



QoS ポリシー アカウンティング

QoS ポリシー アカウンティング機能は、システムのトラフィックを正確に考慮するうえで役立ちます。また、加入者に Quality of Service (QoS) 設定を割り当てる際の柔軟性も向上します。さらに、QoS アカウンティング ハイ アベイラビリティ機能により、予定および予定外のルート プロセッサ (RP) スイッチオーバーが発生しても、QoS アカウンティング統計情報が確実に存続するようになり、RADIUS アカウンティング課金サーバはアカウンティングカウンタの報告を確実に続行します。このモジュールでは、QoS ポリシー アカウンティングを設定する方法、加入者テンプレートを使用する方法、および加入者アカウンティング精度を有効にする方法について説明します。

- [機能情報の確認, 1 ページ](#)
- [QoS ポリシー アカウンティングの前提条件, 2 ページ](#)
- [QoS ポリシー アカウンティングに関する制約事項, 2 ページ](#)
- [QoS ポリシー アカウンティングについて, 5 ページ](#)
- [QoS ポリシー アカウンティングの使用方法, 25 ページ](#)
- [QoS ポリシー アカウンティングの設定例, 29 ページ](#)
- [その他の関連資料, 30 ページ](#)
- [QoS ポリシー アカウンティングの機能情報, 31 ページ](#)

機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェアリリースの [Bug Search Tool](#) およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよびシスコソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

QoS ポリシー アカウンティングの前提条件

- PPP over Ethernet (PPPoE) または PPP over Ethernet over ATM (PPPoEoA) セッションがイネーブルであること。
- RADIUS サーバが設定されていること。
- 認証、認可、およびアカウンティング (AAA) がイネーブルであること。
- RADIUS サーバ上に加入者のユーザプロファイルが作成されていること。
- ポリシー マップが設定されていること。
- サービス テンプレートが設定されていること。
- トラフィック クラスが作成されていること。
- ステートフル スイッチオーバー (SSO) および In-Service Software Upgrades (ISSU) の前提条件が満たされていること。詳細については、『Cisco IOS High Availability Configuration Guide』を参照してください。

QoS ポリシー アカウンティングに関する制約事項

- システム フェールオーバーでは、次の処理が行われます。
 - ポリシー マップで静的に設定された QoS アカウンティングの場合、QoS アカウンティング統計情報がゼロにリセットされます。
 - サービス テンプレートを使って動的に設定された QoS アカウンティングの場合、新しいアクティブ ルート プロセッサ (RP) ではセッションが存在しなくなります。



(注) Cisco IOS XE Release 3.5S 以降のリリースでは、サービス テンプレートを介して有効になるアカウンティング サービス用のハイアベイラビリティ (HA) サポートを使用できます。そのため、システム フェールオーバーの際に QoS アカウンティング統計情報とサービス セッションが維持され、新しくアクティブになった RP で使用可能になります。

- QoS ポリシー アカウンティング サービスではマルチキャストがサポートされていません。
- サービス テンプレートでは、次の QoS アクションがサポートされていません。
 - account

- fair-queue
 - netflow-sampler
 - random-detect
- サービス テンプレートでは、次の QoS フィルタがサポートされていません。
- atm
 - class-map
 - cos
 - destination-address
 - discard-class
 - fr-de
 - fr-dlci
 - input-interface
 - mpls
 - not
 - packet
 - source-address
 - vlan
- サービス テンプレート定義の行は、Cisco IOS CLI で許容される最大設定行の長さを超えてはなりません。この範囲内に抑えるために、シェル変数名を短くしなければならない場合があります。
- セッションでアクティブになっているテンプレートサービスを変更することはできません。代わりに、それを非アクティブ化して、別のテンプレートサービスをアクティブにすることができます。
- テンプレート サービスがアクティブである場合、セッションでアクティブな QoS ポリシーを変更するために従来の複合パラメータ化ストリングを使用することはできません。
- IP アドレスのパラメータ化は、IPv4 および名前付き ACL（リマークなし）でのみサポートされます。パラメータ化サービス アクティベーションで指定される IP アドレスは常に、「permit ip network mask any」および「permit ip any network mask」という固定パターンで、複製された ACL に追加されます。
- サービステンプレートは PPP セッションでのみサポートされ、サブインターフェイスでサービス テンプレートをアクティブにしてはなりません。
- セッションでアクティブにできる Turbo Button サービスは常に 1 つだけです。Turbo Button サービスとは、親ポリシーの class-default で（子ポリシーを変更する）「service-policy xxxx」以外の QoS アクションを変更するあらゆるサービスのことです。

- シェル変数、QoS クラスマップ、およびアクセスコントロールリスト (ACL) の名前には、次の文字を使用してはなりません。
 - !
 - \$
 - #
 - -
 - 、
 - >
 - <
- グループ アカウンティングの場合 (サービス テンプレートで \$`_acctgrp` を使用する場合) にのみ、アカウンティング レコードにサービス名がエコー出力されます。
- セッションでアクティブな IN/OUT QoS ポリシー名は、以前にアクティブであった QoS ポリシー (または、最後のマルチサービス認可変更 (CoA) あるいは `Access-Accept` で指定された静的 QoS ポリシー) を連結することで作成されます。
- 同じサービス テンプレートからインスタンス化された2つのテンプレートサービスを、セッションで同時に有効にすることはできません。ただし、相互関係のない複数のサービス テンプレートからインスタンス化された複数のテンプレートサービスを、セッションで同時にアクティブにすることは可能です。
- ローカルに終端される PPP および Layer 2 Tunneling Protocol (L2TP) アクセス コンセントレータ (LAC) 上の PPP 転送セッションにのみ、テンプレートサービスのサポートを使用できます。
- LAC 上の PPP 転送セッションに対し、`Access-Accept` を介してテンプレート サービスを適用するには、次の設定を使用します。
 - `vpdn authen-before-forward`
 - 認証プロファイルではなく、ユーザ認可プロファイル (PPP 認証後に受け取る `Access-Accept`) 内でのみ、テンプレート サービスを指定します。
- 親 `class-default` の下の子ポリシー (2 レベルのみ) と親ポリシー (Turbo Button サービス) でのみ、テンプレート サービスをアクティブにします。
- デフォルト QoS ポリシーの階層は2レベル (親+`class-default` の子) に制限されているため、`class-default` 以外のクラスの下に子ポリシーを設定することはできません。
- テンプレート サービスを子レベルでアクティブにするには、デフォルト親ポリシー `class-default` の下で子ポリシーを設定する必要があります。
- 構文エラー チェックによるロールバックのみがサポートされます。
- 1つの CoA メッセージに複数のサービス アクティブ化または非アクティブ化が含まれている場合、いずれかの操作 (アクティブ化または非アクティブ化) が失敗すると、CoA 処理開始前の状態にセッションを戻すために、CoA は以前のすべての操作をロールバックする (取り

