

Embedded Event Manager System Events and Configuration Examples

- System Policies, on page 1
- Events, on page 3
- Configuration Examples, on page 4

System Policies

The following table lists the EEM system policies, the events they monitor, and the corresponding automated actions taken by the system.

Table 1: EEM System Policies and Actions

Event	Description / Action
__BootupPortLoopback	Do CallHome, Error-disable affected ports, log error testing on affected ports after 1 consecutive failure of GOLD "BootupPortLoopback" test.
__PortLoopback	Do CallHome, log error in Syslog/OBFL/Exception Log, and disable further HM testing on affected ports after 10 consecutive failures of GOLD "PortLoopback" test.
__RewriteEngineLoopback	Do CallHome, log error in Syslog/OBFL/Exception Log, and disable further HM testing on affected ports after 10 consecutive failures of GOLD "RewriteEngine" test.
__asicmem	Do CallHome and log error when GOLD "AsicMemory" test fails. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. **Note** To avoid a kernel panic when the test fails, you can override the EEM system policy.
__asic_register_check	Do CallHome, log error, and disable further HM testing for that ASIC device/instance after 20 consecutive failures of GOLD "ASICRegisterCheck" test.

Embedded Event Manager System Events and Configuration Examples

Event	Description / Action
__compact_flash	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "CompactFlash" test.
__crypto_device	Do CallHome and log error when GOLD "CryptoDevice" test fails.
__eobc_port_loopback	Do CallHome and log error when GOLD "EOBCPortLoopback" test fails.
__ethpm_debug_1	Action: none
__ethpm_debug_2	Action: none
__ethpm_debug_3	Action: none
__ethpm_debug_4	Action: none
__ethpm_link_flap	More than 30 link flaps in a 420-second interval. Action: Error. Disable the port.
__external_compact_flash	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "ExternalCompactFlash" test.
__fpgareg	Do CallHome, log error, disable further HM testing after 20 consecutive failures of GOLD "FpgaRegTest" test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. **Note** To avoid a kernel panic when the test fails, you can override the EEM system policy.
__L2ACLRedirect	Do CallHome, log error, disable further HM testing after 10 consecutive failures of L2ACLRedirect test. As the issue causing the test failure may be transient, attempt recovery reload through kernel panic. **Note** To avoid a kernel panic when the test fails, you can override the EEM system policy.
__lcm_module_failure	Power cycle two times and then power down.
__management_port_loopback	Do CallHome and log error when GOLD "ManagementPortLoopback" test fails.
__nvram	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "NVRAM" test.
__pfm_fanabsent_all_systemfan	Shuts down if both fan trays (f1 and f2) are absent for 2 minutes.
__pfm_fanbad_all_systemfan	Syslog when fan goes bad.
__pfm_fanbad_any_singlefan	Syslog when fan goes bad.
__pfm_power_over_budget	Syslog warning for insufficient power overbudget.
__pfm_tempev_major	TempSensor Major Threshold. Action: Shutdown.

Event	Description / Action
__pfm_tempev_minor	TempSensor Minor Threshold. Action: Syslog.
__primary_bootrom	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "PrimaryBootROM" test.
__pwr_mgmt_bus	Do CallHome, log error, and disable further HM testing for the module or spine-card after 20 consecutive failures of GOLD "PwrMgmtBus" test.
__real_time_clock	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "RealTimeClock" test.
__secondary_bootrom	Do CallHome, log error, and disable further HM testing after 20 consecutive failures of GOLD "SecondaryBootROM" test.
__spine_control_bus	Do CallHome, log error, and disable further HM testing for that module or spine-card after 20 consecutive failures of GOLD "SpineControlBus" test.
__standby_fabric_loopback	Do CallHome, log error, and disable further HM testing after 10 consecutive failures.
__status_bus	Do CallHome, log error, and disable further HM testing after 5 consecutive failures of GOLD "StatusBus" test.
__system_mgmt_bus	Do Call Home, log error, and disable further HM testing for that fan or power supply after 20 consecutive failures of GOLD "SystemMgmtBus" test.
__usb	Do Call Home and log error when GOLD "USB" test fails.

