



## E から event manager までのコマンド

---

- [E から event manager までのコマンド, 1 ページ](#)

## E から event manager までのコマンド

## event application

EEM イベントパブリッシュアプリケーションプログラミングインターフェイス (API) 経由で発生したイベント単位で起動される Embedded Event Manager (EEM) アプレットのイベント条件を指定するには、アプレットコンフィギュレーションモードで **event application** コマンドを使用します。アプリケーションイベント条件を削除するには、このコマンドの **no** 形式を使用します。

**event** [**tag** *event-tag*] **application** **subsystem** *subsystem-id* **type** *event-type* [**maxrun** *maxruntime-number*]  
**no** [**tag** *event-tag*] **event application** **subsystem** *subsystem-id* **type** *event-type* [**maxrun** *maxruntime-number*]

### 構文の説明

<b>tag</b>	(任意) <b>trigger</b> コマンドで使用できる <i>event-tag</i> 引数を使用してタグを指定し、アプレットの複数のイベント文をサポートします。
<i>event-tag</i>	(任意) タグを識別するストリング。
<b>2subsystem</b>	アプリケーションイベントをパブリッシュするサブシステムの ID を指定します。
<i>subsystem-id</i>	サブシステムを識別する 1 ~ 4294967295 の範囲の番号。EEM ポリシーによりイベントがパブリッシュされる場合、ポリシー用に予約されている <i>subsystem-id</i> は 798 です。
<b>type</b>	指定されたイベント内のイベントタイプを指定します。
<i>event-type</i>	1 ~ 4294967295 の範囲の整数。
<b>maxrun</b>	(任意) アプレットの最大実行時間を指定します。 <b>maxrun</b> キーワードが指定されている場合、 <i>maxruntime-number</i> 値を指定する必要があります。 <b>maxrun</b> キーワードが指定されていない場合、デフォルトのアプレットランタイムは 20 秒です。
<i>maxruntime-number</i>	(任意) <i>sssssss[.mmm]</i> 形式で指定される秒数。 <i>sssssss</i> は、0 ~ 31536000 の秒数を表す整数で、 <i>mmm</i> は 0 ~ 999 のミリ秒数を表す整数である必要があります。

コマンド モデル **EEM** イベント標準は指定されなければ (**config-applet**)

## コマンド履歴

リリース	変更内容
12.2(25)S	このコマンドが導入されました。
12.3(14)T	このコマンドが、Cisco IOS Release 12.3(14)T に統合されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェア モジュラリティ イメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされます。このトレインの特定の 12.2SX リリースにおけるサポートは、フィチャセット、プラットフォーム、およびプラットフォームハードウェアによって異なります。
12.4(20)T	<b>tag</b> および <b>maxrun</b> キーワードが、アプレット内の複数のイベント文をサポートするために追加されました。

## 使用上のガイドライン

EEM イベントは、アプリケーションがサブシステム ID およびアプリケーション イベントタイプに一致するイベント仕様で EEM イベント パブリッシュ API をコールしたときにトリガーされます。

## 例

次に、EventPublish\_A という名前のポリシーが 20 秒ごとに実行され、番号が 1 の既知の EEM イベントタイプにイベントをパブリッシュする例を示します。EventPublish\_B という名前の別のポリシーは、既知の EEM イベントタイプ 1 が発生したときに実行されるように登録されます。EventPublish\_B ポリシーが実行されると、このポリシーにより EventPublish\_A の引数として渡されたデータを含む syslog に、メッセージが出力されます。

```
Router(config)# event manager applet EventPublish_A
Router(config-applet)# event timer watchdog time 20.0
Router(config-applet)# action 1.0 syslog msg "Applet EventPublish_A"
Router(config-applet)# action 2.0 publish-event sub-system 798 type 1 arg1 twenty
Router(config-applet)# exit
Router(config)# event manager applet EventPublish_B
Router(config-applet)# event application subsystem 798 type 1
Router(config-applet)# action 1.0 syslog msg "Applet EventPublish_B arg1
$_application_data1"
```

## 関連コマンド

コマンド	説明
<b>event manager applet</b>	Embedded Event Manager にイベント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。

## event identity

認証、許可、またはインターフェイス上で通常のトラフィックが流れ始めた後にイベントをパブリッシュするには、アプレット コンフィギュレーション モードで **event identity** コマンドを使用します。イベントのパブリッシュをディセーブルにするには、このコマンドの **no** 形式を使用します。

```
event [tag event-tag] identity interface {type number|regexp interface-name} [maxrun maxruntime-number]
[aaa-attribute attribute-name] [authc {all|fail|success}] [authz {all|fail|success}] [authc-complete]
[mac-address mac-address]
```

```
no event identity
```

### 構文の説明

<b>tag</b>	(任意) trigger コマンドで使用できる event-tag 引数を使用してタグを指定し、アプレットの複数のイベント文をサポートします。
<i>event-tag</i>	(任意) タグを識別するストリング。
<b>interface</b>	インターフェイスを指定します。
<i>type number</i>	インターフェイスのタイプと番号。
<b>regexp</b> <i>interface-name</i>	インターフェイス名と照合する正規表現パターンを指定します。
<b>maxrun</b>	(任意) アプレットの最大実行時間を指定します。maxrun キーワードが指定されている場合、maxruntime-number 値を指定する必要があります。maxrun キーワードが指定されていない場合、デフォルトのアプレットランタイムは 20 秒です。
<i>maxruntime-number</i>	(任意) ssssssss[.mmm] 形式で指定される秒数。ssssssss は、0 ~ 31536000 の秒数を表す整数で、mmm は 0 ~ 999 のミリ秒数を表す整数である必要があります。
<b>aaa-attribute</b>	(任意) AAA 属性用の正規表現パターンを指定します。
<i>attribute-name</i>	(任意) AAA 属性名。

<b>authc</b>	<p>(任意) 成功した認証、失敗した認証、または成功と失敗の両方の認証で、イベントをトリガーします。次のいずれかを指定する必要があります。</p> <ul style="list-style-type: none"> <li>• <b>all</b> : すべてのケースの認証でイベントをトリガーします。</li> <li>• <b>fail</b> : 認証が失敗した場合にイベントをトリガーします。</li> <li>• <b>success</b> : 認証が成功した場合にイベントをトリガーします。</li> </ul>
<b>authz</b>	<p>(任意) 成功した許可、失敗した許可、または成功と失敗の両方の許可で、イベントをトリガーします。次のいずれかを指定する必要があります。</p> <ul style="list-style-type: none"> <li>• <b>all</b> : すべてのケースの許可でイベントをトリガーします。</li> <li>• <b>fail</b> : 許可が失敗した場合にイベントをトリガーします。</li> <li>• <b>success</b> : 許可が成功した場合にイベントをトリガーします。</li> </ul>
<b>authz-complete</b>	<p>(任意) インターフェイスに接続されたデバイスが完全に認証、許可され、通常のトラフィックがそのインターフェイスで流れ始めたときにイベントをトリガーします。</p>
<b>mac-address</b>	<p>(任意) MAC アドレスを指定します。</p>
<i>mac-address</i>	<p>(任意) MAC アドレス。</p>

