



# Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定

輻輳回避技術では、トラフィック フローをモニタすることにより、共通ネットワークのボトルネックでの輻輳を予測し回避します。発生した後に輻輳を制御する輻輳管理技術に対し、回避技術は輻輳が発生する前に実行されます。

輻輳回避は、パケットのドロップにより行われます。Cisco IOS XR ソフトウェアでは、パケットをドロップする次の Quality of Service (QoS) 輻輳回避技術がサポートされます。

- ランダム早期検出 (RED)
- 重み付けランダム早期検出 (WRED)
- テール ドロップ

このモジュールでは、次の輻輳回避技術に関連する概念および作業について説明します。

## ラインカード、SIP および SPA のサポート

機能	ASR 9000 イーサネット ラインカード	ASR 9000 用 SIP 700
ランダム早期検出	あり	あり
重み付けランダム早期検出	あり	あり
テール ドロップ	あり	あり

## Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定の機能履歴

リリース	変更箇所
リリース 3.7.2	輻輳回避機能が ASR 9000 イーサネット ラインカードで導入されました。 ランダム早期検出、重み付けランダム早期検出、およびテール ドロップの各機能が ASR 9000 イーサネット ラインカードで導入されました。
リリース 3.9.0	ランダム早期検出、重み付けランダム早期検出、およびテール ドロップの各機能が ASR 9000 用 SIP 700 でサポートされました。

## 内容

- 「Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定の前提条件」 (P.94)
- 「Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定方法」 (P.96)
- 「その他の関連資料」 (P.106)

# Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定の前提条件

ネットワークでの QoS 輻輳回避を設定するには、次の前提条件を満たす必要があります。

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンドリファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

# Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定に関する情報

このマニュアルの QoS 輻輳回避技術を設定するには、次の概念を理解している必要があります。

- 「ランダム早期検出と TCP」(P.94)
- 「重み付けランダム早期検出による優先トラフィック処理」(P.94)
- 「テール ドロップと FIFO キュー」(P.95)

## ランダム早期検出と TCP

RED 輻輳回避技術は、TCP の輻輳制御メカニズムを利用しています。高輻輳期間の前にランダムにパケットをドロップすることにより、RED はパケットの送信元に、その伝送レートを低下させるよう指示します。パケット送信元が TCP を使用している場合、送信元はすべてのパケットが宛先に届くようになるまで伝送レートを下げます。これは輻輳が解消されたことを示します。TCP にパケットの送信速度を下げさせる手段として RED を使用できます。TCP は停止するだけでなく、素早く再起動して、ネットワークがサポート可能なレートに伝送レートを対応させます。

RED は時間の損失を分散させて、トラフィックのバーストを吸収しながら通常の低いキューの深さを維持します。インターフェイスでイネーブルにすると、RED は、設定時に選択したレートで輻輳が発生した場合にパケットのドロップを開始します。

## 重み付けランダム早期検出による優先トラフィック処理

WRED は、特定的一致基準（DSCP、廃棄クラスなど）に基づいて単一のクラス内の複数の RED プロファイルを定義する機能を提供します。これにより、パケットの相対的な重要性に基づいて、異なるドロップ優先順位を設定できます。WRED では、インターフェイスで輻輳が発生し始めると、よりプライオリティが低いトラフィックを選択的に廃棄し、異なるサービスクラスに対して差別化したパフォーマンス特性を提供できます。重み付けされていない RED の動作が行われるように、ドロップの決定を行う際に IP precedence を無視するよう WRED を設定できます。

WRED は、輻輳の可能性を早期検出し、複数のトラフィック クラスを実現します。また、グローバル同期期に対して保護されます。そのため、WRED は輻輳が発生すると予測される任意の出力インターフェイスに有効です。

ただし、WRED は通常は、ネットワークのエッジではなくネットワークのコア ルータで使用されません。エッジ ルータは、ネットワークに送信されるパケットに IP precedence を割り当てます。WRED は、これらの優先順位を使用して、異なるタイプのトラフィックの処理方法を定義します。

