



Configuring the Embedded Event Manager

This chapter contains the following sections:

- [Information About Embedded Event Manager, page 1](#)
- [Configuring Embedded Event Manager, page 5](#)
- [Verifying the Embedded Event Manager Configuration, page 16](#)
- [Configuration Examples for Embedded Event Manager, page 17](#)
- [Additional References, page 18](#)
- [Feature History for EEM, page 18](#)

Information About Embedded Event Manager

The ability to detect and handle critical events in the Cisco NX-OS system is important for high availability. The Embedded Event Manager (EEM) provides a central, policy-driven framework to detect and handle events in the system by monitoring events that occur on your device and taking action to recover or troubleshoot these events, based on your configuration..

EEM consists of three major components:

Event statements

Events to monitor from another Cisco NX-OS component that may require some action, workaround, or notification.

Action statements

An action that EEM can take, such as sending an e-mail or disabling an interface, to recover from an event.

Policies

An event paired with one or more actions to troubleshoot or recover from the event.

Without EEM, each individual component is responsible for detecting and handling its own events. For example, if a port flaps frequently, the policy of "putting it into errDisable state" is built into ETHPM.

Embedded Event Manager Policies

An EEM policy consists of an event statement and one or more action statements. The event statement defines the event to look for as well as the filtering characteristics for the event. The action statement defines the action EEM takes when the event occurs.

For example, you can configure an EEM policy to identify when a card is removed from the device and log the details related to the card removal. By setting up an event statement that tells the system to look for all instances of card removal and then with an action statement that tells the system to log the details.

You can configure EEM policies using the command line interface (CLI) or a VSH script.

EEM gives you a device-wide view of policy management. Once EEM policies are configured, the corresponding actions are triggered. All actions (system or user-configured) for triggered events are tracked and maintained by the system.

Preconfigured System Policies

Cisco NX-OS has a number of preconfigured system policies. These system policies define many common events and actions for the device. System policy names begin with two underscore characters (__).

Some system policies can be overridden. In these cases, you can configure overrides for either the event or the action. The overrides that you configure take the place of the system policy.



Note

Override policies must include an event statement. Override policies without event statements override all possible events for the system policy.

To view the preconfigured system policies and determine which policies you can override, use the **show event manager system-policy** command.

User-Created Policies

User-created policies allow you to customize EEM policies for your network. If a user policy is created for an event, actions in the policy are triggered only after EEM triggers the system policy actions related to the same event.

Log Files

The log file that contains data that is related to EEM policy matches is maintained in the event_archive_1 log file located in the /log/event_archive_1 directory.

Event Statements

Any device activity for which some action, such as a workaround or notification, is taken is considered an event by EEM. In many cases, events are related to faults in the device, such as when an interface or a fan malfunctions.

Event statements specify which event or events triggers a policy to run.

**Tip**

YOu can configure EEM to trigger an EEM policy that is based on a combination of events by creating and differentiating multiple EEM events in the policy and then defining a combination of events to trigger a custom action.

EEM defines event filters so that only critical events or multiple occurrences of an event within a specified time period trigger an associated action.

Some commands or internal events trigger other commands internally. These commands are not visible, but will still match the event specification that triggers an action. You cannot prevent these commands from triggering an action, but you can check which event triggered an action.

Supported Events

EEM supports the following events in event statements:

- Counter events
- Fan absent events
- Fan bad events
- Memory thresholds events
- Events being used in overridden system policies.
- SNMP notification events
- Syslog events
- System manager events
- Temperature events
- Track events

Action Statements

Action statements describe the action that is triggered by a policy when an event occurs. Each policy can have multiple action statements. If no action is associated with a policy, EEM still observes events but takes no actions.

In order for triggered events to process default actions, you must configure the EEM policy to allow the default action. For example, if you match a CLI command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute.

**Note**

When configuring action statements within your user policy or overriding policy, it is important that you confirm that action statements do not negate each other or adversely affect the associated system policy.

Supported Actions

EEM supports the following actions in action statements:

- Execute any CLI commands

- Update a counter
- Reload the device
- Generate a syslog message
- Generate an SNMP notification
- Use the default action for the system policy

VSH Script Policies

You can write policies in a VSH script, by using a text editor. Policies that are written using a VSH script have an event statement and action statement(s) just as other policies, and these policies can either augment or override system policies.