コマンド デフォルト      デフォルトでは、イベントはパブリッシュされません。

コマンド モード      アプレット コンフィギュレーション (config-applet)

## コマンド履歴

リリース	変更内容
12.2(52)SE	このコマンドが導入されました。
12.2(54)SG	このコマンドが、Cisco IOS Release 12.2(54)SG に統合されました。

## 使用上のガイドライン

インターフェイスを指定する必要があります。任意またはすべての他のキーワードを指定できません。キーワードは任意の組み合わせで使用できます。

## 例

次に、認証が成功または失敗し、インターフェイスに接続されているデバイスが完全に認証、許可され、そのインターフェイスで通常のトラフィックが流れ始めたときにイベントをパブリッシュする例を示します。

```
Router(config)# event manager applet identity
Router(config-applet)# event identity interface fastethernet0 authz all athuz-complete
Router(config-applet)#
```

## 関連コマンド

コマンド	説明
<b>event manager applet</b>	Embedded Event Manager にイベント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。

## event ipsla

Embedded Event Manager (EEM) のアプレットに対して IP SLA 操作がトリガーされたときにイベントをパブリッシュするには、アプレットコンフィギュレーションモードで **event ipsla** コマンドを使用します。IP SLA の応答がトリガーされたときのイベントのパブリッシュをディセーブルにするには、このコマンドの **no** 形式を使用します。

**event** [**tag** *event-tag*] **ipsla** {**group-name** *name* [**operation-id** *operation-id-value*]} **operation-id** *operation-id-value* [**group-name** *name*] [**dest-ip-address** *ip-address*] [**reaction-type** *type*] [**maxrun** *maxruntime-number*]

**no event** [**tag** *event-tag*] **ipsla**

### 構文の説明

<b>tag</b>	(任意) <b>trigger</b> コマンドで使用できる <i>event-tag</i> 引数を使用してタグを指定し、アプレットの複数のイベント文をサポートします。
<i>event-tag</i>	(任意) タグを識別するストリング。
<b>group-name</b>	IP SLA グループ ID を指定します。
<i>name</i>	IP SLA グループの名前。
<b>operation-id</b>	IP SLA 動作 ID を指定します。
<i>operation-id-value</i>	番号の範囲は 1 ~ 2147483647 です。
<b>dest-ip-address</b>	(任意) IP SLA イベントがモニタされる宛先 IP アドレスを指定します。
<i>ip-address</i>	(任意) 宛先ポートの IP アドレスを指定します。
<b>reaction-type</b>	(任意) 指定した IP SLA 動作に対する応答を指定します。



<i>type</i>	<p>(任意) IPSLA の応答のタイプ。次のキーワードの 1 つを指定できます。</p> <ul style="list-style-type: none"><li>• <b>connectionLoss</b> : 監視対象動作で一方向接続損失があった場合に、応答が発生することを指定します。</li><li>• <b>icpif</b> : 一方向の Calculated Planning Impairment Factor (ICPIF) の値が上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。</li><li>• <b>jitterAvg</b> : 平均ラウンドトリップジッター値が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。</li><li>• <b>jitterDSAvg</b> : 平均の一方向の (宛先から送信元) ジッター値が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。</li><li>• <b>jitterSDAvg</b> : 平均の一方向の (送信元から宛先) ジッター値が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。</li><li>• <b>maxOfNegativeDS</b> : 一方向の (宛先から送信元) 最大の負のジッターしきい値に違反すると、応答が発生することを指定します。</li><li>• <b>maxOfNegativeSD</b> : 一方向の (送信元から宛先) 最大の負のジッターしきい値に違反すると、応答が発生することを指定します。</li></ul>
-------------	--

event ipsla



- **maxOfPositiveDS** : 一方向の（宛先から送信元）最大の正のジッターしきい値に違反すると、応答が発生することを指定します。
- **maxOfPositiveSD** : 一方向の（送信元から宛先）最大の正のジッターしきい値に違反すると、応答が発生することを指定します。
- **mos** : 一方向の平均オピニオン評点（MOS）値が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **packetLateArrival** : 一方向の遅延パケット数が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **packetLossDS** : 一方向の（宛先から送信元）パケット損失値が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **packetLossSD** : 一方向の（送信元から宛先）パケット損失値が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **packetMIA** : 一方向の不明パケット数が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **packetOutOfSequence** : 一方向のシーケンスを外れたパケット数が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **rtt** : ラウンドトリップ時間が、上限しきい値または下限しきい値に違反すると、応答が発生することを指定します。
- **timeout** : 監視対象動作で一方向タイムアウトがあった場合に、応答が発生することを指定します。
- **verifyError** : 一方向のエラー検証違反があった場合に、応答が発生することを指定

	します。
<b>maxrun</b>	(任意) アプレットの最大実行時間を指定します。 <b>maxrun</b> キーワードが指定されている場合、 <i>maxruntime-number</i> 値を指定する必要があります。 <b>maxrun</b> キーワードが指定されていない場合、デフォルトのアプレットランタイムは 20 秒です。
<i>maxruntime-number</i>	(任意) <i>sssssss[.mmm]</i> 形式で指定される秒数。 <i>sssssss</i> は、0～31536000 の秒数を表す整数で、 <i>mmm</i> は 0～999 のミリ秒数を表す整数である必要があります。

**コマンド デフォルト**

イベントは IP SLA 動作がトリガーされたときにパブリッシュされません。

**コマンド モード**

アプレット コンフィギュレーション (config-applet)

**コマンド履歴**

リリース	変更内容
12.4(22)T	このコマンドが導入されました。
12.2(33)SRE	このコマンドが、Cisco IOS Release 12.2(33)SRE に統合されました。

**使用上のガイドライン**

EEM イベントは、IP SLA 応答がトリガーされるときにパブリッシュされます。**group-name** または **operation-id** のいずれかを指定する必要があります。残りのパラメータはオプションです。

**例**

次に、IP SLA 動作がトリガーされたときにイベントをパブリッシュする例を示します。この例では、**grp1** という名前のグループが、3 秒ごとに現在のインターフェイスを経由して宛先サーバ 209.165.200.221 に ping を送信します。応答がない場合、アクションはタイムアウトになります。

```
Router# configure terminal
Router(config)# event manager applet EventIPSLA
Router(config-applet)# event ipsla group-name grp1 dest-ip-address 209.165.200.221
reaction-type timeout maxrun 3
```

## 関連コマンド

コマンド	説明
<b>event manager applet</b>	EEM にイベント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。