消す) 必要があります。つまり、CoA のすべての操作が正常に完了しない限り、すべて失敗します。この場合、CoA 否定応答 (NACK) が RADIUS に送信されます。

- **Access-Accept** 処理中にロールバックを機能させるには、加入者サービス マルチ許可処理を設定する必要があります。 **Access-Accept** でいずれかのサービスの処理に失敗すると、**Access-Accept** に含まれるそれ以前のすべてのサービスがロールバックされます (取り消されます)。 **Access-Accept** サービス処理が失敗したとしても、セッションは確立されます。
- プラットフォームまたはデータプレーンでエラーが発生してもロールバックがトリガーされないため、不完全なサービスになる可能性があります。
- テンプレート サービスがセッションで使用またはアクティブ状態になっている場合は、サービス テンプレートを変更しないでください。使用中のテンプレート サービスを表示するには、**show subscriber policy ppm-shim-db** コマンドを使用します。

QoS ポリシー アカウンティングについて

RADIUS は AAA を管理するためのネットワークングプロトコルです。とくに、各 RADIUS アカウンティングメッセージには入力と出力のカウンタが含まれます。カウンタ間の誤差を解決するには、QoS ポリシー アカウンティング機能を利用できます。

グループ単位 QoS ポリシー アカウンティング機能

QoS ポリシーアカウンティング機能は、セッションごとに次の情報を収集して RADIUS サーバに報告します。

- Acct-Session-Id
- 入力および出力パケット数/バイト数/ギガワード数、パケット数、正常に送信されたパケットのバイト数
- Parent-Session-ID
- QoS ポリシーとクラスまたはグループ名 (QoS ポリシー アカウンティング機能がグループでイネーブルにされている場合)
- サービス名
- ユーザ名

QoS ポリシー アカウンティング機能をグループに対してイネーブルにし、グループ名を割り当てると、この機能は次の条件を満たすパケットを集約します。

- 同じグループ内のトラフィック クラス別に分類されたパケット
- 同じターゲットに適用される入力または出力 QoS ポリシーに含まれるパケット

個別のアカウンティングストリーム

トラフィック クラスをグループに割り当てる代わりに AAA 方式リストに割り当てると、トラフィック クラスごとに個別の QoS ポリシー アカウンティングストリームが作成されます。個別のアカウンティングストリームによって、複数のクラスに一致するトラフィックを相互に区別できます。固有のターゲット、方向、ポリシー名、およびクラス名のそれぞれに、固有の RADIUS Acct-Session-Id 値が割り当てられます。

サービス テンプレート

サービス テンプレートを使用すると、新しい QoS ポリシーを CLI で定義することなく、動的に QoS パラメータを変更できます。セッションの開始時や、セッション確立後の任意の時点で QoS ポリシーを変更できます。アクティブな QoS を動的に変更する前に、現在のサービスを非アクティブにする必要があります。

サービス テンプレートを理解するには、次の用語について学習してください。

- サービス テンプレート：
 - Cisco IOS シェル関数です。
 - IN QoS ポリシー マップ定義が含まれます。
 - OUT QoS ポリシー マップ定義が含まれます。
 - プログラムによって呼び出されます。
 - シェル変数のデフォルト値を指定します。
- テンプレート サービス：
 - 括弧を使用する QoS サービス名です。
 - 対応するシェル マップ テンプレート定義があります。
 - サービス テンプレートのシェル関数の実行中に動的に作成されます。
- IN 実効ポリシー マップ
- OUT 実効ポリシー マップ

QoS ポリシー アカウンティング機能は、サービス テンプレート シェル関数で使用される変数のデフォルト値を Cisco IOS シェルがオーバーライドする方法を規定します。シェル マップ内の QoS ポリシー定義には、QoS アクションパラメータ値の代わりにシェル変数が含まれる場合があります。

サービス テンプレートの使用

サービス テンプレートを作成するには、テキストエディタでサービス テンプレートを作成してから、そのテンプレートを CLI にコピーします。シェル マップブロックの内容はテキストとして扱われます。

サービス テンプレート ポリシー マップ (policy map \$_outgoing/\$_incoming) を定義する際に使用できる CLI 支援機能またはプロンプトはありません。たとえば、次の CLI 支援機能にはアクセスできません。

- パーサー自動補完
- コマンド オプション
- 範囲のヘルプ
- 構文チェック



(注) CLI ではエディタを使用できないため、設定を間違えた場合にはサービス テンプレート全体を削除して、最初から設定する必要があります。

サービス テンプレートの確認

テキストエディタでサービス テンプレートを作成する際には、構文チェック機能がありません。したがって、サービス テンプレートをアクティブにする前に、その構文を確認する必要があります。次のサンプル コードは、*voice service1* サービス テンプレートを検証する方法を示しています。独自のテンプレートを検証するには、*voice service1* を該当するサービス テンプレート名に置き換えてください。

```
(shell map voice-service1 police_rate=100000 prec_value=4 queue_size=1)
configure terminal
no policy-map test-svc_IN <----- Removes previous service template verifications.
no policy-map test-svc_OUT <----- Removes previous service template verifications.
no aaa-accounting group test_svc_GRP <----- Removes previous service template
verifications.
end
trigger voice-service1 _incoming=test-svc_IN _outgoing=test-svc_OUT _acctgrp=test-svc_GRP
show policy-map test-svc-IN <-----
Ensure that the output matches the expected service template template service with default
values.
show policy-map test-svc-OUT <-----
Ensure that the output matches the expected service template template service with default
values.
```

サービス テンプレートの削除

サービス テンプレートを削除するには、コマンドラインで以下を入力します。

```
no shell map voice-service1 police_rate=100000 prec_value=4 queue_size=1 in_h=class-default
out_h=class-default
```

ここで、voice-service1 はサービス テンプレートの名前です。

サンプル サービス テンプレート

サービス テンプレート

次に、サービス テンプレートの例を示します。

```
{
  configure terminal
  accounting group $_acctgrp list default
  policy-map $_outgoing
  class voip
    police $police_rate 60625 0 conform-action transmit exceed-action drop violate-action
  drop
  exit
    priority level 1
    queue-limit 8 packets
    set precedence $prec_value
    set cos 6
    aaa-accounting group $_acctgrp
  class voip-control
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

    queue-limit $queue_size packets
    set precedence 6
    aaa-accounting group $_acctgrp
  policy-map $_incoming
  class voip
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
  drop
    set precedence 5
    aaa-accounting group $_acctgrp
  class voip-control
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
  drop
    set precedence 7
    aaa-accounting group $_acctgrp
}
```

アクションパラメータ オーバーライド

アクションパラメータ オーバーライドはサービス テンプレートの一種であり、QoS ポリシーのクラスでポリシング、シェーピング、帯域幅などの QoS アクション用に入力されるパラメータ設定の代わりに、シェル変数を使用します。

テンプレートサービスを非アクティブにすると、システムは以前にアクティブであった QoS ポリシーを復元します。その QoS ポリシーの名前は異なっている場合がありますが、テンプレートサービスがアクティブ化される前にアクティブであった QoS ポリシーと構造的にも機能的にもまったく変わりません。

