



Cisco IOS XE 17（Cisco NCS 520 シリーズ）QoS：ポリシーングおよびシェーピング設定ガイド

初版：2019年11月26日

シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター
0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>



目次

第 1 章

クラスベースのポリシング 1

機能情報の確認 1

機能の概要 1

利点 2

機能制限 2

前提条件 3

設定作業 3

トラフィック ポリシングの設定 3

トラフィック ポリシングの確認 3

トラブルシューティングのヒント 4

トラフィック ポリシングのモニタリングと保守 4

設定例 4

例：トラフィックポリシングを含むサービスポリシーの設定 4

その他の参考資料 6

第 2 章

EFP 使用時のポートシェイパーおよび LLQ 9

機能情報の確認 9

EFP 使用時のポートシェイパーおよび LLQ の制約事項 9

EFP 使用時のポートシェイパーおよび LLQ について 10

イーサネットフロー ポイントと LLQ 10

EFP 使用時のポートシェイパーおよび LLQ の設定方法 10

階層型ポリシー マップの設定 10

LLQ ポリシー マップの設定 13

イーサネット フロー ポイントを使用するメイン インターフェイスでのポート レベル シェーピングの設定	15
例：階層型 QoS の設定	17
その他の参考資料	18

第 3 章

優先順位シェイパー	21
優先順位シェイパーの制約事項	22
優先順位シェイパーの設定	22
優先順位シェイパーの設定例	24
例：優先順位シェイパーの設定	24
優先順位シェイパーの確認	25



第 1 章

クラスベースのポリシング

- 機能情報の確認 (1 ページ)
- 機能の概要 (1 ページ)
- 前提条件 (3 ページ)
- 設定作業 (3 ページ)
- トラフィック ポリシングのモニタリングと保守 (4 ページ)
- 設定例 (4 ページ)
- その他の参考資料 (6 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、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、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。

機能の概要

クラスベースポリシングでは、インターフェイスでのトラフィックの最大送受信レートを制御できます。クラスベース ポリシング設定を含むトラフィック ポリシーをインターフェイスに適用すると、クラスベース ポリシング機能が適用されます。

クラスベース ポリシングは、次のように機能します。

- ユーザ定義の基準に基づいて、トラフィックのクラスの入力転送速度を制限します。
- COS、QoS グループ、および DSCP を設定してパケットをマークします。

利点

レート制限による帯域幅管理

クラスベースポリシングでは、インターフェイスでのトラフィックの最大送受信レートを制御できます。クラスベースポリシングは、多くの場合、ネットワークの端のインターフェイスで、ネットワークを出入りするトラフィックを制限するように設定されます。ほとんどのクラスベースポリシング設定では、レートパラメータ内に収まるトラフィックは送信されますが、パラメータを超えるトラフィックはドロップされるか、異なる優先度で送信されます。

パケットのマーキング

パケットのマーキングにより、ネットワークを複数のプライオリティ レベルまたはサービスクラス (CoS) に区分することができます。パケットにマークが付けられると、これらのマーキングを使用して、ダウンストリームデバイスでのトラフィックを識別および分類できます。

- クラスベースポリシングを使用して、ネットワークに入るパケットの IP プレシデンスを設定します。その後、ネットワーク内のネットワークング デバイスは、調整された IP precedence 値を使用してトラフィックの処理方法を決定できます。
- クラスベースポリシングを使用して、パケットを QoS グループに割り当てます。ルータは QoS グループを使用して、ルータ内のパケットに優先順位を付ける方法を決定します。

トラフィックには、クラスベースポリシング機能を使用せずにマークを付けることができます。クラスベースポリシングを使用せずにトラフィックにマークを付けるには、「Marking Network Traffic」モジュールを参照してください。

機能制限

- Cisco NCS 520 ルータでは、クラスベースポリシングを使用して監視できるのはシスコ エクスプレス フォワーディング (CEF) スイッチングパスだけです。クラスベースポリシング機能を使用するには、パケットを受信するインターフェイスとパケットを送信するインターフェイスの両方でシスコ エクスプレス フォワーディングを設定する必要があります。
- Cisco NCS 520 ルータでは、クラスベースポリシングはデバイスから送信されるパケットまたはデバイス宛てに送信されるパケットには適用できません。
- クラスベースポリシングは、インターフェイスまたはサブインターフェイスで設定できます。
- クラスベースポリシングは、物理インターフェイスまたはサービスインスタンスに適用できます。

