



Automatisation de la réponse pour les événements système

Ce chapitre décrit comment configurer le gestionnaire d'événements intégré (EEG)

- [À propos de l'EEG, à la page 1](#)
- [Lignes directrices pour l'EEG, à la page 3](#)
- [Configurer l'EEG, à la page 3](#)
- [Exemples d'EEG, à la page 11](#)
- [Supervision de l'EEG, à la page 12](#)
- [Historique de l'EEG, à la page 12](#)

À propos de l'EEG

Le service EEG vous permet de déboguer les problèmes et fournit une journalisation à usage général pour le dépannage. Il y a deux composants : les événements auxquels l'EEG répond ou écoute, et les applets de gestionnaire d'événements qui définissent les actions ainsi que les événements auxquels l'EEG répond. Vous pouvez configurer plusieurs applets de gestionnaire d'événements pour répondre à différents événements et effectuer des actions différentes.

Événements pris en charge

L'EEG prend en charge les événements suivants :

- **Journal système** : l'ASA utilise les ID des messages de journal système pour identifier les messages de journal système qui déclenchent un applet de gestionnaire d'événements. Vous pouvez configurer plusieurs événements de journal système, mais les ID des messages de journal système ne peuvent pas se chevaucher dans un seul applet de gestionnaire d'événements.
- **Minuteries** : vous pouvez utiliser des minuteries pour déclencher des événements. Vous ne pouvez configurer chaque minuterie qu'une seule fois pour chaque applet de gestionnaire d'événements. Chaque applet de gestionnaire d'événements peut avoir jusqu'à trois minuteries. Les trois types de minuteries sont les suivants :
 - Les minuteries de surveillance (périodiques) déclenchent un applet de gestionnaire d'événements après la période spécifiée suivant la fin des actions de l'applet et redémarrent automatiquement.

- Les minuteries de compte à rebours (à usage unique) déclenchent un applet de gestionnaire d'événements une fois après la période spécifiée et ne redémarrent pas à moins d'être supprimées, puis rajoutées.
- Les minuteries absolues (une fois par jour) déclenchent un événement une fois par jour à une heure spécifiée et redémarrent automatiquement. Le format de l'heure du jour hh:mm:ss.

Vous ne pouvez configurer qu'un seul événement de minuterie de chaque type pour chaque applet de gestionnaire d'événements.

- **Aucun** : l'événement **Aucun** est déclenché lorsque vous exécutez un applet de gestionnaire d'événements manuellement à l'aide de l'interface de ligne de commande ou d'ASDM.
- **Plantage** : l'événement de plantage est déclenché lorsque l'ASA tombe en panne. Dans certains scénarios, un plantage forcé est déclenché :

Si l'ASA est configuré pour se recharger en cas d'épuisement du bloc et que l'ASA reste à court de mémoire pendant une durée configurée, il envoie le journal système et recueille des données de dépannage. L'ASA force le plantage et déclenche le processus de rechargement pour libérer le bloc de mémoire. Dans une configuration à haute disponibilité, dans de tels cas, le basculement est déclenché. Dans une configuration de grappe, le nœud quitte la grappe.

Quelle que soit la valeur de la commande **output**, les commandes **action** sont dirigées vers le fichier crashinfo. La sortie est générée avant la commande **show tech**.

Actions sur les applets de gestionnaire d'événements

Lorsqu'un applet de gestionnaire d'événements est déclenché, les actions sur cet applet sont effectuées. Chaque action a un numéro utilisé pour spécifier la séquence des actions. Le numéro de séquence doit être unique dans un applet de gestionnaire d'événements. Vous pouvez configurer plusieurs actions pour un applet de gestionnaire d'événements. Les commandes sont des commandes d'interface de ligne de commande typiques, telles que **show blocks**.

Destinations de sortie

Vous pouvez envoyer le résultat des actions à un emplacement spécifié à l'aide de la commande **output**. Une seule valeur de sortie peut être activée à la fois. La valeur par défaut est **output none**. Cette valeur annule toute sortie des commandes **action**. La commande s'exécute en mode de configuration globale en tant qu'utilisateur ayant le niveau de privilège 15 (le plus élevé). La commande ne peut accepter aucune entrée, car elle est désactivée. Vous pouvez envoyer la sortie des commandes **action** de l'interface de ligne de commande à l'un des trois emplacements suivants :

- **None** (Aucun), qui est la valeur par défaut et ignore la sortie.
- **Console**, qui envoie les données de sortie à la console ASA.
- **File** (Fichier), qui envoie la sortie dans un fichier. Les quatre options de fichiers suivantes sont disponibles :
 - **Create a unique file** (Créer un fichier unique), qui crée un nouveau fichier avec un nom unique chaque fois qu'un applet de gestionnaire d'événements est invoqué.
 - **Create/overwrite a file** (Créer/remplacer un fichier), qui remplace un fichier spécifié chaque fois qu'un applet de gestionnaire d'événements est invoqué.