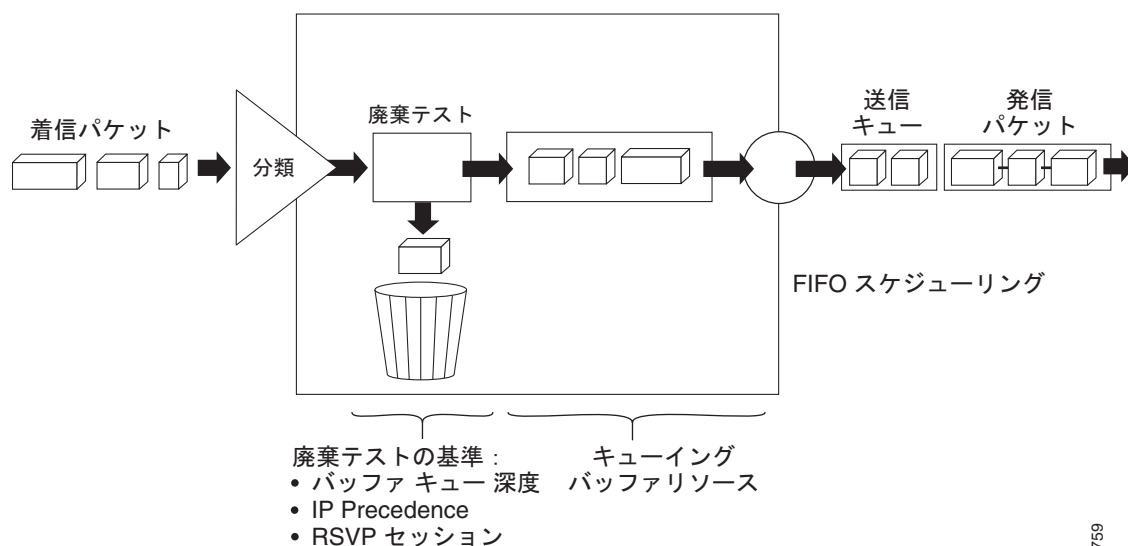
WRED では、異なる分類基準 (IP precedence、MPLS EXP 値など) に対する個別のドロップしきい値 (最小および最大) を設定でき、異なるトラフィック タイプでのパケットのドロップに対し、異なる Quality of Service を実現できます。標準的なトラフィックは、輻輳期間中には、優先度の高いトラフィックよりも頻繁にドロップされる可能性があります。

WRED は非 IP トラフィックを優先順位 0 の最低優先順位として扱います。したがって、非 IP トラフィックは通常、IP トラフィックよりもドロップされやすくなります。

WRED は、ほとんどのトラフィックが TCP/IP トラフィックである場合にだけ有用です。TCP では、破棄されたパケットは輻輳を示しているため、パケットの発信元は伝送レートを低下させます。他のプロトコルでは、パケットの発信元は応答しないか、または破棄されたパケットを同じレートで再送信します。このため、パケットを破棄しても輻輳は低減されません。

図 1 に、WRED の動作方法を示します。

図 1 重み付けランダム早期検出



16759

## WRED のキュー制限

キュー制限は、各キューに使用可能なバッファ数を微調整するために使用されます。これはキューイングクラスでのみ使用できます。デフォルトのキュー制限は、指定されたキューのサービス レートの 100 ms です。サービス レートは、最小保証帯域幅と特定のクラスに暗黙的または明示的に割り当てられた残存帯域幅の合計です。

キュー制限は、8 KB、16 KB、24 KB、32 KB、48 KB、64 KB、96 KB、128 KB、192 KB、256 KB、384 KB、512 KB、768 KB、1024 KB、1536 KB、2048 KB、3072 KB、4196 KB、8192 KB、16394 KB、32768 KB、65536 KB、131072 KB、262144 KB のいずれかの値に丸められます。