この例では、次のパラメータを使用してサービスを生成します。

```
Reserved variable initialization before executing the service template shell function:
$_incoming = voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN
$_outgoing = voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT
$_acctgrp = aaa-accounting group
voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP list default
```


セッションでアクティブな OUT QoS ポリシー :

```
policy-map output_parent
  class class-default
    shape average 10000000
    service-policy output_child
policy-map output_child
  class class-default
```

セッションでアクティブな IN QoS ポリシー :

```
policy-map input_parent
  class class-default
    police 10000000
    service-policy input_child
policy-map input_child
  class-default
```

ターゲットセッションで voice-service1(police_rate=200000,prec_value=5,queue_size=32) をアクティブ化した後、次の OUT ポリシーがアクティブになります。

```
policy-map
output_parent$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

  class class-default
    shape average 10000000
    service-policy
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
policy-map
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

  class voip
    police 200000 60625 0 conform-action transmit exceed-action drop violate-action
drop
  priority level 1
  queue-limit 8 packets
  set precedence 5
  set cos 6
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP

  class voip-control
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

    queue-limit 32 packets
    set precedence 6
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class class-default
```

ターゲットセッションで voice-service1(police_rate=200000,prec_value=5,queue_size=32) をアクティブ化した後、次の IN ポリシーがアクティブになります。

```
policy-map
input_parent$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default

  class class-default
    police cir 10000000 bc 312500 conform-action transmit exceed-action drop
    service-policy
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
policy-map
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
  class voip
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
drop
  set precedence 5
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP

  class voip-control
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
drop
  set precedence 7
```

```

aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class-default

```

アクションパラメータ化のデフォルトパラメータ

アクションパラメータ化デフォルトパラメータはサービステンプレートの一種であり、QoS ポリシーのクラスでポリシング、シェーピング、帯域幅などの QoS アクション用に入力されるパラメータ設定の代わりに、シェル変数を使用します。

テンプレートサービスを非アクティブにすると、システムは以前にアクティブであった QoS ポリシーを復元します。その QoS ポリシーの名前は異なっていますが、テンプレートサービスがアクティブ化される前にアクティブであった QoS ポリシーと構造的にも機能的にもまったく変わりません。

セッションでアクティブな OUT QoS ポリシー :

```

policy-map output_parent
class class-default
  shape average 10000000
  service-policy output_child
policy-map output_child
class class-default

```

セッションでアクティブな IN QoS ポリシー :

```

policy-map input_parent
class class-default
  police 10000000
  service-policy input_child
policy-map input_child
class class-default
ip access-list extended voip-acl
  permit ip 10.1.1.0 0.0.0.255 any
ip access-list extended voip-control-acl
  permit ip 10.2.2.0 0.0.0.255 any
class-map match-any voip
  match access-group name voip-acl
!
class-map match-any voip-control
  match access-group name voip-control-acl
!
shell map voice-service1 police_rate=100000 prec_value=4 queue_size=1 in_h=class-default
out_h=class-default
{
  configure terminal
  accounting group $_acctgrp list default
  policy-map $_outgoing
  class voip
    police $police_rate 60625 0 conform-action transmit exceed-action drop violate-action
  drop
  exit
  priority level 1
  queue-limit 8 packets
  set precedence $prec_value
  set cos 6
  aaa-accounting group $_acctgrp
  class voip-control
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

  queue-limit $queue_size packets
  set precedence 6
  aaa-accounting group $_acctgrp
  policy-map $_incoming
  class voip
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
  drop
}

```

```

        set precedence 5
        aaa-accounting group $_acctgrp
    class voip-control
        police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
drop
        set precedence 7
        aaa-accounting group $_acctgrp
}

```

ターゲットセッションで `voice-service1` をアクティブにした後、次の OUT ポリシーがアクティブになります。

```

policy-map output_parent$class-default$voice-service1<<_OUT$class-default class
  class-default
    shape average 10000000
  service-policy output_child$voice-service1<<_OUT$class-default
policy-map output_child$voice-service1<<_OUT$class-default
  class voip
    police 10000 60625 0 conform-action transmit exceed-action drop violate-action drop
    priority level 1
    queue-limit 8 packets
    set precedence 4
    set cos 6
    aaa-accounting group voice-service1<<_GRP
  class voip-control
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
    queue-limit 16 packets
    set precedence 6
    aaa-accounting group voice-service1<<GRP
  class class-default

```

ターゲットセッションで `voice-service1` をアクティブにした後、次の IN ポリシーがアクティブになります。

```

policy-map input_parent$class-default$voice-service1<<_IN$class-default
  class class-default
    police cir 10000000 bc 312500 conform-action transmit exceed-action drop
  service-policy input_child$voice-service1<<_IN$class-default
policy-map input_child$voice-service1<<_IN$class-default
  class voip
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action drop
    set precedence 5
    aaa-accounting group voice-service1<<_GRP
  class voip-control
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action drop
    set precedence 7
    aaa-accounting group voice-service1<<_GRP
  class class-default

```

クラス名のオーバーライド

クラス名のオーバーライドはサービステンプレートの一種であり、QoS ポリシーのクラスでポリシング、シェーピング、帯域幅などの QoS アクション用に入力されるパラメータ設定の代わりに、シェル変数を使用します。また、サービステンプレートのポリシー定義で、クラス名の代わりにシェル変数が使用されることもあります。シェル変数を使ってクラス名を完全に置き換えることも、一定のプレフィックスを持つ可変サフィックスとして設定することもできます。

テンプレートサービスを非アクティブにすると、システムは以前にアクティブであった QoS ポリシーを復元します。その QoS ポリシーの名前は異なっている可能性があります。テンプレートサービスがアクティブ化される前にアクティブであった QoS ポリシーと構造的にも機能的にもまったく変わりません。

セッションでアクティブな OUT QoS ポリシー：

```

policy-map output_parent
  class class-default
    shape average 10000000

```

```

    service-policy output_child
policy-map output_child
class class-default
セッションでアクティブな IN QoS ポリシー :

policy-map input_parent
class class-default
    police 10000000
    service-policy input_child
policy-map input_child
class class-default
! Pre-configured ACLs/class-maps
ip access-list extended aol_classifier_acl          ! Locally pre-configured
permit ip host 10.1.30.194 any
class-map match-all voice-control-aol_classifier_reference ! Locally pre-configured
    match access-group name aol_classifier_acl
! Other pre-configured ACLs/classes here (e.g., voice-aol_classifier_reference,
voice-t_online, etc.)
! Service template:
shell map voice-aol-service1 prec_value=3 police_rate=100000 class_ref=t_online
in_h=class-default out_h=class-default
{
    configure terminal
accounting group $_acctgrp list default
policy-map $_outgoing
    class voice-control-$class_ref
        police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

        queue-limit 16 packets
        set precedence 6
        aaa-accounting group $_acctgrp
    class voice-$class_ref
        police $police_rate 60625 0 conform-action transmit exceed-action drop violate-action
drop
        priority level 1
        queue-limit 8 packets
        set precedence $prec_value
        set cos 6
        aaa-accounting group $_acctgrp
    policy-map $_incoming
        class voice-control-$class_ref
            police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
drop
        set precedence 7
        aaa-accounting group $_acctgrp
    class voice-$class_ref
        police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
drop
        set precedence $prec_value
        aaa-accounting group $_acctgrp
}

