

瞭解EEM的最佳實踐和有用的指令碼

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[慣例](#)

[最佳實務](#)

[確認適當的驗證已就緒](#)

[新增EEM執行階段和速率限制的限](#)

[避免不按順序執行](#)

[停用分頁](#)

[針對未來可維護性的設計指令碼](#)

[常用EEM邏輯模式](#)

[含有If/Else的分支程式碼路徑](#)

[Loop Over語句](#)

[透過規則運算式\(Regex\)擷取輸出](#)

[實用的EEM指令碼](#)

[跟蹤MAC地址學習的特定MAC地址](#)

[透過SNMP OID監控CPU使用率高](#)

[動態匹配PID並記錄堆疊輸出](#)

[升級交換機](#)

[當IP SLA跟蹤的對象斷開時，將診斷資料轉儲到檔案](#)

[從EEM傳送電子郵件](#)

[依排程關閉連線埠](#)

[達到指定的每秒封包\(PPS\)速率時關閉介面](#)

[相關資訊](#)

簡介

本文檔介紹Cisco IOS® XE裝置上的嵌入式事件管理器(EEM)指令碼配置最佳實踐。

必要條件

需求

思科建議您瞭解並熟悉以下主題：

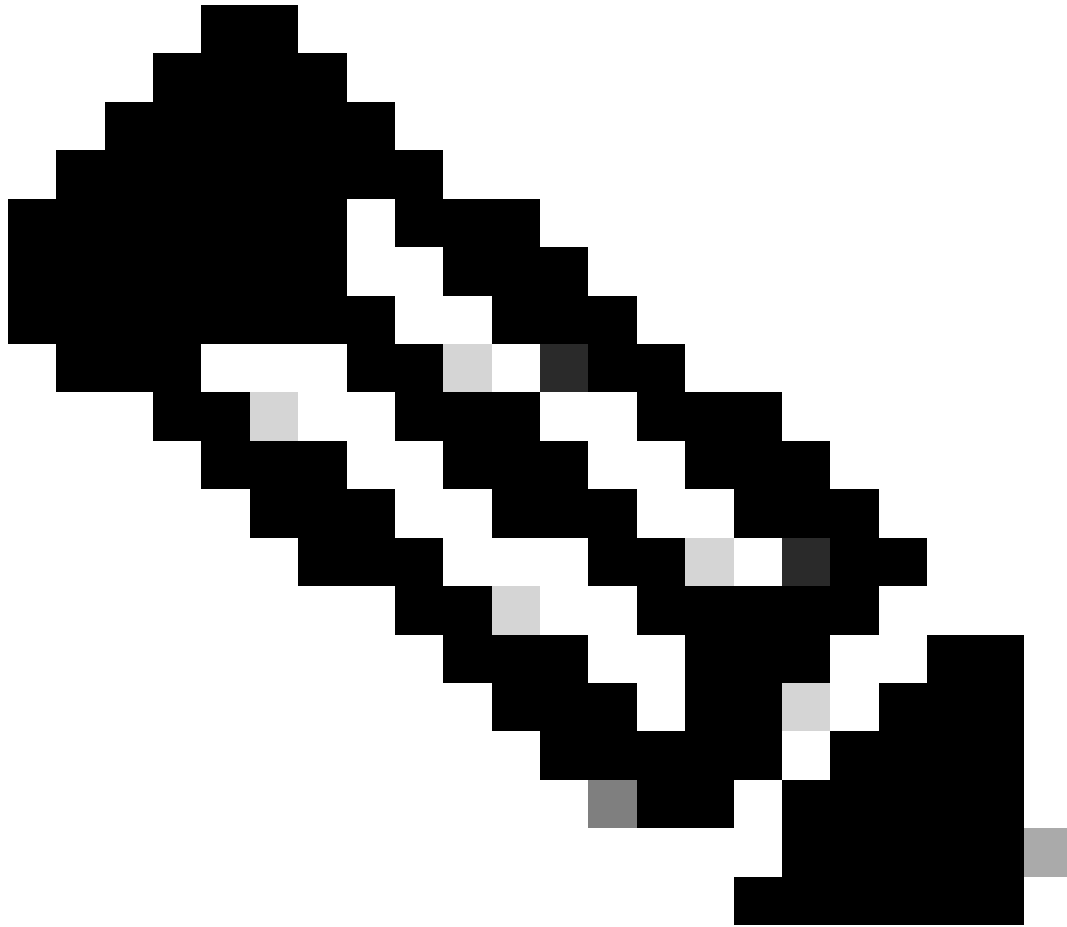
- Cisco IOS和Cisco IOS XE內嵌式事件管理員(EEM)

