Contents

1 Configuring SEM

1.1 Introduction

1.1.1 Overview

The smart embedded manager (SEM) is a network management tool. It is embedded in a device and can be separately deployed. The tool supports command configuration, independent from external network management tools, which makes the tool easier to deploy.

Traditional external network management tools must be connected to devices over a network to manage the devices. If a network fails, affecting network connection, the external network management tools lose effect. SEM can directly manage devices under any circumstance and troubleshoot different types of network or device faults or capture key information.

SEM detects user configuration events in real time. When an event occurs, SEM takes preset actions. The entire procedure is highly customizable. SEM provides fault detection and processing and automatic management functions, which improves the availability of the device and network.

SEM supports a variety of events, including key events of a device such as key syslog, trap and time point, or user input operations such as Command Line Interface (CLI) commands and Simple Network Management Protocol (SNMP) operations, or threshold excess such as interface statistics, SNMP object values, and system resource statistics. SEM supports many actions, including user command running, log sending, and active device reset.

1.1.2 Basic Concepts

- Event

  An event indicates an event that a user cares about. It is configured by users and attached to a policy. One or multiple events can be configured. Each type of events are detected by a specific detector. When conditions are met, corresponding event is triggered by the event detector. For example, to detect the inclusion of the **shutdown** command in the privileged EXEC mode, run the **event tag** *example* **cli pattern** *shutdown* **mode** *exec* command to configure this event to the corresponding command line event detector.

- Action

  An action indicates a measure taken after an event occurs. It is configured by users and attached to a policy. One or multiple actions can be configured. Each type of actions correspond to a type of action configurations. When an event occurs, the policy corresponding to this event runs configured actions in order. For example, to restart a device after an event occurs, run the **action** *example* **reload** command.

- Policy

  A policy organizes relationships between events and between events and actions. A policy must be submitted after it is configured. When an event corresponding to this policy meets a preset rule, this policy is triggered and runs actions in order.

- Event detector

An event detector is embedded in specific business and monitors the business based on user configuration. When the business complies with the user configuration, corresponding event is triggered. The event detector sends an event notification to the smart management server.

- Smart management server

  A smart management server receives the event notification from the event detector and determines whether to trigger a policy based on the event. After the policy is triggered, the smart management server runs the policy in policy management.

- Policy manager

  A policy manager manages SEM policies and runs actions in the policies that are triggered by the smart management server.

- SEM environmental variable

  It indicates a variable that is used by an action of a policy during SEM running. There are three types of SEM environmental variables:

  - Global variable
  - Local system variable
  - Local user variable

  The words and signs located between the United States Dollar sign ($) and the first non-letter, non-digit, and non-underline character are used as the name of a variable to be replaced. Global variables can be used in all policies. Local system variables and local user variables are local variables and can be used in specific policies only. The local system variables are read-only. The local user variables are defined by the actions in the policies during policy running. Therefore, the local system variables start with the underline (_) to avoid confusion with the local user variables.

- Specific application event of SEM

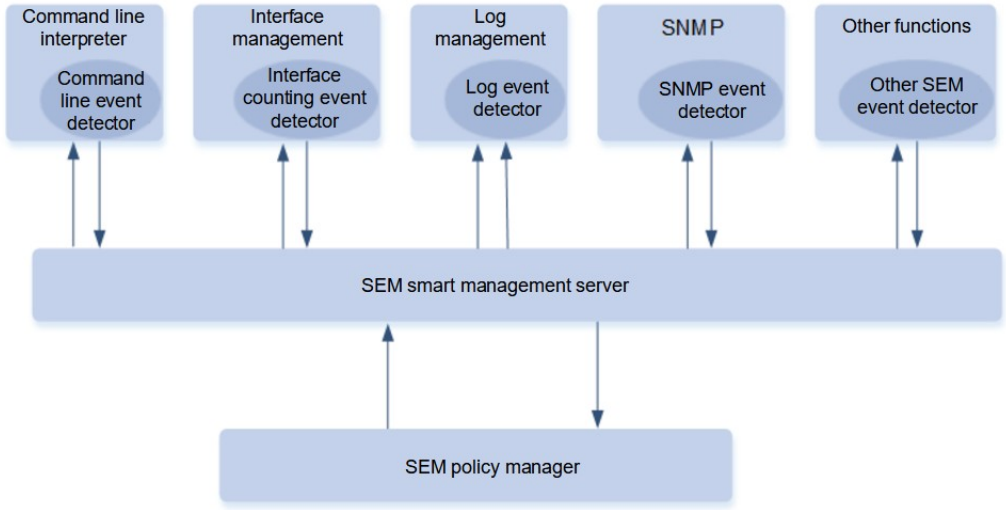
  SEM supports specific application events in the SEM system. These events are used to trigger a policy during running of another policy. Subsets and types are used to distinguish specific application events. Specific application events of SEM are detected by the specific application event detector based on subsets and types, and are released by their actions in the running policy. When an action in the running policy releases a specific application event of a subset and type, the detected event of the subset and type is triggered.