## event manager applet

Embedded Event Manager (EEM) にアプレットを登録してアプレット コンフィギュレーション モードを開始するには、グローバル コンフィギュレーション モードで **event manager applet** コマンドを使用します。アプレットを登録解除するには、このコマンドの **no** 形式を使用します。

**event manager applet** *applet-name* [**authorization bypass**] [**class class-options**] [**trap**]

**no event manager applet** *applet-name* [**authorization bypass**] [**class class-options**] [**trap**]

### 構文の説明

<i>applet-name</i>	アプレット ファイルの名前。
<b>authorization</b>	(任意) アプレットの AAA 許可タイプを指定します。
<b>bypass</b>	(任意) EEM の AAA 許可タイプのバイパスを指定します。
<b>class</b>	(任意) EEM ポリシー クラスを指定します。
<i>class-options</i>	(任意) EEM ポリシー クラス。次のいずれかを指定できます： <ul style="list-style-type: none"> <li>• <b>class-letter</b> : 各ポリシー クラスを識別する A ~ Z の文字。任意の <b>class-letter</b> を 1 つ 指定できます。</li> <li>• <b>default</b> : デフォルトクラスに登録されたポリシーを指定します。</li> </ul>
<b>trap</b>	(任意) ポリシーがトリガーされたときに簡易 ネットワーク管理プロトコル (SNMP) トラップを生成します。

コマンド デフォルト EEM アプレットは登録されません。

コマンド モード グローバル コンフィギュレーション (config)

## コマンド履歴

リリース	変更内容
12.0(26)S	このコマンドが導入されました。
12.3(4)T	このコマンドが Cisco IOS Release 12.3(4)T に統合されました。
12.3(2)XE	このコマンドが、Cisco IOS Release 12.3(2)XE に統合されました。
12.2(25)S	このコマンドが、Cisco IOS Release 12.2(25)S に統合されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェアモジュラリティイメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされません。このトレインの特定の 12.2SX リリースにおけるサポートは、フィーチャセット、プラットフォーム、およびプラットフォームハードウェアによって異なります。
12.4(22)T	<b>class</b> および <b>trap</b> キーワードと <i>class-options</i> 引数が追加されました。
15.0(1)M	コマンドが変更されました。 <b>authorization</b> および <b>bypass</b> キーワードが追加されました。

## 使用上のガイドライン

EEM アプレットは、イベントスクリーニング基準とイベント発生時に実行するアクションを定義する簡潔な方法です。

アプレット コンフィギュレーションでは、**event** コンフィギュレーション コマンドを 1 つだけ使用できます。アプレット コンフィギュレーション サブモードが終了し、**event** コマンドが存在しない場合は、アプレットにイベントが関連付けられていないことを示す警告が表示されます。イベントが指定されていない場合、このアプレットは登録されたと判断されないため、アプレットは表示されません。このアプレットにアクションが割り当てられない場合、イベントはトリガーされますが、アクションは実行されません。1 つのアプレット コンフィギュレーション内で複数の **action** アプレット コンフィギュレーション コマンドが使用できます。 **show event manager policy registered** コマンドを使用して、登録されたアプレットの一覧を表示します。

アプレット コンフィギュレーション モードを終了しないと既存のアプレットが置き換えられないため、EEM アプレットを変更する前に、このコマンドの **no** 形式を使用して登録を解除します。

アプレットコンフィギュレーションモードでアプレットを修正中であっても、既存のアプレットを実行できます。アプレットコンフィギュレーションモードを終了すると、古いアプレットが登録解除され、新しいバージョンが登録されます。

**action** コンフィギュレーションコマンドは、*label* 引数を使用することで一意に識別できます。*label* 引数には任意の文字列値が使用できます。アクションは、*label* 引数をソートキーとして、英数字のキーの昇順にソートされ、この順序で実行されます。

EEM は、ポリシー自体に含まれているイベントの指定内容に基づいて、ポリシーをスケジューリングおよび実行します。アプレットコンフィギュレーションモードが終了するとき、EEM は、入力された **event** コマンドと **action** コマンドを検査し、指定されたイベントの発生時に実行されるようにアプレットを登録します。

EEM ポリシーは、登録されたときに **class class-letter** が指定されている場合はクラスに割り当てられます。クラスなしで登録された EEM ポリシーは、**default** クラスに割り当てられます。**default** をクラスとして保持するスレッドは、スレッドが作業に利用可能であるとき、デフォルトクラスにサービスを提供します。特定のクラス文字に割り当てられたスレッドは、スレッドが作業に利用可能であるとき、クラス文字が一致する任意のポリシーをサービスします。

EEM 実行スレッドが、指定されたクラスのポリシー実行に利用可能でない場合で、クラスのスケジューラールールが設定されている場合は、ポリシーは該当クラスのスレッドが実行可能になるまで待ちます。同じ入力イベントからトリガーされた同期ポリシーは、同一の実行スレッドにスケジューラールールされなければなりません。ポリシーは、**queue\_priority** をキューイング順序として使用し、各クラスの別々のキューにキューイングされます。

ポリシーがトリガーされると、AAA が設定されている場合は、許可のために AAA サーバに接続します。**authorization bypass** キーワードの組み合わせを使用して、AAA サーバへの接続をスキップし、ポリシーをただちに実行することができます。EEM は、AAA バイパスポリシー名をリストに保存します。このリストは、ポリシーがトリガーされたときに検査されます。一致が見つかった場合、AAA 許可はバイパスされます。

EEM ポリシーによって設定されたコマンドの許可を避けるために、EEM は AAA が提供する名前付き方式リストを使用します。これらの名前付き方式リストは、コマンド許可を持たないように設定できます。

次に、AAA の設定例を示します。

この設定は、192.168.10.1 のポート 10000 に TACACS+ サーバを想定しています。TACACS+ サーバがイネーブルでない場合、コンフィギュレーションコマンドは、コンソールで許可されます。ただし、EEM ポリシーとアプレット CLI の相互作用は失敗します。

```
enable password lab
aaa new-model
tacacs-server host 128.107.164.152 port 10000
tacacs-server key cisco
aaa authentication login consoleline none
aaa authorization exec consoleline none
aaa authorization commands 1 consoleline none
aaa authorization commands 15 consoleline none
line con 0
  exec-timeout 0 0
  login authentication consoleline
aaa authentication login default group tacacs+ enable
aaa authorization exec default group tacacs+
aaa authorization commands 1 default group tacacs+
aaa authorization commands 15 default group tacacs+
```