- **Create/append to a file** (Créer/ajouter à un fichier), qui s'ajoute à un fichier spécifié chaque fois qu'un applet de gestionnaire d'événements est invoqué. Si le fichier n'existe pas encore, il est créé.
- **Create a set of files** (Créer un ensemble de fichiers), ce qui crée un ensemble de fichiers portant des noms uniques qui sont tournés chaque fois qu'un applet de gestionnaire d'événements est invoqué.

Lignes directrices pour l'EEG

Cette section décrit les lignes directrices et les limites que vous devez vérifier avant de configurer l'EEG.

Directives relatives au mode contextuel

Non pris en charge en mode de contexte multiple.

Directives supplémentaires

- Lors d'un plantage, l'état de l'ASA est généralement inconnu. Certaines commandes peuvent ne pas être exécutées en toute sécurité dans cette condition.
- Le nom d'un applet de gestionnaire d'événements ne peut pas contenir d'espaces.
- Vous ne pouvez pas modifier les paramètres des événements None (Aucun) et Cashinfo (Renseignements sur le plantage).
- Les performances peuvent être affectées, car les messages du journal système sont envoyés à l'EEG pour traitement.
- La sortie par défaut est **output none** pour chaque applet de gestionnaire d'événements. Pour modifier ce paramètre, vous devez entrer une valeur de sortie différente.
- Vous ne pouvez avoir qu'une seule option de sortie définie pour chaque applet de gestionnaire d'événements.

Configurer l'EEG

La configuration de l'EEG comprend les tâches suivantes :

Procédure

- | | |
|----------------|---|
| Étape 1 | Créer un applet de gestionnaire d'événements et configurer les événements, à la page 4. |
| Étape 2 | Configurer une action et des destinations pour la sortie d'une action, à la page 5. |
| Étape 3 | Exécuter un applet de gestionnaire d'événements, à la page 7. |
| Étape 4 | Suivre l'allocation de mémoire et l'utilisation de la mémoire, à la page 8. |

Créer un applet de gestionnaire d'événements et configurer les événements

Pour créer un applet de gestionnaire d'événements et configurer les événements, procédez comme suit :

Procédure

Étape 1 Créez un applet de gestionnaire d'événements et passez en mode de configuration de l'applet de gestionnaire d'événements.

event manager applet *nom*

Exemple :

```
ciscoasa(config)# event manager applet exampleapplet1
```

L'argument *name* peut comporter jusqu'à 32 caractères alphanumériques. Les espaces ne sont pas autorisées.

Pour supprimer un applet de gestionnaire d'événements, saisissez la forme **no** de cette commande.

Étape 2 Décrivez un applet de gestionnaire d'événements.

description *texte*

Exemple :

```
ciscoasa(config-applet)# description applet1example
```

L'argument *text* peut comporter jusqu'à 256 caractères. Vous pouvez inclure des espaces dans le texte de la description si celui-ci est placé entre guillemets.

Étape 3 Pour configurer un événement donné, saisissez l'une des commandes suivantes. Pour supprimer l'événement configuré, saisissez la forme **no** (non) de chacune des commandes.

- Pour configurer un événement du journal système, identifiez un seul message de journal système ou une plage de messages de journal système qui déclenchent un applet de gestionnaire d'événements.

event syslog id *nnnnnn* [-*nnnnnn*] [**occurs** *n*] [**period** *seconds*]

Exemple :

```
ciscoasa(config-applet)# event syslog id 106201
```

L'argument *nnnnnn* identifie l'ID du message de journal système. La paire mot clé-argument **occurs** *n* indique le nombre de fois que le message du journal système doit se produire pour qu'un applet de gestionnaire d'événements soit invoqué. La valeur par défaut est de 1 occurrence toutes les 0 secondes. Les valeurs valides sont comprises entre 1 et 4294967295. La paire mot clé-argument **period** *seconds* indique le nombre de secondes pendant lesquelles l'événement doit se produire et limite la fréquence d'appel d'un applet de gestionnaire d'événements au maximum une fois dans la période configurée. Les valeurs valides sont comprises entre 0 et 604800. La valeur 0 signifie qu'aucune période n'est définie.

