



Embedded Event Manager の設定

この章は、次の項で構成されています。

- [組み込みイベント マネージャについて \(1 ページ\)](#)
- [Embedded Event Manager ポリシー \(2 ページ\)](#)
- [Embedded Event Manager の前提条件 \(5 ページ\)](#)
- [Embedded Event Manager の注意事項および制約事項 \(5 ページ\)](#)
- [Embedded Event Manager のデフォルト設定 \(6 ページ\)](#)
- [環境変数の定義 \(6 ページ\)](#)
- [CLI によるユーザ ポリシーの定義 \(7 ページ\)](#)
- [イベント文の設定 \(9 ページ\)](#)
- [アクション文の設定 \(11 ページ\)](#)
- [VSH スクリプトによるポリシーの定義 \(14 ページ\)](#)
- [VSH スクリプト ポリシーの登録およびアクティブ化 \(15 ページ\)](#)
- [システム ポリシーの上書き \(16 ページ\)](#)
- [EEM パブリッシャとしての syslog の設定 \(17 ページ\)](#)

組み込みイベント マネージャについて

Cisco NX-OS システム内のクリティカル イベントを検出して処理する機能は、ハイ アベイラビリティにとって重要です。Embedded Event Manager (EEM) は、デバイス上で発生するイベントをモニターし、設定に基づいてこれらのイベントを回復またはトラブルシューティングするためのアクションを実行することによってシステム内のイベントを検出して処理する、中央のポリシー駆動型のフレームワークを提供します。

EEM は次の 3 種類の主要コンポーネントからなります。

イベント文

何らかのアクション、回避策、または通知が必要になる可能性のある、別の Cisco NX-OS コンポーネントからモニターするイベント。

アクション文

電子メールの送信やインターフェイスのディセーブル化などの、イベントから回復するために EEM が実行できるアクション。

ポリシー

イベントのトラブルシューティングまたはイベントからの回復を目的とした1つまたは複数のアクションとペアになったイベント。

EEM を使用しない場合は、個々のコンポーネントが独自のイベントの検出および処理を行います。たとえば、ポートでフラップが頻繁に発生する場合は、「errDisable ステートにする」のポリシーが ETHPM に組み込まれます。

Embedded Event Manager ポリシー

EEM ポリシーは、イベント文および1つまたは複数のアクション文からなります。イベント文では、探すイベントとともに、イベントのフィルタリング特性を定義します。アクション文では、イベントの発生時に EEM が実行するアクションを定義します。

たとえば、いつカードがデバイスから取り外されたかを識別し、カードの取り外しに関する詳細を記録する EEM ポリシーを設定できます。カードの取り外しのインスタンスすべてを探すようにシステムに指示するイベント文および詳細を記録するようにシステムに指示するアクション文を設定します。

コマンドラインインターフェイス (CLI) または VSH スクリプトを使用して EEM ポリシーを設定できます。

EEM からデバイス全体のポリシー管理ビューが得られます。EEM ポリシーが設定されると、対応するアクションがトリガーされます。トリガーされたイベントのすべてのアクション (システムまたはユーザー設定) がシステムによって追跡され、管理されます。

設定済みのシステム ポリシー

Cisco NX-OS には、設定済みのさまざまなシステム ポリシーがあります。これらのシステムポリシーでは、デバイスに関連する多数の一般的なイベントおよびアクションが定義されています。システムポリシー名は、2個の下線記号 (__) から始まります。

一部のシステムポリシーは上書きできます。このような場合、イベントまたはアクションに対する上書きを設定できます。設定した上書き変更がシステムポリシーの代わりになります。



(注) 上書きポリシーにはイベント文を含める必要があります。イベント文が含まれていない上書きポリシーは、システムポリシーで想定されるすべてのイベントを上書きします。

設定済みのシステムポリシーを表示し、上書きできるポリシーを決定するには、**show event manager system-policy** コマンドを使用します。

ユーザー作成ポリシー

ユーザー作成ポリシーを使用すると、ネットワークのEEMポリシーをカスタマイズできます。ユーザーポリシーがイベントに対して作成されると、ポリシーのアクションは、EEMが同じイベントに関連するシステムポリシーアクションをトリガーした後にのみトリガーされます。

ログファイル

EEMポリシーの一致に関連するデータが格納されたログファイルは、`/log/event_archive_1`ディレクトリにある `event_archive_1` ログファイルで維持されます。