```

ターゲットセッションで `voice-aol-service1(class_ref=aol_classifier_reference)` をアクティブにした後、次の OUT ポリシーがアクティブになります。

```

policy-map
output_parent$class-default$voice-aol-service1<class_ref=aol_classifier_reference>_OUT$class-default

class class-default
    shape average 10000000
    service-policy
output_child$voice-aol-service1<class_ref=aol_classifier_reference>_OUT$class-default
policy-map
output_child$voice-aol-service1<class_ref=aol_classifier_reference>_OUT$class-default
    class voice-control-aol_classifier_reference ! Reference to pre-configured class
        police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

        queue-limit 16 packets
        set precedence 6
        aaa-accounting group voice-aol-service1<class_ref=aol_classifier_reference>_GRP
    class voice-aol_classifier_reference ! reference to pre-configured class
        police 100000 60625 0 conform-action transmit exceed-action drop violate-action

```

```

drop
    priority level 1
    queue-limit 8 packets
    set precedence 3
    set cos 6
    aaa-accounting group voice-aol-service1<class_ref=aol_classifier_reference>_GRP
class class-default
ターゲットセッションで voice-aol-service1(class_ref=aol_classifier_reference) をアクティブにした
後、次の IN ポリシーがアクティブになります。

policy-map
input_parent$class-default$voice-aol-service1<class_ref=aol_classifier_reference>_IN$class-default

    class class-default
        police cir 10000000 bc 312500 conform-action transmit exceed-action drop
        service-policy
input_child$voice-aol-service1<class_ref=aol_classifier_reference>_IN$class-default
policy-map input_child$voice-aol-service1<class_ref=aol_classifier_reference>_IN$class-default

    class voice-control-aol_classifier_reference      ! reference to pre-configured class
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
drop
    set precedence 7
    aaa-accounting group voice-aol-service1<class_ref=aol_classifier_reference>_GRP
class voice-aol_classifier_reference      ! reference to pre-configured class
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
drop
    set precedence 3
    aaa-accounting group voice-aol-service1<class_ref=aol_classifier_reference>_GRP
class-default

```

IP アドレスのパラメータ化

IP アドレスパラメータ化はアクションパラメーター化サービステンプレートの一種であり、ACL にエントリを追加することによって動的に分類子を変更できます。ACL に追加されるエントリは、シェル変数に格納される IP アドレスのリストです。

テンプレートサービスを非アクティブにすると、システムは以前にアクティブであった QoS ポリシーを復元します。その QoS ポリシーの名前は異なっている可能性があります。テンプレートサービスがアクティブ化される前にアクティブであった QoS ポリシーと構造的にも機能的にもまったく変わりません。



(注) クラスは動的に作成されないため、事前定義しておく必要があります。

セッションでアクティブな OUT QoS ポリシー :

```

policy-map output_parent
    class class-default
        shape average 10000000
        service-policy output_child
policy-map output_child
class class-default

```

セッションでアクティブな IN QoS ポリシー :

```

policy-map input_parent
    class class-default
        police 10000000
        service-policy input_child
policy-map input_child
    class-default
! Base ACLs:
ip access-list extended IPOne-control-acl      ! Base ACL locally pre-configured

```

```

permit ip any host 10.0.132.118
permit ip host 10.0.132.118 any
permit ip any host 10.1.245.122
permit ip host 10.1.245.122 any
ip access-list extended IPOne-combined-acl      ! Base ACL pre-configured
permit ip any 10.0.132.0 0.0.0.127
permit ip 10.0.132.0 0.0.0.127 any
permit ip any 10.1.245.64 0.0.0.63
permit ip 10.1.245.64 0.0.0.63 any
! Base class-maps:
class-map match-any voice-control              ! Base class map pre-configured
  match access-list name IPOne-control-acl     ! Match on the base ACL
class-map match-any voice                      ! base class-map pre-configured
  match access-list name IPOne-combined-acl   ! Match on the base ACL
! Service template:
shell map voice-toi prec_value=3 police_rate=100000 ip_list=10.2.1.0/28,10.2.1.0/29
in_h=class-default out_h=class-default
{
  configure terminal
  ! Class-map templates:
  classmap-template voice-control $ip_list
  classmap-template voice $ip_list
  ! Service parameter templates:
  policy-map $_outgoing
    class voice-control-$ip_list              ! class names MUST end with -$ip_list
      police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

      queue-limit 16 packets
      set precedence 6
      aaa-accounting group IPOne-aol
    class voice-$ip_list
      police $police_rate 60625 0 conform-action transmit exceed-action drop violate-action
drop
    priority level 1
    queue-limit 8 packets
    set precedence $prec_value
    aaa-accounting group IPOne-aol
  policy-map $_incoming
    class voice-control-$ip_list
      police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
drop
    set precedence 7
    aaa-accounting group IPOne-aol
  class voice-$ip_list
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
drop
    set precedence $prec_value
    aaa-accounting group IPOne-aol

```

ターゲットセッションで **voice-toi(ip_list=10.1.30.0/28,10.1.40.0/29)** をアクティブ化した後、次の OUT QoS ポリシーがアクティブになります。

```

policy-map output_parent$class-default$
voice-toi>ip_list=10.1.30.0/28,10.1.40.0/29<_OUT$class-default
  class class-default
    shape average 10000000
  service-policy output_child$voice-toi>ip_list=10.1.30.0/28,10.1.40.0/29<_OUT$class-default
policy-map output_child$voice-toi>ip_list=10.1.30.0/28,10.1.40.0/29<_OUT$class-default
  class voice-control-10.1.30.0/28,10.1.40.0/29
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop

    queue-limit 16 packets
    set precedence 6
    aaa-accounting group IPOne-aol
  class voice-10.1.30.0/28,10.1.40.0/29
    police 100000 60625 0 conform-action transmit exceed-action drop violate-action
drop
  priority level 1
  queue-limit 8 packets
  set precedence 3
  aaa-accounting group IPOne-aol
class class-default

```

ターゲットセッションで `voice-toi(ip_list=10.1.30.0/28,10.1.40.0/29)` をアクティブ化した後、次の IN QoS ポリシーがアクティブになります。

```
policy-map
input_parent$class-default$voice-toi>ip_list=10.1.30.0/28,10.1.40.0/29<_IN$class-default
  class class-default
    police cir 10000000 bc 312500 conform-action transmit exceed-action drop
  service-policy input_child$voice-toi>ip_list=10.1.30.0/28,10.1.40.0/29<_IN$class-default
policy-map input_child$voice-toi>ip_list=10.1.30.0/28,10.1.40.0/29<_IN$class-default
  class voice-control-10.1.30.0/28,10.1.40.0/29
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
  drop
    set precedence 7
    aaa-accounting group IPOne-aol
    class voice-10.1.30.0/28,10.1.40.0/29
      police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
  drop
    set precedence 3
    aaa-accounting group IPOne-aol
class-default
```