## テール ドロップと FIFO キュー

テール ドロップは、出力キューが満杯のときに、輻輳が削除されるまでパケットをドロップする輻輳回避技術です。テール ドロップでは、すべてのトラフィック フローを平等に扱い、サービス クラス間で区別しません。テール ドロップは、ファーストインファーストアウト (FIFO) キューに入り、下位リンク帯域幅によって決定したレートで転送された未分類のパケットを管理します。

「Cisco ASR 9000 シリーズ ルータでのモジュラ QoS パケット分類とマーキングの設定」モジュールの「デフォルトのトラフィック クラス」の項を参照してください。

# Cisco ASR 9000 シリーズ ルータでのモジュラ QoS 輻輳回避の設定方法

ここでは、次の作業の手順について説明します。

- 「ランダム早期検出の設定」(P.96) (必須)
- 「重み付けランダム早期検出の設定」(P.98) (必須)
- 「テール ドロップの設定」(P.102) (必須)

## ランダム早期検出の設定

任意のクラスで **default** キーワードを指定して **random-detect** コマンドを設定することにより、ランダム早期検出 (RED) を設定できます。この方法で、単一の RED プロファイルがクラスに一致するすべてのパケットに適用されます。

この設定作業は WRED で行う場合と同様ですが、**random-detect precedence** コマンドは設定せずに、RED をイネーブルにするために、**default** キーワードを指定した **random-detect** コマンドを使用する必要があります。

## 制約事項

**class-default** を含む任意のクラスで **random-detect default** コマンドを設定する場合は、次のいずれかのコマンドを設定する必要があります。

- **shape average**
- **bandwidth**
- **bandwidth remaining**

## 手順の概要

1. **configure**
2. **policy-map *policy-name***
3. **class *class-name***
4. **random-detect {*cos value* | **default** | **discard-class** *value* | **dscp** *value* | **exp** *value* | **precedence** *value* | **min-threshold** [*units*] **max-threshold** [*units*] }**
5. **bandwidth {*bandwidth* [*units*] | **percent** *value*}**  
または  
**bandwidth remaining [**percent** *value* | **ratio** *ratio-value*]**
6. **shape average {**percent** *percentage* | *value* [*units*]}**
7. **exit**
8. **exit**
9. **interface *type interface-path-id***

**10. service-policy {input | output} policy-map****11. end**

または  
**commit**

## 手順の詳細

	コマンドまたはアクション	目的
ステップ1	<b>configure</b>  例： RP/0/RSP0/CPU0:router# configure	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	<b>policy-map policy-map-name</b>  例： RP/0/RSP0/CPU0:router(config)# policy-map policy1	ポリシー マップ コンフィギュレーション モードを開始します。  • 1 つ以上のインターフェイスに付加できるポリシー マップを作成または変更し、サービス ポリシーを指定します。
ステップ3	<b>class class-name</b>  例： RP/0/RSP0/CPU0:router(config-pmap)# class class1	ポリシー マップ クラス コンフィギュレーション モードを開始します。  • ポリシーを作成または変更するクラスの名前を指定します。
ステップ4	<b>random-detect {cos value   default   discard-class value   dscp value   exp value   precedence value   min-threshold [units]   max-threshold [units] }</b>  例： RP/0/RSP0/CPU0:router(config-pmap-c)# random-detect default	デフォルトの最小しきい値および最大しきい値を使用した RED をイネーブルにします。
ステップ5	<b>bandwidth {bandwidth [units]   percent value}</b> または <b>bandwidth remaining [percent value   ratio ratio-value]</b>  例： RP/0/RSP0/CPU0:router(config-pmap-c)# bandwidth percent 30 または RP/0/RSP0/CPU0:router(config-pmap-c)# bandwidth remaining percent 20	(任意) ポリシー マップに属しているクラスに割り当てる帯域幅を指定します。  または  (任意) さまざまなクラスに残りの帯域幅を割り当てる方法を指定します。  (注) 非デフォルトクラスには、これらの設定のいずれかが必要です。
ステップ6	<b>shape average {percent percentage   value [units]}</b>  例： RP/0/RSP0/CPU0:router(config-pmap-c)# shape average percent 50	(任意) 指定されたビット レートまたは使用可能な帯域幅のパーセンテージに従い、トラフィックをシェーピングします。

