Exemple de configuration du gestionnaire d'événements intégré ASA

Contenu

Introduction

Ce document décrit Embedded Event Manager (EEM), qui est un outil de dépannage ajouté dans Adaptive Security Appliance (ASA) Version 9.2(1). La fonctionnalité est similaire à Cisco IOS ? basé sur EEM. Il s'agit d'un moyen efficace d'exécuter des commandes CLI basées sur des événements ASA (syslogs) et d'enregistrer les résultats. Ce document couvre une introduction à la fonctionnalité ainsi que quelques exemples d'applets EEM.

Conditions préalables

Conditions requises

L'utilisation du module EEM nécessite que l'ASA soit configuré en mode de contexte unique.

Components Used

Les informations de ce document sont basées sur la version 9.2(1) ou ultérieure d'ASA.

Directives et limitations

Cette section présente les directives et les limites de cette fonction.

Directives du mode contextuel

Actuellement, EEM est uniquement pris en charge sur les pare-feu ASA qui fonctionnent en mode de contexte unique. Les pare-feu configurés en mode de contexte multiple ne sont pas actuellement pris en charge.

Directives relatives au mode pare-feu

EEM est actuellement pris en charge dans les modes de pare-feu routé et transparent.

Directives supplémentaires

- Lorsque l'unité tombe en panne, l'état de l'ASA est généralement inconnu. Certaines commandes peuvent ne pas être exécutées en toute sécurité lorsque l'ASA est dans cette situation.
- Le nom d'une applet de gestionnaire d'événements ne peut pas contenir d'espaces.
- Vous ne pouvez pas modifier les paramètres d'événement None et Crashinfo.
- Les performances peuvent être affectées, car les messages syslog sont envoyés à l'EEM pour être traités.
- Le résultat par défaut est **aucun** pour chaque applet du gestionnaire d'événements. Pour modifier la sortie par défaut, vous devez entrer une valeur de sortie différente.
- Il se peut qu'une seule option de sortie soit définie pour chaque applet du gestionnaire d'événements.

Configuration

La commande **event manager applet** crée/modifie une applet event manager, un processus qui lie les événements aux actions et à la sortie. Le *<nom>* est limité à 32 caractères et ne peut pas contenir d'espaces. Ceci entre dans un sous-mode applet du gestionnaire d'événements.

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

Une **description** peut être ajoutée à une applet. Il n'a qu'un but informatif. Le <*text*> est limité à 256 caractères.

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

Configuration des événements

Plusieurs événements peuvent être ajoutés à une applet qui déclenche l'applet pour appeler les actions qui y sont configurées. Ils sont définis avec le mot clé **event**. Plusieurs événements peuvent être configurés pour chaque applet.

Événements Syslog

Le premier type d'événement pris en charge est **syslog**. L'ASA utilise des ID Syslog afin d'identifier les Syslogs qui déclenchent une applet. Ceci est effectué par le mot clé id, qui peut être un syslog unique ou une plage. Le mot clé facultatif **se produit** indique le nombre de fois que le syslog doit se produire pour que l'applet soit appelée (la valeur par défaut est 1). Le mot clé **période** facultatif indique la durée, en secondes, pendant laquelle l'événement doit se produire. Elle limite la fréquence d'appel de l'applet à une seule fois la période configurée. Une **occurrence** de 5 avec une **période** de 30 signifie que le syslog doit se produire 5 fois dans les 30 secondes avant le déclenchement de l'événement. Si le syslog se produit 11 fois en 30 secondes, l'applet n'est déclenché qu'une seule fois. Une valeur de 0 pour **la période** signifie qu'aucune période n'est définie.

Plusieurs Syslogs peuvent être configurés, mais les plages ne peuvent pas se chevaucher.

```
ASA(config-applet)# [no] event syslog id
```

```
ASA(config-applet)# no event syslog id
```

La valeur **de** la **occurrence** <*n*> a une plage autorisée comprise entre 1 et 4294967295. La valeur de **période** <seconds> a une plage autorisée de 0 à 604800. Une valeur 0 (zéro) signifie qu'aucun point n'est configuré.