(注) 次の設定が動的に作成されます。

```
! Internally created ACLs:
ip access-list extended IPOne-control-acl-10.1.30.0/28,10.1.40.0/29
  permit ip any host 10.0.132.118
  permit ip host 10.0.132.118 any
  permit ip any host 10.1.245.122
  permit ip host 10.1.245.122 any
  permit ip 10.1.30.0 0.0.0.15 any ! ACEs derived from $ip_list
  permit ip any 10.1.30.0 0.0.0.15
  permit ip 10.1.40.0 0.0.0.7 any
  permit ip any 10.1.40.0 0.0.0.7
ip access-list extended IPOne-combined-acl-10.1.30.0/28,10.1.40.0/29
  permit ip any 10.0.132.0 0.0.0.127
  permit ip 10.0.132.0 0.0.0.127 any
  permit ip any 10.1.245.64 0.0.0.63
  permit ip 10.1.245.64 0.0.0.63 any
  permit ip 10.1.30.0 0.0.0.15 any ! ACEs derived from $ip_list
  permit ip any 10.1.30.0 0.0.0.15
  permit ip 10.1.40.0 0.0.0.7 any
  permit ip any 0.0.0.7 10.1.40.0
! internally created class-maps:
class-map match-any voice-control-10.1.30.0/28,10.1.40.0/29
  match access-group name IPOne-control-acl-10.1.30.0/28,10.1.40.0/29
class-map match-any voice-10.1.30.0/28,10.1.40.0/29
  match access-group name IPOne-combined-acl-10.1.30.0/28,10.1.40.0/29
```

Turbo Button サービス

Turbo Button サービスはアクションパラメータ化サービステンプレートの一種であり、入力親クラス `class-default` のポリシーパラメータと、出力親クラス `class-default` のシェーピングパラメータだけを動的に変更できます。

次に、Turbo Button サービスのサービステンプレートを作成する例を示します。

セッションでアクティブな OUT QoS ポリシー :

```
policy-map output_parent
  class class-default
    shape average 10000000
  service-policy output_child
```

```

policy-map output_child
class class-default
セッションでアクティブな IN QoS ポリシー :

policy-map input_parent
class class-default
  police 10000000
  service-policy input_child
policy-map input_child
class class-default
shell map turbo-button in_police_val=20000000 $out_shape=20000000
configure terminal
accounting group $_acctgrp list default
policy-map $_outgoing
class class-default
shape average $out_shape
aaa-accounting group $_acctgrp
policy-map $_incoming
class class-default
police $in_police_val
aaa-accounting group $_acctgrp

```

Turbo Button の有効化

次に、デフォルト値を使用して Turbo Button サービスをアクティブにする例を示します。

セッションでアクティブな OUT QoS ポリシー :

```

policy-map output_parent
class class-default
  shape average 10000000
  service-policy output_child
policy-map output_child
class class-default

```

セッションでアクティブな IN QoS ポリシー :

```

policy-map input_parent
class class-default
  police 10000000
  service-policy input_child
policy-map input_child
class class-default
accounting group turbo-button<< list default

accounting group turbo-button>> list default
! Service outgoing:
policy-map turbo-button>>_OUT
class class-default
shape average 20000000
aaa-accounting group turbo-button>> list default
! Service incoming:
policy-map turbo-button>>_IN
class class-default
police 20000000
aaa-accounting group turbo-button>> list default

```

ターゲットセッションでサービスをアクティブにした後、次の OUT ポリシーがアクティブになります。

```

policy-map output_parent$turbo-button>>_OUT$
class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

class class-default
  shape average 20000000
aaa-accounting group turbo-button>> list default
service-policy
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
policy-map
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
class void

```



```

police 200000 60625 0 conform-action transmit exceed-action drop violate-action drop
priority level 1
queue-limit 8 packets
set precedence 5
set cos 6

aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class voip-control
police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
queue-limit 32 packets
set precedence 6
aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class class-default

```

ターゲットセッションでサービスをアクティブにした後、次の IN ポリシーがアクティブになります。

```

policy-map input_parent$turbo-button>
< IN$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
class class-default
  police cir 20000000 bc 312500 conform-action transmit exceed-action drop
  aaa-accounting group turbo-button>< list default

service-policy
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
policy-map
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
  class voip
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action
  drop
    set precedence 5
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
  class voip-control
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action
  drop
    set precedence 7
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class-default

```

Turbo Button の非アクティブ化

次に、デフォルト値 VSA 252 0c turbo-button() を使用して Turbo Button サービスを非アクティブにする例を示します。

セッションでアクティブな OUT QoS ポリシー :

```

policy-map output_parent
class class-default
  shape average 10000000
  service-policy output_child
policy-map output_child
class class-default

```

セッションでアクティブな IN QoS ポリシー :

```

policy-map input_parent
class class-default
  police 10000000
  service-policy input_child
policy-map input_child
class-default

```

ターゲットセッションでサービスをアクティブにした後、次の OUT ポリシーがアクティブになります。

```

policy-map
output_parent$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

```

```

class class-default
  shape average 10000000
  service-policy
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
policy-map
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

class voip
  police 200000 60625 0 conform-action transmit exceed-action drop violate-action drop
  priority level 1
  queue-limit 8 packets
  set precedence 5
  set cos 6
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class voip-control
  police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
  queue-limit 32 packets
  set precedence 6
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP

class class-default

```

ターゲットセッションでサービスをアクティブにした後、次の IN ポリシーがアクティブになります。

```

policy-map
input_parent$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default

  class class-default
    police cir 10000000 bc 312500 conform-action transmit exceed-action drop
    service-policy
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default

policy-map
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
class voip
  police 200000 9216 0 conform-action transmit exceed-action transmit violate-action drop
  set precedence 5
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class voip-control
  police 112000 21000 0 conform-action transmit exceed-action transmit violate-action drop

  set precedence 7
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class-default

```

Turbo Button のオーバーライド

次に、デフォルト値 VSA 250 Aturbo-button(in_police_val=30000000, out_shape_val=30000000) (Access-Accept からアクティブにする場合) または VSA 252 0b turbo-button(in_police_val=30000000, out_shape_val=30000000) (CoA からアクティブにする場合) を使用して Turbo Button サービスをアクティブにする例を示します。

セッションでアクティブな OUT QoS ポリシー :

```

policy-map output_parent
  class class-default
    shape average 10000000
    service-policy output_child
policy-map output_child
class class-default

```

セッションでアクティブな IN QoS ポリシー :

```

policy-map input_parent
  class class-default
    police 10000000