	コマンドまたはアクション	目的
ステップ7	<code>exit</code>  <b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# exit	ルータをポリシー マップ コンフィギュレーション モードに戻します。
ステップ8	<code>exit</code>  <b>例:</b> RP/0/RSP0/CPU0:router(config-pmap)# exit	ルータをグローバル コンフィギュレーション モードに戻します。
ステップ9	<code>interface type interface-path-id</code>  <b>例:</b> RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/0	コンフィギュレーション モードを開始し、インターフェイスを設定します。
ステップ10	<code>service-policy {input   output} policy-map</code>  <b>例:</b> RP/0/RSP0/CPU0:router(config-if)# service-policy output policyl1	インターフェイスのサービス ポリシーとして使用する入力インターフェイスまたは出力インターフェイスにポリシーマップを付加します。  <ul style="list-style-type: none"> <li>この例では、トラフィック ポリシーでそのインターフェイスから送信されるすべてのトラフィックを評価します。</li> </ul>
ステップ11	<code>end</code> または <code>commit</code>  <b>例:</b> RP/0/RSP0/CPU0:router(config-cmap)# end または RP/0/RSP0/CPU0:router(config-cmap)# commit	設定変更を保存します。  <ul style="list-style-type: none"> <li><b>end</b> コマンドを実行すると、変更をコミットするように要求されます。  Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:   <ul style="list-style-type: none"> <li><b>yes</b> と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。</li> <li><b>no</b> と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。</li> <li><b>cancel</b> と入力すると、現在のコンフィギュレーション セッションが継続します。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。</li> </ul> </li> <li>実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーション セッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## 重み付けランダム早期検出の設定

WRED は、指定した基準（CoS、DSCP、EXP、廃棄クラス、優先順位など）に基づいてパケットを選択的にドロップします。WRED は、これらの一致基準を使用して、異なるタイプのトラフィックの処理方法を決定します。

**random-detect** コマンドと異なる CoS、DSCP、EXP、および廃棄クラスの値を使用して、WRED を設定します。値には、そのフィールドにおいて有効な値の範囲またはリストを指定できます。また、最小キューしきい値および最大キューしきい値を使用して、ドロップするポイントを決定できます。

パケットが着信すると、次の処理が行われます。

- キュー サイズが最小キューしきい値よりも小さい場合、着信パケットはキューイングされます。
- キュー サイズがそのトラフィック タイプの最小キューしきい値と、インターフェイスの最大しきい値の間の場合、そのトラフィック タイプのパケット ドロップ確率に応じて、パケットはドロップされるか、キューイングされます。
- キュー サイズが最大しきい値を超える場合、パケットはドロップされます。

## 制約事項

**random-detect dscp** コマンドを設定する場合は、**shape average**、**bandwidth**、および **bandwidth remaining** のいずれかのコマンドを設定する必要があります。

クラスごとに 2 つの最小しきい値および最大しきい値（それぞれ異なる一致基準）のみを設定できます。

## 手順の概要

1. **configure**
2. **policy-map** *policy-name*
3. **class** *class-name*
4. **random-detect dscp** *dscp-value min-threshold [units] max-threshold [units]*
5. **bandwidth** {*bandwidth [units] | percent value*}  
または  
**bandwidth remaining** [*percent value | ratio ratio-value*]
6. **shape average** {*percent | value [units]*}
7. **queue-limit** *value [units]*
8. **exit**
9. **interface** *type interface-path-id*
10. **service-policy** {*input | output*} *policy-map*
11. **end**  
または  
**commit**