- SEM naming counter

  SEM supports a naming counter in the SEM system. Change in the SEM naming counter is detected by the counting event detector of SEM. The value of the SEM naming counter is operated by the counter action in the running policy. The SEM naming counter can be used for policy trigger counting and numerical statistics.

1.1.3 Principle

Figure 1-1 SEM Structure



As shown in Figure 1-1, SEM provides different types of event detectors. These events detectors are embedded in different business to monitor the business in real time. The detectors try to match event triggering conditions in user configurations with business running events. If a match is hit, the event detector notifies the smart management server of the event.

On the basis of the event configuration in the policy, the smart management server determines whether to trigger a policy to run. If the event triggering conditions are met, the smart management server runs the policy in the policy manager.

The policy in the policy manager runs preset actions in order based on user configuration.

1. Event Detectors Supported by SEM

SEM supports a variety of event detectors. These event detectors are embedded in different business. They detect the running business, and decide whether any event occurs. SEM supports the following detector types:

- Counting event detector

  The counting event detector detects the naming counter in SEM. When the naming counter exceeds the specified threshold, a counter event is triggered. After the counter event is triggered, the counting event detector stops working temporarily, and resumes until the counter reaches the threshold. The value of the naming counter is changed by the action of the running counter and can thus be accumulated by other policies. When the counter reaches a threshold, a counter event is triggered, which triggers the policy to run. During the running of this policy, the value of the counter is reset to realize circulation.

- Interface counting event detector

  The interface counting event detector detects interface statistics of a device. By periodic collection and statistics, the detector determines whether the statistics of an interface exceeds a threshold. If yes, an interface counting event is triggered. After the interface counting event is triggered, the interface counting event detector stops working temporarily and resumes until the interface counter reaches the recovery threshold or the suspension time exceeds the recovery cycle.

- Syslog event detector

  The syslog event detector detects log information of a device and uses regular expressions to match the log information. If the log information is matched using the regular expressions, a syslog event is triggered.

- Timer event detector

  The timer event detector detects time-related events.

  - Countdown timer event: The timer time is the number of seconds before which a policy takes effect. When the effective time reaches the number of such seconds, the timer event is triggered.

- Track event detector

  The track event detector detects tracks. Track refers to status change of a reliable network service (RNS). The track event detector detects change in the status of a tracked entity. When the status of a tracked entity changes, the track event is triggered.

2. Policy Actions Supported by SEM

SEM supports different actions. An action runs after a policy is triggered by an event. The actions include information collection and device/network troubleshooting. SEM supports the following actions:

- Run CLI commands: Run commands set by users.
- Send syslog: Send specified log messages.
- Operate naming counter: Operate the naming counter in SEM.
- Reset device: Reset devices.
- Wait during policy running: Wait for some time during policy running.
- Configure environmental variables: Configure local policy variables.

3. Management of the SEM Smart Management Server

The SEM smart management server provides management interfaces to allow users to view SEM running information and perform management. Users can view different types of SEM information, including:

- Supported detector types
- Versions of the smart management server and event detectors
- User configuration and submitted policies
- Event history
- Currently defined counter

1.2 Configuration Task Summary

SEM configuration includes the following tasks:

(1) Configuring SEM Policies

(2) (Optional) Configuring Parameters of SEM Policies

1.3 Enabling the SEM Function

1.3.1 Overview

SEM is separately deployed in a device independent from external network management tools. SEM allows customers to customize different types of events and policies so that customers can freely obtain or monitor their desired event information and actions taken for the events.

1.3.2 Restrictions and Guidelines

- The syslog function must be enabled by using the **logging** command before syslog actions are configured.
- The device must be equipped with a normal standby engine before actions are switched to the standby engine.
- Run the **action reload** command with caution. It may cause repeated restart of the system or repeated active/standby switchover.

1.3.3 Configuring SEM Policies

1. Overview

By configuring SEM policies, users can monitor events based on actual needs.