After you define your VSH script policy, copy it to the device and activate it.

Licensing Requirements for Embedded Event Manager

This feature does not require a license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Prerequisites for Embedded Event Manager

You must have network-admin privileges to configure EEM.

Guidelines and Limitations for Embedded Event Manager

When you plan your EEM configuration, consider the following:

- The maximum number of configurable EEM policies is 500.
- Action statements within your user policy or overriding policy should not negate each other or adversely affect the associated system policy.
- If you want to allow a triggered event to process any default actions, you must configure the EEM policy to allow the default action. For example, if you match a command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute.
- An override policy that consists of an event statement and no action statement triggers no action and no notification of failures.
- An override policy without an event statement overrides all possible events in the system policy.
- In regular command expressions: all keywords must be expanded, and only the asterisk (*) symbol can be used for replace the arguments.
- EEM event correlation supports up to four event statements in a single policy. The event types can be the same or different, but only these event types are supported: cli, counter, snmp, syslog, and track.

- When more than one event statement is included in an EEM policy, each event statement must have a **tag** keyword with a unique tag argument.
- EEM event correlation does not override the system default policies.
- Default action execution is not supported for policies that are configured with tagged events.
- If your event specification matches a CLI pattern, you can use SSH-style wild card characters.
For example, if you want to match all show commands, enter the **show *** command. Entering the **show . *** command does not work.
- If your event specification is a regular expression for a matching syslog message, you can use a proper regular expression.
For example, if you want to detect ADMIN_DOWN events on any port where a syslog is generated, use **.ADMIN_DOWN.**. Entering the **ADMIN_DOWN** command does not work.
- In the event specification for a syslog, the regex does not match any syslog message that is generated as an action of an EEM policy.
- If an EEM event matches a **show** command in the CLI and you want the output for that **show** command to display on the screen (and to not be blocked by the EEM policy), you must specify the **event-default** command for the first action for the EEM policy.

Default Settings for Embedded Event Manager

Table 1: Default EEM Parameters

Parameters	Default
System Policies	Active

Configuring Embedded Event Manager

Defining an Environment Variable

Defining an environment variable is an optional step but is useful for configuring common values for repeated use in multiple policies.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.

	Command or Action	Purpose
Step 2	event manager environment <i>variable-name</i> <i>variable-value</i> Example: <pre>switch(config) # event manager environment emailto "admin@anyplace.com"</pre>	Creates an environment variable for EEM. The <i>variable-name</i> can be any case-sensitive, alphanumeric string up to 29 characters. The <i>variable-value</i> can be any quoted case-sensitive, alphanumeric string up to 39 characters.
Step 3	show event manager environment { <i>variable-name</i> all} Example: <pre>switch(config) # show event manager environment all</pre>	(Optional) Displays information about the configured environment variables.
Step 4	copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Configure a User Policy.

Defining a User Policy Using the CLI

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	event manager applet <i>applet-name</i> Example: <pre>switch(config)# event manager applet monitorShutdown switch(config-applet)#</pre>	Registers the applet with EEM and enters applet configuration mode. The <i>applet-name</i> can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	description <i>policy-description</i> Example: <pre>switch(config-applet)# description "Monitors interface shutdown."</pre>	(Optional) Configures a descriptive string for the policy. The string can be any alphanumeric string up to 80 characters. Enclose the string in quotation marks.

	Command or Action	Purpose
Step 4	event <i>event-statement</i> Example: switch(config-applet)# event cli match "shutdown"	Configures the event statement for the policy.
Step 5	tag <i>tag</i> { and andnot or } <i>tag</i> [and andnot or { <i>tag</i> }] { happens <i>occurs in seconds</i> } Example: switch(config-applet)# tag one or two happens 1 in 10000	(Optional) Correlates multiple events in the policy. The range for the <i>occurs</i> argument is from 1 to 4294967295. The range for the <i>seconds</i> argument is from 0 to 4294967295 seconds.
Step 6	action <i>number</i> [. <i>number2</i>] <i>action-statement</i> Example: switch(config-applet)# action 1.0 cli show interface e 3/1	Configures an action statement for the policy. Repeat this step for multiple action statements.
Step 7	show event manager policy-state <i>name</i> [<i>module module-id</i>] Example: switch(config-applet)# show event manager policy-state monitorShutdown	(Optional) Displays information about the status of the configured policy.
Step 8	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Configure event statements and action statements.