## 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><code>configure</code></p> <p>例： RP/0/RSP0/CPU0:router# configure</p>	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	<p><code>policy-map policy-name</code></p> <p>例： RP/0/RSP0/CPU0:router(config)# policy-map policy1</p>	<p>ポリシー マップ コンフィギュレーション モードを開始します。</p> <ul style="list-style-type: none"> <li>1 つ以上のインターフェイスに付加できるポリシー マップを作成または変更し、サービス ポリシーを指定します。</li> </ul>
ステップ 3	<p><code>class class-name</code></p> <p>例： RP/0/RSP0/CPU0:router(config-pmap)# class class1</p>	<p>ポリシー マップ クラス コンフィギュレーション モードを開始します。</p> <ul style="list-style-type: none"> <li>ポリシーを作成または変更するクラスの名前を指定します。</li> </ul>
ステップ 4	<p><code>random-detect dscp dscp-value min-threshold [units] max-threshold [units]</code></p> <p>例： RP/0/RSP0/CPU0:router(config-pmap-c)# random-detect dscp af11 1000000 bytes 2000000 bytes</p>	<p>DSCP 値の最小および最大パケットしきい値を変更します。</p> <ul style="list-style-type: none"> <li>WRED をイネーブルにします。</li> <li><i>dscp-value</i> : DSCP 値を設定する 0 ~ 63 の数。数値の代わりに、予約済みキーワードも指定できます。</li> <li><i>min-threshold</i> : 指定した単位の最小しきい値。平均キューの長さが最小しきい値に達すると、WRED では、指定された DSCP の値で一部のパケットがランダムにドロップされます。</li> <li><i>max-threshold</i> : 指定した単位の最大しきい値。キューの平均の長さが最大しきい値を超えると、WRED は指定した DSCP 値のすべてのパケットをドロップします。</li> <li><i>units</i> : しきい値の単位。これには、<b>bytes</b>、<b>gbytes</b>、<b>kbytes</b>、<b>mbytes</b>、<b>ms</b> (ミリ秒)、<b>packets</b>、または <b>us</b> (マイクロ秒) を指定できます。デフォルトは <b>packets</b> です。</li> <li>次に、DSCP AF11 のパケットで、WRED の最小しきい値が 1,000,000 バイト、最大しきい値が 2,000,000 バイトの場合の例を示します。</li> </ul>



	コマンドまたはアクション	目的
ステップ5	<p><b>bandwidth</b> {<i>bandwidth</i> [units]   <b>percent</b> value}</p> <p>または</p> <p><b>bandwidth remaining</b> [<b>percent</b> value   <b>ratio</b> ratio-value]</p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# bandwidth percent 30 または RP/0/RSP0/CPU0:router(config-pmap-c)# bandwidth remaining percent 20</p>	<p>(任意) ポリシー マップに属しているクラスに割り当てる帯域幅を指定します。</p> <p>または</p> <p>(任意) さまざまなクラスに残りの帯域幅を割り当てる方法を指定します。</p> <p><b>(注)</b> 非デフォルト クラスには、これらの設定のいずれかが必要です。</p>
ステップ6	<p><b>shape average</b> {<b>percent</b> percentage   value [units]}</p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# shape average percent 50</p>	<p>(任意) 指定されたビット レートまたは使用可能な帯域幅のパーセンテージに従い、トラフィックをシェーピングします。</p>
ステップ7	<p><b>queue-limit</b> value [units]</p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# queue-limit 50 ms</p>	<p>(任意) キュー制限を変更して、各キューで使用可能なバッファ量を微調整します。デフォルトのキュー制限は、指定されたキュー クラスのサービス レートの 100 ms です。</p>
ステップ8	<p><b>exit</b></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap)# exit</p>	<p>ルータをグローバル コンフィギュレーション モードに戻します。</p>
ステップ9	<p><b>interface</b> type interface-path-id</p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config)# interface gigabitethernet 0/2/0/0</p>	<p>コンフィギュレーション モードを開始し、インターフェイスを設定します。</p>

