.100 = 2 !--- 2 -> commandSource. Show command was executed.
```

```
6506E#term mon
6506E#debug snmp packet
SNMP packet debugging is on
```

```
6506E#conf t
Enter configuration commands, one per line. End with CNTL/Z.
6506E(config)#exit
```

```
22:57:37: SNMP: Queuing packet to 10.198.28.80
22:57:37: SNMP: V2 Trap, reqid 2, errstat 0, erridx 0
sysUpTime.0 = 8261709
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.108 = 1
!--- 1 -> commandLine. Executed via CLI. ccmHistoryEventEntry.4.108 = 2 !--- 2 -> commandSource
ccmHistoryEventEntry.5.108 = 3 !--- 3 -> running. Change was destined to the running
configuration.
```

[関連情報](#)

- [テクニカル サポートとドキュメント – Cisco Systems](#)