**authorization**、**class** および **trap** キーワードは任意の組み合わせで使用できます。

## 例

次に、IPSLAping1 という名前の EEM アプレットが登録され、指定された SNMP オブジェクト ID の値と完全一致する（正常な IP SLA ICMP エコー動作を表す）場合に実行される例を示します（これは **ping** コマンドに相当します）。エコー操作が失敗した場合は 4 つのアクションがトリガーされ、イベントモニタリングは 2 回目の失敗後までディセーブルにされます。サーバへの ICMP エコー動作が失敗したことを示すメッセージが syslog に送信され、SNMP トラップが生成され、EEM はアプリケーション固有のイベントをパブリッシュし、IPSLA1F というカウンタが値 1 で増分されます。

```
Router(config)# event manager applet IPSLAping1
Router(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.42.1.2.9.1.6.4 get-type exact
entry-op eq entry-val 1 exit-op eq exit-val 2 poll-interval 5
Router(config-applet)# action 1.0 syslog priority critical msg "Server IP echo failed:
OID=${_snmp_oid_val}"
Router(config-applet)# action 1.1 snmp-trap strdata "EEM detected server reachability
failure to 10.1.88.9"
Router(config-applet)# action 1.2 publish-event sub-system 88000101 type 1 arg1 10.1.88.9
arg2 IPSLAEcho arg3 fail
Router(config-applet)# action 1.3 counter name _IPSLA1F value 1 op inc
```

次に、名前 one、クラス A でアプレットを登録し、タイマーイベントディテクタが 10 秒ごとにイベントをトリガーするアプレット コンフィギュレーションモードを開始する例を示します。イベントがトリガーされると、**action syslog** コマンドにより、syslog にメッセージ「hello world」が書き込まれます。

```
Router(config)# event manager applet one class A
Router(config-applet)# event timer watchdog time 10
Router(config-applet)# action syslog syslog msg "hello world"
Router(config-applet)# exit
```

次に、名前 one、クラス A でアプレットを登録するときに、AAA 許可をバイパスする例を示します。

```
Router(config)# event manager applet one class A authorization bypass
Router(config-applet)#
```

## 関連コマンド

コマンド	説明
<b>show event manager policy registered</b>	登録されている EEM ポリシーを表示します。

## event manager detector routing

ルーティング イベント ディテクタがモニタリング イベントを開始する際の遅延時間を設定するには、グローバル コンフィギュレーション モードで **event manager detector routing** コマンドを使用します。遅延時間をディセーブルにするには、このコマンドの **no** 形式を使用します。

**event manager detector routing bootup-delay** *delay-time*

**no event manager detector routing**

### 構文の説明

<b>bootup-delay</b>	起動後にモニタリングをオンにする時間遅延を指定します。
<i>delay-time</i>	秒数およびオプションのミリ秒数を ssssssssss [.mmm] 形式で表した数字。秒数の範囲は 0 ～ 4294967295 です。ミリ秒の範囲は 0 ～ 999 です。ミリ秒だけを使用する場合は、0.mmm の形式でミリ秒を指定します。

### コマンド デフォルト

ルーティング イベント ディテクタ コマンドは設定されません。

### コマンド モード

グローバル コンフィギュレーション (config)

### コマンド履歴

リリース	変更内容
12.4(22)T	このコマンドが導入されました。
12.2(33)SRE	このコマンドが、Cisco IOS Release 12.2(33)SRE に統合されました。

### 使用上のガイドライン

起動後に遅延時間が経過してからルーティング アップデートをオンにするように設定するには、**event manager detector routing** コマンドを使用します。設定された場合、ルーティング イベント ディテクタは、起動遅延時間後にイベントのモニタリングを開始します。起動遅延時間に到達すると、ルーティング アップデートがオンになり、ポリシーはトリガーを開始します。

## 例

次に、ルーティング アップデートをオンにための遅延時間を設定する例を示します。

```
Router(config)# event manager detector routing bootup-delay 800
```

## 関連コマンド

コマンド	説明
<b>event manager detector rpc</b>	RPC イベント ディテクタ コマンドを使用して EEM アプレットを受け入れるようにルータを設定します。

## event manager directory user

ユーザライブラリファイルまたはユーザ定義の Embedded Event Manager (EEM) ポリシーを保存するディレクトリを指定するには、グローバルコンフィギュレーションコマンドで **event manager directory user** コマンドを使用します。ユーザライブラリファイルまたはユーザ定義の EEM ポリシーを保存するディレクトリの使用をディセーブルにするには、このコマンドの **no** 形式を使用します。

**event manager directory user** {*library path*| *policy path*}

**no event manager directory user** {*library path*| *policy path*}

### 構文の説明

<b>library</b>	ディレクトリを使用してユーザライブラリファイルを保存するように指定します。
<b>policy</b>	ディレクトリを使用してユーザ定義の EEM ポリシーを保存するように指定します。
<i>path</i>	フラッシュ デバイスのユーザ ディレクトリの絶対パス名。

### コマンド デフォルト

ユーザライブラリファイルまたはユーザ定義の EEM ポリシーを保存するディレクトリは指定されていません。

### コマンド モード

グローバル コンフィギュレーション

### コマンド履歴

リリース	変更内容
12.3(14)T	このコマンドが導入されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェアモジュラリティ イメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。

リリース	変更内容
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされます。このトレインの特定の12.2SXリリースにおけるサポートは、フィーチャセット、プラットフォーム、およびプラットフォームハードウェアによって異なります。

## 使用上のガイドライン

ユーザライブラリディレクトリは、EEMポリシーの作成に関連するユーザライブラリファイルを保存するために必要です。EEMポリシーを記述する予定がない場合は、ユーザライブラリディレクトリを作成する必要はありません。

Cisco IOS Release 12.3(14)T以降のリリースでは、ソフトウェアは Tool Command Language (Tcl) のスクリプト言語を使用して作成されたポリシーファイルをサポートします。Tclは、EEMをネットワークデバイスにインストールする際に、Cisco IOS ソフトウェアイメージで提供されます。.tcl 拡張子を持つファイルは、EEMポリシー、Tclライブラリファイル、または「tclindex」という名前の特別なTclライブラリインデックスファイルのいずれかです。tclindexファイルには、ユーザ関数名のリストと、ユーザ関数を含むライブラリファイルが含まれます。EEMは、Tclがtclindexファイルの処理を開始すると、ユーザライブラリディレクトリを検索します。

EEMに対して識別する前にユーザライブラリディレクトリを作成するには、特権EXECモードで **mkdir** コマンドを使用します。ユーザライブラリディレクトリを作成したら、**copy** コマンドを使用して.tclライブラリファイルをユーザライブラリディレクトリにコピーできます。

ユーザポリシーディレクトリは、ユーザ定義のポリシーファイルを保存するために必要です。EEMポリシーを記述する予定がない場合は、ユーザポリシーディレクトリを作成する必要はありません。EEMは、**event manager policy policy-filename type user** コマンドが入力されたときにユーザポリシーディレクトリを検索します。

EEMに対して識別する前にユーザポリシーディレクトリを作成するには、特権EXECモードで **mkdir** コマンドを使用します。ユーザポリシーディレクトリを作成したら、**copy** コマンドを使用して、ポリシーファイルをユーザポリシーディレクトリにコピーできます。