コマンドまたはアクション	目的
<p>ステップ 10 <code>service-policy {input   output} policy-map</code></p> <p><b>例 :</b>  RP/0/RSP0/CPU0:router(config-if)#  service-policy output policy1</p>	<p>インターフェイスのサービス ポリシーとして使用する入力インターフェイスまたは出力インターフェイスにポリシーマップを付加します。</p> <ul style="list-style-type: none"> <li>この例では、トラフィック ポリシーでそのインターフェイスから送信されるすべてのトラフィックを評価します。</li> <li></li> </ul>
<p>ステップ 11 <code>end</code>  または  <code>commit</code></p> <p><b>例 :</b>  RP/0/RSP0/CPU0:router(config-cmap)# end  または  RP/0/RSP0/CPU0:router(config-cmap)# commit</p>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <li><b>end</b> コマンドを実行すると、変更をコミットするように要求されます。   Uncommitted changes found, commit them before exiting(yes/no/cancel)?  [cancel]: <ul style="list-style-type: none"> <li><b>yes</b> と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。</li> <li><b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。</li> <li><b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。</li> </ul> </li> <li>実行コンフィギュレーション ファイルに変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## テール ドロップの設定

クラスの一致基準を満たすパケットは、サービスを提供されるまで、クラス用に予約されたキューに蓄積されます。**queue-limit** コマンドを使用して、クラスの最大しきい値を定義します。最大しきい値に達したとき、クラス キューへの待機パケットはテール ドロップ (パケット ドロップ) します。

**queue-limit** の値には、**queue\_bandwidth** の基準値として、キューの保証サービス レート (GSR) が使用されます。クラスに帯域幅の割合が関連付けられている場合、**queue-limit** は、そのクラス用に予約された帯域幅の割合に設定されます。

キューの GSR がゼロの場合は、次を使用してデフォルトの **queue-limit** を計算します。

- 非階層型ポリシー内のキューのインターフェイス帯域幅の 1%。
- 階層型ポリシー内のキューの最小の親シェーピング レートおよびインターフェイス レートの 1%。



(注) デフォルトの **queue-limit** は、キューの帯域幅の 100 ms のバイトに設定されます。デフォルトのキュー制限 (バイト単位) の計算には、次の式が使用されます。

$$\text{bytes} = (100 \text{ ms} / 1000 \text{ ms}) \times \text{queue\_bandwidth (kbps)} / 8$$

## 制約事項

- **queue-limit** コマンドを設定する場合は、**priority**、**shape average**、**bandwidth**、または **bandwidth remaining** のいずれかのコマンドを設定する必要があります。

## 手順の概要

1. **configure**
2. **policy-map** *policy-name*
3. **class** *class-name*
4. **queue-limit** *value* [*units*]
5. **class** *class-name*
6. **bandwidth** {*bandwidth* [*units*] | **percent** *value*}
- または
- bandwidth remaining** [**percent** *value* | **ratio** *ratio-value*]
7. **exit**
8. **exit**
9. **interface** *type interface-path-id*
10. **service-policy** {**input** | **output**} *policy-map*
11. **end**
- または
- commit**

## 手順の詳細

	コマンドまたはアクション	目的
ステップ1	<b>configure</b>  例: RP/0/RSP0/CPU0:router# configure	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	<b>policy-map</b> <i>policy-name</i>  例: RP/0/RSP0/CPU0:router(config)# policy-map policy1	ポリシー マップ コンフィギュレーション モードを開始します。  • 1 つ以上のインターフェイスに付加できるポリシー マップを作成または変更し、サービス ポリシーを指定します。