Events

The following table describes the EEM events you can use on the device.

Table 2: EEM Events Lookup Table

EEM Event	Description
application	Publishes an application-specific event.
cli	CLI command is entered that matches a pattern with a wildcard.
counter	EEM counter reaches a specified value or range.
fanabsent	System fan tray is absent.
fanbad	System fan generates a fault.
fib	Monitors routes or TCAM usage in the unicast FIB.
gold	GOLD test failure condition is hit.

EEM Event	Description
interface	Interface counter exceeds a threshold.
memory	Available system memory exceeds a threshold.
module	Specified module enters the selected status.
module-failure	Module failure is generated.
none	Runs the policy event without any events specified.
oir	Online insertion or removal occurs.
policy-default	Default parameters and thresholds are used for the events in the system policy you override.
poweroverbudget	Platform software detects a power budget condition.
snmp	SNMP object ID (OID) state changes.
storm-control	Platform software detects an Ethernet packet storm condition.
syslog	Monitors syslog messages and invokes the policy based on the search string in the policy.
sysmgr	System manager generates an event.
temperature	Temperature level in the system exceeds a threshold.
timer	Specified time is reached.
track	Tracked object changes state.

Configuration Examples

Command Line Interface Events

- Monitor Interface Shutdown
- Monitor Module Powerdown
- Add Trigger to Initiate a Rollback

Monitor Interface Shutdown

This example shows how to monitor an interface shutdown:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorShutdown
switch(config-applet)#
```

```
switch(config-applet)# description "Monitors interface shutdown."
switch(config-applet)# event cli match "conf t; interface *; shutdown"
switch(config-applet)# action 1.0 cli show interface e 3/1
switch(config)# copy running-config startup-config
```

Note Outputs of **show** commands entered as part of EEM policy are archived in the logflash as text files with the "eem_archive_" prefix. To view the archived output, use the **show file logflash:eem_archive_***n* command.

Monitor Module Powerdown

This example shows how to monitor a module powerdown:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# event manager applet monitorPoweroff
switch(config-applet)#
switch(config-applet)# description "Monitors module power down."
switch(config-applet)# event cli match "conf t; poweroff *"
switch(config-applet)# action 1.0 cli show module
switch(config)# copy running-config startup-config
```

Add Trigger to Initiate a Rollback

This example shows how to add a trigger to initiate a rollback:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)#
switch(config)# event manager applet rollbackTrigger
switch(config-applet)#
switch(config-applet)# description "Rollback trigger."
switch(config-applet)# event cli match "rollback *"
switch(config-applet)# action 1.0 cli copy running-config bootflash:last_config
switch(config)# copy running-config startup-config
```

Override (Disable) Major Thresholds

- Prevent Shutdown When Reaching a Major Threshold
- Disable One Bad Sensor
- Disable Multiple Bad Sensors
- Override (Disable) an Entire Module
- Override (Disable) Multiple Modules and Sensors
- Enable One Sensor While Disabling All Remaining Sensors of All Modules
- Enable Multiple Sensors While Disabling All Remaining Sensors of All Modules
- Enable All Sensors of One Module While Disabling All Sensors of the Remaining Modules
- Enable Combination of Sensors on Modules While Disabling All Sensors of the Remaining Modules

Prevent Shutdown When Reaching a Major Threshold

This example shows how to prevent a shutdown caused by reaching a major threshold:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Disable One Bad Sensor

This example shows how to disable only sensor 3 on module 2 when sensor 3 is malfunctioning (all other sensors are unaffected):

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Disable Multiple Bad Sensors

This example shows how to disable sensors 5, 6, and 7 on module 2 when these sensors are malfunctioning (all other sensors are unaffected):

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 5 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 6 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Override (Disable) an Entire Module

This example shows how to disable module 2 when it is malfunctioning:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Override (Disable) Multiple Modules and Sensors

This example shows how to disable sensors 3, 4, and 7 on module 2 and all sensors on module 3 when they are malfunctioning:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_tempev_major
switch(config)# end
```