前提条件

Cisco NCS 520 ルータでは、クラスベースポリシングを使用するには、その前にインターフェイスでシスコ エクスプレス フォワーディング (CEF) を設定する必要があります。

設定作業

トラフィック ポリシングの設定

コマンド	目的
<pre>Device(config-pmap-c)# police bps burst-normal burst-max conform-action action exceed-action action violate-action action</pre>	<p>トラフィック クラスによる最大帯域幅の使用を指定します。</p> <p>(注) クラスベース ポリシング機能は、トークンバケットメカニズムと連動します。現在、トークンバケットアルゴリズムには、シングルトークンバケットアルゴリズムとツー トークンバケットアルゴリズムの2種類があります。シングルトークンバケットシステムは、violate-action オプションが指定されない場合に使用されます。ツー トークンバケットシステムは、violate-action オプションが指定される場合に使用されます。</p>

トラフィック ポリシングの確認

クラスベースポリシング機能がインターフェイスで設定されていることを確認するには、**show policy-map interface EXEC** コマンドを使用します。この機能がインターフェイスで設定されている場合、**show policy-map interface** コマンド出力にポリシング統計情報が表示されます。

```
Device# show policy-map interface
Ethernet1/7
  service-policy output: x
    class-map: a (match-all)
      0 packets, 0 bytes
      5 minute rate 0 bps
      match: ip precedence 0
      police:
        1000000 bps, 10000 limit, 10000 extended limit
        conformed 0 packets, 0 bytes; action: transmit
        exceeded 0 packets, 0 bytes; action: drop
        conformed 0 bps, exceed 0 bps, violate 0 bps
```

トラブルシューティングのヒント

- インターフェイスタイプをチェックします。インターフェイスが、このモジュールの[機能制限 \(2 ページ\)](#) のサポートされていないインターフェイスの説明に記載されていないことを確認してください。
- Cisco NCS 520 シリーズ ルータでの入力クラスベースポリシングの場合は、クラスベースポリシングが設定されているインターフェイスで CEF が設定されていることを確認してください。
- Cisco NCS 520 シリーズ ルータでの出力クラスベースポリシングの場合は、着信トラフィックが CEF 交換であることを確認してください。クラスベース ポリシングは、CEF スイッチングがイネーブルになっていない限り、スイッチング パスで使用できません。

トラフィック ポリシングのモニタリングと保守

コマンド	目的
Device# show policy-map	設定されたすべてのポリシー マップを表示します。
Device# show policy-map <i>policy-map-name</i>	ユーザ指定ポリシー マップを表示します。
Device# show policy-map interface	インターフェイスに適用されたすべての入力および出力ポリシーの統計情報および設定を表示します。
Device# show policy-map interface service instance	ポートチャネルにおける特定のサービスインスタンスに関するポリシー マップ情報を表示します。

設定例

例：トラフィックポリシングを含むサービスポリシーの設定

次の例では、ファストイーサネットインターフェイス 0/0 から出るすべてのパケットに関して、平均レートを 8000 ビット/秒、ノーマルバーストサイズを 1000 バイト、超過バーストサイズを 1000 バイトに指定したクラスベースポリシングを設定します。

```
class-map access-match
match access-group 1
exit
policy-map police-setting
class access-match
police 8000 1000 1000 conform-action transmit exceed-action set-qos-transmit 1
violate-action drop
exit
```



```
exit
service-policy input police-setting
```

ファストイーサネットインターフェイス 0/0 から出る一連のパケットの処理方法は、パケットのサイズ、および準拠トークンバケットと超過トークンバケットに残っているバイト数に応じて異なります。一連のパケットは、次のルールに基づいてポリシングされます。

- 前のパケットが T1 に到達し、現在のパケットが T に到達した場合、バケットはトークン到達レートに基づいて T-T1 に相当するビット数で更新されます。リフィルトークンは、準拠バケットに置かれます。トークンが準拠バケットでオーバーフローになると、超過バケットにオーバーフロートークンが置かれます。トークンの到達レートは次のように計算されます。