Configuring Event Statements

Use one of the following commands in EEM configuration mode (config-applet) to configure an event statement:

Before You Begin

Define a user policy.

Procedure

	Command or Action	Purpose
Step 1	event cli [tag tag] match expression [count repeats time seconds] Example: <pre>switch(config-applet) # event cli match "shutdown"</pre>	Triggers an event if you enter a command that matches the regular expression. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>repeats</i> range is from 1 to 65000. The <i>time</i> range is from 0 to 4294967295, where 0 indicates no time limit.
Step 2	event counter [tag tag] name counter entry-val entry entry-op {eq ge gt le lt ne} { exit-val exit exit-op {eq ge gt le lt ne} } Example: <pre>switch(config-applet) # event counter name mycounter entry-val 20 gt</pre>	Triggers an event if the counter crosses the entry threshold based on the entry operation. The event resets immediately. Optionally, you can configure the event to reset after the counter passes the exit threshold. The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy. The <i>counter</i> name can be any case-sensitive, alphanumeric string up to 28 characters. The <i>entry</i> and <i>exit</i> value ranges are from 0 to 2147483647.
Step 3	event fanabsent [fan number] time seconds Example: <pre>switch(config-applet) # event fanabsent time 300</pre>	Triggers an event if a fan is removed from the device for more than the configured time, in seconds. The <i>number</i> range is from 1 to 1 and is module-dependent. The <i>seconds</i> range is from 10 to 64000.
Step 4	event fanbad [fan number] time seconds Example: <pre>switch(config-applet) # event fanbad time 3000</pre>	Triggers an event if a fan fails for more than the configured time, in seconds. The <i>number</i> range is module-dependent. The <i>seconds</i> range is from 10 to 64000.
Step 5	event memory { critical minor severe } Example: <pre>switch(config-applet) # event memory critical</pre>	Triggers an event if a memory threshold is crossed.
Step 6	event policy-default count repeats [time seconds] Example: <pre>switch(config-applet) # event policy-default count 3</pre>	Uses the event configured in the system policy. Use this option for overriding policies. The <i>repeats</i> range is from 1 to 65000. The <i>seconds</i> range is from 0 to 4294967295, where 0 indicates no time limit.

	Command or Action	Purpose
Step 7	<p>event snmp [tag tag] oid oid get-type {exact next} entry-op {eq ge gt le lt ne} entry-val entry [exit-comb {and or}] exit-op {eq ge gt le lt ne} exit-val exit exit-time time polling-interval interval</p> <p>Example: <pre>switch(config-applet) # event snmp oid 1.3.6.1.2.1.31.1.1.1.6 get-type next entry-op lt 300 entry-val 0 exit-op eq 400 exit-time 30 polling-interval 300</pre></p>	<p>Triggers an event if the SNMP OID crosses the entry threshold based on the entry operation. The event resets immediately, or optionally you can configure the event to reset after the counter passes the exit threshold. The OID is in dotted decimal notation.</p> <p>The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy.</p> <p>The entry and exit value ranges are from 0 to 18446744073709551615.</p> <p>The time, in seconds, is from 0 to 2147483647.</p> <p>The interval, in seconds, is from 0 to 2147483647.</p>
Step 8	<p>event sysmgr memory [module module-num] major major-percent minor minor-percent clear clear-percent</p> <p>Example: <pre>switch(config-applet) # event sysmgr memory minor 80</pre></p>	<p>Triggers an event if the specified system manager memory threshold is exceeded.</p> <p>The percent range is from 1 to 99.</p>
Step 9	<p>event temperature [module slot] [sensor number] threshold {any down up}</p> <p>Example: <pre>switch(config-applet) # event temperature module 2 threshold any</pre></p>	<p>Triggers an event if the temperature sensor exceeds the configured threshold.</p> <p>The sensor range is from 1 to 18.</p>
Step 10	<p>event track [tag tag] object-number state {any down up}</p> <p>Example: <pre>switch(config-applet) # event track 1 state down</pre></p>	<p>Triggers an event if the tracked object is in the configured state.</p> <p>The tag tag keyword-argument pair identifies this specific event when multiple events are included in the policy.</p> <p>The object-number range is from 1 to 500.</p>

What to Do Next

Configure action statements.