## 例

次に、ユーザライブラリファイルを保存するディレクトリとして disk0:/user\_library を指定する例を示します。

```
Router(config)# event manager directory user library disk0:/user_library
```

## 関連コマンド

コマンド	説明
<b>copy</b>	コピー元からコピー先に任意のファイルをコピーします。
<b>event manager policy</b>	EEMポリシーをEEMに登録します。

コマンド	説明
<b>mkdir</b>	クラス C フラッシュ ファイル システムに新しいディレクトリを作成します。

## event manager environment

Embedded Event Manager (EEM) 環境変数を設定するには、グローバル コンフィギュレーション モードで **event manager environment** コマンドを使用します。EEM 環境変数をディセーブルにするには、このコマンドの **no** 形式を使用します。

**event manager environment** *variable-name string*

**no event manager environment** *variable-name*

### 構文の説明

<i>variable-name</i>	EEM 環境変数に割り当てる名前。
<i>string</i>	環境変数 <i>variable-name</i> に配置する一連の文字列 (埋め込みスペースを含む)。

### コマンド デフォルト

EEM 環境変数は設定されていません。

### コマンド モード

グローバル コンフィギュレーション

### コマンド履歴

リリース	変更内容
12.2(25)S	このコマンドが導入されました。
12.3(14)T	このコマンドが、Cisco IOS Release 12.3(14)T に統合されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェア モジュラリティ イメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされません。このトレインの特定の 12.2SX リリースにおけるサポートは、フィーチャセット、プラットフォーム、およびプラットフォームハードウェアによって異なります。

**使用上のガイドライン** 通例として、シスコで定義されているすべての環境変数の名前は、他の変数と区別するためにアンダースコア文字で始まります（\_show\_cmd など）。

*string* 引数で埋め込みスペースをサポートするために、このコマンドでは *variable-name* 引数の後ろから行の最後までを、*string* 引数の一部と解釈します。

設定後、EEM 環境変数の名前と値を表示するには、**show event manager environment** コマンドを使用します。

**例** 次に、**event manager environment** コマンドが EEM 環境変数のセットを定義する例を示します。

```
Router(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-7
Router(config)# event manager environment _show_cmd show version
```

#### 関連コマンド

コマンド	説明
<b>show event manager environment</b>	すべての EEM 環境変数の名前および値を表示します。



## event manager history size

Embedded Event Manager (EEM) 履歴テーブルのサイズを変更するには、グローバルコンフィギュレーションモードで **event manager history size** コマンドを使用します。デフォルトの履歴テーブルのサイズに戻すには、このコマンドの **no** 形式を使用します。

**event manager history size {events| traps} [ size ]**

**no event manager history size {events| traps}**

### 構文の説明

<b>events</b>	EEM イベント履歴テーブルのサイズを変更します。
<b>traps</b>	EEM 簡易ネットワーク管理プロトコル (SNMP) トラップ履歴テーブルのサイズを変更します。
<i>size</i>	(任意) 履歴テーブルのエントリ数を指定する 1 ~ 50 の範囲の整数。デフォルトは 50 です。

### コマンド デフォルト

履歴テーブルのサイズは 50 エントリです。

### コマンド モード

グローバル コンフィギュレーション

### コマンド履歴

リリース	変更内容
12.2(25)S	このコマンドが導入されました。
12.3(14)T	このコマンドが、Cisco IOS Release 12.3(14)T に統合されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェア モジュラリティ イメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。

リリース	変更内容
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされます。このトレインの特定の 12.2SX リリースにおけるサポートは、フィーチャセット、プラットフォーム、およびプラットフォームハードウェアによって異なります。

## 例

次に、**event manager history size** コマンドが SNMP トラップ履歴テーブルのサイズを 30 エントリに変更する例を示します。

```
Router(config)# event manager history size traps 30
```

## 関連コマンド

コマンド	説明
<b>show event manager history events</b>	トリガーされた EEM イベントを表示します。
<b>show event manager history traps</b>	送信された EEM SNMP トラップを表示します。

## event manager run

登録済みの Embedded Event Manager (EEM) ポリシーを手動で実行するには、特権 EXEC モードで **event manager run** コマンドを使用します。

```
event manager run policy-filename [[ parameter1 ] [ parameter2 ]... [ parameter15 ]]
```

### 構文の説明

<i>policy-filename</i>	ポリシー ファイルの名前。
<i>parameter</i>	(任意) スクリプトに渡されるパラメータ。最大 15 個のパラメータを指定できます。パラメータは、英数字のストリングである必要があります。引用符、埋め込みスペースや特殊文字は使用できません。

### コマンド デフォルト

登録済みの EEM ポリシーは実行されません。

### コマンド モード

特権 EXEC (#)

### コマンド履歴

リリース	変更内容
12.3(14)T	このコマンドが導入されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェア モジュールリティ イメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされました。このトレインの特定の 12.2SX リリースにおけるサポートは、フィーチャセット、プラットフォーム、およびプラットフォーム ハードウェアによって異なります。
12.4(20)T	parameter 引数が追加されました。最大 15 のパラメータ値を指定でき、引数はレジストリ コールで指定できます。

**使用上のガイドライン** また、このコマンドを使用することでパラメータをイベントポリシーで使用し、レジストリコールで引数を指定できます。

EEMは、通常、ポリシーそのものに含まれるイベント仕様に基づいてポリシーをスケジューリングし、実行します。 **event manager run** コマンドを使用することで、ポリシーを手動で実行できます。手動でポリシーを実行するには、最初に **event none** コマンドを設定する必要があります。None イベントディテクタは、None イベントをパブリッシュする場合は引数を含みます。このコマンドには、**no** 形式はありません。

**例** 次に、policy-manual.tcl という名前の EEM ポリシーを手動で実行する例を示します。

```
Router# event manager run policy-manual.tcl
```

各パラメータは組み込みの総数 (`$_none_argc`) と、続けて組み込みのリスト (`$_none_arg1`、`$_none_arg2`、および `$_none_arg3`) で構成されます。次に、アプレットおよび Tool Tcl スクリプトの例を示します。

**例**

```
event manager applet none_parameter_test
 event none
 action 1 syslog msg "Number of Arguments is $_none_argc"
 action 2 syslog msg "Argument 1 is $_none_arg1"
 action 3 syslog msg "Argument 2 is $_none_arg2"
 action 4 syslog msg "Argument 3 is $_none_arg3"
end
Router# event manager run none_parameter_test 11 22 33
01:26:03: %HA_EM-6-LOG: none_parameter_test: Number of Arguments is 3
01:26:03: %HA_EM-6-LOG: none_parameter_test: Argument 1 is 11
01:26:03: %HA_EM-6-LOG: none_parameter_test: Argument 2 is 22
01:26:03: %HA_EM-6-LOG: none_parameter_test: Argument 3 is 33
```