イベント文

対応策、通知など、一部のアクションが実行されるデバイスアクティビティは、EEMによってイベントと見なされます。イベントは通常、インターフェイスやファンの誤動作といったデバイスの障害に関連します。

イベント文は、どのイベントがポリシー実行のトリガーになるかを指定します。



ヒント ポリシー内に複数のEEMイベントを作成し、区別してから、カスタムアクションをトリガーするためのイベントの組み合わせを定義することで、イベントの組み合わせに基づいたEEMポリシーをトリガーするようにEEMを設定できます。

EEMではイベントフィルタを定義して、クリティカルイベントまたは指定された時間内で繰り返し発生したイベントだけが関連付けられたアクションのトリガーになるようにします。

一部のコマンドまたは内部イベントが他のコマンドを内部的にトリガーします。これらのコマンドは表示されませんが、引き続きアクションをトリガーするイベント指定と一致します。これらのコマンドがアクションをトリガーするのを防ぐことはできませんが、どのイベントがアクションを引き起こしたかを確認できます。

サポートされるイベント

EEMはイベント文で次のイベントをサポートします。

- カウンタ イベント
- ファン欠損イベント
- ファン不良イベント
- メモリしきい値イベント
- 上書きされたシステムポリシーで使用されるイベント
- SNMP通知イベント
- syslog イベント
- システムマネージャ イベント

- 温度イベント
- 追跡イベント

アクション文

アクション文は、イベントが発生したときに、ポリシーによってトリガーされるアクションを説明します。各ポリシーに複数のアクション文を設定できます。ポリシーにアクションを関連付けなかった場合、EEM はイベント観察を続けますが、アクションは実行されません。

トリガーされたイベントがデフォルトアクションを処理するために、デフォルトアクションを許可する EEM ポリシーを設定する必要があります。たとえば、一致文で CLI コマンドを照合する場合、EEM ポリシーに `event-default` アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。



-
- (注) ユーザーポリシーまたは上書きポリシー内のアクション文を設定する場合、アクション文が、相互に否定したり、関連付けられたシステムポリシーに悪影響を与えるようなことがないように確認することが重要です。
-

サポートされるアクション

EEM がアクション文でサポートするアクションは、次のとおりです。

- CLI コマンドの実行
- カウンタのアップデート
- デバイスのリロード
- syslog メッセージの生成
- SNMP 通知の生成
- システム ポリシー用デフォルトアクションの使用

VSH スクリプト ポリシー

テキストエディタを使用して、VSH スクリプトでポリシーを作成できます。VSH スクリプトを使用して作成されたポリシーには、他のポリシーと同様にイベント文とアクション文が含まれます。また、これらのポリシーはシステムポリシーを拡張するか、または無効にすることができます。

VSH スクリプトポリシーを定義したら、それをデバイスにコピーしてアクティブにします。

Embedded Event Manager の前提条件

EEM を設定するには、network-admin の権限が必要です。

Embedded Event Manager の注意事項および制約事項

EEM の設定を計画するときは、次の点を考慮します。

- 設定可能な EEM ポリシーの最大数は 500 です。
- ユーザポリシーまたは上書きポリシー内のアクション文が、相互に否定したり、関連付けられたシステムポリシーに悪影響を与えたりするようなことがないようにする必要があります。
- 発生したイベントでデフォルトのアクションを処理できるようにするには、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。たとえば、一致文でコマンドを照合する場合、EEM ポリシーに `event-default` アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。
- イベント文が指定されていて、アクション文が指定されていない上書きポリシーを設定した場合、アクションは開始されません。また、障害も通知されません。
- 上書きポリシーにイベント文が含まれていないと、システムポリシーで可能性のあるイベントがすべて上書きされます。
- 通常コマンドの表現の場合：すべてのキーワードを拡張する必要があり、アスタリスク (*) 記号のみが引数の置換に使用できます。
- EEM イベント相関は 1 つのポリシーに最大 4 つのイベント文をサポートします。イベントタイプは同じでも別でもかまいませんが、サポートされるイベントタイプは、cli、カウンタ、snmp、syslog、追跡だけです。
- 複数のイベント文が EEM ポリシーに存在する場合は、各イベント文に `tag` キーワードと一意な `tag` 引数が必要です。
- EEM イベント相関はシステムのデフォルト ポリシーを上書きしません。
- デフォルトアクション実行は、タグ付きのイベントで設定されているポリシーではサポートされません。
- イベント指定が CLI のパターンと一致する場合、SSH 形式のワイルドカード文字を使用できます。
たとえば、すべての `show` コマンドを照合する場合は、`show *` コマンドを入力します。
`show . *` コマンドを入力すると、機能しません。
- イベント指定が一致する syslog メッセージの正規表現の場合、適切な正規表現を使用できます。