(パケット間の時間 (= T - T1) X ポリシング レート) / 8 バイト

- 準拠バケット B のバイト数が 0 以上の場合、パケットは準拠し、パケットで準拠処理が実行されます。パケットが準拠している場合、B バイトが準拠バケットから削除され、準拠処理が実行されます。このシナリオでは、超過バケットには影響ありません。
- 準拠バケット B のバイト数が 0 未満の場合、超過トークンバケットでパケットによるバイトがチェックされます。超過バケット B のバイト数が 0 以上の場合、超過処理が実行され、超過トークンバケットから B バイトが削除されます。このシナリオでは、準拠バケットからバイトは削除されません。
- 超過バケット B のバイト数が 0 未満の場合、パケットはレートに違反しているため、違反処理が実行されます。パケットに対する処理が完了します。

この例では、初期トークンバケットはフルの 1000 バイトで開始します。450 バイトのパケットを受信すると、準拠トークンバケットに使用可能なバイトが十分あるため、パケットは準拠しています。パケットにより準拠処理（送信）が実行され、450 バイトが準拠トークンバケットから削除されます（残り 550 バイト）。

次のバケットが 0.25 秒後に到達し、準拠トークンバケットに 250 バイト追加された場合

$((0.25 \times 8000)/8)$ 、準拠トークンバケットには 800 バイト残ります。次のパケットが 900 バイトの場合、準拠トークンバケットでは 800 バイトしか使用できないため、パケットは準拠していません。

フルの 1000 バイトで始まる超過トークンバケット（超過バーストサイズで指定）に使用可能なバイトがあるかどうかチェックされます。超過トークンバケットには使用可能なバイトが十分あるため、超過処理（QoS 送信値を 1 に設定）が実行され、超過バケットから 900 バイトが取られ、超過トークンバケットの残りは 100 バイトになります。

次のパケットが 0.40 秒後に到達し、トークンバケットに 400 バイトが追加されます ($(.40 \times 8000)/8$)。これで、準拠トークンバケットには 1000 バイトあり（準拠バケットで使用可能な最大トークン数）、200 バイトが準拠トークンバケットをオーバーフローします（準拠トークンバケットの容量を満たすために必要なのは 200 バイトだけのため）。これらのオーバーフローバイトは、超過トークンバケットに置かれ、超過トークンバケットに 300 バイト与えられます。

着信パケットが 1000 バイトの場合、準拠トークンバケットで使用可能なバイト数が十分あるため、パケットは準拠します。パケットにより準拠処理（送信）が実行され、1000 バイトが準拠トークンバケットから削除されます（残り 0 バイト）。

次のパケットが 0.20 秒後に到達し、トークンバケットに 200 バイトが追加されます（ $(.20 \times 8000)/8$ ）。これで、準拠バケットの中身は 200 バイトになります。着信パケットが 400 バイトの場合、準拠トークンバケットでは 200 バイトしか使用できないため、パケットは準拠していません。同様に、超過バケットで使用可能なバイト数は 300 バイトだけなので、パケットは超過しません。したがって、パケットは違反となり、違反処理（ドロップ）が実行されます。

その他の参考資料

関連資料

関連項目	マニュアルタイトル
QoS コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『Cisco IOS Quality of Service Solutions Command Reference』
モジュラ Quality of Service (QoS) コマンドラインインターフェイス (CLI) (MQC)、階層型ポリシー、ポリシーマップ	「Applying QoS Features Using the MQC」モジュール
トラフィックのポリシングとシェーピング	「Policing and Shaping Overview」モジュール

標準

標準	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 IOS XE ソフトウェアリリース、およびフィチャーセットの MIB の場所を検索しダウンロードするには、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
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>★枠で囲まれた Technical Assistance の場合★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。</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>



第 2 章

EFP 使用時のポートシェイパーおよび LLQ

EFP 使用時のポートシェイパーおよび LLQ の機能を使用して、ネットワーク設計者は、イーサネットフローポイント (EFP) を含むポートでポートポリシーおよびクラスポリシーを設定できます。これらのポリシーは、EFP間での低遅延キューイング (LLQ) とトラフィックの優先順位付けをサポートしています。

- [機能情報の確認 \(9 ページ\)](#)
- [EFP 使用時のポートシェイパーおよび LLQ の制約事項 \(9 ページ\)](#)
- [EFP 使用時のポートシェイパーおよび LLQ について \(10 ページ\)](#)
- [EFP 使用時のポートシェイパーおよび LLQ の設定方法 \(10 ページ\)](#)
- [その他の参考資料 \(18 ページ\)](#)