```

```

    service-policy input_child
policy-map input_child
  class-default
accounting group turbo-button>in_police_val=30000000#out_shape_val=30000000 list default

! Service outgoing:
policy-map turbo-button>in_police_val=30000000#out_shape_val=30000000<_OUT
  class class-default
    shape average 30000000
  accounting group turbo-button>in_police_val=30000000#out_shape_val=30000000
! Service incoming:
policy-map turbo-button>in_police_val=30000000#out_shape_val=30000000<_IN
  class class-default
    police 30000000
  accounting group turbo-button>in_police_val=30000000#out_shape_val=30000000

```

ターゲットセッションでサービスをアクティブにした後、次の OUT ポリシーがアクティブになります。

```

policy-map output_parent$turbo-button>
in_police_val=30000000#out_shape_val=30000000<_OUT$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
class class-default
  shape average 20000000
  accounting group turbo-button>in_police_val=30000000#out_shape_val=30000000
  service-policy
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

  policy-map
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
class voip
  police 200000 60625 0 conform-action transmit exceed-action drop violate-action drop
  priority level 1
  queue-limit 8 packets
  set precedence 5
  set cos 6
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class voip-control
  police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
  queue-limit 32 packets
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class class-default

```

ターゲットセッションでサービスをアクティブにした後、次の IN ポリシーがアクティブになります。

```

policy-map
input_parent$turbo-button>in_police_val=30000000#out_shape_val=30000000<_IN$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
class class-default
  police cir 20000000 bc 312500 conform-action transmit exceed-action drop
  accounting group turbo-button>in_police_val=30000000#out_shape_val=30000000
  service-policy
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
policy-map
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
class voip
  police 200000 9216 0 conform-action transmit exceed-action transmit violate-action drop
  set precedence 5
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class voip-control
  police 112000 21000 0 conform-action transmit exceed-action transmit violate-action drop
  set precedence 7
  aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
class class-default

```

例 : Turbo Button のオーバーライドの非アクティブ化

次に、デフォルト値 VSA 252 0c turbo-button (in_police_val=30000000, out_shape_val=30000000) を使用して Turbo Button のオーバーライドを非アクティブにする例を示します。

セッションでアクティブな OUT QoS ポリシー :

```
policy-map output_parent
  class class-default
    shape average 10000000
    service-policy output_child
policy-map output_child
  class class-default
```

セッションでアクティブな IN QoS ポリシー :

```
policy-map input_parent
  class class-default
    police 10000000
    service-policy input_child
policy-map input_child
  class-default
```

ターゲットセッションでサービスをアクティブにした後、次の OUT ポリシーがアクティブになります。

```
policy-map
output_parent$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

  class class-default
    shape average 10000000
    service-policy
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default
policy-map
output_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_OUT$class-default

  class voip
    police 200000 60625 0 conform-action transmit exceed-action drop violate-action drop
    priority level 1
    queue-limit 8 packets
    set precedence 5
    set cos 6
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
  class voip-control
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
    queue-limit 32 packets
    set precedence 6
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
  class class-default
```

ターゲットセッションでサービスをアクティブにした後、次の IN ポリシーがアクティブになります。

```
policy-map
input_parent$class-default$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default

  class class-default
    police cir 10000000 bc 312500 conform-action transmit exceed-action drop
    service-policy
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
policy-map
input_child$voice-service1>police_rate=200000#prec_value=5#queue_size=32<_IN$class-default
  class voip
    police 200000 9216 0 conform-action transmit exceed-action transmit violate-action drop
    set precedence 5
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
  class voip-control
    police 112000 21000 0 conform-action transmit exceed-action transmit violate-action drop

    set precedence 7
    aaa-accounting group voice-service1>police_rate=200000#prec_value=5#queue_size=32<_GRP
  class-default
```

例：中間アカウンティング インターバルのオーバーライド

中間アカウンティング インターバル オーバーライドはアクションパラメータ化サービス テンプレートの一つであり、アカウンティング方式リストの定義で `interim interval` 値の代わりにシェル変数を使用して `account interim` 値を動的に変更できます。

次に、デフォルト値 `VSA 252 0b voice-service1(policy_rate=200000,prec_value=5,acct_interval=600)` を使用してアカウンティング グループをオーバーライドする例を示します。

この例では、次のパラメータを使用してサービスを生成します。

```
! Global AAA method list and accounting group parameters
aaa accounting network list-600
  action-type start-stop periodic interval 600
  accounting group voice-service1>policy_rate=200000#prec_value=5#acct_interval=600 <_GRP
list list-600
! OUT policy-map:
policy-map voice-service1>policy_rate=200000#prec_value=5#acct_interval=600 <_OUT
  class voip
    police 200000 60625 0 conform-action transmit exceed-action drop violate-action drop
    priority level 1
    queue-limit 8 packets
    set precedence 5
    set cos 6
  aaa-accounting group voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
<_GRP
  class voip-control
    police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
    queue-limit 32 packets
    set precedence 6
  aaa-accounting group
```

```
OUT:
policy-map output_parent
  class class-default
    shape average 10000000
    service-policy output_child
policy-map output_child
  class class-default
IN:
policy-map input_parent
  class class-default
    police 10000000
    service-policy input_child
policy-map input_child
  class-default
```

ターゲットセッションでサービスをアクティブにした後、次の OUT ポリシーがアクティブになります。

```
policy-map
output_parent$class-default$voice-service1>policy_rate=200000#prev_value=5#acct_interval=600
<_OUT$class-default
  class class-default
    shape average 10000000
  service-policy output_child$voice-service1>policy_rate=200000#prev_value=5#acct_interval=600
<_OUT$class-default
  policy-map output_child$voice-service1>policy_rate=200000#prev_value=5#acct_interval=600
<_OUT$class-default
  class voip
    police 200000 60625 0 conform-action transmit exceed-action drop violate-action drop
    priority level 1
    queue-limit 8 packets
    set precedence 5
    set cos 6
  aaa-accounting group voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
<_GRP
  class voip-control
```

```

police 112000 1000 0 conform-action transmit exceed-action drop violate-action drop
queue-limit 32 packets
set precedence 6
aaa-accounting group voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
< GRP
class class-default

```

ターゲットセッションでサービスをアクティブにした後、次の IN ポリシーがアクティブになります。

```

policy-map
input_parent$class-default$voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
< IN$class-default
class class-default
police cir 10000000 bc 312500 conform-action transmit exceed-action drop
service-policy input_child$voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
< IN$class-default
policy-map input_child$voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
< IN$class-default
class voip
police 200000 9216 0 conform-action transmit exceed-action transmit violate-action drop
set precedence 5
aaa-accounting group voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
< GRP
class voip-control
police 112000 21000 0 conform-action transmit exceed-action transmit violate-action drop
set precedence 7
aaa-accounting group voice-service1>policy_rate=200000#prec_value=5#acct_interval=600
< GRP
class class-default