たとえば、syslog が生成されているポート上で ADMIN_DOWN イベントを検出するには、**.ADMIN_DOWN.** を使用します。ADMIN_DOWN コマンドを入力すると、機能しません。

- syslog のイベント指定では、regex は、EEM ポリシーのアクションとして生成される syslog メッセージと一致しません。
- EEM イベントが CLI の **show** コマンドと一致し、画面に表示するために（および EEM ポリシーによってブロックされないために）**show** コマンドの出力が必要な場合は、EEM ポリシーの最初のアクションに対して、**event-default** コマンドを指定する必要があります。
- Cisco Nexus 3500 シリーズ スイッチは、Cisco NX-OS リリース 7.0(3)I7(2) およびそれ以前のリリースの Embedded Event Manager をサポートしていません。

Embedded Event Manager のデフォルト設定

表 1: デフォルトの EEM パラメータ

パラメータ	デフォルト
システム ポリシー	アクティブ

環境変数の定義

環境変数の定義はオプションの手順ですが、複数のポリシーで繰り返し使用する共通の値を設定する場合に役立ちます。

手順の概要

1. **configure terminal**
2. **event manager environment** *variable-name variable-value*
3. (任意) **show event manager environment** {*variable-name* | **all**}
4. (任意) **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： switch# configure terminal switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	event manager environment <i>variable-name variable-value</i>	EEM 用の環境変数を作成します。

	コマンドまたはアクション	目的
	例 : <pre>switch(config) # event manager environment emailto "admin@anyplace.com"</pre>	<i>variable-name</i> は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。 <i>variable-value</i> は大文字と小文字が区別され、引用符で囲んだ最大 39 文字の英数字を使用できます。
ステップ 3	(任意) show event manager environment <i>{variable-name all}</i> 例 : <pre>switch(config) # show event manager environment all</pre>	設定した環境変数に関する情報を表示します。
ステップ 4	(任意) copy running-config startup-config 例 : <pre>switch(config)# copy running-config startup-config</pre>	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

ユーザー ポリシーを設定します。

CLI によるユーザ ポリシーの定義

手順の概要

1. **configure terminal**
2. **event manager applet** *applet-name*
3. (任意) **description** *policy-description*
4. **event** *event-statement*
5. (任意) **tag** *tag* {**and** | **andnot** | **or**} *tag* [**and** | **andnot** | **or** {*tag*}] {**happens occurs in seconds**}
6. **action** *number*[.*number2*] *action-statement*
7. (任意) **show event manager policy-state** *name* [**module** *module-id*]
8. (任意) **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : <pre>switch# configure terminal switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 2	event manager applet <i>applet-name</i> 例： switch(config)# event manager applet monitorShutdown switch(config-applet)#	EEM にアプレットを登録し、アプレット コンフィギュレーションモードを開始します。 applet-name は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。
ステップ 3	(任意) description <i>policy-description</i> 例： switch(config-applet)# description "Monitors interface shutdown."	ポリシーの説明になるストリングを設定します。 string には最大 80 文字の英数字を使用できます。ストリングは引用符で囲みます。
ステップ 4	event <i>event-statement</i> 例： switch(config-applet)# event cli match "shutdown"	ポリシーのイベント文を設定します。
ステップ 5	(任意) tag <i>tag</i> { and andnot or } <i>tag</i> [and andnot or { <i>tag</i> }] { happens occurs in seconds } 例： switch(config-applet)# tag one or two happens 1 in 10000	ポリシー内の複数のイベントを相互に関連付けます。 occurs 引数の範囲は 1 ~ 4294967295 です。 seconds 引数の範囲は 0 ~ 4294967295 秒です。
ステップ 6	action <i>number</i> [<i>number2</i>] <i>action-statement</i> 例： switch(config-applet)# action 1.0 cli show interface e 3/1	ポリシーのアクション文を設定します。アクション文が複数ある場合、このステップを繰り返します。
ステップ 7	(任意) show event manager policy-state <i>name</i> [module <i>module-id</i>] 例： switch(config-applet)# show event manager policy-state monitorShutdown	設定したポリシーの状態に関する情報を表示します。
ステップ 8	(任意) copy running-config startup-config 例： switch(config)# copy running-config startup-config	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