ポリシーの場合、**event\_reqinfo** はオプションパラメータを文字列で返し、その文字列はその後ポリシーによって処理されます。

**例**

```
none_paramter_test.tcl
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
# query the event info
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
    set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
action_syslog priority info msg "Number of Arguments is $arr_einfo(argc)"
if {$_cerrno != 0} {
    set result [format \
        "component=%s; subsys err=%s; posix err=%s;\n%s" \
        $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
    error $result
}
```

```

action_syslog priority info msg "Argument 1 is $arr_einfo(arg1)"
if {$_cerrno != 0} {
  set result [format \
    "component=%s; subsys err=%s; posix err=%s;\n%s" \
    $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
  error $result
}
action_syslog priority info msg "Argument 2 is $arr_einfo(arg2)"
if {$_cerrno != 0} {
  set result [format \
    "component=%s; subsys err=%s; posix err=%s;\n%s" \
    $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
  error $result
}
action_syslog priority info msg "Argument 3 is $arr_einfo(arg3)"
if {$_cerrno != 0} {
  set result [format \
    "component=%s; subsys err=%s; posix err=%s;\n%s" \
    $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
  error $result
}
}
jubjub#event manager run none_parameter_test.tcl 1 2 3
01:26:03: %HA_EM-6-LOG: tmpsys:/eem_policy/none_parameter_test.tcl: Number of Arguments is
3
01:26:03: %HA_EM-6-LOG: tmpsys:/eem_policy/none_parameter_test.tcl: Argument 1 is 1
01:26:03: %HA_EM-6-LOG: tmpsys:/eem_policy/none_parameter_test.tcl: Argument 2 is 2
01:26:03: %HA_EM-6-LOG: tmpsys:/eem_policy/none_parameter_test.tcl: Argument 3 is 3

```

---

**関連コマンド**

コマンド	説明
<b>event manager applet</b>	EEM に EEM 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。
<b>event manager policy</b>	EEM ポリシーを EEM に登録します。
<b>event none</b>	EEM に登録して手動で起動される EEM ポリ シーを指定します。
<b>show event manager policy registered</b>	登録済みの EEM ポリシーを表示します。

## event manager scheduler clear

実行中または実行を保留中の Embedded Event Manager (EEM) ポリシーをクリアするには、特権 EXEC モードで **event manager scheduler clear** コマンドを使用します。

```
event manager scheduler clear {all| policy job-id| queue-type {applet| call-home| axp| script} [class
class-options]} [processor {rp_primary| rp_standby}]
```

### 構文の説明

<b>all</b>	現在実行中または実行保留中のキューにあるすべてのポリシーをクリアします。
<b>policy</b>	ジョブ ID で指定された EEM ポリシーをクリアします。
<i>job-id</i>	キューにある各ポリシーを識別する 1 ~ 4294967295 の範囲の番号。
<b>queue-type</b>	EEM ポリシーのキュータイプをクリアします。
<b>applet</b>	EEM キュータイプのアプレットを指定します。
<b>call-home</b>	EEM キュータイプ の Call Home ポリシーを指定します。
<b>axp</b>	EEM キュータイプ のアプリケーション拡張プラットフォームを指定します。
<b>script</b>	EEM 実行スレッドで Tcl スクリプトを実行するように指定します。
<b>class</b>	指定したクラスの EEM ポリシーをクリアします。

<i>class-options</i>	<p>EEM ポリシー クラスを指定します。次のうちの 1 つまたはすべてを指定できます。</p> <ul style="list-style-type: none"> <li>• <b>class-letter</b> : 各ポリシー クラスを識別する A ~ Z の文字。 <i>class-letter</i> の複数のインスタンスを指定できます。</li> <li>• <b>default</b> : デフォルト クラス。 クラスなしで登録された EEM ポリシーは、デフォルト クラスに割り当てられます。</li> <li>• <b>range class-range</b> : EEM ポリシー クラスを範囲で指定します。 A ~ Z. の範囲を指定できます。 <b>range class-range</b> の複数のインスタンスを指定できます。</li> </ul>
<b>processor</b>	(任意) プロセッサがコマンドを実行するように指定します。
<b>rp_primary</b>	(任意) デフォルトの RP を示します。 ポリシーは、イベント相関によりポリシーがスケジュールされる場合は、プライマリ RP で実行されます。
<b>rp_standby</b>	(任意) スタンバイ RP を示します。 ポリシーは、イベント相関によりポリシーがスケジュールされる場合は、スタンバイ RP で実行されます。

## コマンド モード

特権 EXEC (#)

## コマンド履歴

リリース	変更内容
12.4(20)T	このコマンドが導入されました。
12.4(22)T	<b>queue-type</b> および <b>processor</b> キーワードと、 <i>class-letter</i> および <i>class-range</i> 引数が追加されました。
12.2(33)SRE	このコマンドが、 Cisco IOS Release 12.2(33)SRE に統合されました。

**使用上のガイドライン** **show event manager policy pending** コマンドを使用して、サーバ実行キューで保留されているポリシーを表示します。

**show event manager policy active** コマンドを使用して、実行中のポリシーを表示します。

**event manager scheduler clear** コマンドを使用して、サーバのポリシーまたはポリシー キューをクリアします。

**class** キーワードの場合、**class-letter**、**default**、または **range class-range** のいずれかのオプションを指定する必要があります。同じ CLI 文でこれらのオプションをすべて指定できます。

**例** 次に、実行保留中の EEM ポリシーをクリアする例を示します。 **show** コマンドは、ポリシーをクリアする前後の出力例を表示します。

```
Router# show event manager policy pending
no. job id status time of event          event type      name
1   1      pend  Thu Sep 7 02:54:04 2006  syslog         applet: one
2   2      pend  Thu Sep 7 02:54:04 2006  syslog         applet: two
3   3      pend  Thu Sep 7 02:54:04 2006  syslog         applet: three
```

```
Router# event manager scheduler clear policy 2
Router# show event manager policy pending

no. job id status time of event          event type      name
1   1      pend  Thu Sep 7 02:54:04 2006  syslog         applet: one
3   3      pend  Thu Sep 7 02:54:04 2006  syslog         applet: three
```

#### 関連コマンド

コマンド	説明
<b>event manager policy</b>	EEM ポリシーを EEM に登録します。
<b>show event manager policy pending</b>	実行保留中の EEM ポリシーを表示します。



## event manager scheduler hold

EEM スケジューラのスケジュールされた Embedded Event Manager (EEM) ポリシー イベントまたは イベント キューをホールドするには、特権 EXEC モードで **event manager scheduler hold** コマンドを使用します。ポリシー イベントまたは イベント キューを再開するには、**event manager scheduler release** コマンドを使用します。

```
event manager scheduler hold {all| policy job-id| queue-type {applet| call-home| axp| script} [class class-options]} [processor {rp_primary| rp_standby}]
```

