ASA Embedded Event Manager Configuration Example

Document ID: 117883

Contributed by Cisco TAC Engineers.
Jul 16, 2014

Contents

Introduction

This document describes Embedded Event Manager (EEM), which is a troubleshooting tool that was added in Adaptive Security Appliance (ASA) Version 9.2(1). The functionality is similar to Cisco IOS® based EEM. It is a powerful way to run CLI commands based on ASA events (syslogs) and save the output. This document covers an introduction to the feature as well as some example EEM applets.

Prerequisites

Requirements

The use of EEM requires that the ASA is configured in single context mode.

Components Used

The information in this document is based on ASA Version 9.2(1) or later.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

Context Mode Guidelines

EEM is currently only supported on ASA firewalls that run in single context mode. Firewalls configured in multiple context mode are not currently supported.

Firewall Mode Guidelines

EEM is currently supported in both routed and transparent firewall modes.

Additional Guidelines

- While the unit crashes, the state of the ASA is generally unknown. Some commands might not be safe to run while the ASA is in this condition.
- The name of an event manager applet cannot contain spaces.
- You cannot modify the None event and Crashinfo event parameters.
- Performance might be affected because syslog messages are sent to the EEM to be processed.
- The default output is ***output none*** for each event manager applet. In order to change the default output, you must enter a different output value.
- You might have only one output option defined for each event manager applet.

Configure

The ***event manager applet*** command creates/edits an event manager applet, a process that links events with actions and output. The <*name*> is limited to 32 characters and cannot have spaces. This enters an event manager applet submode.

```
ASA(config)# [no] event manager applet <name>
```

A ***description*** can be added to an applet. This is for informational purposes only. The <*text*> is limited to 256 characters.

```
ASA(config-applet)# [no] description <text>
```

Event Configuration

Various events might be added to an applet that trigger the applet to invoke the actions that are configured on it. They are defined with the ***event*** keyword. Multiple events might be configured for each applet.

Syslog Events

The first event type that is supported is ***syslog***. The ASA uses syslog IDs in order to identify syslogs that trigger an applet. This is completed through the id keyword, which might be a single syslog or a range. The optional ***occurs*** keyword indicates the number of times that the syslog must occur for the applet to be invoked (default is 1). The optional ***period*** keyword indicates the amount of time, in seconds, that the event must occur in. It limits the frequency of the applet invocation to at most once the configured period. An ***occurs*** of 5 with a ***period*** of 30, means the syslog must occur 5 times within 30 seconds before the event is triggered. If the syslog occurs 11 times in 30 seconds, the applet is only triggered once. A value of 0 for ***period*** means that no period is defined.

Multiple syslogs can be configured, but the ranges cannot overlap.

```
ASA(config−applet)# [no] event syslog id <nnnnnn>[−<nnnnnn>] [occurs <n>]
 [period <seconds>]ASA(config−applet)# no event syslog id <nnnnnn>[−<nnnnnn>]
```

The ***occurs*** value <*n*> has an allowable range of 1 to 4294967295. The ***period*** value <seconds> has an allowable range of 0 to 604800. A 0 (zero) value means no period is configured.

Syslog Events Example

In this example, EEM takes action when it detects a low memory block condition. If the available 1550 byte blocks become depleted, it gathers ***show blocks pool 1550 dump*** and saves to the disk. It does this, at most, once every 10 minutes.

```
event manager applet depletedblock
 description "Take a snapshot of block output when it is depleted"
 event syslog id 321007 period 600
 action 1 cli command "show blocks pool 1550 dump"
 output file rotate 10
```

Periodic Events

EEM can also be configured to do an action periodically. When you configure a timer−based event, use the ***timer*** keyword in the events configuration. There are 3 timer based options:

- absolute − The first timer is an ***absolute*** timer that triggers the applet once per day at the specified time and automatically restarts.

  ```
  ASA(config−applet)# [no] event timer absolute time <hh:mm:ss>
  ASA(config−applet)# no event timer absolute
  ```
- countdown − The second timer is a ***countdown*** timer that triggers the applet once and does not restart unless removed and re−added.

  ```
  ASA(config−applet)# [no] event timer countdown time <seconds>
  ASA(config−applet)# no event timer countdown
  ```
- watchdog − The third timer is a ***watchdog*** timer that triggers the applet once per configured period and restarts automatically.

  ```
  ASA(config−applet)# [no] event timer watchdog time <seconds>
  ASA(config−applet)# no event timer watchdog
  ```

Periodic Events Example

For example, this event configuration pings 192.168.1.100 every 1 minute. This could be used to ensure a VPN tunnel is kept up and operational even during periods of idle traffic. It uses the ***watchdog*** timer to execute every 60 seconds.

```
event manager applet period−event
 description "Run a command once per minute"
 event timer watchdog time 60
 action 0 cli command "ping 192.168.1.100"
 output none
```

