

# EEM – EMBEDDED EVENT MANAGER

## Definition

Embedded Event Manager (EEM) allows you to have event tracking and management functionality directly on the Cisco IOS device, rather than on an external device. By having the configuration locally, actions can still be taken, even if the connection to an external monitoring station is unavailable. Really useful for troubleshooting

## Policy creation

1 – Select event

2 – Define detector options

3 – Define variables (optional)

4 – Define actions

## Applet creation

```
event manager applet XXXXXX
```

## Event example

```
event syslog pattern ".*UPDOWN.*FastEthernet1/0.*"
event none
event track 99 state any
event timer cron cron-entry "15 13 * * 1-5"
```

## Detector options

The different **detector** we can use (like the one in blue above) are the following :

## Define actions

Here are some actions we can take if the applet match the previous event.

## Actions requirement

Track - Define objects  
SNMP - Trap Basic snmp config, enable traps event-manager  
Syslog - Logging enabled

none	allows an event to be run manually
cli	screening CLI input for regex
counter	watch a named counter
interface	generic interface counters / threshold
ioswdsysmon	watchdog – CPU / thresholds..
ipsla	watch for SLA events
nf	watch for NetFlow events
oir	hardware – online insertion removal events
resource	event from embedded resource manager
rf	dual RP – redundancy framework
routing	changes to RIB
snmp	monitor a MIB object for values / thresholds
syslog	screen for regex match
timer	absolute time of day, countdown, watchdog, CRON
track	tracking object event

cli	execute a CLI command
gets/puts	used to send to or pull from tty.
mail/syslog/snmp-trap/cns-event	used to send messages
increment/decrement/append	changing variables
if/else/elseif/while/end/break...	conditional operators
wait	pause for a period of time
track	read or set a tracking object
regex	match
reload / force-switchover	system actions

## Verification commands

```
#show event manager history
#show event manager statistics
#show event manager history events
#show event manager detector
#show event manager directory user
#show event manager environment
#show event manager history traps
#show event manager scheduler
```

```
Router# show event manager history events
No. Time of Event Event Type Name
1 Fri Aug13 21:42:57 2004 snmp applet: SAAping1
2 Fri Aug13 22:20:29 2004 snmp applet: SAAping1
3 Wed Aug18 21:54:48 2004 snmp applet: SAAping1
4 Wed Aug18 22:06:38 2004 snmp applet: SAAping1
5 Wed Aug18 22:30:58 2004 snmp applet: SAAping1
6 Wed Aug18 22:34:58 2004 snmp applet: SAAping1
7 Wed Aug18 22:51:18 2004 snmp applet: SAAping1
8 Wed Aug18 22:51:18 2004 application applet: CustAppl
```

# EEM – EMBEDDED EVENT MANAGER

## Examples

In this page I post some personal examples using the most used event / actions

### Description

This script unshut a defined interface Ser0/0/0 and save the configuration.

Using the event “none”, he has to be triggered using the following command :  
#event manager applet NOSHUT run

### Script

```
event manager applet NOSHUT
event none
action 1.0 cli command "enable"
action 2.0 cli command "config term"
action 3.0 cli command "interface Ser0/0/0"
action 4.0 cli command "no shut"
action 5.0 cli command "end"
action 6.0 cli command "write mem"
```

### Description

This script take some CPU logs in order to find the impacting process when the CPU is higher 80%.

Logs are collected into the file  
flash:high\_cpu.txt

### Script

```
event manager applet HIGHCPU
event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.3.1 get-type exact entry-op ge entry-val 80 poll-interval 0.5
action 1.0 cli command "enable"
action 2.0 cli command "sh clock | append file flash:high_cpu.txt"
action 3.0 cli command "show proc cpu sorted | ex 0.00 | append file flash:high_cpu.txt"
action 4.0 cli command "show memory | append file flash:high_cpu.txt"
action 5.0 cli command "show proc cpu history | append file flash:high_cpu.txt"
action 6.0 cli command "show log | append file flash:high_cpu.txt"
end
```

### Description

This script is the same as the previous but the logs are taken at periodic time using the cron event.

Every monday at 6h10, 7h10, 8h10  
====> "10 6,7,8 \* \* 1"

#### Cron timer

minute.....0 – 59  
hour.....0 – 23  
day of month.....1 – 31  
month of year.....1 – 12  
day of week.....0 – 6 (Sunday == 0)

### Script

```
event manager applet HIGHCPU_CRON
event timer cron cron-entry "10 6,7,8 * * 1"
action 1.0 cli command "enable"
action 2.0 cli command "sh clock | append file flash:high_cpu.txt"
action 3.0 cli command "show proc cpu sorted | ex 0.00 | append file flash:high_cpu.txt"
action 4.0 cli command "show memory | append file flash:high_cpu.txt"
action 5.0 cli command "show proc cpu history | append file flash:high_cpu.txt"
action 6.0 cli command "show log | append file flash:high_cpu.txt"
action 7.0 cli command "end"
Exit

Examples :
"1 1 1 1 *" 1:01 AM, 1 January
"1 9 * * 1-5" 9:01 AM, Monday-Friday
"5 8,16 * * 0,6" 8:05 AM or 4:05 PM, Saturday or Sunday
```

### Description

This script is shutting/unshutting an interface if the text “test” is appearing in the syslog console.

“logging on” must be configured.

### Script

```
event manager applet PATTERN_TEST
event syslog pattern "test"
action 1.0 cli command "enable"
action 2.0 cli command "config t"
action 3.0 cli command "int f0/0"
action 4.0 cli command "shut"
action 5.0 cli command "no shut"
```

### Description

This script is hiding the Applet from the show run.

### Script

```
event manager applet NoAppletsHere
event cli pattern "show run" sync yes
action 111 cli command "enable"
action 112 cli command "show run | excl applet|event|action"
action 113 puts "$_cli_result"
action 114 set _exit_status "0"
```

### Description

This script is displaying a syslog message if the route from IP SLA is down

### Script

```
ip sla 11
icmp-echo 198.19.8.4
timeout 500
frequency 3
!
ip sla schedule 11 life forever start-time now
!
track 1 ip sla 11 reachability
delay down 8 up 10

event manager applet email_server_unreachable
event track 1 state down
action 1.0 syslog msg "Houston we have a problem. Ping failed, server unreachable!"
```