### 構文の説明

<b>all</b>	EEM スケジューラのすべての EEM ポリシー イベントまたは イベント キューをホールドします。
<b>policy</b>	ジョブ ID で指定された EEM スケジューラの EEM ポリシー イベントまたは イベント キューをホールドします。
<i>job-id</i>	キューにある各ポリシーを識別する 1 ～ 4294967295 の範囲の番号。
<b>queue-type</b>	EEM キュータイプに基づいて EEM ポリシー イベントまたは イベント キューをホールドします。
<b>applet</b>	EEM キュータイプのアプレットを指定します。
<b>call-home</b>	EEM キュータイプの Call Home ポリシーを指定します。
<b>axp</b>	EEM キュータイプのアプリケーション拡張プラットフォームを指定します。
<b>script</b>	EEM 実行スレッドで Tcl スクリプトを実行するように指定します。
<b>class</b>	EEM ポリシーのクラスを指定します。

<i>class-options</i>	EEM ポリシー クラスを指定します。次のうちの 1 つまたはすべてを指定できます。 <ul style="list-style-type: none"> <li>• <b>class-letter</b> : 各ポリシー クラスを識別する A ~ Z の文字。 <i>class-letter</i> の複数のインスタンスを指定できます。</li> <li>• <b>default</b> : デフォルトクラス。クラスなしで登録された EEM ポリシーは、デフォルトクラスに割り当てられます。</li> <li>• <b>range class-range</b> : EEM ポリシー クラスを範囲で指定します。A ~ Z の範囲を指定できます。 <b>range class-range</b> の複数のインスタンスを指定できます。</li> </ul>
<b>processor</b>	(任意) プロセッサがコマンドを実行するように指定します。
<b>rp_primary</b>	(任意) デフォルトの RP を示します。ポリシーは、イベント相関によりポリシーがスケジュールされる場合は、プライマリ RP で実行されます。
<b>rp_standby</b>	(任意) スタンバイ RP を示します。ポリシーは、イベント相関によりポリシーがスケジュールされる場合は、スタンバイ RP で実行されます。

## コマンド モード

特権 EXEC (#)

## コマンド履歴

リリース	変更内容
12.4(22)T	このコマンドが導入されました。
12.2(33)SRE	このコマンドが、Cisco IOS Release 12.2(33)SRE に統合されました。

## 使用上のガイドライン

**show event manager policy pending** コマンドを使用して、サーバ実行キューで保留されているポリシーを表示します。

**event manager scheduler hold** コマンドを使用して、サーバのポリシーまたはポリシー キューをホールドします。

**class** キーワードの場合、*class-letter*、**default**、または **range class-range** のいずれかのオプションを指定する必要があります。同じ CLI 文でこれらのオプションをすべて指定できます。

## 例

次に、EEM スケジューラのスケジュールされたポリシーイベントをホールドする例を示します。**show** コマンドは、ポリシー イベントをホールドする前後の出力例を表示します。

```
Router# show event manager policy pending
no. job id status time of event      event type      name
1  1      pend  Thu Sep 7  02:54:04 2006  syslog         applet: one
2  2      pend  Thu Sep 7  02:54:04 2006  syslog         applet: two
3  3      pend  Thu Sep 7  02:54:04 2006  syslog         applet: three
Router# event manager scheduler hold policy 2
Router# show event manager policy pending

no. job id status time of event      event type      name
1  1      pend  Thu Sep 7  02:54:04 2006  syslog         applet: one
2  2      held  Thu Sep 7  02:54:04 2006  syslog         applet: two
3  3      pend  Thu Sep 7  02:54:04 2006  syslog         applet: three
```

## 関連コマンド

コマンド	説明
<b>event manager policy</b>	EEM ポリシーを EEM に登録します。
<b>event manager scheduler release</b>	ポリシー イベントまたはイベント キューを再開します。
<b>show event manager policy pending</b>	実行保留中の EEM ポリシーを表示します。

## event manager scheduler modify

Embedded Event Manager (EEM) ポリシーのスケジューリング パラメータを変更するには、特権 EXEC モードで **event manager scheduler modify** コマンドを使用します。

```
event manager scheduler modify {all| policy job-id queue-type {applet| call-home| axp| script}} {class
class-options [queue-priority {high| last| low| normal}]} [queue-priority {high| last| low| normal}] [class
class-options]} [processor {rp_primary| rp_standby}]
```

### 構文の説明

<b>all</b>	現在実行中または実行保留中のキューにあるすべてのポリシーを変更します。
<b>policy</b>	ジョブ ID で指定された EEM ポリシーを変更します。
<i>job-id</i>	キューにある各ポリシーを識別する 1 ～ 4294967295 の範囲の番号。
<b>queue-type</b>	EEM ポリシーのキュー タイプを変更します。
<b>applet</b>	EEM キュータイプのアプレットを指定します。
<b>call-home</b>	EEM キュータイプ の Call Home ポリシーを指定します。
<b>axp</b>	EEM キュータイプ のアプリケーション拡張プラットフォームを指定します。
<b>script</b>	EEM 実行スレッドで Tcl スクリプトを実行するように指定します。
<b>class</b>	EEM ポリシーのクラスを変更します。
<i>class-options</i>	EEM ポリシー クラスを指定します。次のうちの 1 つまたはすべてを指定できます。 <ul style="list-style-type: none"> <li>• <b>class-letter</b> : 各ポリシー クラスを識別する A ～ Z の文字。 <i>class-letter</i> の複数のインスタンスを指定できます。</li> <li>• <b>default</b> : デフォルトクラス。 クラスなしで登録された EEM ポリシーは、デフォルトクラスに割り当てられます。</li> </ul>

<b>queue-priority</b>	(任意) EEM ポリシーのキューイング順序のプライオリティを変更します。
<b>high</b>	(任意) キューのプライオリティを高に指定します。
<b>last</b>	(任意) キューのプライオリティを最後に指定します。
<b>low</b>	(任意) キューのプライオリティを低に指定します。
<b>normal</b>	(任意) キューのプライオリティを通常に指定します。
<b>processor</b>	(任意) プロセッサがコマンドを実行するように指定します。
<b>rp_primary</b>	(任意) デフォルトの RP を示します。ポリシーは、イベント関連によりポリシーがスケジュールされる場合は、プライマリ RP で実行されます。
<b>rp_standby</b>	(任意) スタンバイ RP を示します。ポリシーは、イベント関連によりポリシーがスケジュールされる場合は、スタンバイ RP で実行されます。

## コマンドモード

特権 EXEC (#)

## コマンド履歴

リリース	変更内容
12.4(22)T	このコマンドが導入されました。
12.2(33)SRE	このコマンドが、Cisco IOS Release 12.2(33)SRE に統合されました。

