目 录

1 QoS 概述

QoS 即服务质量。对于网络业务，影响服务质量的因素包括传输的带宽、传送的时延、数据的丢包率等。在网络中可以通过保证传输的带宽、降低传送的时延、降低数据的丢包率以及时延抖动等措施来提高服务质量。网络资源总是有限的，在保证某类业务的服务质量的同时，可能就是在损害其它业务的服务质量。因此，网络管理者需要根据各种业务的特点来对网络资源进行合理的规划和分配，从而使网络资源得到高效利用。

1.1 QoS服务模型简介

通常 QoS 提供以下三种服务模型：

- Best-Effort service（尽力而为服务模型）
- Integrated service（综合服务模型，简称 IntServ）
- Differentiated service（区分服务模型，简称 DiffServ）

1.1.1 Best-Effort 服务模型

Best-Effort 是一个单一的服务模型，也是最简单的服务模型。对 Best-Effort 服务模型，网络尽最大的可能性来发送报文。但对时延、可靠性等性能不提供任何保证。

Best-Effort 服务模型是网络的缺省服务模型，通过 FIFO 队列来实现。它适用于绝大多数网络应用，如 FTP、E-Mail 等。

1.1.2 IntServ 服务模型

IntServ 是一个综合服务模型，它可以满足多种 QoS 需求。该模型使用 RSVP 协议，RSVP 运行在从源端到目的端的每个设备上，可以监视每个流，以防止其消耗资源过多。这种体系能够明确区分并保证每一个业务流的服务质量，为网络提供最细粒度化的服务质量区分。

但是，IntServ 模型对设备的要求很高，当网络中的数据流数量很大时，设备的存储和处理能力会遇到很大的压力。IntServ 模型可扩展性很差，难以在 Internet 核心网络实施。

1.1.3 DiffServ 服务模型

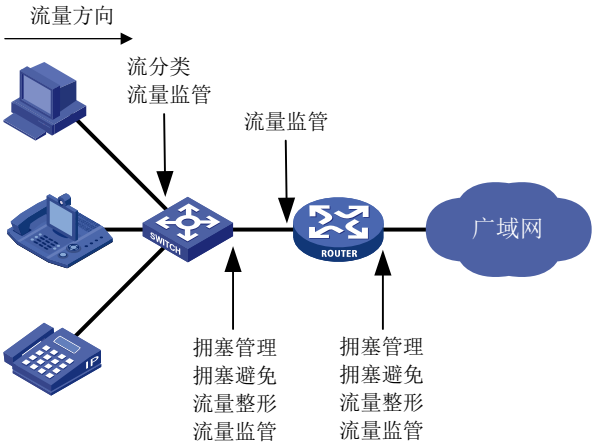
DiffServ 是一个多服务模型，它可以满足不同的 QoS 需求。与 IntServ 不同，它不需要通知网络为每个业务预留资源。区分服务实现简单，扩展性较好。

本文提到的技术都是基于 DiffServ 服务模型。

1.2 QoS技术在网络中的位置

QoS 技术包括流分类、流量监管、流量整形、限速、拥塞管理、拥塞避免等。下面对常用的技术进行简单地介绍。

图1-1 常用 QoS 技术在网络中的位置



如图 1-1 所示，流分类、流量监管、流量整形、拥塞管理和拥塞避免主要完成如下功能：

- 流分类：采用一定的规则识别符合某类特征的报文，它是对网络业务进行区分服务的前提和基础。
- 流量监管：对进入或流出设备的特定流量进行监管，以保护网络资源不受损害。可以作用在接口入方向和出方向。
- 流量整形：一种主动调整流的输出速率的流量控制措施，用来使流量适配下游设备可供给的网络资源，避免不必要的报文丢弃，通常作用在接口出方向。
- 拥塞管理：当拥塞发生时制定一个资源的调度策略，决定报文转发的处理次序，通常作用在接口出方向。
- 拥塞避免：监督网络资源的使用情况，当发现拥塞有加剧的趋势时采取主动丢弃报文的策略，通过调整队列长度来解除网络的过载，通常作用在接口出方向。