如果您尚不熟悉此功能，請首先閱讀[EEM功能概述](#)。

採用元件

本文中的資訊係根據以下軟體和硬體版本：

- Cisco Catalyst 9300、9400和9500交換機
 - Cisco IOS軟體版本16.X或17.X
-



注意：這些指令碼不受Cisco TAC支援，僅按實際提供用於教育目的。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

慣例

有關文檔規則的資訊，請參閱[Cisco技術提示規則](#)。

最佳實務

本節介紹設計和實施EEM指令碼時觀察到的一些最常見問題。有關EEM最佳實踐的更多資訊，請參閱參考部分下參考的EEM最佳實踐文檔。

確認適當的驗證已就緒

如果您的裝置使用AAA，則必須確保裝置上配置的EEM指令碼是使用能夠運行指令碼中命令的AAA使用者配置的，或者使用指令碼定義中的authorization bypass 命令配置了授權繞行。

新增EEM執行階段和速率限制的制限

預設情況下，EEM指令碼最多可以運行20秒。如果您設計的指令碼需要較長的時間才能運行，或者必須在兩次命令執行之間等待，請在applet事件觸發器上指定maxrun值以更改預設執行計時器。

此外，還必須考慮觸發EEM指令碼的事件可以運行的頻率。如果觸發指令碼的條件是在短時間內快速出現的情況（例如，MAC抖動的系統日誌觸發器），則在EEM指令碼中包含速率限制條件非常重要，這樣可以防止並行執行的次數過多，並防止裝置資源耗盡。

避免不按順序執行

如EEM文檔中所述，操作語句的執行順序由其標籤控制(例如，操作0001 cli命令enable的標籤為0001)。此標籤值不是數字，而是字母數字。操作按升序字母數字鍵序列排序，使用label引數作為排序鍵，且按此順序運行。根據您構造動作標籤的方式，這可能導致意外的執行順序。

請考慮以下示例：

```
event manager applet test authorization bypass
event timer watchdog time 60 maxrun 60
action 13 syslog msg "You would expect to see this message first"
action 120 syslog msg "This message prints first"
```

由於字母數字比較中的120在13之前，因此此指令碼不會按預期順序運行。若要避免這種情況，請使用下列填補系統：

```
event manager applet test authorization bypass
event timer watchdog time 60 maxrun 60
action 0010 syslog msg "This message appears first"
action 0020 syslog msg "This message appears second"
action 0120 syslog msg "This message appears third"
```

由於此處有填充，編號語句按預期順序計算。每個標籤之間的增量為10，允許稍後根據需要在EEM指令碼中插入其他語句，而無需為所有後續語句重新編號。

停用分頁

EEM查詢裝置提示以確定命令輸出完成的時間。如果命令輸出的資料多於一個螢幕上可以顯示的資料（按照終端長度配置），則可能會阻止EEM指令碼的完成（並最終透過maxrun計時器終止），因為在檢視輸出的所有頁面之前，不會顯示裝置提示。在檢查大型輸出的EEM指令碼的開始處配置術語len 0。

針對未來可維護性的設計指令碼

當您設計EEM指令集時，請保留動作標籤之間間隙，以便將來更輕鬆地更新EEM指令集邏輯。如果有適當的間隙(即，兩個語句(如操作0010和操作0020留有9個可插入的標籤的間隙)，可以根據需要增加新語句，而無需重新編號或重新檢查操作標籤，並確保繼續按預期順序執行操作。

