Exemplo de configuração do ASA Embedded Event Manager

Contents

Introduction

Este documento descreve o Embedded Event Manager (EEM), que é uma ferramenta de solução de problemas que foi adicionada no Adaptive Security Appliance (ASA) versão 9.2(1). A funcionalidade é semelhante ao Cisco IOS? EEM com base. É uma maneira poderosa de executar comandos CLI com base em eventos ASA (syslogs) e salvar a saída. Este documento aborda uma introdução ao recurso, bem como alguns exemplos de miniaplicativos EEM.

Prerequisites

Requirements

O uso do EEM requer que o ASA esteja configurado no modo de contexto único.

Componentes Utilizados

As informações neste documento são baseadas no ASA versão 9.2(1) ou posterior.

Diretrizes e limitações

Esta seção inclui as diretrizes e limitações para este recurso.

Diretrizes do modo de contexto

Atualmente, o EEM só é suportado em firewalls ASA executados em modo de contexto único. Os firewalls configurados em modo de contexto múltiplo não são suportados no momento.

Diretrizes do modo de firewall

Atualmente, o EEM é suportado nos modos de firewall roteado e transparente.

Diretrizes adicionais

- Enquanto a unidade trava, o estado do ASA é geralmente desconhecido. Alguns comandos podem não ser seguros de execução enquanto o ASA estiver nessa condição.
- O nome de um miniaplicativo do gerenciador de eventos não pode conter espaços.
- Não é possível modificar os parâmetros do evento Nenhum e do evento Crashinfo.
- O desempenho pode ser afetado porque as mensagens de syslog são enviadas ao EEM para serem processadas.
- A saída padrão é **output none** para cada applet do event manager. Para alterar a saída padrão, você deve inserir um valor de saída diferente.
- Você pode ter apenas uma opção de saída definida para cada miniaplicativo do gerenciador de eventos.

Configurar

O comando **event manager applet** cria/edita um applet do event manager, um processo que vincula eventos a ações e saídas. O *<nome>* é limitado a 32 caracteres e não pode ter espaços. Insere um submodo applet do event manager.

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

Uma **descrição** pode ser adicionada a um miniaplicativo. Isso é apenas para fins informativos. O <*text*> é limitado a 256 caracteres.

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

Configuração do evento

Vários eventos podem ser adicionados a um miniaplicativo que aciona o miniaplicativo para chamar as ações configuradas nele. Eles são definidos com a palavra-chave **event**. Vários eventos podem ser configurados para cada miniaplicativo.

Eventos de Syslog

O primeiro tipo de evento suportado é **syslog**. O ASA usa IDs de syslog para identificar syslogs que disparam um applet. Isso é concluído por meio da palavra-chave id, que pode ser um único syslog ou um intervalo. A palavra-chave opcional **ocorre** indica o número de vezes que o syslog deve ocorrer para o miniaplicativo ser chamado (o padrão é 1). A palavra-chave **period** opcional indica a quantidade de tempo, em segundos, em que o evento deve ocorrer. Limita a frequência de chamada do applet para, no máximo, o período configurado. Uma **ocorrência** de 5 com um **período** de 30, significa que o syslog deve ocorrer 5 vezes em 30 segundos antes do evento ser disparado. Se o syslog ocorrer 11 vezes em 30 segundos, o miniaplicativo será acionado apenas uma vez. Um valor de 0 para o **período** significa que nenhum período é definido.

Vários syslogs podem ser configurados, mas os intervalos não podem se sobrepor.

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

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

O valor **de ocorrência** <*n*> tem um intervalo permitido de 1 a 4294967295. O valor do **período** <seconds> tem um intervalo permitido de 0 a 604800. Um valor 0 (zero) significa que nenhum período está configurado.

Exemplo de eventos de syslog

Neste exemplo, o EEM age quando detecta uma condição de bloco de memória baixa. Se os blocos de 1550 bytes disponíveis forem esgotados, ele coletará o **show block pool 1550 dump** e salvará no disco. Ele faz isso, no máximo, a cada 10 minutos.

```
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
```

Eventos periódicos

O EEM também pode ser configurado para executar uma ação periodicamente. Ao configurar um evento baseado em temporizador, use a palavra-chave **timer** na configuração de eventos. Há três opções baseadas em temporizador:

- absolute - O primeiro temporizador é um temporizador **absoluto** que aciona o applet uma vez por dia na hora especificada e é reiniciado automaticamente.
  ```
  ASA(config-applet)# [no] event timer absolute time
  ```

  ```
  ASA(config-applet)# no event timer absolute
  ```
- countdown - O segundo temporizador é um temporizador **de contagem regressiva** que aciona o applet uma vez e não é reiniciado a menos que removido e adicionado novamente.
  ```
  ASA(config-applet)# [no] event timer countdown time
  ```

  ```
  ASA(config-applet)# no event timer countdown
  ```
- watchdog - O terceiro temporizador é um temporizador **watchdog** que aciona o applet uma vez por período configurado e é reiniciado automaticamente.
  ```
  ASA(config-applet)# [no] event timer watchdog time
  ```