- Pour configurer un événement afin qu'il se produise une fois par période configurée et redémarre automatiquement.

event timer watchdog time *seconds*

Exemple :

```
ciscoasa(config-applet)# event timer watchdog time 30
```

Le nombre de secondes peut aller de 1 à 604800.

- Pour configurer un événement afin qu'il se produise une seule fois et ne redémarre pas à moins qu'il ne soit supprimé, puis rajouté.

event timer countdown time seconds

Exemple :

```
ciscoasa(config-applet)# event timer countdown time 60
```

Le nombre de secondes peut aller de 1 à 604800. Utilisez la forme **no** de cette commande pour supprimer un événement de minuterie de compte à rebours.

Remarque

Cette minuterie se réexécute lorsque vous redémarrez s'il s'agit de la configuration de démarrage.

- Pour configurer un événement afin qu'il se produise une fois par jour à une heure précise et qu'il redémarre automatiquement.

event timer absolute time hh:mm:ss

Exemple :

```
ciscoasa(config-applet)# event timer absolute time 10:30:20
```

Le format de l'heure du jour hh:mm:ss. La plage horaire va de 00:00:00 (minuit) à 23:59:59.

- Déclenchez un événement de plantage lorsque l'ASA tombe en panne.

event crashinfo

Exemple :

```
ciscoasa(config-applet)# event crashinfo
```

Quelle que soit la valeur de la commande **output**, les commandes **action** sont dirigées vers le fichier crashinfo. La sortie est générée avant la commande **show tech**.

Configurer une action et des destinations pour la sortie d'une action

Pour configurer une action et des destinations spécifiques pour l'envoi de la sortie d'une action, procédez comme suit :

Procédure

Étape 1 Configurez une action sur un applet de gestionnaire d'événements.

action *n* cli command "*command*"

Exemple :

```
ciscoasa(config-applet)# action 1 cli command "show version"
```

L'option *n* est un ID d'action. Les ID valides vont de 0 à 4294967295. La valeur de l'option *command* doit être entre guillemets, sinon une erreur se produit si la commande comporte plusieurs mots. La commande s'exécute en mode de configuration globale en tant qu'utilisateur ayant le niveau de privilège 15 (le plus élevé). La commande ne peut accepter aucune entrée, car elle est désactivée. Utilisez l'option **noconfirm** si la commande la propose.

Étape 2 Choisissez l'une des options de destination de sortie disponibles. Utilisez la forme **no** dde chaque commande pour supprimer une destination de sortie.

- L'option **None** annule toute sortie des commandes **action**, ce qui correspond au paramètre par défaut :

output none

Exemple :

```
ciscoasa(config-applet)# output none
```

- L'option **Console** envoie le résultat des commandes **action** à la console.

output console

Exemple :

```
ciscoasa(config-applet)# output console
```

Remarque

L'exécution de cette commande affecte les performances.

- L'option **New File** envoie la sortie des commandes **action** vers un nouveau fichier pour chaque applet de gestionnaire d'événements invoqué.

output file new

Exemple :

```
ciscoasa(config-applet)# output file new
```

Le nom de fichier a le format *eem-Applet-horodatage.log*, dans lequel *Applet* est le nom de l'Applet du gestionnaire d'événements et *Horodatage* est un horodatage datant au format *YYMMDD-hhmmss*.

- L'option **New Set of Rotated Files** crée un ensemble de fichiers qui sont tournés. 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 premier fichier.

output file rotate *n*

Exemple :

```
ciscoasa(config-applet)# output file rotate 50
```

Le fichier le plus récent est indiqué par 0, et le fichier le plus ancien est indiqué par le nombre le plus élevé ($n-1$). L'option n est la valeur de rotation. Les valeurs valides sont comprises entre 2 et 100. Le format du nom de fichier est *eem-Applet-x.log*, dans lequel *Applet* est le nom de l'applet et x est le numéro de fichier.

- L'option **Single Overwritten File** écrit la sortie de la commande **action** dans un fichier unique, qui est remplacé chaque fois.

output file overwrite *filename*

Exemple :

```
ciscoasa(config-applet)# output file overwrite examplefile1
```

L'argument *filename* est un nom de fichier local (pour l'ASA). Cette commande peut également utiliser des fichiers ciblés FTP, TFTP et SMB.