2. Restrictions and Guidelines

- When multiple events are configured for a policy, the events are automatically arranged in the alphabetical order of tags. You are advised to use the naming method based on a sequencing rule, for example, 01_cli, 02_timer, and 03_counter.
- The relationship between events refers to the relationship of the current event with the combination of all other events. Only when the first event is triggered, SEM checks whether to trigger a policy. If users try to configure juxtaposition for other events except the first event, the default juxtaposition is an AND relation. In this case, the juxtaposition of the first event is ignored.
- In either of the following scenarios, the **rollback** command can be used to roll back the policy configuration.
  - A new configured policy has not been submitted.
  - A configured policy has been submitted and registered, but changes to the policy have not been submitted.
- If no action is configured for a submitted policy, the policy can be registered. However, no action is taken when the policy is triggered.
- Generally, multiple actions are configured for a policy. When the policy is triggered, the actions are run in the alphabetical order of the *label* parameter.
- To save the I/O of CLI commands, run the **policy record** command.
  The **smart manager policy record** command does not save the command I/O as well.
- If the policy is being edited but is submitted, run the **list-config** command to display the configuration not submitted in the currently edited policy. The **commit** command is not included in the displayed content. If the policy is submitted, the configuration of the submitted policy is displayed. The **commit** command is included in the displayed content.
- The submittable policies are divided into new policies and edited and registered policies. When a registered

but unedited policy is not submitted, a notification will be displayed. If users want to quit the changes to the policy configuration before submitting the changes, run the **rollback** command to roll back the policy configuration.

- During submission, the validity of the policy configuration is checked. If the checking fails, the policy configuration fails to be submitted. In this case, the policy is not registered and remains in the editing status. For example, if no event is configured for the policy, the policy submission checking fails.

3. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure an SEM policy and enter the SEM configuration mode.

smart manager applet *applet-name*

No policy is configured by default.

(4) Configure events. Configure at least one of the following tasks.

- Configure a counter monitoring event.

  event tag *event-name* [**correlate** { **andnot** | **and** | **or** }] **counter name** *counter-name* **entry-op** *operator* **entry-val** *entry-value* **exit-op** *operator* **exit-val** *exit-value*

  No counter monitoring event is configured by default.

- Configure an interface monitoring event.

  event tag *event-name* [**correlate** { **andnot** | **and** | **or** }] **interface name** *interface-type interface-number* **parameter** { **link_up** | **link_down** }

  No interface monitoring event is configured by default.

- Configure a log monitoring event.

  event tag *event-name* [**correlate** { **andnot** | **and** | **or** }] **syslog pattern** *regular-expression* [**priority** *priority-level*] [**occurs** *num-occurrences*] [**period** *period-value*] [**skip** { **yes** | **no** }]

  No log monitoring event is configured by default.

- Configure a timer monitoring event.

  event tag *event-name* [**correlate** { **andnot** | **and** | **or** }] **timer**

  No timer monitoring event is configured by default.

- Configure a track monitoring event.

  event tag *event-name* [**correlate** { **andnot** | **and** | **or** }] **track** [**state** { **up** | **down** }] [*track-id*]

  No track monitoring event is configured by default.

A complete policy is composed of events and actions. If no event is configured for the policy, an error occurs during policy submission, resulting in submission failure.

(5) Configure actions. Configure one of the following tasks.

- Run CLI commands of a device.

action *action-label* **cli command** *cli-string* [**pattern** *pattern-string*]

- Operate SEM counters.

  action *action-label* **counter name** *counter-name* **op** { **dec** | **inc** | **nop** | **set** } **value** *counter-value*

- Terminate a policy script and set the exiting status.

  action *action-label* **exit** [*result*]

- Restart the device.

  action *action-label* **reload**

- Set local variables.

  action *action- label* **set** *variable-name variable-value*

- Carry out logging.

  action *action-label* **syslog** [**priority** *priority-level*] **msg** *syslog-message* [**facility** *mnemonics*]

- Pause a policy script.

  action *action-label* **wait** *wait-time*

No action is configured by default.

(6) Enable the function of recording CLI action output.

policy record

The recording function is not enabled for CLI command action output by default.

(7) (Optional) Configure description.

description *string*

No description is configured for an SEM policy by default.

(8) (Optional) Display current policy configuration.

list-config

(9) (Optional) Roll back configuration.

rollback

(10) Submit configuration.

commit

1.3.4 Configuring Parameters of SEM Policies

1. Restrictions and Guidelines

Modification to policy description takes effect immediately without submission.

2. Procedure

(1) Enter the privileged EXEC mode.

enable

(2) Enter the global configuration mode.

configure terminal

(3) Configure upper limits of SEM parameters. Configure at least one of the following tasks.

- Configure the upper limits of SEM detector parameters.

  smart manager detector { **counter** | **interface** | **timer** } **event-number** *detector-number*

  By default, the maximum number of events for the counting detector, the maximum number of events for the interface detector, and the maximum number of events for the timer detector are **256**, respectively.

- Configure the maximum number of SEM global variables.

  smart manager global-variant number *global-variant-number*

  By default, the maximum number of global variables is **512**.