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、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリースノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、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。

EFP 使用時のポートシェイパーおよび LLQ の制約事項

- EFP が存在するポートシェイパーでは、ポートでのポリシーを最初に適用してから、EFP でのポリシーを適用する必要があります。最初に EFP にポリシーを適用してからポートにポリシーを適用すると、機能しません。
- ポート上でクラスベースポリシーを設定した場合、イーサネットフローポイント (EFP) でサービスポリシーが設定できません。

- BDI へのサービス ポリシーの適用はサポートされていません。
- 優先順位クラスが存在しない場合の帯域幅余剰パーセンテージ (BRP) の使用では、使用可能な帯域幅が反復的な方法で割り当てられます。たとえば、それぞれのクラスマップで設定された共有の割合に従って最初の BRP クラスに帯域幅が割り当てられ、残りの帯域幅は、帯域幅が使い果たされるまで、他のすべての BRP クラスに対して繰り返し割り当てられます。
- ポートがダウンしている場合、手動でポート速度を変更すると、予期しない動作が発生する可能性があります。
- 帯域幅のステートメントは、親ポリシーから削除する前に、子ポリシーから削除する必要があります。

EFP 使用時のポートシェイパーおよび LLQ について

イーサネット フローポイントと LLQ

イーサネットフローポイント (EFP) は、プロバイダーエッジ (PE) ルータにある転送判断ポイントであり、インターフェイス内の多数のレイヤ2のフロー判断に関する自由度をネットワーク設計者に提供します。1つの物理ポートに複数の EFP が設定できます (設定数は1デバイスからそれ以上までさまざまです)。EFP は、インターフェイス上の Ethernet Virtual Connection (EVC : イーサネット仮想コネクション) の論理境界点です。複数のユーザネットワークインターフェイス (UNI) を使用する EVC では、EVC が経路するすべてのデバイスの関連する入出力インターフェイスに EFP が必要です。

ネットワーク設計者は、ポート レベルシェーピング機能を含む出力 HQoS を使用して、EFP を含むポートのポート ポリシーおよびクラス ポリシーを設定することができます。これらのポリシーは、EFP 間での低遅延キューイング (LLQ) とトラフィックの優先順位付けをサポートしています。

LLQ を設定する方法については、*QoS 輻輳管理の構成ガイド*を参照してください。

EFP 使用時のポートシェイパーおよび LLQ の設定方法

EFP 機能を使用する場合にポートシェイパーおよび LLQ を設定するには、まず低遅延キューイング (LLQ) をサポートする階層型ポリシーマップまたは階層のないポリシーマップのいずれかを作成し、EFP インターフェイスに適用します。

階層型ポリシー マップの設定

階層型ポリシーマップを設定するには、親ポリシーに適用する子ポリシーを作成します。その後、親ポリシーはインターフェイスに適用されます。

ステップ 1 enable

例 :

```
Device> enable
```

特権 EXEC モードを有効にします。

- 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 policy-map *policy-map-name*

例 :

```
Device(config)# policy-map child-llq
```

子ポリシーを作成または変更して、QoS 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。

- `child-llq` は子ポリシー マップの名前です。

ステップ 4 class *class-map-name*

例 :

```
Device(config-pmap)# class qos-1
```

ポリシー マップに対して指定されたトラフィック クラスを割り当て、QoS 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。

- `qos-1` は、設定済みのクラスマップの名前です。これは QoS アクションを定義する対象となるトラフィッククラスです。

ステップ 5 bandwidth percent *percent*

例 :

```
Device(config-pmap-c)# bandwidth percent 20
```

(オプション) クラスレベルのキューの帯域幅パーセンテージを指定します。これは、プライオリティトラフィックによって使用されていない余分な帯域幅を判断し、非プライオリティ キューに割り当てるために輻輳時に使用します。

ステップ 6 exit

例 :

```
Device(config-pmap-c)# exit
```

QoS ポリシーマップ クラス コンフィギュレーション モードを終了します。

ステップ 7 **class class-map-name**

例 :

```
Device(config-pmap)# class qos-2
```

ポリシー マップに対して指定されたトラフィック クラスを割り当て、QoS 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。