イベント文およびアクション文を設定します。

イベント文の設定

イベント文を設定するには、EEM コンフィギュレーションモード (config-applet) で次のいずれかのコマンドを使用します。

始める前に

ユーザー ポリシーを定義します。

手順の概要

1. **event cli** [tag tag] match expression [count repeats | time seconds
2. **event counter** [tag tag] name counter entry-val entry entry-op {eq | ge | gt | le | lt | ne} { exit-val exit exit-op {eq | ge | gt | le | lt | ne}
3. **event fanabsent** [fan number] time seconds
4. **event fanbad** [fan number] time seconds
5. **event memory** {critical | minor | severe}
6. **event policy-default** count repeats [time seconds]
7. **event snmp** [tag tag] oid oid get-type {exact | next} entry-op {eq | ge | gt | le | lt | ne} entry-val entry [exit-comb {and | or}]exit-op {eq | ge | gt | le | lt | ne} exit-val exit exit-time time polling-interval interval
8. **event sysmgr memory** [module module-num] major major-percent minor minor-percent clear clear-percent
9. **event temperature** [module slot] [sensor number] threshold {any | down | up}
10. **event track** [tag tag] object-number state {any | down | up

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	event cli [tag tag] match expression [count repeats time seconds 例 : <pre>switch(config-applet) # event cli match "shutdown"</pre>	正規表現と一致するコマンドが入力された場合に、イベントを発生させます。 tag tag キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。 <i>repeats</i> の範囲は 1 ~ 65000 です。 <i>time</i> の範囲は 0 ~ 4294967295 です。0 は無制限を示します。
ステップ 2	event counter [tag tag] name counter entry-val entry entry-op {eq ge gt le lt ne} { exit-val exit exit-op {eq ge gt le lt ne} 例 : <pre>switch(config-applet) # event counter name mycounter entry-val 20 gt</pre>	カウンタが、開始演算子に基づいて開始のしきい値を超えた場合にイベントを発生させます。イベントはただちにリセットされます。任意で、カウンタが終了のしきい値を超えたあとでリセットされるように、イベントを設定できます。

	コマンドまたはアクション	目的
		<p>tag tag キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。</p> <p>counter name は大文字と小文字を区別し、最大 28 の英数字を使用できます。</p> <p>entry および exit の値の範囲は 0 ~ 2147483647 です。</p>
ステップ 3	<p>event fanabsent [fan number] time seconds</p> <p>例 :</p> <pre>switch(config-applet) # event fanabsent time 300</pre>	<p>秒数で設定された時間を超えて、ファンがデバイスから取り外されている場合に、イベントを発生させます。</p> <p>number の範囲はモジュールに依存します。</p> <p>seconds の範囲は 10 ~ 64000 です。</p>
ステップ 4	<p>event fanbad [fan number] time seconds</p> <p>例 :</p> <pre>switch(config-applet) # event fanbad time 3000</pre>	<p>秒数で設定された時間を超えて、ファンが故障状態の場合に、イベントを発生させます。</p> <p>number の範囲はモジュールに依存します。</p> <p>seconds の範囲は 10 ~ 64000 です。</p>
ステップ 5	<p>event memory {critical minor severe}</p> <p>例 :</p> <pre>switch(config-applet) # event memory critical</pre>	<p>メモリのしきい値を超えた場合にイベントを発生させます。</p>
ステップ 6	<p>event policy-default count repeats [time seconds]</p> <p>例 :</p> <pre>switch(config-applet) # event policy-default count 3</pre>	<p>システム ポリシーで設定されているイベントを使用します。このオプションは、ポリシーを上書きする場合に使用します。</p> <p>repeats の範囲は 1 ~ 65000 です。</p> <p>seconds の範囲は 0 ~ 4294967295 秒です。0 は無制限を示します。</p>
ステップ 7	<p>event snmp [tag tag] oid oid get-type {exact next} entry-op {eq ge gt le lt ne} entry-val entry [exit-comb {and or}]exit-op {eq ge gt le lt ne} exit-val exit exit-time time polling-interval interval</p> <p>例 :</p> <pre>switch(config-applet) # event snmp oid 1.3.6.1.2.1.31.1.1.1.6 get-type next entry-op lt 300 entry-val 0 exit-op eq 400 exit-time 30 polling-interval 300</pre>	<p>SNMPOID が、開始演算子に基づいて開始のしきい値を超えた場合にイベントを発生させます。イベントはただちにリセットされます。または任意で、カウンタが終了のしきい値を超えたあとでリセットされるように、イベントを設定できます。OID はドット付き 10 進表記です。</p> <p>tag tag キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。</p>