- Configure the upper limits of SEM policy parameters.

  smart manager policy { **action-number** *policy-number* | **config-number** *policy-number* | **event-number** *policy-number* | **register-number** *policy-number* | **trigger-delay-number** *policy-number* }

  By default, the maximum number of actions in a policy is **64**; the maximum numbers of policy detectors and registered policies are **128**, respectively; and the maximum numbers of configured policies and policy delayed triggers are **256**, respectively.

- Configure the upper limits of SEM configuration instance parameters.

  smart manager record { **size-of-instance** *record-number* | **size-of-policy** *record-number* }

  By default, the maximum number of policy instances is **50** and the maximum size of a policy file is **1024** KB.

- Configure the upper limits of SEM scheduler parameters.

  smart manager schdulr { **pending-number** *schdulr-number* | **running-number** *schdulr-number* }

  By default, the maximum number of wait policies of the scheduler and the maximum number of policies run by the scheduler are **128**, respectively.

1.4 Monitoring

Run the **show** command to check the running status of a configured function to verify the configuration effect.

Table 1-1 SEM Monitoring

Command	Purpose
show smart manager detector [**all** | *detector-name*] [**statistics**]	Displays SEM detector information.
show smart manager history events [**detailed**] [**maximum** *number*]	Displays SEM history information.
show smart manager policy all	Displays all policies and policy submission information.
show smart manager policy registered [**policy** *policy-name*] [**event-type** *event-name*] [**class** *class-options*] [**statistics**]	Displays information of registered policies.
show smart manager version	Display SEM versions.

1.5 Configuration Examples

1.5.1 Configuring an Interface Monitoring Event

1. Requirements

Device A is directly connected to Device B through GigabitEthernet 0/1. Due to the problem of the line or Device B, Device A often receives many consecutive error frames, which affects communication. After GigabitEthernet 0/1 of Device A is shut down and restarted, the problem is solved.

2. Topology

Figure 1-1 Topology of an Interface Monitoring Event

3. Notes

(1) Create a policy.

(2) Configure an interface monitoring event.

(3) Configure CLI command actions.

(4) Submit the policy configuration.

4. Procedure

(1) Create a policy with the name policy_interface on Device A.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA (config)# smart manager applet policy_interface
```

(2) Configure an event with the name event_1 in the policy_interface policy and set the type to interface. The specific parameter configuration is as follows:

Configure the interface counting type as link_down and set the interface status to down.

Configure the interface name as GigabitEthernet 0/1 to specify the interface to be detected.

```
DeviceA (sem-applet)# event tag event_1 interface name GigabitEthernet 0/1
parameter link_down
```

(3) Configure actions of the policy_interface policy.

Configure action_1 to run the **enable** command to enter the privileged EXEC mode.

```
DeviceA (sem-applet)#action action_1 cli command "enable"
```

Configure action_2 to run the **configure terminal** command to enter the global configuration mode.

```
DeviceA (sem-applet)#action action_2 cli command "configure terminal"
```

Configure action_3 to run the **interface GigabitEthernet 0/1** command to enter the interface configuration mode.

```
DeviceA (sem-applet)#action action_3 cli command "interface GigabitEthernet
0/1"
```

Configure action_4 to run the **no shutdown** command to set the interface status to up.

```
DeviceA (sem-applet)#action action_4 cli command "no shutdown"
```

(4) Submit the policy.

```
DeviceA(sem-applet)# commit
```

(5) Exit the policy editing status.

```
DeviceA(sem-applet)# exit
```

5. Verification

Run the **show smart manager policy registered** command to display a registered SEM policy.

```
DeviceA# show smart manager policy registered
 No.  Name                                                              Type
Event Type
   1  policy_interface                                                  applet
interface
  event_1: interface: parameter link_down name GigabitEthernet 0/1
  action action_1 cli command "enable"
  action action_2 cli command "configure terminal"
  action action_3 cli command "interface GigabitEthernet 0/1"
  action action_4 cli command "no shutdown"
```

Run the **show smart manager policy active** command to display an active policy.

```
DeviceA# show smart manager policy active
Key: P - Priority        :L - Low, H - High, N - Normal
     S - Scheduling node :A - Active, P - Pending

No.  Job id   P S Status     Time Of Event        Event Type    Policy Name
1    2        N A suspend    2021-03-19 18:05:59  interface     policy_interface
```

Set GigabitEthernet 0/1 to the down status to verify whether the policy takes effect.

```
DeviceA# configure terminal
DeviceA (config)# interfce GigabitEthernet 0/1
DeviceA (config-if-GigabitEthernet 0/1)#shutdown
*Mar 19 18:06:19: %SEM-6-PLCY_TRG: Policy policy_1 id 2 is triggered by event 2,
and the policy instance id is 1.
```

Run the **show smart manager history events detailed** command to display SEM event history.