- qos-2 は、設定済みのクラスマップの名前です。これは QoS アクションを定義する対象となるトラフィッククラスです。

ステップ 8 **bandwidth percent percent**

例 :

```
Device(config-pmap-c)# bandwidth percent 80
```

(オプション) クラスレベルのキューの帯域幅パーセンテージを指定します。これは、プライオリティトラフィックによって使用されていない余分な帯域幅を判断し、非プライオリティ キューに割り当てるために輻輳時に使用します。

ステップ 9 **exit**

例 :

```
Device(config-pmap-c)# exit
```

QoS ポリシーマップ クラス コンフィギュレーション モードを終了します。

ステップ 10 **policy-map policy-map-name**

例 :

```
Device(config-pmap)# policy-map parent-llq
```

親ポリシーを作成または変更します。

- parent-llq は親ポリシー マップの名前です。

ステップ 11 **class class-default**

例 :

```
Device(config-pmap)# class class-default
```

親 class-default クラスを設定または変更し、QoS 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。

- 親ポリシーには class-default クラスのみ設定できます。他のトラフィック クラスは設定しないでください。

ステップ 12 `service-policy policy-map-name`

例 :

```
Device(config-pmap-c)# service-policy child-llq
```

親の `class-default` クラスに子ポリシーを適用します。

- `child-llq` は、ステップ 1 で設定した子ポリシー マップの名前です。

LLQ ポリシー マップの設定

ステップ 1 `enable`

例 :

```
Device> enable
```

特権 EXEC モードを有効にします。

- 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 `policy-map policy-map-name`

例 :

```
Device(config)# policy-map llq-flat
```

ポリシーを作成し、QoS 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。

ステップ 4 `class class-map-name`

例 :

```
Device(config-pmap)# class qos-group
```

ポリシー マップに対して指定されたトラフィック クラスを割り当て、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。

ステップ 5 `priority`

例 :

```
Device(config-pmap-c)# priority
```

クラスベース重み付け均等化キューイング (CBWFQ) に絶対優先キューイング (PQ) を指定し、LLQ を設定します。

ステップ 6 **exit**

例 :

```
Device(config-pmap-c)# exit
```

QoS ポリシーマップ クラス コンフィギュレーション モードを終了します。

ステップ 7 **class class-map-name**

例 :

```
Device(config-pmap)# class qos-group
```

ポリシー マップに対して指定されたトラフィック クラスを割り当て、QoS 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。

ステップ 8 **shape average value**

例 :

```
Device(config-pmap-c)# shape average 200000000
```

認定情報レートに 200 Mb/s を指定して、シェイプ エンティティを設定します。

ステップ 9 **exit**

例 :

```
Device(config-pmap-c)# exit
```

QoS ポリシーマップ クラス コンフィギュレーション モードを終了します。

ステップ 10 **class class-map-name**

例 :

```
Device(config-pmap)# class qos-group
```

ポリシー マップに対して指定されたトラフィック クラスを割り当て、QoS 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。

ステップ 11 **bandwidth percent**

例 :

```
Device(config-pmap-c)# bandwidth 4000000
```

(オプション) クラスレベルのキューの帯域幅パーセンテージを指定します。これは、プライオリティトラフィックによって使用されていない余分な帯域幅を判断し、非プライオリティ キューに割り当てるために輻輳時に使用します。

ステップ 12 **exit**

例 :

```
Device(config-pmap-c)# exit
```

QoS ポリシーマップ クラス コンフィギュレーション モードを終了します。

イーサネット フロー ポイントを使用するメインインターフェイスでのポート レベル シェーピングの設定

EFPS を使用するメインインターフェイスにポート レベル シェーピングを設定するには、まずインターフェイスで自動ネゴシエーションプロトコルをイネーブル化し、インターフェイスにポリシー マップを適用して、最後にイーサネット サービス インスタンスを設定します。