Exemple d'événements Syslog

Dans cet exemple, EEM prend des mesures lorsqu'il détecte une condition de bloc de mémoire insuffisante. Si les blocs de 1 550 octets disponibles sont épuisés, il collecte **show blocks pool 1550 dump** et les enregistre sur le disque. Il le fait, au plus une fois toutes les 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
```

Événements périodiques

Le module EEM peut également être configuré pour effectuer une action périodique. Lorsque vous configurez un événement basé sur le minuteur, utilisez le mot clé **timer** dans la configuration des événements. Il existe trois options basées sur le compteur :

- absolute - Le premier minuteur est un minuteur **absolu** qui déclenche l'applet une fois par jour à l'heure spécifiée et redémarre automatiquement.
  ```
  ASA(config-applet)# [no] event timer absolute time
  ```

  ```
  ASA(config-applet)# no event timer absolute
  ```
- compte à rebours - Le second compteur est un compteur **à** rebours qui déclenche l'applet une fois et ne redémarre que si supprimé et réajouté.
  ```
  ASA(config-applet)# [no] event timer countdown time
  ```

  ```
  ASA(config-applet)# no event timer countdown
  ```
- watchdog - Le troisième minuteur est un minuteur **watchdog** qui déclenche l'applet une fois par période configurée et redémarre automatiquement.
  ```
  ASA(config-applet)# [no] event timer watchdog time
  ```

  ```
  ASA(config-applet)# no event timer watchdog
  ```

Exemple d'événements périodiques

Par exemple, cette configuration d'événement envoie une requête ping à 192.168.1.100 toutes les 1 minute. Ceci pourrait être utilisé pour s'assurer qu'un tunnel VPN est maintenu et opérationnel même pendant les périodes de trafic inactif. Il utilise le minuteur **de surveillance** pour s'exécuter toutes les 60 secondes.

```
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
```

Cette applet enregistre les informations d'allocation de bloc de mémoire toutes les heures et écrit la sortie dans un ensemble rotatif de fichiers journaux, puisqu'elle conserve une journée de journaux. Il utilise le minuteur **de surveillance** pour s'exécuter toutes les 1 heure.

```
event manager applet blockcheck
description "Log block usage"
event timer watchdog time 3600
output rotate 24
action 1 cli command "show blocks old"
```

Ces applets désactivent l'interface donnée (Gig 0/0) entre minuit et 3h du matin. Il utilise le minuteur **absolu** pour s'exécuter une fois par jour.

```
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"
```

Événement manuel

Ces applets EEM peuvent également être appelées manuellement. Pour ce faire, l'applet doit configurer **event none**. Afin d'exécuter une applet manuellement, entrez la commande **d'exécution du gestionnaire d'événements** suivie du nom de l'applet. Si l'applet est configuré pour un mécanisme de déclenchement d'événement en dehors de 'none', la tentative d'exécution manuelle génère une erreur. Avec l'utilisation de l'un des exemples précédents, 's'est appauvri', vous voyez :

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

Exemple d'événement manuel

Les événements manuels peuvent être utilisés de la même manière qu'une macro. Par exemple, un événement manuel peut être utilisé pour exécuter quelques commandes dans l'ordre. Dans cet exemple, il enregistre la configuration, envoie une requête ping à un hôte et efface tous les désactivations.

```
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
```

Événement de panne

L'événement **crashinfo** déclenche une applet lorsqu'une panne se produit sur l'ASA. Quelle que soit la valeur de la commande **output**, les commandes **action** sont dirigées vers le fichier crashinfo. La sortie est générée avant que la partie **show tech** de crashinfo ne soit générée.

> **Avertissement :** Lorsque l'ASA est en panne, l'état de la boîte est généralement inconnu. Certaines commandes CLI peuvent ne pas être exécutées en toute sécurité lorsque l'unité est dans cette condition.

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

Configuration de l'action

Lorsque l'applet est déclenché, les actions de l'applet sont exécutées. Chaque **action** a un ordinal qui est utilisé pour spécifier l'ordre des actions. Plusieurs actions peuvent être configurées par applet ; mais chaque ordinal ne peut être utilisé qu'une seule fois. Les commandes sont des commandes CLI typiques, telles que **show blocks**. Les devis sont fortement recommandés, mais ne sont pas obligatoires.