```
DeviceA# show smart manager history events detailed
No.      Job id   Event Type       Time                      Policy name
1        2        interface        2021-03-22 17:19:38       policy_interface
         Class: default, Policy Type: applet
```

6. Configuration Files

Device configuration file

```
hostname DeviceA
!
smart manager applet policy_interface
 event tag event_1 interface name GigabitEthernet 0/1 parameter link_down
 action action_1 cli command "enable"
 action action_2 cli command "configure terminal"
 action action_3 cli command "interface GigabitEthernet 0/1"
 action action_4 cli command "no shutdown"
!
end
```

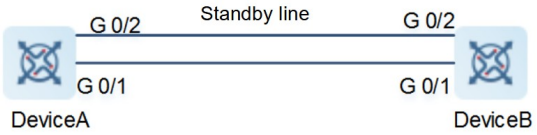
1.5.2 Configuring a Counter Monitoring Event

1. Requirements

Device A is directly connected to Device B through GigabitEthernet 0/1. A low-bandwidth standby line is created between Device A and Device B through GigabitEthernet 0/2. Due to the problem of the line or Device B, Device A often receives many consecutive error frames on GigabitEthernet 0/1, which affects communication. After GigabitEthernet 0/1 of Device A is shut down and restarted, the problem is solved. If the counter monitoring event is triggered more than 50 times, GigabitEthernet 0/1 is shut down permanently and the standby line is used.

2. Topology

Figure 1-1 Topology of a Counter Monitoring Event



3. Notes

(1) Configure an interface monitoring event, including interface counting event, CLI actions, and counter actions.

(2) Configure a counter monitoring event for detecting interface monitoring events.

4. Procedure

(1) Create a policy with the name policy_interface on Device A.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA(config)# smart manager applet policy_interface
```

(2) Configure an event with the name event_1 in the policy_interface policy and set the type to interface. The specific parameter configuration is as follows:

Configure the interface counting type as link_down and set the interface status to down.

Configure the interface name as GigabitEthernet 0/1 to specify the interface to be detected.

```
DeviceA (sem-applet)# event tag event_1 interface name GigabitEthernet 0/1
parameter link_down
```

(3) Configure actions of the policy_interface policy.

Configure action_1 to run the **enable** command to enter the privileged EXEC mode.

```
DeviceA (sem-applet)#action action_1 cli command "enable"
```

Configure action_2 to run the **configure terminal** command to enter the global configuration mode.

```
DeviceA (sem-applet)#action action_2 cli command "configure terminal"
```

Configure action_3 to run the **interface GigabitEthernet 0/1** command to enter the interface configuration mode.

```
DeviceA (sem-applet)#action action_3 cli command "interface GigabitEthernet
0/1"
```

Configure action_4 to run the **no shutdown** command to set the interface status to up.

```
DeviceA (sem-applet)#action action_4 cli command "no shutdown"
```

Configure action_5, and increase the value of the naming counter for the SEM policy name of policy_interface by 1.

```
DeviceA(sem-applet)# action action_5 counter name policy_interface op inc
value 1
```

(4) Submit the policy.

```
DeviceA(sem-applet)# commit
```

(5) Exit the policy editing status.

```
DeviceA(sem-applet)# exit
```

(6) Create a policy with the name policy_counter.

```
DeviceA(config)# smart manager applet policy_counter
```

(7) Configure an event with the name event_counter in the policy_counter policy and set the event type to SEM counter. The specific parameter configuration is as follows:

Set the counter name to policy_interface, which is the policy name in step (1) .

Set the counter trigger parameter **entry-op ge entry-val** to **6**, indicating that the event is triggered when the value of the SEM counter is equal to or greater than 6.

Set the counter restoration parameter **exit-op gt exit-val** to **5**, indicating that event monitoring is restored immediately when the value of the SEM counter is greater than 5.

```
DeviceA(sem-applet)# event tag event_counter counter name policy_interface
entry-op ge entry-val 6 exit-op gt exit-val 5
```

(8) Configure actions of the policy_counter policy.

Configure action_counter_1 to set the value of the counter policy_interface to 0.

```
DeviceA(sem-applet)# action action_counter_1 counter name policy_interface op
set value 0
```

Configure action_counter_2 to run the **enable** command to enter the privileged EXEC mode.

```
DeviceA(sem-applet)# action action_counter_2 cli command "enable"
```

Configure action_counter_3 to run the **configure terminal** command to enter the global configuration mode.

```
DeviceA(sem-applet)# action action_counter_3 cli command "configure terminal"
```

Configure action_counter_4 to run the **interface GigabitEthernet 0/1** command to enter the interface configuration mode.

```
DeviceA(sem-applet)# action action_counter_4 cli command "interface
GigabitEthernet 0/1"
```