Enable One Sensor While Disabling All Remaining Sensors of All Modules

This example shows how to disable all sensors on all modules except sensor 4 on module 9:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Enable Multiple Sensors While Disabling All Remaining Sensors of All Modules

This example shows how to disable all sensors on all modules except sensors 4, 6, and 7 on module 9:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
```

```
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 4 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 6 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

Enable All Sensors of One Module While Disabling All Sensors of the Remaining Modules

This example shows how to disable all sensors on all modules except all sensors on module 9:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 9 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Enable Combination of Sensors on Modules While Disabling All Sensors of the Remaining Modules

This example shows how to disable all sensors on all modules except sensors 3, 4, and 7 on module 2 and all sensors on module 3:

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 3 threshold major
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 4 threshold major
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_tempev_major
switch(config-applet)# event temperature module 2 sensor 7 threshold major
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet5 override __pfm_tempev_major
switch(config-applet)# event temperature module 3 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

Override (Disable) Shutdown for Fan Tray Removal

- Override (Disable) Shutdown for Removal of a Specified Fan Tray
- Override (Disable) Shutdown for Removal of One or More Fan Trays
- Override (Disable) Shutdown for Removal of All Fan Trays Except One from a Set of Fan Trays
- Override (Disable) Shutdown for Removal of Fan Trays Except for a Specified Set of Fan Trays
- Override (Disable) Shutdown for Removal of All Fan Trays Except One
- Override (Disable) Shutdown for Removal of Multiple Specified Fan Trays

Override (Disable) Shutdown for Removal of a Specified Fan Tray

This example shows how to disable a shutdown so that you can remove a specified fan tray (fan tray 3):

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config) no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

Override (Disable) Shutdown for Removal of One or More Fan Trays

This example shows how to disable a shutdown so that you can remove one or more (or all) fan trays:

```
switch# configure terminal
switch(config)# event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
```

Override (Disable) Shutdown for Removal of All Fan Trays Except One from a Set of Fan Trays

This example shows how to disable a shutdown so that you can remove all fan trays except one from a set of fan trays (fan trays 2, 3, or 4):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
```

```
switch(config-applet)# action 3 policy-default
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet4 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# action 4 policy-default
switch(config-applet)# end
```

Override (Disable) Shutdown for Removal of Fan Trays Except for a Specified Set of Fan Trays

This example shows how to disable a shutdown so that you can remove fans except for a specified set of fan trays (fan trays 2, 3, and 4):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2,3,4 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Override (Disable) Shutdown for Removal of All Fan Trays Except One

This example shows how to disable a shutdown so that you can remove all fan trays except one (fan tray 2):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# action 2 policy-default
switch(config-applet)# end
```

Override (Disable) Shutdown for Removal of Multiple Specified Fan Trays

This example shows how to disable a shutdown so that you can remove multiple specified fan trays (fan trays 2, 3, and 4):

```
switch# configure terminal
switch(config)# event manager applet myapplet1 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 2 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet2 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 3 time 60
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet myapplet3 override __pfm_fanabsent_any_singlefan
switch(config-applet)# event fanabsent fan 4 time 60
switch(config-applet)# end
```

This example shows how to revert to the default configuration:

```
switch# configure terminal
switch(config)# no event manager applet myappletname override __pfm_fanabsent_any_singlefan
switch(config)# end
```

Create Supplemental Policy

- Create Supplemental Policy for the Fan Tray Absent Event
- Create Supplemental Policy for the Temperature Threshold Event

Create Supplemental Policy for the Fan Tray Absent Event

This example shows how to create a supplemental policy using the **event fanabsent** command:

[**no**] **event fanabsent** [**fan** *fan-tray-number*] **time** *time-interval*