```
ASA(config-applet)# [no] action
```

```
ASA(config-applet)# no action
```

La valeur de l'identificateur d'action *<n>* est comprise entre 0 et 4294967295. La valeur de la *<commande>* doit être citée, sinon une erreur se produit si la commande se compose de plusieurs mots. La commande est exécutée en mode de configuration en tant qu'utilisateur avec un niveau de privilège 15 (le plus élevé). La commande peut ne pas accepter d'entrée ; comme entrée sera désactivée si une commande a l'option **noconfirm**. Cela doit être utilisé car les commandes ne sont pas traitées de manière interactive.

Configuration de sortie

La sortie des actions peut être dirigée vers un emplacement spécifié via la commande **output**. Une seule valeur de sortie peut être activée à la fois. La valeur par défaut est **output none**. Cette valeur ignore tout résultat des commandes action.

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

La commande **output console** envoie le résultat des commandes action à la console.

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

La commande **output file** dirige le résultat des commandes action vers les fichiers. Quatre options peuvent être utilisées. La **nouvelle** option écrit la sortie de l'applet dans un nouveau fichier pour chaque appel. Le *nom de fichier* a le format **eem-*<applet>*-*<timestamp>*.log**. Où *<applet>* est le nom de l'applet et *<timestamp>* est un horodatage daté au format *AAAA-MJJ-hmss*.

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

L'option **rotate** est utilisée pour créer un ensemble de fichiers qui sont tournés comme le mécanisme de rotation du journal de Linux. Le format du nom de fichier est **eem-*<applet>*-*<x>*.log**. Où *<applet>* est le nom de l'applet et *<x>* le numéro de fichier. Le fichier le plus récent est indiqué par le numéro 0 (zéro) et le fichier le plus ancien par le numéro le plus élevé (*<n>*-1). Lorsqu'un nouveau fichier doit être écrit, le fichier le plus ancien est supprimé et tous les fichiers suivants sont renumérotés avant l'écriture du 0e fichier.

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

La valeur de rotation *<n>* est comprise entre 2 et 100.

L'option **overwrite** est utilisée pour toujours écrire la sortie de la commande action dans un fichier unique tronqué à chaque fois.

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

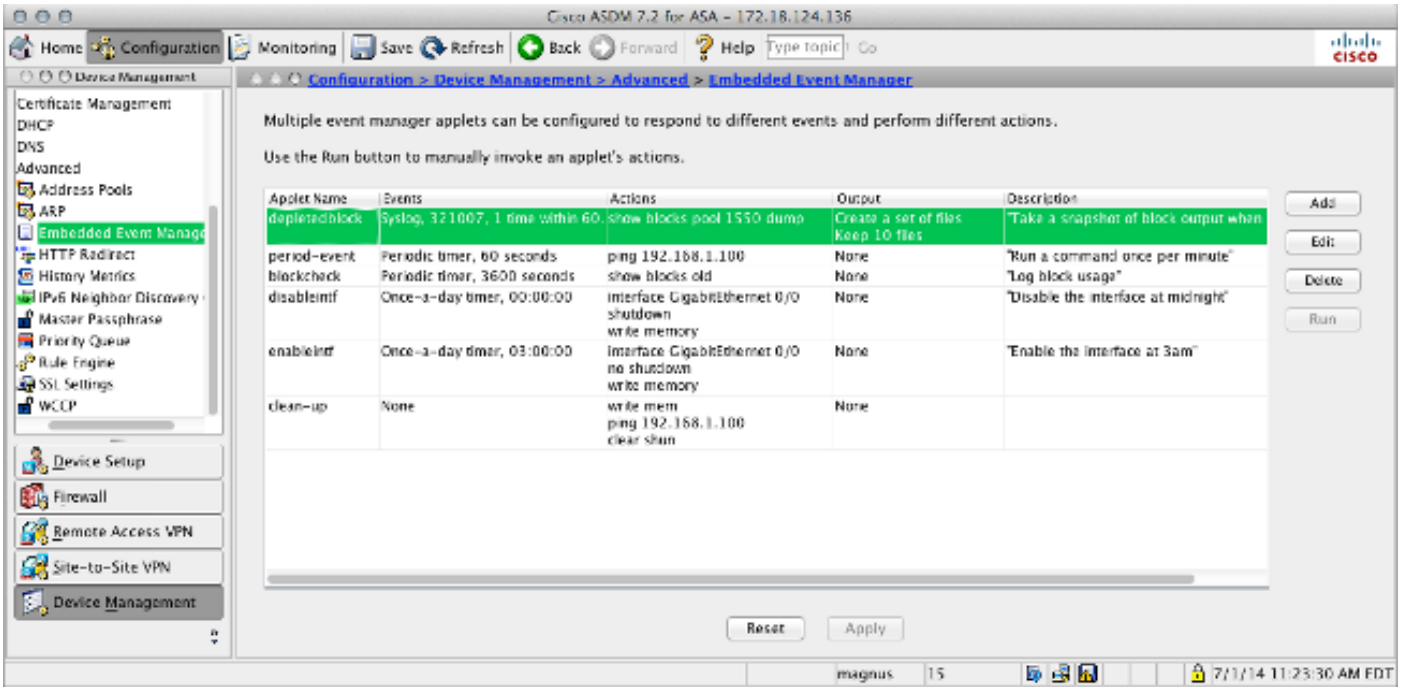