Configure action_counter_5 to run the **shutdown** command to set the interface status to down permanently.

```
DeviceA(sem-applet)# action action_counter_5 cli command "shutdown"
```

(9) Submit the policy.

```
DeviceA(sem-applet)# commit
```

(10) Exit the policy editing status.

```
DeviceA(sem-applet)# exit
```

5. Verification

Run the **show smart manager policy registered** command to display a registered SEM policy.

```
DeviceA# show smart manager policy registered
No.  Name                                                          Type
Event Type
   1  policy_interface                                            applet
interface
  event_1: interface: parameter link_down name GigabitEthernet 0/1
  action action_1 cli command "enable"
  action action_2 cli command "configure terminal"
  action action_3 cli command "interface GigabitEthernet 0/1"
  action action_4 cli command "no shutdown"
    policy record per-instance 50 per-policy 1000

   2  policy_counter                                              applet
counter
  event_counter: counter: name policy_interface entry-op ge entry-val 6 exit-op
gt exit-val 5
  action 1 counter name policy_interface op set value 0
  action action_counter_1 cli command "enable"
  action action_counter_2 cli command "configure terminal"
  action action_counter_3 cli command "interface GigabitEthernet 0/1"
  action action_counter_4 cli command "shutdown"
  action action_counter_5 cli command "end"
```

Set GigabitEthernet 0/1 to the down status five times. Run the **show smart manager detector** command to display the monitoring event information.

```
DeviceA# show smart manager detector counter statistics
```

```
detector counter events number: 1
    event id: 3, event name: event_counter
      counter name: policy_interface
      counter value: 5
      status: enable
      trigger times: 0
      trigger policy times: 0
```

Set GigabitEthernet 0/1 to down status again to trigger the policy_counter policy, and run the **show smart manager detector** command to display the monitoring event information.

```
DeviceA# show smart manager detector counter statistics
detector counter events number: 1
    event id: 3, event name: event_counter
      counter name: policy_interface
      counter value: 1
      status: enable
      trigger times: 1
      trigger policy times: 1
```

Run the **show smart manager history events detailed** command to display SEM event history.

```
DeviceA# show smart manager history events detailed
No.      Job id   Event Type       Time                     Policy name
1        65       interface        2021-03-24 11:27:30      policy_interface
         Class: default, Policy Type: applet
2        54       counter          2021-03-24 11:27:29      policy_counter
         Class: default, Policy Type: applet
3        64       interface        2021-03-24 11:27:29      policy_interface
         Class: default, Policy Type: applet
4        63       interface        2021-03-24 11:24:43      policy_interface
         Class: default, Policy Type: applet
5        62       interface        2021-03-24 11:24:41      policy_interface
         Class: default, Policy Type: applet
6        61       interface        2021-03-24 11:24:29      policy_interface
         Class: default, Policy Type: applet
7        60       interface        2021-03-24 11:24:25      policy_interface
         Class: default, Policy Type: applet
8        59       interface        2021-03-24 11:24:07      policy_interface
         Class: default, Policy Type: applet
```

6. Configuration Files

Device A configuration file

```
hostname DeviceA
!
smart manager applet policy_interface
 event tag event_1 interface name GigabitEthernet 0/1 parameter link_down
 action action_1 cli command "enable"
 action action_2 cli command "configure terminal"
```

```
 action action_3 cli command "interface GigabitEthernet 0/1"
 action action_4 cli command "no shutdown"
 action action_5 counter name policy_interface op inc value 1
!
smart manager applet policy_counter
 event tag event_counter counter name policy_interface entry-op ge entry-val 6
exit-op gt exit-val 5
 action 1 counter name policy_interface op set value 0
 action action_counter_1 cli command "enable"
 action action_counter_2 cli command "configure terminal"
 action action_counter_3 cli command "interface GigabitEthernet 0/1"
 action action_counter_4 cli command "shutdown"
 action action_counter_5 cli command "end"
 commit
!
end
```

1.5.3 Configuring a Syslog Monitoring Event

1. Requirements

Though the interface status is down, Device A sends logs. When the logs are detected, the interface status must be restored to up.

2. Topology

Figure 1-1 Topology of a Syslog Monitoring Event

3. Notes

(1) Create a policy.

(2) Configure a syslog monitoring event.

(3) Configure actions.

(4) Configure policy trigger parameters.

(5) Submit the policy configuration.