If you have already configured action statements or choose not to, complete any of the optional tasks:

- Define a policy using a VSH script. Then, register and activate a VSH script policy.
- Configure memory thresholds
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Configuring Action Statements

You can configure an action by using one of the following commands in EEM configuration mode (config-applet):



Note

If you want to allow a triggered event to process any default actions, you must configure the EEM policy to allow the default action.

For example, if you match a command in a match statement, you must add the event-default action statement to the EEM policy or EEM does not allow the command to execute. You can use the **terminal event-manager bypass** command to allow all EEM policies with matches to execute the command.

Before You Begin

Define a user policy.

Procedure

	Command or Action	Purpose
Step 1	action <i>number</i> [. <i>number2</i>] cli <i>command1</i> [<i>command2</i> .] [local] Example: <pre>switch(config-applet) # action 1.0 cli "show interface e 3/1"</pre>	Runs the configured commands. You can optionally run the commands on the module where the event occurred. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits. The range for <i>number2</i> is from 0 to 9.
Step 2	action <i>number</i> [. <i>number2</i>] counter name <i>counter value val op</i> { dec inc nop set } Example: <pre>switch(config-applet) # action 2.0 counter name mycounter value 20 op inc</pre>	Modifies the counter by the configured value and operation. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits. The range for <i>number2</i> is from 0 to 9. The <i>counter</i> can be any case-sensitive, alphanumeric string up to 28 characters. The <i>val</i> can be an integer from 0 to 2147483647 or a substituted parameter.
Step 3	action <i>number</i> [. <i>number2</i>] event-default Example: <pre>switch(config-applet) # action 1.0 event-default</pre>	Completes the default action for the associated event. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits. The range for <i>number2</i> is from 0 to 9.
Step 4	action <i>number</i> [. <i>number2</i>] policy-default Example: <pre>switch(config-applet) # action 1.0 policy-default</pre>	Completes the default action for the policy that you are overriding. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits.

	Command or Action	Purpose
		The range for <i>number2</i> is from 0 to 9.
Step 5	action <i>number</i> [. <i>number2</i>] reload [module <i>slot</i> [- <i>slot</i>]] Example: <pre>switch(config-applet) # action 1.0 reload module 3-5</pre>	Forces one or more modules to the entire system to reload. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits. The range for <i>number2</i> is from 0 to 9.
Step 6	action <i>number</i> [. <i>number2</i>] snmp-trap [intdata1 <i>integer-data1</i>] [intdata2 <i>integer-data2</i>] [strdata <i>string-data</i>] Example: <pre>switch(config-applet) # action 1.0 snmp-trap strdata "temperature problem"</pre>	Sends an SNMP trap with the configured data. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits. The range for <i>number2</i> is from 0 to 9. The <i>data</i> elements can be any number up to 80 digits. The <i>string</i> can be any alphanumeric string up to 80 characters.
Step 7	action <i>number</i> [. <i>number2</i>] syslog [priority <i>prio-val</i>] msg <i>error-message</i> Example: <pre>switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"</pre>	Sends a customized syslog message at the configured priority. The action label is in the format <i>number1.number2</i> . The <i>number</i> can be any number from 1 to 16 digits. The range for <i>number2</i> is from 0 to 9. The <i>error-message</i> can be any quoted alphanumeric string up to 80 characters.

What to Do Next

Configure event statements.

If you have already configured event statements or choose not to, complete any of the optional tasks:

- Define a policy using a VSH script. Then, register and activate a VSH script policy.
- Configure memory thresholds
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Defining a Policy Using a VSH Script

This is an optional task. Complete the following steps if you are using a VSH script to write EEM policies:

Procedure

-
- Step 1** In a text editor, list the commands that define the policy.
- Step 2** Name the text file and save it.
- Step 3** Copy the file to the following system directory: bootflash://eem/user_script_policies
-

What to Do Next

Register and activate a VSH script policy.