- L'option **Single Appended File** écrit la sortie de la commande **action** dans un fichier unique, mais ce fichier est ajouté à chaque fois.

output file append *filename*

Exemple :

```
ciscoasa(config-applet)# output file append examplefile1
```

L'argument *filename* est un nom de fichier local (pour l'ASA).

Exécuter un applet de gestionnaire d'événements

Pour exécuter un applet de gestionnaire d'événements, procédez comme suit :

Procédure

Exécutez un applet de gestionnaire d'événements.

event manager run *applet*

Exemple :

```
ciscoasa# event manager run exampleapplet1
```

Si vous exécutez un applet de gestionnaire d'événements qui n'a pas été configuré avec la commande **event none**, une erreur se produit. L'argument *applet* est le nom de l'applet de gestionnaire d'événements.

Suivre l'allocation de mémoire et l'utilisation de la mémoire

Pour consigner l'allocation de mémoire et l'utilisation de la mémoire, procédez comme suit :

Procédure

Étape 1 Activez la journalisation de la mémoire.

memory logging [1024-4194304] [**wrap**] [**size** [1-2147483647]] [**process** *process-name*] [**context** *context-name*]

Exemple :

```
ciscoasa(config)# memory logging 202980
```

Le seul argument requis est le nombre d'entrées dans le tampon de journalisation de la mémoire. L'option **wrap** demande à l'utilitaire de journalisation de la mémoire d'enregistrer le tampon lorsqu'il arrive à saturation. Il ne peut être enregistré qu'une seule fois.

Si le tampon de journalisation de la mémoire arrive à saturation plusieurs fois, il peut être remplacé. Lorsque le tampon arrive à saturation, un déclencheur est envoyé au gestionnaire d'événements pour permettre l'enregistrement des données. L'option **size** surveille une taille particulière. L'option **process** surveille un processus particulier.

Remarque

Le processus CheckHeaps est complètement ignoré en tant que processus, car il utilise l'allocation de mémoire d'une manière non standard.

L'option **context** enregistre la journalisation de la mémoire pour un contexte virtuel donné sous le nom indiqué.

Pour modifier les paramètres de journalisation de la mémoire, vous devez la désactiver, puis la réactiver.

Étape 2 Affichez les résultats de la journalisation de la mémoire.

show memory logging [**brief** | **wrap**]
show memory logging include [**address**] [**caller**] [**operator**] [**size**] [**process**] [**time**] [**context**]

Exemple :

```
ciscoasa# show memory logging
Number of free           6
Number of calloc         0
Number of malloc         8
```

```

Number of realloc-new          0
Number of realloc-free        0
Number of realloc-null        0
Number of realloc-same        0
Number of calloc-fail         0
Number of malloc-fail         0
Number of realloc-fail        0
Total operations 14
Buffer size: 50 (3688 x2 bytes)
process=[ci/console] time=[13:26:33.407] oper=[malloc]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000016466ea 0x0000000002124542
0x000000000131911a 0x0000000000442bfd process=[ci/console] time=[13:26:33.407] oper=[free]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000021246ef 0x00000000013193e8
0x0000000000443455 0x0000000001318f5b
process=[CMGR Server Process] time=[13:26:35.964] oper=[malloc]
addr=0x00007fff2cd0aa00 size=16 @ 0x00000000016466ea 0x0000000002124542
0x000000000182774d 0x000000000182cc8a process=[CMGR Server Process]
time=[13:26:35.964] oper=[malloc]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000016466ea 0x0000000002124542
0x0000000000bfff9a 0x0000000000bfff606 process=[CMGR Server Process]
time=[13:26:35.964] oper=[free]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000021246ef 0x0000000000bfff3d8
0x0000000000bfff606 0x000000000182ccb0
process=[CMGR Server Process] time=[13:26:35.964] oper=[malloc]
addr=0x00007fff224b9460 size=40 @ 0x00000000016466ea 0x0000000002124542
0x0000000001834188 0x000000000182ce83
process=[CMGR Server Process] time=[13:26:37.964] oper=[free]
addr=0x00007fff2cd0aa00 size=16 @ 0x00000000021246ef 0x0000000001827098
0x000000000182c08d 0x000000000182c262 process=[CMGR Server Process]
time=[13:26:37.964] oper=[free]
addr=0x00007fff224b9460 size=40 @ 0x00000000021246ef 0x000000000182711b
0x000000000182c08d 0x000000000182c262 process=[CMGR Server Process]
time=[13:26:38.464] oper=[malloc]
addr=0x00007fff2cd0aa00 size=16 @ 0x00000000016466ea 0x0000000002124542
0x000000000182774d 0x000000000182cc8a process=[CMGR Server Process]
time=[13:26:38.464] oper=[malloc]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000016466ea 0x0000000002124542
0x0000000000bfff9a 0x0000000000bfff606 process=[CMGR Server Process]
time=[13:26:38.464] oper=[free]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000021246ef 0x0000000000bfff3d8
0x0000000000bfff606 0x000000000182ccb0
process=[CMGR Server Process] time=[13:26:38.464] oper=[malloc]
addr=0x00007fff224b9460 size=40 @ 0x00000000016466ea 0x0000000002124542
0x0000000001834188 0x000000000182ce83
process=[ci/console] time=[13:26:38.557] oper=[malloc]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000016466ea 0x0000000002124542
0x000000000131911a 0x0000000000442bfd process=[ci/console] time=[13:26:38.557] oper=[free]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000021246ef 0x00000000013193e8
0x0000000000443455 0x0000000001318f5b

ciscoasa# show memory logging include process operation size
Number of free                6
Number of calloc              0
Number of malloc              8
Number of realloc-new         0
Number of realloc-free        0
Number of realloc-null        0
Number of realloc-same        0
Number of calloc-fail         0
Number of malloc-fail         0
Number of realloc-fail        0
Total operations 14
Buffer size: 50 (3688 x2 bytes)
process=[ci/console] oper=[malloc] size=72 process=[ci/console] oper=[free]