您需要在EEM指令碼的開頭運行一些常用命令。這可能包括：

- 將terminal length設定為0
- 進入啟用模式
- 啟用命令輸出的自動時間戳

這是本文檔中顯示的示例中的常見模式，其中許多指令碼都以相同的3個action語句開始配置此模式。

常用EEM邏輯模式

本節介紹EEM指令碼中使用的一些常見邏輯模式和語法塊。這裡的示例不是完整的指令碼，而是說明如何使用特定功能建立複雜的EEM指令碼的演示。

含有If/Else的分支程式碼路徑

EEM變數可用於控制EEM指令碼的執行流。請考慮此EEM指令碼：

```
event manager applet snmp_cpu authorization bypass
event timer watchdog time 60
action 0010 info type snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.3 get-type exact
action 0020 if $_info_snmp_value ge "50"
action 0030 syslog msg "This syslog message is sent if CPU utilization is above 50%"
action 0040 elseif $_info_snmp_value ge "30"
action 0050 syslog msg "This syslog message is sent if CPU utilization is above 30% and below 50%"
action 0060 else
action 0070 syslog msg "This syslog message is sent if CPU utilization is below 30%"
action 0080 end
```

此指令碼每分鐘運行一次。檢查SNMP OID的CPU使用率值，然後根據OID值輸入三個不同執行路徑之一。類似的語句可用於任何其他合法EEM變數，以在EEM指令碼中構建複雜的執行流。

Loop Over語句

執行回圈可用來大幅縮短EEM指令集，並使它們更容易推理。請考慮以下指令碼，該指令碼旨在將Te2/1/15的介面統計資訊在1分鐘內拉出6次，以檢查是否存在使用率較高的小時間段：

```
event manager applet int_util_check auth bypass
event timer watchdog time 300 maxrun 120
action 0001 cli command "enable"
action 0002 cli command "term exec prompt timestamp"
action 0003 cli command "term length 0"
action 0010 syslog msg "Running iteration 1 of command"
action 0020 cli command "show interface te2/1/15 | append flash:interface_util.txt"
action 0030 wait 10
action 0040 syslog msg "Running iteration 2 of command"
action 0050 cli command "show interface te2/1/15 | append flash:interface_util.txt"
action 0060 wait 10
action 0070 syslog msg "Running iteration 3 of command"
action 0080 cli command "show interface te2/1/15 | append flash:interface_util.txt"
action 0090 wait 10
action 0100 syslog msg "Running iteration 4 of command"
action 0110 cli command "show interface te2/1/15 | append flash:interface_util.txt"
action 0120 wait 10
action 0130 syslog msg "Running iteration 5 of command"
action 0140 cli command "show interface te2/1/15 | append flash:interface_util.txt"
action 0150 wait 10
action 0160 syslog msg "Running iteration 6 of command"
action 0170 cli command "show interface te2/1/15 | append flash:interface_util.txt"
```

使用EEM環路構造，可以顯著縮短此指令碼：

```
event manager applet int_util_check auth bypass
event timer watchdog time 300 maxrun 120
action 0001 cli command "enable"
action 0002 cli command "term exec prompt timestamp"
action 0003 cli command "term length 0"
action 0010 set loop_iteration 1
action 0020 while $loop_iteration le 6
action 0030 syslog msg "Running iteration $loop_iteration of command"
action 0040 cli command "show interface te2/1/15 | append flash:interface_util.txt"
action 0050 wait 10
action 0060 increment loop_iteration 1
action 0070 end
```

透過規則運算式(Regex)擷取輸出

EEM regexp語句可用於從命令輸出中提取值以用於後續命令，並在EEM指令碼本身中啟用動態命令建立。有關從show proc cpu的輸出中提取SNMP引擎PID的示例，請參閱此代碼塊 | i SNMP engine，並將其列印到syslog消息。此提取的值也可用於需要PID運行的其他命令中。

```
event manager applet check_pid auth bypass
event none
```

```
action 0010 cli command "show proc cpu | i SNMP ENGINE"
action 0020 regexp "^[ ]*([0-9]+) .*" $_cli_result match match1
action 0030 syslog msg "Found SNMP Engine PID $match1"
```

實用的EEM指令碼

跟蹤MAC地址學習的特定MAC地址

在本例中，跟蹤MAC地址b4e9.b0d3.6a41。該指令碼每30秒檢查一次，以檢視是否已在ARP或MAC表中獲知指定的MAC地址。如果看到MAC，指令碼將執行以下操作：