This applet records memory block allocation information every hour and writes the output to a rotating set of log files, since it keeps a day's worth of logs. It uses the ***watchdog*** timer to execute every 1 hour.

```
event manager applet blockcheck
 description "Log block usage"
```

```
 event timer watchdog time 3600
 output rotate 24
 action 1 cli command "show blocks old"
```

These applets disable the given interface (Gig 0/0) between midnight and 3 a.m. It uses the ***absolute*** timer to execute once per day.

```
event manager applet disableintf
 description "Disable the interface at midnight"
 event timer absolute time 0:00:00
 output none
 action 1 cli command "interface GigabitEthernet 0/0"
 action 2 cli command "shutdown"
 action 3 cli command "write memory"
!
event manager applet enableintf
 description "Enable the interface at 3am"
 event timer absolute time 3:00:00
 output none
 action 1 cli command "interface GigabitEthernet 0/0"
 action 2 cli command "no shutdown"
 action 3 cli command "write memory"
```

Manual Event

These EEM applets might also be invoked manually. In order to do this, the applet must configure ***event none***. In order to run an applet manually, enter the ***event manager run*** command followed by the name of the applet. If the applet is configured for any event trigger mechanism aside from 'none', the attempt to run it manually generates an error. With the use of one of the previous examples, 'depletedblock', you see:

```
ASA# event manager run depletedblock
ERROR: Applet not configured with 'event none'
```

Manual Event Example

Manual events can be used in a similar fashion to a macro. For example, a manual event could be used to execute a few commands in order. In this example, it saves the configuration, pings a host, and clears all shuns.

```
event manager applet clean−up
 event none
 action 0 cli command "write mem"
 action 1 cli command "ping 192.168.1.100"
 action 2 cli command "clear shun"
 output none
```

Crash Event

The ***crashinfo*** event triggers an applet when a crash occurs on the ASA. Regardless of the value of the ***output*** command, the ***action*** commands are directed to the crashinfo file. The output is generated before the ***show tech*** portion of the crashinfo is generated.

Warning: When the ASA is crashing, the state of the box is generally unknown. Some CLI commands might not be safe to run when the unit is in this condition.

```
ASA(config−applet)# [no] event crashinfo
```

Action Configuration

When the applet is triggered, the actions on the applet are performed. Each ***action*** has an ordinal that is used to specify the order of the actions. Multiple actions can be configured per applet; but each ordinal can only be used once. The commands are typical CLI commands, such as ***show blocks***. The quotes are strongly recommended, but are not required.

```
ASA(config-applet)# [no] action <n> cli command "<command>"ASA(config-applet)# no action <n>
```

The value of the action identifier <*n*> has a range of 0 to 4294967295. The value of the <*command*> must be quoted, otherwise an error occurs if the command consists of more than one word. The command is executed in configuration mode as a user with privilege level 15 (the highest). The command might not accept any input; as input will be disabled if a command has the ***noconfirm*** option. That should be used since the commands are not processed interactively.

Output Configuration

The output from the actions can be directed to a specified location via the ***output*** command. Only one output value can be enabled at any one time. The default value is ***output none***. This value discards any output from the action commands.

```
ASA(config-applet)# [no] output none
```

The ***output console*** command sends the output of the action commands to the console.

```
ASA(config-applet)# [no] output console
```

The ***output file*** command directs the output of the action commands to files. There are four options that can be used. The ***new*** option writes the output of the applet to a new file for each invocation. The *filename* has the format of ***eem−<applet>−<timestamp>.log***. Where <*applet*> is the name of the applet and <*timestamp*> is a dated timestamp in the format of *YYYYMMDD−hhmmss*.

```
ASA(config-applet)# [no] output file new
```

The ***rotate*** option is used to create a set of files that are rotated similar to Linux's log rotate mechanism. The filename format is ***eem−<applet>−<x>.log***. Where <*applet*> is the name of the applet, and <*x*> is the file number. The newest file is indicated by number 0 (zero), and the oldest file is indicated by the highest number (<*n*>−1). When a new file is to be written, the oldest file is deleted and all the subsequent files are renumbered before the 0th file is written.

```
ASA(config-applet)# [no] output file rotate <n>
```

The rotate value <*n*> has a range of 2 to 100.

The ***overwrite*** option is used to always write the action command output to a single file that is truncated every time.

```
ASA(config-applet)# [no] output file overwrite <filename>
```

The ***append*** option is used to always write the action command output to a single file, but that file is appended to every time.