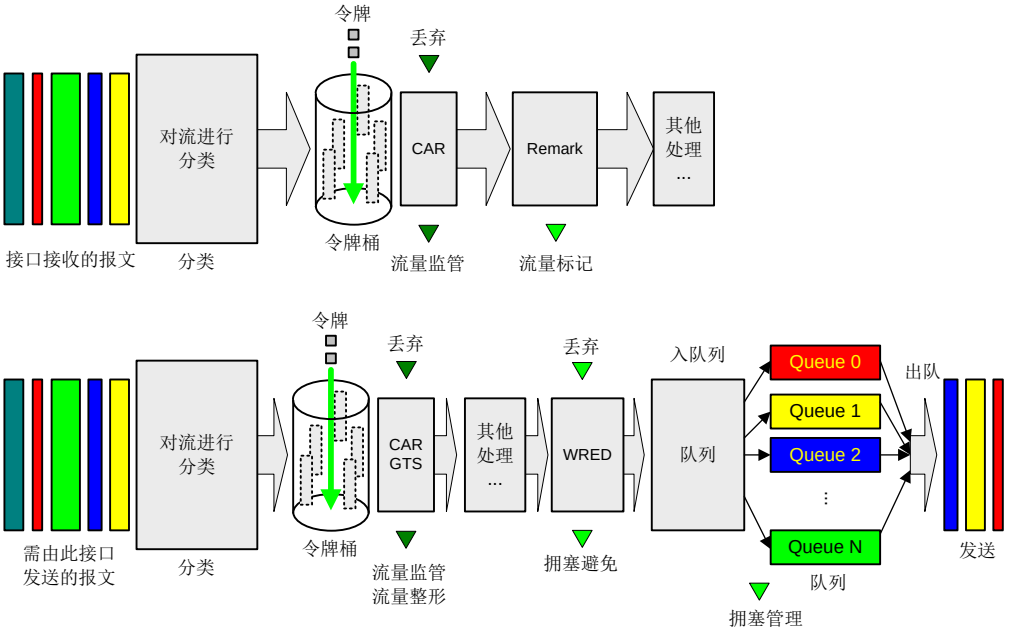
1.3 QoS技术在设备中的处理顺序

图 1-2 简要描述了各种 QoS 技术在网络设备中的处理顺序。

(1) 首先通过流分类对各种业务进行识别和区分，它是后续各种动作的基础；

(2) 通过各种动作对特定的业务进行处理。这些动作需要和流分类关联起来才有意义。具体采取何种动作，与所处的阶段以及网络当前的负载状况有关。例如，当报文进入网络时进行流量监管；流出节点之前进行流量整形；拥塞时对队列进行拥塞管理；拥塞加剧时采取拥塞避免措施等。

图1-2 各 QoS 技术在同一网络设备中的处理顺序



1.4 QoS配置方式

QoS 的配置方式分为 MQC 方式（模块化 QoS 配置，Modular QoS Configuration）和非 MQC 方式。

MQC 方式通过 QoS 策略定义不同类别的流量要采取的动作，并将 QoS 策略应用到不同的目标位置（例如接口）来实现对业务流量的控制。

非 MQC 方式则通过直接在目标位置上配置 QoS 参数来实现对业务流量的控制。例如，在接口上配置限速功能来达到限制接口流量的目的。

有些 QoS 功能只能使用其中一种方式来配置，有些使用两种方式都可以进行配置。在实际应用中，两种配置方式也可以结合起来使用。

2 QoS 策略

2.1 QoS策略简介

QoS 策略由如下部分组成：

- 类，定义了对报文进行识别的规则。
- 流行为，定义了一组针对类识别后的报文所做的 QoS 动作。

通过将类和流行为关联起来，QoS 策略可对符合分类规则的报文执行流行为中定义的动作。

用户可以在一个策略中定义多个类与流行为的绑定关系。

2.2 QoS策略配置任务简介

QoS 策略配置任务如下：

(1) 定义类

(2) 定义流行为

(3) 定义策略

(4) （可选）配置策略嵌套

(5) 应用策略

- 基于接口应用 QoS 策略
- 基于 PVC 应用 QoS 策略
- 基于 PW 应用 QoS 策略
- 基于全局应用 QoS 策略
- 基于控制平面应用 QoS 策略
- 基于管理口控制平面应用 QoS 策略
- 基于上线用户应用 QoS 策略

(6) （可选）配置接口流速统计时间

2.3 定义类

(1) 进入系统视图。

system-view

(2) 创建类，并进入类视图。

traffic classifier *classifier-name* [**operator** { **and** | **or** }]