ステップ 1 enable

例 :

```
Device> enable
```

特権 EXEC モードを有効にします。

- 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 interface *type number*

例 :

```
Device(config)# interface GigabitEthernet 0/0/1
```

インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。

- インターフェイスのタイプ番号を入力します。

ステップ 4 no ip address

例 :

```
Device(config-if)# no ip address
```

インターフェイス上で IP ルーティングをディセーブル化します。

ステップ 5 negotiation auto

イーサネットフローポイントを使用するメインインターフェイスでのポートレベルシェーピングの設定

例：

```
Device(config-if)# negotiation auto
```

ギガビットイーサネットインターフェイスの速度、デュプレックス、および自動フロー制御を自動ネゴシエーションプロトコルで設定できるようにします。

ステップ 6 **service-policy output *policy-map-name***

例：

```
Device(config-if)# service-policy output parent-llq
```

インターフェイスの入力方向または出力方向にアタッチするポリシー マップの名前を指定します。

- 階層型ポリシー マップまたは階層のないポリシー マップの名前を入力できます。

ステップ 7 **service instance *id* ethernet**

例：

```
Device(config-if)# service instance 1 ethernet
```

インターフェイス上でイーサネット サービス インスタンスを設定し、サービス インスタンス コンフィギュレーション モードを開始します。

ステップ 8 **encapsulation dot1q *vlan-id***

例：

```
Device(config-if-srv)# encapsulation dot1q 100
```

インターフェイスの 802.1Q フレーム入力をサービス インスタンスにマッピングするための一致基準を定義します。

ステップ 9 **bridge-domain *bridge-domain-id***

例：

```
Device(config-if-srv)# bridge-domain 100
```

ブリッジ ドメインをサービス インスタンスにバインドします。

ステップ 10 **exit**

例：

```
Device(config-if-srv)# exit
```

サービス インスタンス コンフィギュレーション モードを終了します。

ステップ 11 **service instance *id* ethernet**

例：

```
Device(config-if)# service instance 2 ethernet
```

インターフェイス上でイーサネット サービス インスタンスを設定し、サービス インスタンス コンフィギュレーション モードを開始します。

ステップ 12 **encapsulation dot1q vlan-id**

例：

```
Device(config-if-srv)# encapsulation dot1q 101
```

インターフェイスの 802.1Q フレーム入力をサービス インスタンスにマッピングするための一致基準を定義します。

ステップ 13 **bridge-domain bridge-domain-id**

例：

```
Device(config-if-srv)# bridge-domain 101
```

ブリッジ ドメインをサービス インスタンスにバインドします。

ステップ 14 **exit**

例：

```
Device(config-if-srv)# exit
```

QoS ポリシーマップ クラス コンフィギュレーション モードを終了します。

ステップ 15 **end**

例：

```
Device(config-if)# end
```

(任意) インターフェイス コンフィギュレーション モードを終了します。

例：階層型 QoS の設定

ルータは、最大 3 つのレベルの階層型 QoS ポリシーをサポートしています。

- ポートポリシー：class-default のみを使用し、物理インターフェイスに初期設定として適用する必要があります。
- EFP ポリシー：class-default で親ポリシーが一致し、QoS-group 値で子ポリシーが一致する、ネストされたポリシーを使用します。

次の例では、ポートポリシーおよび EFP ポリシーを使用した階層型 QoS 設定を示します。

```
policy-map parent1  
class class-default
```

```

shape average 1000000000
!
policy-map EFP30
class class-default
shape average 500000000
service-policy pq_test
!
end

```

設定を確認するには、**show run policy-map** を使用します。

```

router#sh run policy-map pq_test
Building configuration...

Current configuration : 590 bytes
!
policy-map pq_test
class qg1
priority level 1 percent 20
queue-limit percent 10
class qg0
priority level 2 percent 10
queue-limit percent 10
class qg2
bandwidth remaining percent 15
queue-limit percent 10
class qg3
bandwidth remaining percent 20
queue-limit percent 10
class qg4
bandwidth remaining percent 25
queue-limit percent 10
class qg5
bandwidth remaining percent 10
queue-limit percent 10
class qg6
bandwidth remaining percent 25
queue-limit percent 10
class class-default
bandwidth remaining percent 5
queue-limit percent 10
!
end

```

その他の参考資料

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 : コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『 Cisco IOS QoS Command Reference 』

関連項目	マニュアルタイトル
ポリシングとシェーピング	「Policing and Shaping Overview」モジュール
クラス マップ	「Applying QoS Features Using the MQC」モジュール
ポリシー マップ	「Applying QoS Features Using the MQC」モジュール
低遅延キューイング	QoS 輻輳管理の構成ガイド

標準および RFC

標準	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	--

MIB

MIB	MIB のリンク
新しい MIB または変更された MIB はサポートされていません。また、既存の MIB に対するサポートに変更はありません。	<p>選択したプラットフォーム、Cisco IOS XE ソフトウェア リリース、およびフィーチャセットの MIB の場所を検索しダウンロードするには、次の URL にある Cisco MIB Locator を使用します。</p> <p>http://www.cisco.com/go/mibs</p>