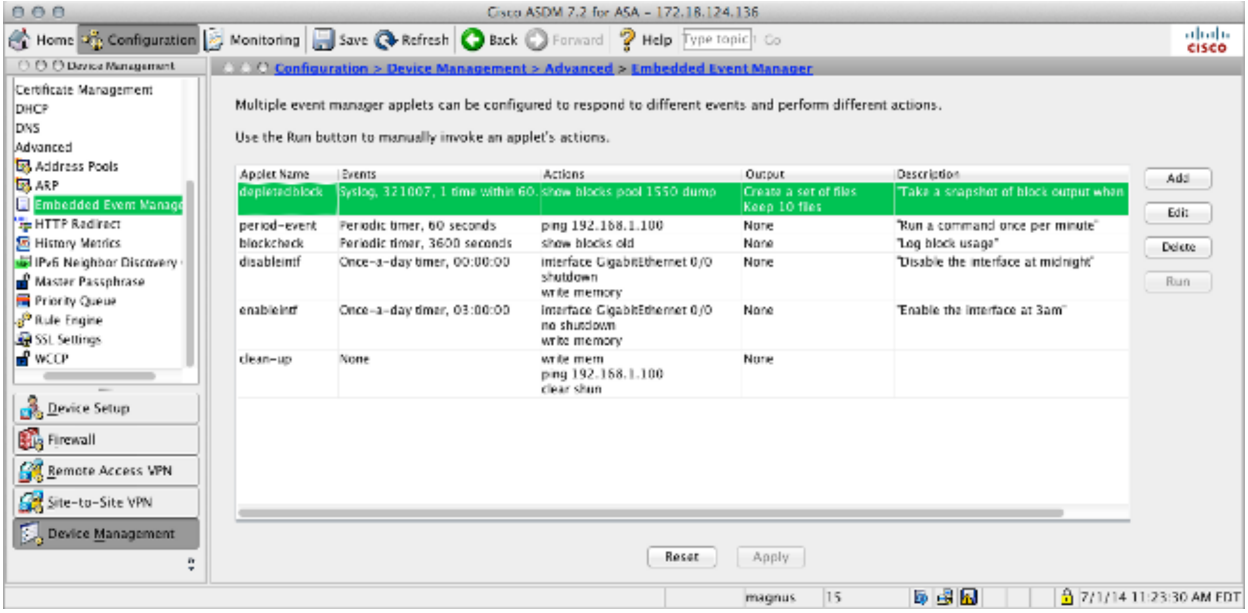
```
ASA(config-applet)# [no] output file append <filename>
```

The <*filename*> argument is a local (to the ASA) file name. The overwrite command might also use ***ftp***:, ***tftp:***

and ***smb:*** targeted files.

ASDM Configuration

EEM can also be configured from within ASDM. Choose ***Configuration*** > ***Device Management*** > ***Advanced*** > ***Embedded Event Manager***. In this section of ASDM, you can configure your EEM applets with the same parameters discussed previously. After you configure an applet, click ***Apply*** to push the configuration to the ASA.



Multiple event manager applets can be configured to respond to different events and perform different actions.

Use the Run button to manually invoke an applet's actions.

Applet Name	Events	Actions	Output	Description
depletedblock	Syslog, 321007, 1 time within 60.	show blocks pool 1550 dump	Create a set of files Keep 10 files	"Take a snapshot of block output when
period-event	Periodic timer, 60 seconds	ping 192.168.1.100	None	"Run a command once per minute"
blockcheck	Periodic timer, 3600 seconds	show blocks old	None	"Log block usage"
disableintf	Once-a-day timer, 00:00:00	interface GigabitEthernet 0/0 shutdown write memory	None	"Disable the interface at midnight"
enableintf	Once-a-day timer, 03:00:00	interface GigabitEthernet 0/0 no shutdown write memory	None	"Enable the interface at 3am"
clean-up	None	write mem ping 192.168.1.100 clear shun	None	

Verify

Exec Mode Commands

Use this section to confirm that your configuration works properly.

All of these commands are used in exec mode.

This command shows the running configuration of the event manager system.

```
ASA# show running-config event manager
```

This command executes an event manager applet that has been configured with ***event none***. If you run an applet that has not been configured with ***event none***, an error is reported.

```
ASA# event manager run <applet>
```

This command shows information about the configured applets, which includes hit counts and when the applet was last invoked

```
ASA# event manager applet period-event, hits 1, last 2014/07/01 10:51:52
  last file none
  event watchdog 60 secs, left 54 secs, hits 1, last 2014/07/01 10:51:52
  action 0 cli command "ping 192.168.1.100", hits 1, last 2014/07/01 10:51:52
```

Event manager uses the standard counters. Due to limitations within the ***show counter*** CLI, the ***eem*** keyword is used for protocol filtering.

```
ASA# show counters protocol eem
```

The Output Interpreter Tool (registered customers only) supports certain ***show*** commands. Use the Output Interpreter Tool in order to view an analysis of ***show*** command output.

Debug

Enter these commands in order to debug the EEM and display the output.

Note: Refer to Important Information on Debug Commands before you use ***debug*** commands.

```
ASA# [no] debug event manager <n>
ASA# show debug event manager
```

Troubleshoot

There is currently no specific troubleshooting information available for this configuration. If it does not operate as expected, use the debugging and verification steps listed in the previous section in order to determine if an error has occured.