In addition to the default policy, this example shows how to execute the policy myappletname and action 3 if fan tray 1 is absent for 60 seconds:

```
switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event fanabsent fan 1 time 60
switch(config-applet)# action 3 cli "show env fan"
switch(config-applet)# end
```

Create Supplemental Policy for the Temperature Threshold Event

This example shows how to create a supplemental policy using the **event temperature** command:

[**no**] **event temperature** [**mod** *module-number*] [**sensor** *sensor-number*] **threshold** {**major** | **minor** | **any**}

In addition to the default policy, this example shows how to execute the policy myappletname and action 1 if the temperature crosses the minor threshold on sensor 3 of module 2:

```
switch# configure terminal
switch(config)# event manager applet myappletname
switch(config-applet)# event temperature module 2 sensor 3 threshold minor
switch(config-applet)# action 1 cli "show environ temperature"
switch(config-applet)# end
```

Power Over-Budget Policy

The power over-budget policy gets triggered when the available power capacity drops below zero and the device is no longer able to keep the previously powered-up modules in the powered-up state. The default action is to print a syslog to notify the user of the occurrence of power over budget.

You can enable an additional action to power down modules until the available power recovers from the red (negative) zone.

- Shut Down Modules
- Shut Down Specified List of Modules

Shut Down Modules

If you do not specify any modules, the power over-budget shutdown starts from slot 1 and shuts down modules until the power recovers from the red (negative) zone. Empty slots and slots that contain a supervisor, standby supervisor, spine, or crossbar are skipped.

This example shows how to shut down modules starting from module 1 when the available power drops below zero:

```
switch# configure terminal
switch(config)# event manager applet <myappletname4a> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 4 overbudgetshut
switch(config-applet)# end
```

Shut Down Specified List of Modules

You can specify a list of modules that the power over-budget action uses to shut down modules until the power recovers from the red (negative) zone. Empty slots and slots that contain a supervisor, standby supervisor, spine, or crossbar are skipped.

This example shows how to shut down modules from a specified list of modules (1, 2, 7, 8) when the available power drops below zero:

```
switch# configure terminal
switch(config)# event manager applet <myappletname4b> override __pfm_power_over_budget
switch(config-applet)# event poweroverbudget
switch(config-applet)# action 5 overbudgetshut module 1,2,7,8
switch(config-applet)# end
```

Select Modules to Shut Down

- Use Parameter Substitution to Select Nonoverridden Modules to Shut Down
- Use the Policy Default to Select Nonoverridden Modules to Shut Down

Use Parameter Substitution to Select Nonoverridden Modules to Shut Down

This example shows how to use parameter substitution to select the nonoverridden modules to shut down when a major threshold is exceeded:

```
switch# configure terminal
switch(config)# event manager applet my5b1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet my5b2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 8 threshold major
switch(config-applet)# action 6 forceshut module my_module_list reset "temperature-sensor
policy trigger"
switch(config-applet)# end
```

To create event manager parameters, use the **event manager environment** command. To display the values of event manager parameters, use the **show event manager environment all** command.

Use the Policy Default to Select Nonoverridden Modules to Shut Down

This example shows how to use the policy default to select the nonoverridden modules to shut down when a major threshold is exceeded:

```
switch# configure terminal
switch(config)# event manager applet my5a1 override __pfm_tempev_major
switch(config-applet)# end
switch# configure terminal
switch(config)# event manager applet my5a2 override __pfm_tempev_major
switch(config-applet)# event temperature module 1-3 sensor 4 threshold major
switch(config-applet)# action 5 policy-default
switch(config-applet)# end
```

Online Insertion Removal Event

The Online Insertion Removal (OIR) event is a system event that is triggered when a device is inserted or removed while the system is powered on.

- The **event oir** command is used to configure OIR events.
- The *device-type* can be **fan**, **module**, or **powersupply**.
- The *event-type* can be **insert**, **remove**, or **anyoir** (insert or remove).
- The optional *device-number* specifies a single device; if omitted, all devices are selected.