4. Procedure

(1) Create a policy with the name policy_syslog on Device A.

```
Device> enable
DeviceA# configure terminal
DeviceA(config)# smart manager applet policy_syslog
```

(2) Configure an event with the name event_syslog1 in the policy_syslog policy, set the type to syslog, and put logs with the content "LINEPROTO-5-UPDOWN" within the detection scope.

```
DeviceA(sem-applet)# event tag event_1 syslog pattern "LINEPROTO-5-UPDOWN"
priority critical
```

(3) Configure an event with the name event_syslog2 in the policy_syslog policy, set the type to syslog, and put logs with the content "GigabitEthernet 0/1" within the detection scope.

```
DeviceA(sem-applet)# event tag event_syslog2 syslog pattern "GigabitEthernet
0/1"
```

(4) Configure an event with the name event_syslog3 in the policy_syslog policy, set the type to syslog, and put logs with the content "changed state to down" within the detection scope. The conditional relationship between event_syslog3 and event_syslog1/event_syslog2 is an AND relation.

```
DeviceA(sem-applet)# event correlate and tag event_syslog3 syslog pattern
"changed state to down"
```

(5) Configure actions of the policy_syslog policy.

Configure action_syslog_1 to run the **enable** command to enter the privileged EXEC mode.

```
DeviceA(sem-applet)# action action_syslog_1 cli command "enable"
```

Configure action_syslog_2 to run the **configure terminal** command to enter the global configuration mode.

```
DeviceA(sem-applet)# action action_syslog_2 cli command "configure terminal"
```

Configure action_syslog_3 to run the **interface GigabitEthernet 0/1** command to enter the interface configuration mode.

```
DeviceA(sem-applet)# action action_syslog_3 cli command "interface
GigabitEthernet 0/1"
```

Configure action_syslog_4 to wait for five seconds.

```
DeviceA(sem-applet)# action action_syslog_4 wait 5
```

Configure action_syslog_5 to run the **no shutdown** command to set the interface status to up.

```
DeviceA(sem-applet)# action action_syslog_5 cli command "no shutdown"
```

Configure action_syslog_6 to run the **end** command to exit the privileged EXEC mode.

```
DeviceA(sem-applet)# action action_syslog_6 cli command "end"
```

Configure action_syslog_7 to run the **show interfaces status | include UP** command to display the interface status.

```
DeviceA(sem-applet)# action action_syslog_7 cli command "show interfaces
status | include UP"
```

(6) Submit the policy.

```
DeviceA(sem-applet)# commit
```

(7) Exit the policy editing status.

```
DeviceA(sem-applet)# exit
```

5. Verification

Run the **show smart manager policy registered** command to display a registered SEM policy.

```
DeviceA# show smart manager policy registered
 No.  Name                                                                Type
Event Type
   1  policy_syslog                                                       applet
syslog
  event_syslog1: syslog: pattern "LINEPROTO-5-UPDOWN"
  event_syslog2: syslog: pattern "GigabitEthernet 0/1"
  event_syslog3: syslog: pattern "changed state to down"
  action action_syslog_1 cli command "enable"
  action action_syslog_2 cli command "configure terminal"
  action action_syslog_3 cli command "interface GigabitEthernet 0/1"
  action action_syslog_4 wait 5
  action action_syslog_5 cli command "no shutdown"
  action action_syslog_6 cli command "end"
  action action_syslog_7 cli command "show interfaces status | include UP"
```

Run the **show smart manager policy active** command to display an active policy.

```
DeviceA# show smart manager policy active
Key: P - Priority        :L - Low, H - High, N - Normal
     S - Scheduling node :A - Active, P - Pending

No.  Job id   P S Status     Time Of Event        Event Type    Policy Name
1    10       N A suspend    2021-03-23 11:01:49  syslog        policy_syslog
```

Set GigabitEthernet 0/1 to the down status to verify whether the policy takes effect.

```
DeviceA# configure terminal
DeviceA(config)# interfce GigabitEthernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)#shutdown
*Mar 23 11:02:03: %LINK-5-CHANGED: Interface GigabitEthernet 0/1, changed state
to administratively down.
*Mar 23 11:02:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet
0/1, changed state to down.
*Mar 23 11:02:03: %SEM-6-PLCY_TRG: Policy policy_syslog id 7 is triggered by
event 10, and the policy instance id is 12.
*Mar 23 11:02:11: %LINK-3-UPDOWN: Interface GigabitEthernet 0/1, changed state to
up.
*Mar 23 11:02:11: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet
0/1, changed state to up.
```

Run the **show smart manager history events detailed** command to display SEM event history.

```
DeviceA# show smart manager history events detailed
No.      Job id   Event Type       Time                      Policy name
1        10       syslog           2021-03-23 11:02:03       policy_syslog
         Class: default, Policy Type: applet
```

6. Configuration Files

```
hostname DeviceA
!
```

```
smart manager applet policy_syslog
 event tag event_syslog1 syslog pattern "LINEPROTO-5-UPDOWN"
 event tag event_syslog2 syslog pattern "GigabitEthernet 0/1"
 event correlate and tag event_syslog3 syslog pattern "changed state to down"
 action action_syslog_1 cli command "enable"
 action action_syslog_2 cli command "configure terminal"
 action action_syslog_3 cli command "interface GigabitEthernet 0/1"
 action action_syslog_4 wait 5
 action action_syslog_5 cli command "no shutdown"
 action action_syslog_6 cli command "end"
 action action_syslog_7 cli command "show interfaces status | include UP"
 commit
!
end
```

1.5.4 Configuring a Track Monitoring Event

1. Requirements

To facilitate management, a log is printed when the status of a specified or random tracked entity of Device A changes.