```

加入者アカウンティング精度

加入者アカウンティング精度機能により、アカウンティング停止レコード内の I/O のパケット/バイト統計情報の精度が確実に 1 秒以内になります。

以下のイベントが発生すると、加入者アカウンティングデータが認証、許可、およびアカウンティング (AAA) サーバに送信されます。

- セッションまたはサービスのライフタイム中の設定済みインターバル
- サービス ログオフ
- セッション切断

加入者アカウンティング精度機能の値を設定するには、**subscriber accounting accuracy milliseconds** コマンドを使用します。

認可変更 (CoA) 要求応答

CoA 要求応答は、CoA に関する QoS アカウンティング レコードが送信される前に、CoA イベントごとに CoA-ACK を送信します。CoA には、単一または複数のサービスのアクティブ化または非アクティブ化が含まれることがあります。

セッションへのサービスのインストールが失敗した場合は、次の結果になります。

- CoA 全体が失敗します。

- Policy Manager が RADIUS サーバに CoA-NAK を送信します。
- 以前のサービス コンフィギュレーションが復元されます。

障害が検出される前に 1 つ以上のサービスがインストールされた場合は、次の結果になります。

- CoA 全体が失敗します。
- サービスが取り消されます。
- Policy Manager が RADIUS サーバに CoA-NAK を送信します。
- 以前のサービス コンフィギュレーションが復元されます。

マルチサービス CoA は、次のいずれかで構成されます。

- QoS サービス : Policy Manager は、複数のサービスを 1 つの実効ポリシー マップに結合します。すべてのサービスで、1 つの QoS ポリシーのみがセッションに適用されます。ポリシーのインストールが失敗した場合、システムは、以前のポリシー マップを使用するようセッションを復元します。実質的に、セッションは CoA の前の状態に復元されます。
- QoS およびインテリジェント サービス ゲートウェイ (ISG) サービス : Policy Manager は最初に ISG サービスを適用し、次に QoS サービスを適用します。ポリシーのインストールが失敗した場合、システムは、以前のポリシーマップにセッションを復元します。ISG サービスと QoS サービスの両方が、前の状態にロールバックされます。

マルチサービス CoA では、すべてのサービスが正常にインストールされた場合に 1 つの CoA-ACK だけが送信されます。

認可変更ロールバック

CoA ロールバック機能は、QoS ポリシー アカウンティングを、CoA 発行前の状態に復元します。また、CoA ロールバックは、CoA-NAK を使用して適切に RADIUS サーバに応答します。

CoA ロールバック機能が適用される対象は、構文の間違い、およびポリシーのインストール失敗 (アドミッション制御やリソース割り当ての失敗など) です。

CoA が失敗した場合、システムは CoA-NAK を送信しますが、QoS アカウンティング レコードを送信しません。既存のサービスのアカウンティング レコードでは以前のカウンタが保持され、引き続き新しいパケットがカウントされます。

QoS アカウンティング ハイ アベイラビリティ

クラスで QoS アカウンティングがイネーブルにされている場合、ポリシー アカウンティング機能では次の 3 種類のイベントがサポートされます。

- 開始 : 新しいアカウンティング フローを示します。開始レコードには、このフローに固有の統計情報と属性が記録されます。
- 中間 : フロー統計情報が報告される頻度を示します。

- 停止：アカウンティング フローの終了を示します。停止レコードにも、このフローに固有の統計情報と属性が記録されます。

ポリシー アカウンティング機能は、アカウンティング フローの統計情報を収集し、その情報を RADIUS アカウンティング課金サーバに送信します。

QoS アカウンティング ハイ アベイラビリティ機能により、予定または予定外のフェールオーバーが発生しても、開始、中間、および停止アカウンティングレコードには影響が及びません。予定または予定外のフェールオーバーが発生すると、QoS アカウンティング HA 機能は、RADIUS アカウンティング課金サーバへの情報フローを中断せずに RP スイッチオーバーが行われます。また、この機能により、RP スイッチオーバーが発生してもすべてのアクティブセッションですべての QoS サービスが中断せずに続行し、サービス アカウンティング カウンタが存続するようになります。

ポリシー アカウンティング状態の持続

開始、停止、中間アカウンティングがステートフルスイッチオーバー (SSO) や In-Service Software Upgrade (ISSU) の影響を受けないようにするために、Policy Manager はフェールオーバー発生時にすべての QoS サービスおよびパラメータ化された CoA 機能をスタンバイ RP との間で同期させます。さらに、アクティブ RP とスタンバイの RP の間で、動的 QoS 設定およびポーリング間隔が同期されます。

パラメータ化された CoA イベントをスタンバイ RP に同期させるために、Policy Manager は次の機能を実行します。

- スタンバイ RP でプロビジョニング イベントを同期させる CoA リプレイを管理します。
- アクティブ RP とスタンバイ RP の両方で同じサービス テンプレートを 사용합니다。
- アクティブ RP とスタンバイ RP の両方のセッションに適用する同じ名前のポリシー マップとクラス マップを作成します。
- サービス テンプレートのアクティブセッション中に、事前定義された QoS ポリシー マップとクラス マップを 사용합니다。

ポリシー アカウンティング カウンタの持続

QoS アカウンティング HA 機能は、SSO またはフェールオーバーが発生してもポリシー アカウンティング カウンタが存続するようにします。スイッチオーバーが発生すると、スタンバイ RP がアクティブ RP になり、以前のアクティブ RP から統計情報を収集します。新しくアクティブになった RP は、スイッチオーバー後に定期的な更新を受信すると、収集済みの統計情報に定期的な更新の値を加えて、中間レコードを生成します。スイッチオーバー後に定期的な更新が受信されない場合、新しくアクティブになった RP は以前のアクティブ RP から収集した統計情報だけを使って中間レコードを生成します。

SSO と ISSU の詳細については、『Cisco IOS ハイ アベイラビリティ コンフィギュレーション ガイド』を参照してください。

QoS ポリシー アカウンティングの使用法

QoS ポリシー アカウンティングを使用するには、グループまたは AAA 方式リストをトラフィック クラスに割り当ててから、ポリシー アカウンティング用のサービス テンプレートを設定します。そして最後に、加入者アカウンティング精度機能を有効にします。



(注) デフォルトでは、QoS ポリシー アカウンティングはトラフィック クラスに割り当てられていません。

トラフィック クラスへのグループまたは AAA 方式リストの割り当て

はじめる前に

グループまたは AAA 方式リストがすでに存在することを確認します。未定義のグループまたは AAA 方式リストをトラフィック クラスに追加しようとする、エラー メッセージが表示されません。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa authentication ppp *list-name method1***
4. **aaa accounting network *methodlist-name***
5. **action-type start-stop**
6. **periodic interval *minutes***
7. **accounting group *group_name list list-name***
8. **policy-map *policy-map-name***
9. **class class-default**
10. **accounting aaa list *list-name [group-name]***
11. **end**
12. **show policy-map session**
13. **show accounting group *group-name***