シスコのテクニカル サポート

説明	リンク
<p>★枠で囲まれた Technical Assistance の場合★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。</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>



第 3 章

優先順位シェイパー

以前は、ホップ単位動作（PHB）でのキューの優先順位がチャンネルレベルまでのすべての階層に伝播された場合、PHB レベルで優先順位が設定されている PHB クラスのみが、他のサブチャンネルのクラスよりも優先されていました。これを回避するために、優先順位シェイパー機能が実装されました。

優先順位シェイパー機能により、複数のストリームがプライオリティキューから出力される場合に、ストリーム間のパケットドロップのバランスをとることができます。優先順位シェイパーでは、出力 QoS ポリシーがサポートされています。

優先順位シェイパーは、次のいずれかの方法で設定できます。

- 絶対優先：ポリシーごとに1つのシェイパーを適用できます。絶対優先を設定するには、**priorityshaper-value** コマンドを使用します。
- 優先順位レベル：ポリシーごとに2つの優先順位レベルを適用できます。

次の **show run policy-map** コマンドで、優先順位レベルを使用してシェイパーを設定する方法を確認できます。

```
router#show run policy-map pq_test
Building configuration...

Current configuration : 590 bytes
!
policy-map pq_test
class qg1
priority level 1 percent 20
queue-limit percent 10
class qg0
priority level 2 percent 10
queue-limit percent 10
class qg2
bandwidth remaining percent 15
queue-limit percent 10
class qg3
bandwidth remaining percent 20
queue-limit percent 10
class qg4
bandwidth remaining percent 25
queue-limit percent 10
class qg5
```

```
bandwidth remaining percent 10
queue-limit percent 10
class qg6
bandwidth remaining percent 25
queue-limit percent 10
class class-default
bandwidth remaining percent 5
queue-limit percent 10
!
end
```

- [優先順位シェイパーの制約事項 \(22 ページ\)](#)
- [優先順位シェイパーの設定 \(22 ページ\)](#)

優先順位シェイパーの制約事項

- 優先順位シェイパーは、PHB レベルのクラスに対してのみサポートされます。
- 優先順位シェイパーを持つ出力 QoS ポリシーマップは、論理レベルではなく、ポートチャネルのメンバーインターフェイスにのみ適用できます。
- ポリサー設定は、同じクラスマップ下の優先順位シェイパー設定ではサポートされていません。
- Q レベルでの優先順位シェイパー設定では、輻輳中に優先順位トラフィックの遅延が増加します。優先順位トラフィックの遅延を減らすには、より小さなプライオリティキューの値を使用してキュー制限を設定します。
- パケットが 10G インターフェイスから 1G インターフェイスに向かう場合は、バーストが発生しやすくなります。このため、この絶対優先キューのキュー解除レートは、エンキューを上回る場合があります。その結果、他のキューで表示されるパケットカウンタが非常に少なくなります。

優先順位シェイパーの設定

優先順位シェイパーを設定するには、次の手順を実行します。

ステップ 1 enable

例 :

```
Device> enable
```

特権 EXEC モードを有効にします。

- 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 **class-map** *class-map-name*

例 :

```
Device(config)#class-map class_priority
```

クラスマップを設定し、作成するクラスマップの名前を指定します。

ステップ 4 **match qos-group** *number*

例 :

```
Device(config-cmap)# match qos-group 1
```

異なるクラスマップに対して異なる PHB が一致しました。

ステップ 5 **policy-map** *policy-map-name*

例 :

```
Device(config)#policy-map shape_priority
```

ポリシー マップを設定します。

ステップ 6 **class** *class-map-name*

例 :

```
Device(config-pmap)#class class_priority
```

作成するポリシーのクラス名を指定し、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。このクラスが、前に作成したクラスマップと関連付けられます。

ステップ 7 **priority level** <level 1/2> **percent** <percentage 1-100> or **priority level** <level 1/2> <kbps> <burst size>

例 :

```
Device(config-pmap-c)# priority <1-10000000> Kilo Bits per second
```

```
Device(config-pmap-c)# priority Percent <1-100>
```

```
Device(config-pmap-c)# priority level <1-2> <1-10000000> Kilo Bits per second
```

```
Device(config-pmap-c)# priority level <1-2> percent <1-100>
```