	コマンドまたはアクション	目的
		<p><i>entry</i> および <i>exit</i> の値の範囲は 0 ~ 18446744073709551615 です。</p> <p><i>time</i> の範囲は 0 ~ 2147483647 秒です。</p> <p><i>interval</i> の範囲は 0 ~ 2147483647 秒です。</p>
ステップ 8	<p>event sysmgr memory [module <i>module-num</i>] major <i>major-percent</i> minor <i>minor-percent</i> clear <i>clear-percent</i></p> <p>例 :</p> <pre>switch(config-applet) # event sysmgr memory minor 80</pre>	<p>指定したシステム マネージャのメモリのしきい値を超えた場合にイベントを発生させます。</p> <p><i>percent</i> の範囲は 1 ~ 99 です。</p>
ステップ 9	<p>event temperature [module <i>slot</i>] [sensor <i>number</i>] threshold {any down up}</p> <p>例 :</p> <pre>switch(config-applet) # event temperature module 2 threshold any</pre>	<p>温度センサーが設定されたしきい値を超えた場合に、イベントを発生させます。</p> <p><i>sensor</i> の範囲は 1 ~ 18 です。</p>
ステップ 10	<p>event track [tag <i>tag</i>] object-number state {any down up}</p> <p>例 :</p> <pre>switch(config-applet) # event track 1 state down</pre>	<p>トラッキング対象オブジェクトが設定された状態になった場合に、イベントを発生させます。</p> <p>tag <i>tag</i> キーワードと引数のペアは、複数のイベントがポリシーに含まれている場合、この特定のイベントを識別します。</p> <p>指定できる <i>object-number</i> の範囲は 1 ~ 500 です。</p>

次のタスク

アクション文を設定します。

すでにアクション文を設定した場合、または設定しないことを選択した場合は、次のオプション作業のいずれかを実行します。

- VSH スクリプトを使用してポリシーを定義します。その後、VSH スクリプト ポリシーを登録し、アクティブにします。
- メモリのしきい値を設定します。
- EEM パブリッシャとして syslog を設定します。
- EEM 設定を確認します。

アクション文の設定

EEM のコンフィギュレーションモード (`config-applet`) で次のいずれかのコマンドを使用して、アクションを設定できます。



- (注) 発生したイベントでデフォルトのアクションを処理できるようにする場合は、デフォルトのアクションを許可する EEM ポリシーを設定する必要があります。たとえば、一致文でコマンドを照合する場合、EEM ポリシーに `event-default` アクション文を追加する必要があります。この文がないと、EEM ではコマンドを実行できません。 `terminal event-manager bypass` コマンドを使用すると、一致するすべての EEM ポリシーでコマンドを実行できます。

始める前に

ユーザー ポリシーを定義します。

手順の概要

1. `action number[.number2] cli command1[command2.] [local]`
2. `action number[.number2] counter name counter value val op {dec | inc | nop | set}`
3. `action number[.number2] event-default`
4. `action number[.number2] policy-default`
5. `action number[.number2] reload [module slot [- slot]]`
6. `action number[.number2] snmp-trap [intdata1 integer-data1] [intdata2 integer-data2] [strdata string-data]`
7. `action number[.number2] syslog [priority prio-val] msg error-message`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>action number[.number2] cli command1[command2.] [local]</code> 例： <pre>switch(config-applet) # action 1.0 cli "show interface e 3/1"</pre>	設定済みコマンドを実行します。任意で、イベントが発生したモジュール上でコマンドを実行できます。 アクションラベルのフォーマットは <code>number1.number2</code> です。 <code>number</code> には 1～16 桁の任意の番号を指定できます。 <code>number2</code> の範囲は 0～9 です。
ステップ 2	<code>action number[.number2] counter name counter value val op {dec inc nop set}</code> 例： <pre>switch(config-applet) # action 2.0 counter name mycounter value 20 op inc</pre>	設定された値および操作でカウンタを変更します。 アクションラベルのフォーマットは <code>number1.number2</code> です。 <code>number</code> には 1～16 桁の任意の番号を指定できます。 <code>number2</code> の範囲は 0～9 です。 <code>counter</code> は大文字と小文字を区別し、最大 28 文字の英数字を使用できます。