This example shows how to configure the OIR event using the **event oir** command:

event oir *device-type event-type* [*device-number*]

This example shows how to configure the insert event:

```
switch# configure terminal
 switch(config)# event manager applet myoir
 switch(config-applet)# event oir module insert
 switch(config-applet)# action 1 syslog priority critical msg "OIR insert event: A Module
is inserted"
```

This example shows how to configure the remove event:

```
switch# configure terminal
 switch(config)# event manager applet myoir
 switch(config-applet)# event oir module remove
 switch(config-applet)# action 1 syslog priority critical msg "OIR remove event: A Module
is removed"
```

Generate User Syslog

This example shows how to generate a user syslog using the **action syslog** command:

```
 switch# configure terminal
 switch(config)#event manager applet myoir
 witch(config-applet)# event oir module remove
       switch(config-applet)# action 1 syslog priority critical msg "Module is removed"
```

When this event is triggered, the system generates a syslog as follows:

```
switch(config)# 2013 May 20 00:08:27 p1b-57 %$ VDC-1 %$ %EEM_ACTION-2-CRIT: "Module is
removed"
```

Monitor Syslog Messages

```
 switch(config)# event manager applet a1
 switch(config-applet)# event syslog occurs 6 period 4294967 pattern "authentication failed"
```

When this event is triggered, the action defined in the policy is executed.

SNMP Notification

- Poll SNMP OID to Generate an EEM Event
- Send SNMP Notification in Response to an Event in the Event Policy

Poll SNMP OID to Generate an EEM Event

The SNMP object ID (OID) CISCO-SYSTEM-EXT-MIB::cseSysCPUUtilization is used for querying the CPU utilization of the switch:

```
cseSysCPUUtilization OBJECT-TYPE
SYNTAX Gauge32 (0..100 )
UNITS "%"
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"The average utilization of CPU on the active supervisor."
::= { ciscoSysInfoGroup 1 }
```

This example shows the use of an SNMP OID that is polled at an interval of 10 seconds and has a threshold value of 95 percent:

```
switch# configure terminal
switch(config)# event manager applet test_policy
switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.305.1.1.1.0 get-type exact entry-op
gt entry-val 95 exit-op lt exit-val 90 poll-interval 10
```

Send SNMP Notification in Response to an Event in the Event Policy

You can use this type of configuration to cause a critical event trigger to generate an SNMP notification.

This example shows how to send an SNMP notification for an event from the Event Manager applet configuration mode:

```
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "CPU Hogging
 at switch1"
switch(config-applet)# action 1.1 snmp-trap intdata1 100 intdata2 300 strdata "Port Failure
 eth9/1"
```

This configuration triggers an SNMP notification (trap) from the switch to SNMP hosts. The SNMP payload carries the values of user-defined fields intdata1, intdata2, and strdata.

Port Tracking

This example shows how to configure the state of one port to match the state of another port (port tracking).

To configure the port tracking of Ethernet interface 3/23 by Ethernet interface 1/2, follow these steps:

Procedure

Step 1 Create an object to track the status of Ethernet interface 3/23.

Example:

```
switch# configure terminal
switch(config)# track 1 interface ethernet 3/23
switch(config-track)# end
```

Step 2 Configure an EEM event to shut Ethernet interface 1/2 when the tracking object shuts down.

Example:

```
switch(config)# event manager applet track_3_23_down
switch(config-applet)# event track 1 state down
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down shutting down port
eth1/2 due to eth3/23 being down
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli shut
switch(config-applet)# end
```

Step 3 Configure an EEM event to bring up Ethernet interface 1/2 when Ethernet interface 3/23 comes up.