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Router> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Router# configure terminal	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 3	aaa authentication ppp list-name method1 例： Router(config)# aaa authentication ppp group radius	有効な AAA 認証方式を指定します。 • グループ RADIUS により、グローバル RADIUS 認証がイネーブルになります。
ステップ 4	aaa accounting network methodlist-name 例： Router(config)# aaa accounting network list1	RADIUS を使用する場合、サービスの AAA をイネーブルにします。 • クラスまたはグループの暫定インターバルを決定するアルゴリズムは、ここで指定された方式リストを使用します。
ステップ 5	action-type start-stop 例： Router(config)# action-type start-stop	プロセスの開始時に start accounting 通知を送信し、プロセスの終了時に stop accounting 通知を送信します。
ステップ 6	periodic interval minutes 例： Router(config)# periodic interval 1	暫定インターバル値（1 ～ 71,582 分）を方式リストに追加します（指定されている場合）。 • 暫定インターバルを定義しない場合、AAA で定義されたグローバル値が使用されます。 • 方式リストで暫定アップデートがディセーブルにされている場合、方式リストを使用するアカウンティングフローは暫定アップデートを生成しません。
ステップ 7	accounting group group_name list list-name	AAA 方式リスト内のプロパティを設定します。

	コマンドまたはアクション	目的
	例 : <pre>Router(config)# accounting group group_name AAAMethodlist AAAMethodlist1</pre>	<ul style="list-style-type: none"> 既存のトラフィック クラスに対してセッション単位の変更を加えるには、その割り当て先であるグループまたは AAA 方式リスト内のプロパティを一時的に上書きすることができます。これにより、加入者ごとに動的にカスタマイズされた QoS 設定を指定できます。
ステップ 8	policy-map <i>policy-map-name</i> 例 : <pre>Router(config)# policy-map pl</pre>	ポリシー マップを作成します。
ステップ 9	class <i>class-default</i> 例 : <pre>Router(config)# class class-default</pre>	トラフィック クラスを作成します。
ステップ 10	accounting aaa list <i>list-name</i> [<i>group-name</i>] 例 : <pre>Router(config)# accounting aaa list AAAMethodlist1</pre>	グループまたは AAA 方式リストにトラフィック クラスを割り当てます。 <ul style="list-style-type: none"> この例は、(グループを使用せず) 「AAAMethodlist1」リストを使用して、トラフィック クラスのインスタンスに関してイネーブルにされた QoS ポリシー アカウンティング機能を示しています。
ステップ 11	end 例 : <pre>Router(config)# end</pre>	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。
ステップ 12	show policy-map session 例 : <pre>Router# show policy-map session</pre>	(オプション) グループまたは AAA 方式リストを使用するトラフィック クラスに関する QoS ポリシー アカウンティング機能の情報を表示します。
ステップ 13	show accounting group <i>group-name</i> 例 : <pre>Router# show accounting group acc-group1</pre>	(オプション) すべてのグループと方式リストの関連付けを表示します。 <ul style="list-style-type: none"> グループに固有の情報を表示するには、そのグループの名前を入力します。

加入者アカウンティング精度のアクティブ化

手順の概要

1. **enable**
2. **configure terminal**
3. **subscriber accounting accuracy** *milliseconds*
4. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	configure terminal 例： Device# configure terminal	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 3	subscriber accounting accuracy <i>milliseconds</i> 例： Device(config)# subscriber accounting accuracy 1000	加入者アカウンティング精度機能の値を設定します。
ステップ 4	end 例： Device(config)# end	特権 EXEC モードを開始します。

サービス テンプレートのトラブルシューティング

サービス テンプレートのトラブルシューティングを行うために、ルータ上のすべてのテンプレート サービス ポリシー マップの使用状況に関する情報を表示できます。

手順の概要

1. enable
2. show subscriber policy ppm-shim-db

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	enable 例： Router> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	show subscriber policy ppm-shim-db 例： Router(config)# show subscriber policy ppm-shim-db	ルータ上のすべてのテンプレートサービスポリシーマップおよび実効ポリシーマップの参照カウント（使用状況）を表示します。

QoS ポリシー アカウンティングの設定例

例：グループ単位 QoS ポリシー アカウンティング機能の使用

次に、グループ化の例を示します。

```
policy-map my-policy
class voip
police
aaa-accounting group premium-services
class voip-control
police
aaa-accounting group premium-services
```

例：個別アカウンティングストリームの生成

次の例では、「class voip」と「class voip-control」という名前の2つの分類子を示します。これらの分類子は、1つのターゲットに関連付けられた1つのポリシーに割り当てられます。この設定により、2つの別個のQoSポリシーアカウンティングストリームが生成されます。

```
policy-map my-policy
class voip
police 200000
accounting aaa list AAA-LIST
```

```
class voip-control
  police 100000
  accounting aaa list AAA-LIST
```

その他の関連資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『 Cisco IOS Master Commands List, All Releases 』
QoS コマンド	『 Cisco IOS QoS Command Reference 』
Cisco IOS のハイ アベイラビリティ	『 Cisco IOS High Availability Configuration Guide 』

規格

規格	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	--

MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャセットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
RFC 2866	RADIUS アカウンティング

テクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカルサポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

QoS ポリシー アカウンティングの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、特定のソフトウェア リリース トレーンで各機能のサポートが導入されたときのソフトウェア リリースのみを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 1: QoS ポリシー アカウンティングの機能情報

機能名	リリース	機能情報
QoS アカウンティング HA	Cisco IOS XE Release 3.5S	<p>QoS アカウンティング ハイ アベイラビリティ (HA) 機能により、予定済みまたは予定外のルートプロセッサ (RP) スイッチオーバーが発生しても、QoS アカウンティング統計情報が確実に存続するようになり、RADIUS アカウンティング課金サーバはアカウンティングカウンタの報告を確実に続行します。</p> <p>Cisco IOS XE Release 3.5S で、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータにこの機能が導入されました。次のコマンドが変更されました。debug qos accounting</p>

機能名	リリース	機能情報
QoS ポリシー アカウンティング	Cisco IOS XE Release 2.6 Cisco IOS XE Release 3.2S Cisco IOS XE Release 3.8S	<p>QoS ポリシー アカウンティング機能は、システムのトラフィックを正確に考慮するうえで役立ちます。また、QoS 設定を加入者に割り当てる際の柔軟性も向上します。</p> <p>静的 CLI-driven アカウンティングがサポートされます。</p> <p>この機能は、Cisco IOS XE Release 2.6 で Cisco ASR 1000 シリーズ アグリゲーション サービスルータに導入されました。</p> <p>Cisco IOS XE Release 3.2S では、サービス テンプレート、加入者サブセカンド精度、動的 CoA がサポートされ、動的アクティベーションがサービスに作用しなかった場合でもアカウンティングが中断されないようになっています。</p> <p>次のコマンドが追加されました。show subscriber policy ppm-shim-db、subscriber accounting accuracy。</p>