```

```

size=72 process=[CMGR Server Process] oper=[malloc] size=16
process=[CMGR Server Process] oper=[malloc] size=512 process=[CMGR Server Process]
oper=[free] size=512 process=[CMGR Server Process] oper=[malloc] size=40
process=[CMGR Server Process] oper=[free] size=16 process=[CMGR Server Process]
oper=[free] size=40 process=[CMGR Server Process] oper=[malloc] size=16
process=[CMGR Server Process] oper=[malloc] size=512 process=[CMGR Server Process]
oper=[free] size=512 process=[CMGR Server Process] oper=[malloc] size=40
process=[ci/console] oper=[malloc] size=72 process=[ci/console]
oper=[free] size=72 ciscoasa# show memory logging brief
Number of free                6
Number of calloc              0
Number of malloc              8
Number of realloc-new         0
Number of realloc-free        0
Number of realloc-null        0
Number of realloc-same        0
Number of calloc-fail         0
Number of malloc-fail         0
Number of realloc-fail        0
Total operations 14
Buffer size: 50 (3688 x2 bytes)

```

Sans aucune option, **show memory logging** affiche les statistiques, puis les opérations enregistrées. L'option **brief** affiche uniquement les statistiques. L'option **wrap** affiche le tampon lors de l'encapsulation, puis purge les données afin que les données dupliquées n'apparaissent pas ou ne soient pas enregistrées. L'option **include** inclut uniquement les champs spécifiés dans la sortie. Vous pouvez spécifier les champs dans n'importe quel ordre, mais ils s'affichent toujours dans l'ordre suivant :

1. Processus
2. Durée
3. Contexte (sauf en mode unique)
4. Opération (libre/malloc/etc.)
5. Adresse
6. Taille
7. Appelants

Le format de sortie est le suivant :

```

process=[XXX] time=[XXX] context=[XXX] oper=[XXX] address=0xFFFFFFFF size=XX @
XXXXXXXXXX XXXXXXXXXXXX XXXXXXXXXXXX XXXXXXXXXXXX

```

Jusqu'à quatre adresses d'appelants apparaissent. Les types d'opérations sont répertoriés dans la sortie (nombre de...) illustrée dans l'exemple.

Étape 3

Répondez aux événements d'encapsulation de la mémoire.

event memory-logging-wrap

Exemple :

```

ciscoasa(config)# event manager applet memlog
ciscoasa(config)# event memory-logging-wrap
ciscoasa(config)# action 0 cli command "show memory logging wrap"
ciscoasa(config)# output file append disk0:/memlog.log

```

L'exemple montre un applet qui enregistre toutes les allocations de mémoire. Lorsque l'encapsulation est activée pour la journalisation de la mémoire, le journaliseur de mémoire envoie un événement au gestionnaire d'événements afin de déclencher les applets configurés.