L'option **append** est utilisée pour toujours écrire la sortie de la commande action dans un seul fichier, mais ce fichier est ajouté à chaque fois.

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

L'argument *<filename>* est un nom de fichier local (à l'ASA). La commande overwrite peut également utiliser **ftp** :, **tftp:** et **smb :** fichiers ciblés.

Configuration ASDM

EEM peut également être configuré à partir de l'ASDM. Choisissez **Configuration > Device Management > Advanced > Embedded Event Manager**. Dans cette section de l'ASDM, vous pouvez configurer vos applets EEM avec les mêmes paramètres que ceux décrits précédemment. Après avoir configuré une applet, cliquez sur **Apply** pour transmettre la configuration à l'ASA.



Vérification

Commandes du mode d’exécution

Référez-vous à cette section pour vous assurer du bon fonctionnement de votre configuration.

Toutes ces commandes sont utilisées en mode d’exécution.

Cette commande affiche la configuration en cours du système du gestionnaire d'événements.

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

Cette commande exécute une applet du gestionnaire d'événements qui a été configurée avec **event none**. Si vous exécutez une applet qui n'a pas été configurée avec **event none**, une erreur est signalée.

ASA# **event manager run**

Cette commande affiche des informations sur les applets configurés, qui incluent le nombre de coups et le moment où l'applet a été appelé pour la dernière foisASA# 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 10:51:52Le gestionnaire d'événements utilise les compteurs standard. En raison de limitations dans l'interface de ligne de commandeshow counter, le mot clé eem est utilisé pour le filtrage de protocole.

ASA# show counters protocol eem L'Outil d'interprétation de sortie (clients enregistrés seulement) prend en charge certaines commandes d'affichage. Utilisez l'Outil d'interprétation de

sortie afin de visualiser une analyse de commande d'affichage de sortie .DéboguerEntrez ces commandes afin de déboguer le module EEM et afficher le résultat.Note: Référez-vous aux informations importantes sur les commandes de débogage avant d'utiliser les commandes de débogage.

```
ASA# [no] debug event manager
```

ASA# show debug event managerDépannageIl n'existe actuellement aucune information de dépannage spécifique pour cette configuration. S'il ne fonctionne pas comme prévu, utilisez les étapes de débogage et de vérification répertoriées dans la section précédente afin de déterminer si une erreur s'est produite.