Registering and Activating a VSH Script Policy

This is an optional task. Complete the following steps if you are using a VSH script to write EEM policies.

Before You Begin

Define a policy using a VSH script and copy the file to the system directory.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	event manager policy <i>policy-script</i> Example: switch(config)# event manager policy moduleScript	Registers and activates an EEM script policy. The <i>policy-script</i> can be any case-sensitive, alphanumeric string up to 29 characters.
Step 3	event manager policy internal <i>name</i> Example: switch(config)# event manager policy internal moduleScript	(Optional) Registers and activates an EEM script policy. The <i>policy-script</i> can be any case-sensitive alphanumeric string up to 29 characters.
Step 4	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Complete any of the following, depending on your system requirements:

- Configure memory thresholds.
- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Overriding a System Policy

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	show event manager policy-state <i>system-policy</i> Example: <pre>switch(config-applet)# show event manager policy-state __ethpm_link_flap Policy __ethpm_link_flap Cfg count : 5 Cfg time interval : 10.000000 (seconds) Hash default, Count 0</pre>	(Optional) Displays information about the system policy that you want to override, including thresholds. Use the show event manager system-policy command to find the system policy names.
Step 3	event manager applet <i>applet-name</i> override <i>system-policy</i> Example: <pre>switch(config-applet)# event manager applet ethport override __ethpm_link_flap switch(config-applet)#</pre>	Overrides a system policy and enters applet configuration mode. The <i>applet-name</i> can be any case-sensitive, alphanumeric string up to 80 characters. The <i>system-policy</i> must be one of the system policies.
Step 4	description <i>policy-description</i> Example: <pre>switch(config-applet)# description "Overrides link flap policy"</pre>	Configures a descriptive string for the policy. The <i>policy-description</i> can be any case-sensitive, alphanumeric string up to 80 characters, but it must be enclosed in quotation marks.
Step 5	event <i>event-statement</i> Example: <pre>switch(config-applet)# event policy-default count 2 time 1000</pre>	Configures the event statement for the policy.
Step 6	section <i>number</i> <i>action-statement</i> Example: <pre>switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."</pre>	Configures an action statement for the policy. For multiple action statements, repeat this step.

	Command or Action	Purpose
Step 7	show event manager policy-state <i>name</i> Example: <pre>switch(config-applet)# show event manager policy-state ethport</pre>	(Optional) Displays information about the configured policy.
Step 8	copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Configuring Memory Thresholds

Memory thresholds are used to trigger events and set whether the operating system should stop processes if it cannot allocate memory.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	system memory-thresholds <i>minor</i> <i>minor</i> <i>severe</i> <i>severe</i> <i>critical</i> <i>critical</i> Example: <pre>switch(config)# system memory-thresholds minor 60 severe 70 critical 80</pre>	Configures the system memory thresholds that generate EEM memory events. The default values are as follows: <ul style="list-style-type: none"> • Minor—85 • Severe—90 • Critical—95 When these memory thresholds are exceeded, the system generates the following syslogs: <ul style="list-style-type: none"> • 2009 May 7 17:06:30 switch %\$ VDC-1 %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR • 2009 May 7 17:06:30 switch %\$ VDC-1 %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE

	Command or Action	Purpose
		<ul style="list-style-type: none"> • 2009 May 7 17:06:30 switch %\$ VDC-1 %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL • 2009 May 7 17:06:35 switch %\$ VDC-1 %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : MINOR ALERT RECOVERED • 2009 May 7 17:06:35 switch %\$ VDC-1 %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : SEVERE ALERT RECOVERED • 2009 May 7 17:06:35 switch %\$ VDC-1 %\$ %PLATFORM-2-MEMORY_ALERT: Memory Status Alert : CRITICAL ALERT RECOVERED
Step 3	system memory-thresholds threshold critical no-process-kill Example: <pre>switch(config)# system memory-thresholds threshold critical no-process-kill</pre>	(Optional) Configures the system to not stop processes when the memory cannot be allocated. The default value is to allow the system to stop processes, starting with the one that consumes the most memory.
Step 4	show running-config include "system memory" Example: <pre>switch(config)# show running-config include "system memory"</pre>	(Optional) Displays information about the system memory configuration.
Step 5	copy running-config startup-config Example: <pre>switch(config)# copy running-config startup-config</pre>	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Complete any of the following, depending on your system requirements:

- Configure the syslog as an EEM publisher.
- Verify your EEM configuration.