Exemples d'EEG

L'exemple suivant montre un applet de gestionnaire d'événements qui enregistre les renseignements de fuite de bloc toutes les heures et écrit le résultat dans un ensemble rotatif de fichiers journaux, en conservant les journaux d'une journée :

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

L'exemple suivant montre un applet de gestionnaire d'événements qui redémarre l'ASA tous les jours à 1 h, en enregistrant la configuration si nécessaire :

```
ciscoasa(config)# event manager applet dailyreboot
ciscoasa(config-applet)# description "Reboot every night"
ciscoasa(config-applet)# event timer absolute time 1:00:00
ciscoasa(config-applet)# output none
ciscoasa(config-applet)# action 1 cli command "reload save-config noconfirm"
```

L'exemple suivant montre des applets de gestionnaire d'événements qui désactivent l'interface donnée entre minuit et 3 h.

```
ciscoasa(config)# event manager applet disableintf
ciscoasa(config-applet)# description "Disable the interface at midnight"
ciscoasa(config-applet)# event timer absolute time 0:00:00
ciscoasa(config-applet)# output none
ciscoasa(config-applet)# action 1 cli command "interface GigabitEthernet 0/0"
ciscoasa(config-applet)# action 2 cli command "shutdown"
ciscoasa(config-applet)# action 3 cli command "write memory"

ciscoasa(config)# event manager applet enableintf
ciscoasa(config-applet)# description "Enable the interface at 3am"
ciscoasa(config-applet)# event timer absolute time 3:00:00
ciscoasa(config-applet)# output none
ciscoasa(config-applet)# action 1 cli command "interface GigabitEthernet 0/0"
ciscoasa(config-applet)# action 2 cli command "no shutdown"
ciscoasa(config-applet)# action 3 cli command "write memory"
```

Supervision de l'EEG

Consultez les commandes suivantes pour superviser l'EEG.

- **clear configure event manager**

Cette commande supprime la configuration en cours d'exécution du gestionnaire d'événements.

- **clear configure event manager applet *appletname***

Cette commande supprime l'applet de gestionnaire d'événements nommé de la configuration.

- **show counters protocol eem**

Cette commande affiche les compteurs du gestionnaire d'événements.

- **show event manager**

Cette commande affiche des renseignements sur les applets de gestionnaire d'événements configurés, notamment le nombre d'accès et la date de la dernière invocation des applets du gestionnaire d'événements.

- **show memory logging, show memory logging include**

Ces commandes affichent les statistiques sur les allocations de mémoire et l'utilisation de la mémoire.

- **show running-config event manager**

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

Historique de l'EEG

Tableau 1 : Historique de l'EEG

Nom de la caractéristique	Versions de plateforme	Description
Gestionnaire d'événements intégré de Cisco (EEG)	9.2(1)	Le service EEG vous permet de déboguer les problèmes et fournit une journalisation à usage général pour le dépannage. Il y a deux composants : les événements auxquels l'EEG répond ou écoute, et les applets de gestionnaire d'événements qui définissent les actions ainsi que les événements auxquels l'EEG répond. Vous pouvez configurer plusieurs applets de gestionnaire d'événements pour répondre à différents événements et effectuer des actions différentes. Nous avons introduit ou modifié les commandes suivantes : event manager applet, description, event syslog id, event none, event timer {watchdog time <i>seconds</i> countdown time <i>seconds</i> absolute time <i>hh:mm:ss</i>}, event crashinfo, action cli command, output {none console file {append <i>filename</i> new overwrite <i>filename</i> rotate <i>n</i>}}, show running-config event manager, event manager run, show event manager, show counters protocol eem, clear configure event manager, debug event manager, debug menu eem.

Nom de la caractéristique	Versions de plateforme	Description
Suivi de mémoire pour l'EEG	9.4(1)	Nous avons ajouté une nouvelle fonctionnalité de débogage pour consigner les allocations de mémoire et l'utilisation de la mémoire, et pour répondre aux événements d'encapsulation de la journalisation de la mémoire. Nous avons introduit ou modifié les commandes suivantes : memory logging, show memory logging, show memory logging include, event memory-logging-wrap.

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d'incohérence, la version anglaise du présent contenu prévaudra.