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

Exemplo de eventos periódicos

Por exemplo, essa configuração de evento executa ping em 192.168.1.100 a cada 1 minuto. Isso pode ser usado para garantir que um túnel VPN seja mantido ativo e operacional mesmo durante períodos de tráfego ocioso. Ele usa o temporizador **watchdog** para executar a cada 60 segundos.

```
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
```

Este miniaplicativo registra as informações de alocação de bloco de memória a cada hora e grava a saída em um conjunto rotativo de arquivos de log, já que mantém registros válidos por um dia. Ele usa o temporizador **watchdog** para executar a cada 1 hora.

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

Esses miniaplicativos desabilitam a interface especificada (Gig 0/0) entre a meia-noite e as 3 da manhã. Ela usa o temporizador **absoluto** para executar uma vez por dia.

```
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"
```

Evento manual

Esses miniaplicativos EEM também podem ser chamados manualmente. Para fazer isso, o applet deve configurar **event none**. Para executar um applet manualmente, insira o comando **event manager run** seguido do nome do applet. Se o applet estiver configurado para qualquer mecanismo de disparo de eventos além de 'none', a tentativa de executá-lo manualmente gerará um erro. Com o uso de um dos exemplos anteriores, 'esgotetedblock', você vê:

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

Exemplo de evento manual

Os eventos manuais podem ser usados de forma semelhante a uma macro. Por exemplo, um evento manual pode ser usado para executar alguns comandos em ordem. Neste exemplo, ele salva a configuração, faz ping em um host e limpa todos os 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
```

Evento de travamento

O evento **crashinfo** aciona um applet quando ocorre um travamento no ASA. Independentemente do valor do comando **output**, os comandos **action** são direcionados para o arquivo crashinfo. A

saída é gerada antes que a parte **show tech** das informações de travamento seja gerada.

> **aviso:** Quando o ASA está travando, o estado da caixa é geralmente desconhecido. Alguns comandos CLI podem não ser seguros de execução quando a unidade está nessa condição.

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

Configuração da ação

Quando o miniaplicativo é acionado, as ações no miniaplicativo são executadas. Cada **ação** tem um ordinal que é usado para especificar a ordem das ações. Várias ações podem ser configuradas por miniaplicativo; mas cada ordinal só pode ser usado uma vez. Os comandos são comandos CLI típicos, como **show block**. Os orçamentos são altamente recomendados, mas não são obrigatórios.

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

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

O valor do identificador da ação *<n>* tem um intervalo de 0 a 4294967295. O valor do *<comando>* deve ser citado, caso contrário, ocorrerá um erro se o comando consistir em mais de uma palavra. O comando é executado no modo de configuração como um usuário com nível de privilégio 15 (o mais alto). O comando pode não aceitar qualquer entrada; como input será desativado se um comando tiver a opção **noconfirm**. Isso deve ser usado, já que os comandos não são processados interativamente.

Configuração de saída

A saída das ações pode ser direcionada para um local especificado através do comando **output**. Somente um valor de saída pode ser ativado de cada vez. O valor padrão é **output none**. Esse valor descarta qualquer saída dos comandos de ação.

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

O comando **output console** envia a saída dos comandos action para o console.