- 輸出系統日誌消息（當您要確認獲取MAC地址的位置或獲取時間/頻率時，此功能非常有用）。

實作

```
event manager applet mac_trace authorization bypass
event timer watchdog time 30
action 0001 cli command "enable"
action 0002 cli command "term exec prompt timestamp"
action 0003 cli command "term length 0"
action 0010 cli command "show ip arp | in b4e9.b0d3.6a41"
action 0020 regexp ".*(ARPA).*" $_cli_result
action 0030 if $_regexp_result eq 1
action 0040 syslog msg $_cli_result
action 0050 end
action 0060 cli command "show mac add vlan 1 | in b4e9.b0d3.6a41"
action 0070 regexp ".*(DYNAMIC).*" $_cli_result
action 0080 if $_regexp_result eq 1
action 0090 syslog msg $_cli_result
action 0100 end
```

透過SNMP OID監控CPU使用率高

此指令碼監控用於讀取過去5秒內CPU忙碌百分比的SNMP OID。當CPU忙碌超過80%時，指令碼會執行下列動作：

- 從show clock的輸出建立時間戳，然後使用此資訊建立唯一檔名
- 然後，關於進程和軟體狀態的輸出將寫入此檔案
- 嵌入式資料包捕獲(EPC)配置為捕獲發往控制平面的10秒流量並將其寫入檔案。
- 一旦完成EPC捕獲，EPC配置將被刪除，指令碼將退出。

實作

```
event manager applet high-cpu authorization bypass
event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.3 get-type next entry-op gt entry-val 80 poll-interval 1 rat
action 0001 cli command "enable"
```

```

action 0002 cli command "term exec prompt timestamp"
action 0003 cli command "term length 0"
action 0010 syslog msg "High CPU detected, gathering system information."
action 0020 cli command "show clock"
action 0030 regex "([0-9]|[0-9][0-9]):([0-9]|[0-9][0-9]):([0-9]|[0-9][0-9])" $_cli_result match match1
action 0040 string replace "$match" 2 2 "."
action 0050 string replace "$_string_result" 5 5 "."
action 0060 set time $_string_result
action 0070 cli command "show proc cpu sort | append flash:tac-cpu-$time.txt"
action 0080 cli command "show proc cpu hist | append flash:tac-cpu-$time.txt"
action 0090 cli command "show proc cpu platform sorted | append flash:tac-cpu-$time.txt"
action 0100 cli command "show interface | append flash:tac-cpu-$time.txt"
action 0110 cli command "show interface stats | append flash:tac-cpu-$time.txt"
action 0120 cli command "show log | append flash:tac-cpu-$time.txt"
action 0130 cli command "show ip traffic | append flash:tac-cpu-$time.txt"
action 0140 cli command "show users | append flash:tac-cpu-$time.txt"
action 0150 cli command "show platform software fed switch active punt cause summary | append flash:tac-cpu-$time.txt"
action 0160 cli command "show platform software fed switch active cpu-interface | append flash:tac-cpu-$time.txt"
action 0170 cli command "show platform software fed switch active punt cpuq all | append flash:tac-cpu-$time.txt"
action 0180 cli command "no monitor capture tac_cpu"
action 0190 cli command "monitor capture tac_cpu control-plane in match any file location flash:tac-cpu-$time.txt"
action 0200 cli command "monitor capture tac_cpu start" pattern "yes"
action 0210 cli command "yes"
action 0220 wait 10
action 0230 cli command "monitor capture tac_cpu stop"
action 0240 cli command "no monitor capture tac_cpu"

```

動態匹配PID並記錄堆疊輸出

此指令碼查詢SNMP輸入隊列已滿的系統日誌消息並採取以下操作：

- 將show proc cpu sort的輸出記錄到檔案中
- 透過regex提取SNMP引擎進程的PID
- 在後續命令中使用SNMP PID來獲取PID的堆疊資料
- 從配置中刪除指令碼，以便不再執行它

實作

```

event manager applet TAC-SNMP-INPUT-QUEUE-FULL authorization bypass
event syslog pattern "INPUT_QFULL_ERR" ratelimit 40 maxrun 120
action 0010 cli command "en"
action 0020 cli command "show proc cpu sort | append flash:TAC-SNMP.txt"
action 0030 cli command "show proc cpu | i SNMP ENGINE"
action 0040 regexp "^[ ]*([0-9]+) .*" $_cli_result match match1
action 0050 syslog msg "Found SNMP Engine PID $match1"
action 0060 cli command "show stacks $match1 | append flash:TAC-SNMP.txt"
action 0070 syslog msg "$_cli_result"
action 0080 cli command "configure terminal"
action 0090 cli command "no event manager applet TAC-SNMP-INPUT-QUEUE-FULL"
action 0100 cli command "end"