	コマンドまたはアクション	目的
ステップ3	<p><code>class class-name</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap)# class class1</p>	<p>ポリシー マップ クラス コンフィギュレーション モードを開始します。</p> <ul style="list-style-type: none"> <li>ポリシーを作成または変更するクラスの名前を指定します。</li> </ul>
ステップ4	<p><code>queue-limit value [units]</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# queue-limit 1000000 bytes</p>	<p>ポリシー マップに設定したクラス ポリシー用にキューが保持できる最大値を指定または変更します。<i>units</i> 引数のデフォルト値は <b>packets</b> です。</p> <ul style="list-style-type: none"> <li>この例では、キュー制限が 1,000,000 バイトに到達すると、このクラス キューへの待機パケットはドロップされます。</li> </ul>
ステップ5	<p><code>class class-name</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap)# class class2</p>	<p>ポリシーを作成または変更するクラスの名前を指定します。</p> <ul style="list-style-type: none"> <li>この例では、class2 が設定されます。</li> </ul>
ステップ6	<p><code>bandwidth {bandwidth [units]   percent value}</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# bandwidth percent 30</p>	<p>(任意) ポリシー マップに属しているクラスに割り当てる帯域幅を指定します。</p> <ul style="list-style-type: none"> <li>この例では、クラス class2 へのインターフェイス帯域幅の 30% が保証されます。</li> </ul>
ステップ7	<p><code>exit</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap-c)# exit</p>	<p>ルータをポリシー マップ コンフィギュレーション モードに戻します。</p>
ステップ8	<p><code>exit</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-pmap)# exit</p>	<p>ルータをグローバル コンフィギュレーション モードに戻します。</p>
ステップ9	<p><code>interface type interface-path-id</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config)# interface pos 0/2/0/0</p>	<p>コンフィギュレーション モードを開始し、インターフェイスを設定します。</p>
ステップ10	<p><code>service-policy {input   output} policy-map</code></p> <p><b>例:</b> RP/0/RSP0/CPU0:router(config-if)# service-policy output policy1</p>	<p>インターフェイスのサービス ポリシーとして使用する入力インターフェイスまたは出力インターフェイスにポリシー マップを付加します。</p> <ul style="list-style-type: none"> <li>この例では、トラフィック ポリシーでそのインターフェイスから送信されるすべてのトラフィックを評価します。</li> </ul>

コマンドまたはアクション	目的
<p>ステップ 11</p> <pre>end または commit</pre> <p>例:</p> <pre>RP/0/RSP0/CPU0:router(config-cmap)# end または RP/0/RSP0/CPU0:router(config-cmap)# commit</pre>	<p>設定変更を保存します。</p> <ul style="list-style-type: none"> <li>• <b>end</b> コマンドを実行すると、変更をコミットするように要求されます。 Uncommitted changes found, commit them before exiting (yes/no/cancel)? [cancel]: <ul style="list-style-type: none"> <li>– <b>yes</b> と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。</li> <li>– <b>no</b> と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。</li> <li>– <b>cancel</b> と入力すると、現在のコンフィギュレーションセッションが継続します。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。</li> </ul> </li> <li>• 実行コンフィギュレーションファイルに変更を保存し、コンフィギュレーションセッションを継続するには、<b>commit</b> コマンドを使用します。</li> </ul>

## その他の関連資料

ここでは、QoS 輻輳回避の実装に関する関連資料について説明します。

### 関連資料

関連項目	参照先
初期システム起動と設定	『Cisco ASR 9000 Series Aggregation Services Router Getting Started Guide』
マスター コマンド リファレンス	『Cisco ASR 9000 Series Aggregation Services Router Master Command Listing』
QoS コマンド	『Cisco ASR 9000 Series Aggregation Services Router Modular Quality of Service Command Reference』
ユーザ グループとタスク ID	『Cisco ASR 9000 Series Aggregation Services Router System Security Configuration Guide』の「Configuring AAA Services on Cisco ASR 9000 Series Router」モジュール

### 標準

標準	タイトル
この機能によってサポートされる新しい標準または変更された標準はありません。またこの機能による既存標準のサポートに変更はありません。	—

### MIB

MIB	MIB のリンク
—	Cisco IOS XR ソフトウェアを使用して MIB を検索およびダウンロードするには、 <a href="http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml">http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml</a> の URL にある Cisco MIB Locator を使用し、[Cisco Access Products] メニューからプラットフォームを選択します。

### RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

## シスコのテクニカル サポート

説明	リンク
<p>シスコのテクニカル サポート Web サイトには、数千ページに及ぶ検索可能な技術情報があります。製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクもあります。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。</p>	<p><a href="http://www.cisco.com/en/US/support/index.html">http://www.cisco.com/en/US/support/index.html</a></p>