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

O comando **output file** direciona a saída dos comandos action para arquivos. Há quatro opções que podem ser usadas. A **nova** opção grava a saída do applet em um novo arquivo para cada invocação. O *nome de arquivo* tem o formato **eem-*<applet>*-*<timestamp>*.log**. Onde *<applet>* é o nome do applet e *<timestamp>* é um timestamp datado no formato *YYYMMDD-hmss*.

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

A opção **rotate** é usada para criar um conjunto de arquivos que são girados de forma semelhante ao mecanismo de rotação de log do Linux. O formato do nome de arquivo é **eem-*<applet>*-*<x>*.log**. Onde *<applet>* é o nome do applet e *<x>* é o número do arquivo. O arquivo mais recente é indicado pelo número 0 (zero) e o arquivo mais antigo é indicado pelo número mais alto (*<n>*-1). Quando um novo arquivo deve ser gravado, o arquivo mais antigo é excluído e todos os arquivos subsequentes são renumerados antes que o 0º arquivo seja gravado.

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

O valor de rotação *<n>* tem um intervalo de 2 a 100.

A opção **overwrite** é usada para gravar sempre a saída do comando action em um único arquivo que é truncado sempre.

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

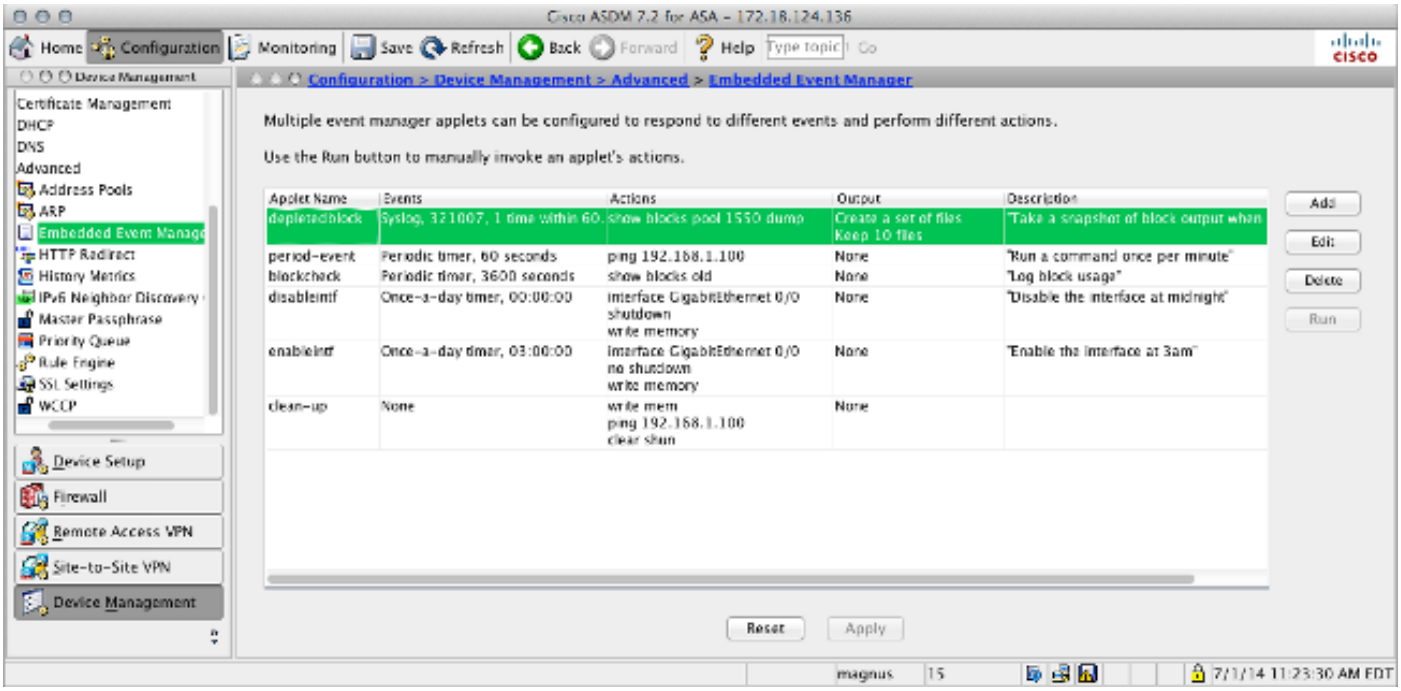
A opção **append** é usada para sempre gravar a saída do comando action em um único arquivo, mas esse arquivo é anexado a cada vez.

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

O argumento *<filename>* é um nome de arquivo local (para o ASA). O comando overwrite também pode usar **ftp**:, **tftp:** e **smb:** arquivos de destino.

Configuração do ASDM

O EEM também pode ser configurado no ASDM. Escolha **Configuration > Device Management > Advanced > Embedded Event Manager**. Nesta seção do ASDM, você pode configurar seus applets EEM com os mesmos parâmetros discutidos anteriormente. Depois de configurar um applet, clique em **Apply** para enviar a configuração para o ASA.



Verificar

Comandos do modo exec

Use esta seção para confirmar se a sua configuração funciona corretamente.

Todos esses comandos são usados no modo exec.

Este comando mostra a configuração atual do sistema do gerenciador de eventos.

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

Este comando executa um applet do gerente de eventos que foi configurado com **event none**. Se você executar um miniaplicativo que não foi configurado com **evento nenhum**, um erro será relatado.

ASA# **event manager run**

Este comando mostra informações sobre os miniaplicativos configurados, o que inclui contagens de acertos e quando o miniaplicativo foi chamado pela última vezASA# 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 10:51:52O gerenciador de eventos usa os contadores padrão. Devido a limitações dentro da CLIshow counter, a palavra-chave eem é usada para filtragem de protocolo.

ASA# show counters protocol eem A ferramenta Output Interpreter (exclusiva para clientes registrados) é compatível com alguns comandos de exibição.. Use a ferramenta Output Interpreter

para visualizar uma análise do resultado gerado pelo comando show..DebugInsira estes comandos para depurar o EEM e exibir a saída.Note: Consulte Informações Importantes sobre Comandos de Depuração antes de usar comandos debug.

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

ASA# show debug event managerTroubleshootAtualmente, não existem informações disponíveis específicas sobre Troubleshooting para esta configuração. Se ele não funcionar como esperado, use as etapas de depuração e verificação listadas na seção anterior para determinar se ocorreu um erro.