

瞭解簡易網路管理通訊協定 (SNMP) 陷阱

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[慣例](#)

[使用SNMP陷阱](#)

[Cisco IOS傳送的陷阱示例](#)

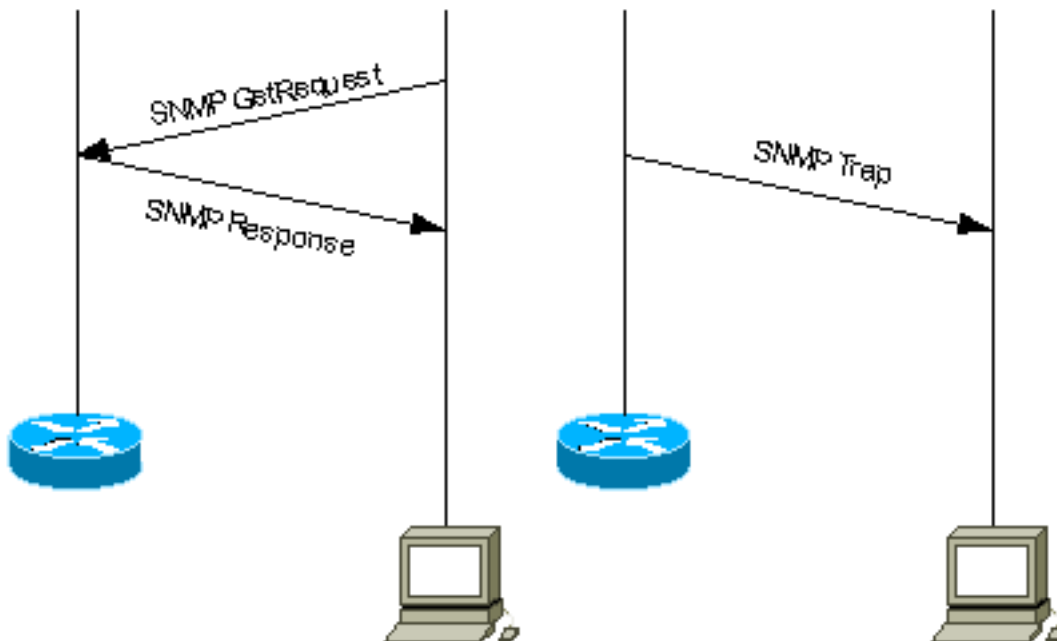
[相關資訊](#)

簡介

本文件提供 SNMP 陷阱的簡介，其中說明 SNMP 陷阱的使用方式，以及 SNMP 陷阱在資料網路管理中扮演的角色。

SNMP陷阱使代理能夠通過未經請求的SNMP消息將重大事件通知管理站。

在此圖中，左側的設定顯示了一個網路管理系統，用於輪詢資訊並獲得響應。右側的設定顯示一個代理，該代理將未經請求的陷阱或非同步陷阱傳送到網路管理系統(NMS)。



必要條件

需求

本文件沒有特定需求。

採用元件

本文件所述內容不限於特定軟體和硬體版本。

慣例

如需文件慣例的詳細資訊，請參閱[思科技術提示慣例](#)。

使用SNMP陷阱

SNMPv1 (簡單網路管理協定) 和SNMPv2c，以及相關的管理資訊庫(MIB)，鼓勵陷阱定向通知。

陷阱導向通知的思想是，如果管理員負責大量裝置，並且每台裝置都有大量對象，則管理員從每台裝置上的每個對象輪詢或請求資訊是不切實際的。解決方案是讓受管裝置上的每個代理在不請求的情況下通知管理器。它通過傳送稱為事件陷阱的消息來實現這一點。

在管理器收到事件後，它會顯示它，並且可以選擇根據事件採取措施。例如，管理器可以直接輪詢代理，或輪詢其他關聯的裝置代理，以便更好地瞭解事件。

陷阱導向型通知通過消除無用的SNMP請求，可顯著節省網路和代理資源。但是，無法完全消除SNMP輪詢。發現和拓撲更改需要SNMP請求。此外，如果裝置出現災難性中斷，受管裝置代理無法傳送陷阱。

RFC 1157中定義了SNMPv1陷阱，包括以下欄位：

- *Enterprise* — 標識生成陷阱的受管對象的型別。
- *Agent address* — 提供生成陷阱的受管對象的地址。
- *通用陷阱類型(Generic trap type)* — 表示多種通用陷阱型別之一。
- *特定陷阱代碼(Specific trap code)* — 表示許多特定陷阱代碼之一。
- *Time stamp* — 提供從上次網路重新初始化到生成陷阱之間經過的時間。
- *變數綁定* — 包含PDU的陷阱的資料欄位。每個變數繫結將特定MIB對象例項與其當前值相關聯。