	コマンドまたはアクション	目的
		<i>val</i> には 0 ~ 2147483647 の整数または置換パラメータを指定できます。
ステップ 3	action number[.number2] event-default 例： <pre>switch(config-applet) # action 1.0 event-default</pre>	関連付けられたイベントのデフォルトアクションを実行します。 アクションラベルのフォーマットは <i>number1.number2</i> です。 <i>number</i> には 1 ~ 16 桁の任意の番号を指定できます。 <i>number2</i> の範囲は 0 ~ 9 です。
ステップ 4	action number[.number2] policy-default 例： <pre>switch(config-applet) # action 1.0 policy-default</pre>	上書きしているポリシーのデフォルトアクションを実行します。 アクションラベルのフォーマットは <i>number1.number2</i> です。 <i>number</i> には 1 ~ 16 桁の任意の番号を指定できます。 <i>number2</i> の範囲は 0 ~ 9 です。
ステップ 5	action number[.number2] reload [module slot [- slot]] 例： <pre>switch(config-applet) # action 1.0 reload module 3-5</pre>	システム全体に 1 つ以上のモジュールをリロードします。 アクションラベルのフォーマットは <i>number1.number2</i> です。 <i>number</i> には 1 ~ 16 桁の任意の番号を指定できます。 <i>number2</i> の範囲は 0 ~ 9 です。
ステップ 6	action number[.number2] snmp-trap [intdata1 integer-data1] [intdata2 integer-data2] [strdata string-data] 例： <pre>switch(config-applet) # action 1.0 snmp-trap strdata "temperature problem"</pre>	設定されたデータを使用して SNMP トラップを送信します。アクションラベルのフォーマットは <i>number1.number2</i> です。 <i>number</i> には 1 ~ 16 桁の任意の番号を指定できます。 <i>number2</i> の範囲は 0 ~ 9 です。 <i>data</i> 要素には 80 桁までの任意の数を指定できます。 <i>string</i> には最大 80 文字の英数字を使用できます。
ステップ 7	action number[.number2] syslog [priority prio-val] msg error-message 例： <pre>switch(config-applet) # action 1.0 syslog priority notifications msg "cpu high"</pre>	設定されたプライオリティで、カスタマイズした syslog メッセージを送信します。 アクションラベルのフォーマットは <i>number1.number2</i> です。 <i>number</i> には 1 ~ 16 桁の任意の番号を指定できます。 <i>number2</i> の範囲は 0 ~ 9 です。

	コマンドまたはアクション	目的
		<i>error-message</i> には最大 80 文字の英数字を引用符で囲んで使用できます。

次のタスク

イベント文を設定します。

すでにイベント文を設定した場合、または設定しないことを選択した場合は、次のオプション作業のいずれかを実行します。

- VSH スクリプトを使用してポリシーを定義します。その後、VSH スクリプト ポリシーを登録し、アクティブにします。
- メモリのしきい値を設定します。
- EEM パブリッシャとして syslog を設定します。
- EEM 設定を確認します。

VSH スクリプトによるポリシーの定義

これはオプションのタスクです。VSH スクリプトを使用して EEM ポリシーを記述する場合は、次の手順を実行します。

手順の概要

1. テキスト エディタで、ポリシーを定義するコマンドリストを指定します。
2. テキスト ファイルに名前をつけて保存します。
3. 次のシステム ディレクトリにファイルをコピーします。bootflash://eem/user_script_policies