```

升級交換機

此指令碼配置為在install add file <file> activate commit命令返回的非標準提示符後進行模式匹配，並響應提示。由於未配置任何觸發事件，因此在需要透過event manager run UPGRADE進行升級時，必須由使用者手動觸發EEM指令碼。maxrun計時器設定為300秒而不是預設值20秒，因為install add命令需要大量時間運行。

實作

```
event manager applet UPGRADE authorization bypass
event none maxrun 300
action 0001 cli command "enable"
action 0002 cli command "term length 0"
action 0020 cli command "install add file flash:cat9k_iosxe.16.06.02.SPA.bin activate commit" pattern "
action 0030 cli command "y" pattern "y\n"
action 0040 syslog msg "Reloading device to upgrade code"
action 0050 cli command "y"
```

當IP SLA跟蹤的對象斷開時，將診斷資料轉儲到檔案

此指令碼在IP SLA對象11關閉並執行下列操作時觸發：

- 收集MAC表、ARP表、系統日誌和路由表
- 將資訊寫入快閃記憶體上的檔案：稱為sla_track.txt

實作

```
ip sla 10
icmp-echo 10.10.10.10 source-ip 10.10.10.10
frequency 10
exit
ip sla schedule 10 life forever start-time now
track 11 ip sla 10 reachability
exit
event manager applet track-10 authorization bypass
event track 11 state down
action 0001 cli command "enable"
action 0002 cli command "term exec prompt timestamp"
action 0003 cli command "term length 0"
action 0010 syslog msg "IP SLA object 10 has gone down"
action 0020 cli command "show mac address-table detail | append flash:sla_track.txt"
action 0030 cli command "show ip arp | append flash:sla_track.txt"
action 0040 cli command "show log | append flash:sla_track.txt"
action 0050 cli command "show ip route | append flash:sla_track.txt"
```

從EEM傳送電子郵件

當看到event syslog pattern語句中描述的模式時，將觸發此指令碼，並執行以下操作：

- 從內部電子郵件伺服器傳送電子郵件（假設內部電子郵件伺服器允許從裝置進行開放式驗證

)。

實作

```
event manager environment email_from email_address@company.test
event manager environment email_server 192.168.1.1
event manager environment email_to dest_address@company.test
event manager applet email_syslog
event syslog pattern "SYSLOG PATTERN HERE" maxrun 60
action 0010 info type routename
action 0020 mail server "$email_server" to "$email_to" from "$email_from" subject "SUBJECT OF EMAIL - S"
```

依排程關閉連線埠

該指令碼在每天下午6點關閉埠Te2/1/15。

實作

```
event manager applet shut_port authorization bypass
event timer cron cron-entry "0 18 * * *"
action 0001 cli command "enable"
action 0002 cli command "term exec prompt timestamp"
action 0003 cli command "term length 0"
action 0010 syslog msg "shutting port Te2/1/15 down"
action 0030 cli command "config t"
action 0040 cli command "int Te2/1/15"
action 0050 cli command "shutdown"
action 0060 cli command "end"
```

達到指定的每秒封包(PPS)速率時關閉介面

此指令碼每秒檢查介面Te2/1/9上TX方向的PPS速率。如果PPS速率超過100，它將採取以下操作：

- 將接口的show int輸出記錄到syslog。
- 關閉介面。

實作

```
event manager applet disable_link authorization bypass
event interface name te2/1/9 parameter transmit_rate_pps entry-op ge entry-val 100 poll-interval 1 entry-type value
action 0001 cli command "enable"
action 0002 cli command "term length 0"
action 0010 syslog msg "Detecting high input rate on interface te2/1/9. Shutting interface down."
action 0020 cli command "show int te2/1/9"
```

```
action 0030 syslog msg $_cli_result
action 0040 cli command "config t"
action 0050 cli command "int te2/1/9"
action 0060 cli command "shutdown"
action 0070 cli command "end"
```

相關資訊

- [Cisco EEM最佳實踐](#)
- [思科技術支援與下載](#)

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。