標準通用陷阱為：coldStart、warmStart、linkDown、linkUp、authenticationFailure、egpNeighborLoss。對於通用SNMPv1陷阱，*Enterprise*欄位包含傳送陷阱的裝置的[sysObjectID](#)。對於供應商特定的陷阱，*一般陷阱類型*欄位設定為enterpriseSpecific(6)。思科以非傳統方式實施其自己的特定陷阱。Cisco沒有將*Enterprise*陷阱欄位仍保留為[sysObjectID](#)，也沒有使用*Specific trap code*來標識所有Cisco裝置支援的所有特定陷阱，而是使用各種陷阱Enterprise和Specific trap code欄位實施陷阱標識。您可以從SNMP物件導覽器看到[實際值](#)。此外，Cisco通過新增更多繫結變數，重新定義了[CISCO-GENERAL-TRAPS MIB中的某些通用陷阱](#)。對於這些陷阱，*Generic trap type*保持相同，且不設定為enterpriseSpecific(6)。

在SNMPv2c中，陷阱定義為NOTIFICATION，其格式與SNMPv1不同。它具有以下引數：

- *sysUpTime* — 這與SNMPv1陷阱中的時間戳相同。
- [snmpTrapOID](#) — 陷阱標識欄位。對於一般陷阱，值在RFC 1907中定義，對於供應商特定陷阱*snmpTrapOID*，基本上是SNMPv1 *Enterprise*引數和兩個附加子識別符號「0」和SNMPv1 *Specific trap code*引數的串聯。

- *VarBindList* — 這是變數繫結的清單。

為了讓管理系統瞭解代理向其傳送的陷阱，管理系統必須知道對象識別符號(OID)所定義的內容。因此，它必須載入該陷阱的MIB。這樣可以提供正確的OID資訊，以便網路管理系統能夠瞭解傳送到它的陷阱。

有關特定MIB中Cisco裝置支援的陷阱，請參閱[Cisco SNMP Object Navigator](#)。這將列出特定MIB可用的陷阱。為了接收這些陷阱之一，您的Cisco IOS®軟體版本必須支援列出的MIB。要瞭解您的Cisco裝置支援哪些MIB，請訪問www.cisco.com/go/mibs。必須將MIB載入到網路管理系統中。這通常稱為編譯。有關NMS平台上的MIB編譯的資訊，請參閱《網路管理系統》(例如，HP OpenView或NetView)使用手冊。另請參閱[SNMP:有關MIB和MIB編譯器](#)以及[載入MIB的常見問題](#)。

此外，裝置不會向網路管理系統傳送陷阱，除非其配置為這樣做。裝置必須知道它應該傳送陷阱。陷阱目標通常由IP地址定義，但如果裝置設定為查詢域名系統(DNS)伺服器，則它可以是主機名。在更高版本的Cisco IOS軟體中，裝置管理員可以選擇他們希望傳送哪些陷阱。有關如何為SNMP配置Cisco裝置以及如何傳送陷阱的資訊，請參閱相應裝置配置指南和[基本撥號NMS實施指南](#)、[Cisco IOS SNMP支援陷阱](#)以及如何配置它們以及[如何支援和配置Cisco CatalystOS SNMP陷阱](#)。

注意：管理器通常在UDP埠號162上接收SNMP通知 (TRAP和INFORM)。

Cisco IOS傳送的陷阱示例

本節包含由Cisco IOS傳送的陷阱的一些示例，與`debug snmp packet`一起使用。

SNMPv1通用陷阱，由Cisco重新定義：

```
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
```

此輸出顯示[CISCO-GENERAL-TRAPS](#) MIB的Cisco重新定義的[linkUp陷阱](#)，其中包含四個繫結變數。它具有以下欄位：

- *Enterprise* = products.45([sysObjectID](#) of the device sending trap, 在本例中為c7507路由器)
- *通用陷阱類型* = 3(linkUp)
- *特定陷阱代碼* = 0

SNMPv1 Cisco特定陷阱：

```
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
```

此輸出顯示了來自[CISCO-SYSLOG-MIB](#)的Cisco特定的[clogMessageGenerated陷阱](#)，其中包含五個繫結變數。它具有以下欄位：

- *Enterprise* = clogMessageGenerated陷阱的企業值
- 通用陷阱類型= 6 (企業特定)
- 特定陷阱代碼= 1 (clogMessageGenerated的特定陷阱代碼)

SNMPv2c Cisco特定陷阱：

```
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
```

此輸出顯示來自[CISCO-CONFIG-MAN-MIB](#)的Cisco特定的[ciscoConfigManEvent](#) SNMPv2c通知，包含三個繫結變數：

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

如果對裝置配置進行了任何更改，則可以使用此陷阱。最後兩個元件的值會判斷是否已發出show命令，或是已接觸組態。

```
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.
```

[相關資訊](#)

- [技術支援與文件 - Cisco Systems](#)