2. Topology

Figure 1-1 Topology of a Track Monitoring Event

3. Notes

(1) Create a policy.

(2) Configure a track monitoring event.

(3) Configure actions.

(4) Submit the policy configuration.

4. Procedure

(1) Create a policy with the name policy_track on Device A.

```
DeviceA> enable
DeviceA# configure terminal
DeviceA(config)# smart manager applet policy_track
```

(2) Configure an event with the name event_track in the policy_track policy, set the type to track, and detect all tracked objects in the down status.

```
DeviceA(sem-applet)# event tag event_track track state down
```

(3) Configure actions of the policy_track policy.

Configure action_track1 to print "track obj down".

```
DeviceA(sem-applet)# action action_tarck1 syslog msg “track obj down.”
```

Configure action_track2 to run the **enable** command to enter the privileged EXEC mode.

```
DeviceA(sem-applet)# action action_track2 cli command "enable"
```

Configure action_track3 to run the **configure terminal** command to enter the global configuration mode.

```
DeviceA(sem-applet)# action action_track3 cli command "configure terminal"
```

Configure action_track4 to run the **interface GigabitEthernet 0/1** command to enter the interface configuration mode.

```
DeviceA(sem-applet)# action action_track4 cli command "interface
GigabitEthernet 0/1"
```

Configure action_track5 to run the **no shutdown** command to set the interface status to up.

```
DeviceA(sem-applet)# action action_track5 cli command "no shutdown"
```

(4) Submit the policy.

```
DeviceA(sem-applet)# commit
```

(5) Exit the policy editing status.

```
DeviceA(sem-applet)# exit
```

(6) Configure a track monitoring event.

```
DeviceA(config)# track 1 Interface GigabitEthernet 0/1 line-protocol
DeviceA(config-track)# delay down 10
```

5. Verification

Run the **show smart manager policy registered** command to display a registered SEM policy.

```
DeviceA# show smart manager policy registered
No.  Name                                                                Type
Event Type
  1  policy_track                                                        applet
track
  event_track: track: all state down
  action action_track1 syslog msg "track obj down"
  action action_track2 cli command "enable"
  action action_track3 cli command "configure terminal"
  action action_track4 cli command "interface GigabitEthernet 0/1"
  action action_track5 cli command "no shutdown"
```

Run the **show smart manager policy active** command to display an active policy.

```
DeviceA# show smart manager policy active
Key: P - Priority        :L - Low, H - High, N - Normal
     S - Scheduling node :A - Active, P - Pending

No.  Job id   P S Status     Time Of Event        Event Type    Policy Name
1    4        N A suspend    2021-03-22 17:21:49  track         policy_track
```

Set GigabitEthernet 0/1 to the down status to verify whether the policy takes effect.

```
DeviceA# configure terminal
DeviceA(config)# interface gigabitEthernet 0/1
DeviceA(config-if-GigabitEthernet 0/1)# shutdown
DeviceA(config-if-GigabitEthernet 0/1)# end
*Mar 22 17:21:59: %SEM-6-PLCY_TRG: Policy track id 4 is triggered by event 4, and the policy instance id is 8.
*Mar 22 17:21:59: %SEM-6-COMM: track: track obj down
*Mar 22 17:22:01: %LINK-3-UPDOWN: Interface GigabitEthernet 0/1, changed state to up.
*Mar 22 17:22:01: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet 0/1, changed state to up.
```

Run the **show smart manager history events detailed** command to display SEM event history.

```
DeviceA# show smart manager history events detailed
No.      Job id   Event Type       Time                      Policy name
1        4        track            2021-03-22 17:21:59       track
         Class: default, Policy Type: applet
```

6. Configuration Files

Device A configuration file

```
hostname DeviceA
!
smart manager applet policy_track
 event tag event_track track state down
action action_track1 syslog msg "track obj down"
 action action_track2 cli command "enable"
 action action_track3 cli command "configure terminal"
 action action_track4 cli command "interface GigabitEthernet 0/1"
 action action_track5 cli command "no shutdown"
commit
!
track 1 Interface GigabitEthernet 0/1 line-protocol
 delay down 10
!
end
```