手順の詳細

ステップ 1 テキスト エディタで、ポリシーを定義するコマンドリストを指定します。

ステップ 2 テキスト ファイルに名前をつけて保存します。

ステップ 3 次のシステム ディレクトリにファイルをコピーします。bootflash://eem/user_script_policies

次のタスク

VSH スクリプト ポリシーを登録してアクティブにします。

VSH スクリプト ポリシーの登録およびアクティブ化

これはオプションのタスクです。VSH スクリプトを使用して EEM ポリシーを記述する場合は、次の手順を実行します。

始める前に

ポリシーを VSH スクリプトを使用して定義し、システム ディレクトリにファイルをコピーします。

手順の概要

1. **configure terminal**
2. **event manager policy *policy-script***
3. (任意) **event manager policy internal *name***
4. (任意) **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： <pre>switch# configure terminal switch(config)#</pre>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	event manager policy <i>policy-script</i> 例： <pre>switch(config)# event manager policy moduleScript</pre>	EEM スクリプト ポリシーを登録してアクティブにします。 <i>policy-script</i> は大文字と小文字を区別し、最大 29 文字の英数字を使用できます。
ステップ 3	(任意) event manager policy internal <i>name</i> 例： <pre>switch(config)# event manager policy internal moduleScript</pre>	EEM スクリプト ポリシーを登録してアクティブにします。 <i>policy-script</i> は大文字と小文字を区別し、最大 29 の英数字を使用できます。
ステップ 4	(任意) copy running-config startup-config 例： <pre>switch(config)# copy running-config startup-config</pre>	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

システム要件に応じて、次のいずれかを実行します。

- メモリのしきい値を設定します。

- EEM パブリッシャとして syslog を設定します。
- EEM 設定を確認します。

システム ポリシーの上書き

手順の概要

1. **configure terminal**
2. (任意) **show event manager policy-state system-policy**
3. **event manager applet applet-name override system-policy**
4. **description policy-description**
5. **event event-statement**
6. **section number action-statement**
7. (任意) **show event manager policy-state name**
8. (任意) **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： switch# configure terminal switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	(任意) show event manager policy-state system-policy 例： switch(config-applet)# show event manager policy-state __ethpm_link_flap Policy __ethpm_link_flap Cfg count : 5 Cfg time interval : 10.000000 (seconds) Hash default, Count 0	上書きするシステムポリシーの情報をしきい値を含めて表示します。 show event manager system-policy コマンドを使用して、システムポリシーの名前を探します。
ステップ 3	event manager applet applet-name override system-policy 例： switch(config-applet)# event manager applet ethport override __ethpm_link_flap switch(config-applet)#	システムポリシーを上書きし、アプレットコンフィギュレーションモードを開始します。 <i>applet-name</i> は大文字と小文字を区別し、最大 80 文字の英数字を使用できます。 <i>system-policy</i> は、システムポリシーの 1 つにする必要があります。
ステップ 4	description policy-description 例：	ポリシーの説明になるストリングを設定します。

	コマンドまたはアクション	目的
	switch(config-applet)# description "Overrides link flap policy"	<i>policy-description</i> は大文字と小文字を区別し、最大 80 文字の英数字を使用できますが、引用符で囲む必要があります。
ステップ 5	event <i>event-statement</i> 例： switch(config-applet)# event policy-default count 2 time 1000	ポリシーのイベント文を設定します。
ステップ 6	section <i>number action-statement</i> 例： switch(config-applet)# action 1.0 syslog priority warnings msg "Link is flapping."	ポリシーのアクション文を設定します。複数のアクション文では、この手順を繰り返します。
ステップ 7	(任意) show event manager policy-state <i>name</i> 例： switch(config-applet)# show event manager policy-state ethport	設定したポリシーに関する情報を表示します。
ステップ 8	(任意) copy running-config startup-config 例： switch(config)# copy running-config startup-config	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

EEM パブリッシャとしての syslog の設定

EEM パブリッシャとして syslog を設定すると、スイッチから syslog メッセージをモニターできます。



(注) syslog メッセージをモニターする検索文字列の最大数は 10 です。

始める前に

- EEM が syslog による登録で利用できることを確認します。
- syslog デーモンが設定され、実行されていることを確認します。

手順の概要

1. **configure terminal**
2. **event manager applet** *applet-name*

3. **event syslog** [**tag tag**] { **occurs number** | **period seconds** | **pattern msg-text** | **priority priority**}
4. (任意) **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例： switch# configure terminal switch(config)#	グローバル コンフィギュレーション モードを開始します。
ステップ 2	event manager applet applet-name 例： switch(config)# event manager applet abc switch (config-appliet)#	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 3	event syslog [tag tag] { occurs number period seconds pattern msg-text priority priority } 例： switch(config-appliet)# event syslog occurs 10	EEM にアプレットを登録し、アプレット コンフィギュレーション モードを開始します。
ステップ 4	(任意) copy running-config startup-config 例： switch(config)# copy running-config startup-config	リブートおよびリスタート時に実行コンフィギュレーションをスタートアップコンフィギュレーションにコピーして、変更を継続的に保存します。

次のタスク

EEM 設定を確認します。

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。