## 使用上のガイドライン

**show event manager policy pending** コマンドを使用して、サーバ実行キューで保留されているポリシーを表示します。

**event manager scheduler modify** コマンドを使用して、ポリシーのスケジューリングパラメータを変更します。

**class** キーワードの場合、*class-letter* または **default** のいずれかのオプションを指定する必要があります。同じ CLI 文で両方のオプションを指定できます。

## 例

次に、EEM ポリシーのスケジューリングパラメータを変更する例を示します。**show** コマンドは、スケジューリングパラメータを変更する前後の出力例を表示します。

```
Router# show event manager policy pending
no. class status time of event event type name
1 default pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 default pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 B pend Thu Sep 7 02:54:04 2006 syslog applet: three

Router# event manager scheduler modify all class A
Router# show event manager policy pending
no. class status time of event event type name
1 A pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 A pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 A pend Thu Sep 7 02:54:04 2006 syslog applet: three
```

## 関連コマンド

コマンド	説明
<b>event manager policy</b>	EEM ポリシーを EEM に登録します。
<b>show event manager policy pending</b>	実行保留中の EEM ポリシーを表示します。

## event manager scheduler release

Embedded Event Manager (EEM) ポリシーの実行を再開するには、特権 EXEC モードで **event manager scheduler release** コマンドを使用します。

```
event manager scheduler release {all| policy policy-id| queue-type {applet| call-home| axp| script} [class class-options]} [processor {rp_primary| rp_standby}]
```

### 構文の説明

<b>all</b>	すべての EEM ポリシーの実行を再開します。
<b>policy</b>	ポリシー ID で指定された EEM ポリシーを再開します。
<i>policy-id</i>	キューにある各ポリシーを識別する 1 ～ 4294967295 の範囲の番号。
<b>queue-type</b>	EEM キュー タイプに基づいてポリシーの実行を再開します。
<b>applet</b>	EEM アプレット ポリシーを指定します。
<b>call-home</b>	Call Home ポリシーを指定します。
<b>axp</b>	Application Extension Platform (AXP) ポリシーを指定します。
<b>script</b>	EEM スクリプト ポリシーを指定します。
<b>class</b>	EEM ポリシー クラスを指定します。
<i>class-options</i>	<p>EEM ポリシー クラス。 次のうちの 1 つまたはすべてを指定できます。</p> <ul style="list-style-type: none"> <li>• <b>class-letter</b> : 各ポリシー クラスを識別する A ～ Z の文字。 <i>class-letter</i> の複数のインスタンスを指定できます。</li> <li>• <b>default</b> : デフォルト クラスに登録されたポリシーを指定します。</li> <li>• <b>range class-letter-range</b> : EEM ポリシー クラスを範囲で指定します。 <b>range class-letter-range</b> の複数のインスタンスを指定できます。 <i>class-letter-range</i> で使用する文字は大文字にする必要があります。</li> </ul>

<b>processor</b>	プロセッサがコマンドを実行するように指定します。デフォルト値はプライマリ RP です。
<b>rp_primary</b>	プライマリ RP を示します。
<b>rp_standby</b>	スタンバイ RP を示します。

コマンド デフォルト      ディセーブル

コマンド モード      特権 EXEC (#)

コマンド履歴	リリース	変更内容
	12.4(22)T	このコマンドが導入されました。

**使用上のガイドライン** **event manager scheduler hold** コマンドを使用してホールドした EEM ポリシーを解放するには、**event manager scheduler release** コマンドを使用します。

*class-letter*、**default**、および **range class-letter-range** のいずれかのオプションを指定する必要があります。同じ CLI 文でこれらのオプションをすべて指定できます。

**例**      次に、すべての EEM ポリシーの実行を再開する例を示します。

```
Router# event manager scheduler release all
```

次に、クラス A ~ E のポリシーの実行を再開する例を示します。

```
Router# event manager scheduler release queue-type script class range A-E
```

関連コマンド	コマンド	説明
	<b>event manager scheduler hold</b>	EEM ポリシーのスケジューリング実行をホールドします。



## event manager scheduler suspend

Embedded Event Manager (EEM) ポリシーのスケジューリング実行をただちに停止するには、グローバル コンフィギュレーション モードで **event manager scheduler suspend** コマンドを使用します。EEM ポリシーのスケジューリングを再開するには、このコマンドの **no** 形式を使用します。

**event manager scheduler suspend**

**no event manager scheduler suspend**

### 構文の説明

このコマンドには引数またはキーワードはありません。

### コマンド デフォルト

ポリシー スケジューリングはアクティブです。

### コマンド モード

グローバル コンフィギュレーション

### コマンド履歴

リリース	変更内容
12.2(25)S	このコマンドが導入されました。
12.3(14)T	このコマンドが、Cisco IOS Release 12.3(14)T に統合されました。
12.2(28)SB	このコマンドが、Cisco IOS Release 12.2(28)SB に統合されました。
12.2(18)SXF4	このコマンドが Cisco IOS Release 12.2(18)SXF4 に統合され、ソフトウェア モジュール イメージだけをサポートするようになりました。
12.2(33)SRA	このコマンドが、Cisco IOS Release 12.2(33)SRA に統合されました。
12.2(18)SXF5	このコマンドが、Cisco IOS Release 12.2(18)SXF5 に統合されました。
12.2SX	このコマンドは、Cisco IOS Release 12.2SX トレインでサポートされません。このトレインの特定の 12.2SX リリースにおけるサポートは、フィーチャセット、プラットフォーム、およびプラットフォームハードウェアによって異なります。

### 使用上のガイドライン

**event manager scheduler suspend** コマンドを使用して、すべてのポリシーのスケジューリング要求を停止します。このコマンドの **no** 形式を入力するまで、スケジューリングは行われません。こ

のコマンドの **no** 形式を使用すると、ポリシーのスケジューリングが再開され、保留中のポリシーが実行されます。

ポリシーを1つずつ登録解除せずに、ポリシーの実行をただちに停止する必要がある状況として、次のものが挙げられます。

- セキュリティ上の理由：システムのセキュリティが侵害されていると思われる場合。
- パフォーマンス上の理由：他の機能に多くのCPUサイクルを割り当てるため、ポリシーの実行を一時的に停止する場合。

## 例

次の **event manager scheduler suspend** コマンドの例では、ポリシーのスケジュールがディセーブルになります。

```
Router(config)# event manager scheduler suspend
May 19 14:31:22.439: fm_server[12330]: %HA_EM-6-FMS_POLICY_EXEC: fh_io_msg: Policy execution
has been suspended
```

次の **event manager scheduler suspend** コマンドの例では、ポリシーのスケジュールがイネーブルになります。

```
Router(config)# no event manager scheduler suspend
May 19 14:31:40.449: fm_server[12330]: %HA_EM-6-FMS_POLICY_EXEC: fh_io_msg: Policy execution
has been resumed
```

## 関連コマンド

コマンド	説明
<b>event manager policy</b>	EEM ポリシーを EEM に登録します。