(3) 定义匹配数据包的规则。

if-match [**not**] *match-criteria*

缺省情况下，未定义匹配数据包的规则。

具体规则的介绍，请参见“QoS 命令”中的 **if-match** 命令。

2.4 定义流行为

1. 配置限制和指导

当一个流行为中配置了多个动作时，如果其中某个动作不生效，则整个流行为都不会生效。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建流行为，并进入流行为视图。

traffic behavior *behavior-name*

(3) 配置流行为的动作。

缺省情况下，未配置流行为的动作。

流行为动作就是对符合流分类的报文做出相应的 QoS 动作，例如流量监管、流量过滤、重标记、流量统计等，具体情况请参见本文相关章节。

2.5 定义策略

(1) 进入系统视图。

system-view

(2) 创建 QoS 策略，并进入策略视图。

qos policy *policy-name*

(3) 为类指定流行为。

classifier *classifier-name* **behavior** *behavior-name* [**insert-before** *before-classifier-name*]

缺省情况下，未指定类对应的流行为。

2.6 配置策略嵌套

1. 功能简介

QoS 策略分为两种：父策略和子策略，其中父策略即为普通的 QoS 策略。通过在父策略流行为视图下创建一个新的策略，即创建子策略，可以实现策略嵌套功能。

配置策略嵌套后，**traffic classifier** 命令定义的某一类流量，除了执行父策略中定义的流行为外，还会由子策略再次对该类流量进行分类，并执行子策略中定义的流行为。

2. 配置限制和指导

配置策略嵌套时需要注意的是：

- 如果子策略中配置了 CBQ，那么父策略中必须配置 GTS，并且配置的父策略 GTS 带宽必须大于等于子策略 CBQ 带宽，否则配置失败。
- 如果父策略的 GTS 配置采用百分比形式，则子策略 CBQ 带宽配置必须采用百分比形式，不允许采用绝对值形式。
- 如果父策略的 GTS 配置采用绝对值形式，则子策略 CBQ 带宽配置既可以采用百分比形式，也可以采用绝对值形式。

- 子策略中不允许配置 GTS。

3. 配置准备

配置策略嵌套时，请先定义子策略。关于定义子策略配置，请参见“2.5 定义策略”。

4. 配置步骤

(1) 进入系统视图。

system-view

(2) 定义父策略的类。

a. 创建父策略的类，并进入父策略的类视图。

traffic classifier *classifier-name* [**operator** { **and** | **or** }]

b. 定义父策略匹配数据包的规则。

if-match [**not**] *match-criteria*

缺省情况下，未定义匹配数据包的规则。

具体规则的配置介绍，请参见 QoS 命令中的 **if-match** 命令。

c. 退回系统视图。

quit

(3) 在父策略流行为中嵌套子策略。

a. 创建父策略流行为，并进入父策略的流行为视图。

traffic behavior *behavior-name*

b. 指定子策略，配置策略嵌套。

traffic-policy *policy-name*

c. 退出流行为视图。

quit

(4) 创建父策略，并进入父策略视图。

qos policy *policy-name*

(5) 在父策略中为类指定采用的流行为。

classifier *classifier-name* **behavior** *behavior-name*

缺省情况下，没有为类指定流行为。

2.7 应用策略

2.7.1 设备支持的策略应用位置

QoS 策略支持应用在如下位置：

- 基于接口应用 QoS 策略，支持在入方向和出方向应用。
- 基于 PVC 应用 QoS 策略，支持在入方向和出方向应用。关于 PVC 相关内容的介绍，请参见“二层技术—广域网接入配置指导”中的“ATM”。
- 基于 PW 应用 QoS 策略，仅支持在出方向应用。关于 PW 相关内容的介绍，请参见“MPLS 配置指导”中的“MPLS L2VPN”或“VPLS”。
- 基于全局应用 QoS 策略，支持在入方向和出方向应用。

- 基于控制平面应用 QoS 策略，仅支持在入方向应用。
- 基于管理口控制平面应用 QoS 策略，仅支持在入方向应用。
- 基于上线用户应用 QoS 策略，支持在入方向和出方向应用。

2.7.2 策略应用限制和指导

QoS 策略应用后：

- 用户仍然可以修改 QoS 策略中的流分类规则和流行为，以及二者的对应关系。当流分类规则中使用 ACL 匹配报文时，允许删除或修改该 ACL（包括向该 ACL 中添加、删除和修改匹配规则）。
- 如果一个流行为中配置了多个动作，而其中某个动作未生效，则该 CB 对（即通过 **classifier behavior** 命令关联的一个流分类和一个流行为）都不会生效。