指定されたプライオリティ レベルでトラフィック クラスにプライオリティを割り当てます。

(注) **level** は、優先順位クラスに割り当てられた優先順位のレベルです。有効な値は、1 (高優先順位) または 2 (低優先順位) です。デフォルト値は 1 です。同じポリシー マップ内の異なる 2 つのクラスに同じプライオリティ レベルを指定しないでください。

ステップ 8 **interface** *interface-type interface-number*

例 :

```
Device(config)# interface gigabitethernet 0/0/1
```

ポリシーマップを適用するポートを指定し、インターフェイス コンフィギュレーション モードの開始を許可します。指定できるインターフェイスは、物理ポートです。

ステップ 9 **service-policy output** *policy-map-name*

例：

```
Device(config-if)# service instance 1 ethernet
Device(config-if-srv)# service-policy output shape_priority
```

インターフェイスに出力ポリシーを適用します。

(注) サービスインスタンスにサービスポリシーを適用することもできます。

ステップ10 end

例：

```
Device(config)#end
```

特権 EXEC モードに戻ります。

優先順位シェイパーの設定例

この項では、優先順位シェイパーの設定例を示します。

例：優先順位シェイパーの設定

次に、優先順位シェイパーの設定例を示します。

```
Device(config)#class-map match-any class_level1
Device(config-cmap)#match qos-group 1
Device(config-cmap)#match qos-group 2
Device(config-cmap)#class-map match-any class_level2
Device(config-cmap)#match qos-group 3
Device(config-cmap)#match qos-group 4
Device(config-cmap)#class-map match-any class_bw
Device(config-cmap)#match qos-group 5
Device(config-cmap)#end

.
Device(config)#policy-map shape_priority
Device(config-pmap)#class class_level1
Device(config-pmap-c)#priority level 1 per 10
Device(config-pmap-c)#class class_level2
Device(config-pmap-c)#priority level 2 per 20
Device(config-pmap-c)#class class_bw
Device(config-pmap-c)#bandwidth remaining percent 70
Device(config-pmap-c)#end

Device(config)#interface GigabitEthernet0/0/3
Device(config-if)#load-interval 30
Device(config-if)#service-policy input shape_priority
Device(config-if)#end
```

優先順位シェイパーの確認

優先順位シェイパー機能がインターフェイスで設定されていることを確認するには、次のコマンドを使用します。

```
Device# show policy-map interface TenGigabitEthernet0/0/2
show policy-map interface TenGigabitEthernet0/0/2
TenGigabitEthernet0/0/2

Service-policy output: shape_priority

queue stats for all priority classes:
Queueing
priority level 1
queue limit 3932 us/ 49152 bytes
(queue depth/total drops/no-buffer drops) 49476/44577300/0
(pkts output/bytes output) 2348138/1202246656

queue stats for all priority classes:
Queueing
priority level 2
queue limit 1966 us/ 49152 bytes
(queue depth/total drops/no-buffer drops) 51072/42228358/0
(pkts output/bytes output) 4697080/2404904960

Class-map: class_priority (match-any)
46925438 packets, 24025824256 bytes
30 second offered rate 1871849000 bps, drop rate 1778171000 bps
Match: qos-group 1
Match: qos-group 2
Priority: 10% (100000 kbps), burst bytes 2500000, b/w exceed drops: 44577300

Priority Level: 1

Class-map: class_priority_level2 (match-any)
46925438 packets, 24025824256 bytes
30 second offered rate 1871849000 bps, drop rate 1684485000 bps
Match: qos-group 3
Match: qos-group 4
Priority: 20% (200000 kbps), burst bytes 5000000, b/w exceed drops: 42228358

Priority Level: 2

Class-map: class_bw (match-any)
23462719 packets, 12012912128 bytes
30 second offered rate 935925000 bps, drop rate 281045000 bps
Match: qos-group 5
Queueing
queue limit 393 us/ 49152 bytes
(queue depth/total drops/no-buffer drops) 49476/7045085/0
(pkts output/bytes output) 16417634/8405828608
bandwidth remaining 70%

Class-map: class-default (match-any)
0 packets, 0 bytes
30 second offered rate 0000 bps, drop rate 0000 bps
Match: any

queue limit 393 us/ 49152 bytes
(queue depth/total drops/no-buffer drops) 0/0/0
(pkts output/bytes output) 0/0
```