Configuring Syslog as an EEM Publisher

Configuring syslog as an EEM publisher allows you to monitor syslog messages from the switch.

**Note**

The maximum number of searchable strings to monitor syslog messages is 10.

Before You Begin

- Confirm that EEM is available for registration by the syslog.
- Confirm that the syslog daemon is configured and executed.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: switch# configure terminal switch(config)#	Enters global configuration mode.
Step 2	event manager applet <i>applet-name</i> Example: switch(config)# event manager applet abc switch (config-applet)#	Registers an applet with EEM and enters applet configuration mode.
Step 3	event syslog [tag <i>tag</i>] {occurs <i>number</i> period <i>seconds</i> pattern <i>msg-text</i> priority <i>priority</i>} Example: switch(config-applet)# event syslog occurs 10	Registers an applet with EEM and enters applet configuration mode.
Step 4	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

What to Do Next

Verify your EEM configuration.

Verifying the Embedded Event Manager Configuration

Use one of the following commands to verify the configuration:

Command	Purpose
show event manager environment [<i>variable-name</i> all]	Displays information about the event manager environment variables.

Command	Purpose
show event manager event-types [<i>event</i> all module <i>slot</i>]	Displays information about the event manager event types.
show event manager history events [detail] [maximum <i>num-events</i>] [severity { catastrophic minor moderate severe }]	Displays the history of events for all policies.
show event manager policy internal [<i>policy-name</i>] [inactive]	Displays information about the configured policies.
show event manager policy-state <i>policy-name</i>	Displays information about the policy state, including thresholds.
show event manager script system [<i>policy-name</i> all]	Displays information about the script policies.
show event manager system-policy [all]	Displays information about the predefined system policies.
show running-config eem	Displays information about the running configuration for EEM.
show startup-config eem	Displays information about the startup configuration for EEM.

Configuration Examples for Embedded Event Manager

The following example shows how to override the `__lcm_module_failure` system policy by changing the threshold for only module 3 hitless upgrade failures. It also sends a syslog message. The settings in the system policy, `__lcm_module_failure`, apply in all other cases.

```
event manager applet example2 override __lcm_module_failure
event module-failure type hitless-upgrade-failure module 3 count 2
  action 1 syslog priority errors msg module 3 "upgrade is not a hitless upgrade!"
  action 2 policy-default
```

The following example shows how to override the `__ethpm_link_flap` system policy and shut down the interface:

```
event manager applet ethport override __ethpm_link_flap
  event policy-default count 2 time 1000
  action 1 cli conf t
  action 2 cli int et1/1
  action 3 cli no shut
```

The following example shows how to create an EEM policy that allows the command to execute but triggers an SNMP notification when a user enters configuration mode on the device:

```
event manager applet TEST
  event cli match "conf t"
  action 1.0 snmp-trap strdata "Configuration change"
  action 2.0 event-default
```

**Note**

You must add the **event-default** action statement to the EEM policy or EEM does not allow the command to execute.

The following example shows how to correlate multiple events in an EEM policy and execute the policy based on a combination of the event triggers. In this example, the EEM policy is triggered if one of the specified syslog patterns occurs within 120 seconds.

```
event manager applet eem-correlate
  event syslog tag one pattern "copy bootflash:.* running-config.*"
  event syslog tag two pattern "copy run start"
  event syslog tag three pattern "hello"
  tag one or two or three happens 1 in 120
  action 1.0 reload module 1
```

Additional References

Related Documents

Related Topic	Document Title
EEM commands	<i>Cisco Nexus 3000 Series NX-OS System Management Command Reference</i>

Standards

There are no new or modified standards supported by this feature, and support for existing standards has not been modified by this feature.

Feature History for EEM

Table 2: Feature History for EEM

Feature Name	Release	Feature Information
EEM	5.0(3)U3(1)	Feature added.