Example:

```
switch# configure terminal
switch(config)# event manager applet track_3_23_up
switch(config-applet)# event track 1 state up
switch(config-applet)# action 1 syslog msg EEM applet track_3_23_down bringing up port
eth1/2 due to eth3/23 being up
switch(config-applet)# action 2 cli conf term
switch(config-applet)# action 3 cli interface ethernet 1/2
switch(config-applet)# action 4 cli no shut
switch(config-applet)# end
```

Register an EEM Policy with the EEM

This example shows how to register an EEM policy with the EEM:

Basic Switch Configuration

Note The severity keyword is deprecated and only the following patterns are allowed:

[0-9 a-zA-Z][0-9 a-zA-Z]*[-_ ,:/0-9a-zA-Z]*

```
event manager applet vpc_check_peer_at_startup
event track 101 state up
action 1.0 cli copy bootflash:eem/user_script_policies/load_schedules running-config
feature scheduler
!!## 2 x dummy loopbacks are required ##!!
interface loopback 101
interface loopback 102
track 1 list boolean or
object 13
```

```
object 12
object 102
track 2 list boolean and
object 13
object 12
track 12 interface Ethernet 2/24 line-protocol
track 13 interface port-channel 3000 line-protocol
track 101 interface loopback 101 line-protocol
track 102 interface loopback 102 line-protocol
```



Note In this example, port channel 3000 is the vPC peer link, and Ethernet 2/24 is the vPC keepalive link.

You need to copy the following files to the bootflash:

- A directory called: /eem/user_script_policies needs to be created on the supervisor bootflash.
- These five files need to be created and loaded into the above directory:
  - load_schedules
  - remove_vpc_if_peer_failed
  - clean_up
  - unload_schedules
  - restore_vpc

Configuration for the load_schedules File

```
feature scheduler
configure terminal
scheduler job name vpc_check
configure terminal
event manager policy remove_vpc_if_peer_failed
end
configure terminal
scheduler job name clean_up
configure terminal
event manager policy clean_up
end
configure terminal
scheduler job name trigger
configure terminal
interface loopback 102
shutdown
no shutdown
end
configure terminal
scheduler schedule name load_vpc_check
time start +00:00:04
job name vpc_check
scheduler schedule name trigger_vpc_check
time start +00:00:05
job name trigger
scheduler schedule name load_clean_up
time start +00:00:08
job name clean_up
scheduler schedule name trigger_clean_up
```

```
time start +00:00:10
job name trigger
```

Configuration for the remove_vpc_if_peer_failed File

```
event manager applet remove_vpc_if_peer_failed
event track 1 state down
action 1.0 cli show run vpc > bootflash://sup-active/eem/user_script_policies/vpc_saved.cfg
action 2.0 cli show run vpc > bootflash://sup-standby/eem/user_script_policies/vpc_saved.cfg
action 3.0 cli configure terminal
action 4.0 cli no feature vpc
action 5.0 syslog msg severity alert "##### WARNING!!!! PEER SWITCH FAILED TO COME ONLINE.
 VPC CONFIG REMOVED #####"
action 6.0 cli event manager policy restore_vpc
action 7.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 8.0 cli no event manager applet remove_vpc_if_peer_failed
action 9.0 cli end
```

Configuration for the clean_up File

```
event manager applet clean_up
event track 102 state up
action 1.0 cli configure terminal
action 2.0 cli no event manager applet remove_vpc_if_peer_failed
action 3.0 cli copy bootflash:eem/user_script_policies/unload_schedules running
action 4.0 cli no event manager applet clean_up
action 5.0 end
```

Configuration for the unload_schedules File

```
no scheduler schedule name load_vpc_check
no scheduler schedule name trigger_vpc_check
no scheduler schedule name load_clean_up
no scheduler schedule name trigger_clean_up
no scheduler job name vpc_check
no scheduler job name trigger
no scheduler job name clean_up
```

Configuration for the restore_vpc File

```
event manager applet restore_vpc
event track 2 state up
action 1.0 cli copy bootflash:eem/user_script_policies/vpc_saved.cfg running-config
action 1.0 syslog priority alerts msg VPC PEER DETECTED. VPC CONFIG RESTORED
action 3.0 cli configure terminal
action 4.0 cli copy bootflash:eem/user_script_policies/unload_schedules running-config
action 5.0 cli no event manager applet restore_vpc
action 6.0 cli end
```

Embedded Event Manager System Events and Configuration Examples