2.7.3 基于接口应用 QoS 策略

1. 配置限制和指导

基于接口应用 QoS 策略时需要注意的是：

- 如果应用某 QoS 策略时未指定 **preorder**，则设备优先执行已配置了 **preorder** 的 QoS 策略的流行为。
- 在接口相同方向上，如果一份流量同时被多个 QoS 策略的类匹配，且 QoS 策略流行为的类型相同，优先执行已配置了 **preorder** 的 QoS 策略的流行为。
- QoS 策略应用在出方向时，对设备发出的协议报文不起作用，以确保这些报文在策略误配置时仍然能够正常发出，维持设备的正常运行。常见的本地协议报文如下：链路维护报文、RIP、LDP、SSH 等。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 在接口上应用已创建的 QoS 策略。

qos apply policy *policy-name* { **inbound** | **outbound** } [**preorder** *preorder-value*]

缺省情况下，未在接口上应用 QoS 策略。

2.7.4 基于 PVC 应用 QoS 策略

1. 配置限制和指导

基于 PVC 应用 QoS 策略时需要注意的是：

- 一个 QoS 策略可以应用于多个 PVC，但在 PVC 的每个方向（出和入两个方向）只能应用一个策略。

- 应用在出方向时，QoS 策略对设备发出的协议报文不起作用，以确保这些报文在策略误配置时仍然能够正常发出，维持设备的正常运行。常见的本地协议报文如下：链路维护报文、IS-IS、OSPF、RIP、BGP、LDP、RSVP、SSH 等。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(3) 在 PVC 上应用已创建的 QoS 策略。

qos apply policy *policy-name* { **inbound** | **outbound** }

缺省情况下，未在 PVC 上应用 QoS 策略。

2.7.5 基于 PW 应用 QoS 策略

1. 配置限制和指导

一个策略可以应用于多条 PW。只能在 PW 的出方向上应用 QoS 策略。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 PW 视图。请选择其中一项进行配置。

- 请依次执行以下命令进入交叉连接 PW 视图。

  xconnect-group *group-name*

  connection *connection-name*

  peer *ip-address* **pw-id** *pw-id* [**in-label** *label-value* **out-label** *label-value*] [**pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI LDP PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling ldp

  peer *ip-address* [**pw-id** *pw-id*] [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI 静态 PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling static

  peer *ip-address* [**pw-id** *pw-id*] [**in-label** *label-value* **out-label** *label-value* [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *]

(3) 在 PW 上应用已创建的 QoS 策略。

qos apply policy *policy-name* **outbound**

缺省情况下，未在 PW 上应用 QoS 策略。

2.7.6 基于全局应用 QoS 策略

1. 功能简介

基于全局应用 QoS 策略后可以对设备所有接口上的流量进行管理。

2. 配置限制和指导

基于全局应用 QoS 策略时，该 QoS 策略会被所有单板应用，如果某个单板 QACL 资源不足，将导致 QoS 策略应用失败。此时需要先执行 **undo qos apply policy global** 命令删除基于全局应用的 QoS 策略，待预留足够资源后，再将 QoS 策略应用到全局。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 全局应用已创建的 QoS 策略。

qos apply policy *policy-name* **global** { **inbound** | **outbound** } [**preorder** *preorder-value*]

缺省情况下，未在全局应用 QoS 策略。

2.7.7 基于控制平面应用 QoS 策略

1. 功能简介

设备上存在用户平面和控制平面：

- 用户平面（User Plane）：是指对报文进行收发、交换的处理单元，它的主要工作是转发报文。在设备上，与之相对应的核心物理实体就是各种专用转发芯片，它们有极高的处理速度和很强的数据吞吐能力。
- 控制平面（Control Plane）：是指运行大部分路由交换协议进程的处理单元，它的主要工作是进行协议报文的解析和协议的计算。在设备上，与之相对应的核心物理实体就是 CPU，它具备灵活的报文处理能力，但数据吞吐能力有限。

用户平面接收到无法识别或处理的报文会送到控制平面进行进一步处理。如果上送控制平面的报文速率超过了控制平面的处理能力，那么上送控制平面的报文会得不到正确转发或及时处理，从而影响协议的正常运行。

为了解决此问题，用户可以把 QoS 策略应用在控制平面上，通过对上送控制平面的报文进行过滤、限速等 QoS 处理，达到保护控制平面正常报文的收发、维护控制平面正常处理状态的目的。

预定义的 QoS 策略中通过协议类型或者协议组类型来标识各种上送控制平面的报文类型，用户也可以在流分类视图下通过 **if-match** 命令引用这些协议类型或者协议组类型来进行报文分类，然后根据需要为这些报文重新配置流行为。系统预定义的 QoS 策略信息可以通过 **display qos policy control-plane pre-defined** 命令查看。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入控制平面视图。

（独立运行模式）

control-plane slot *slot-number*

（IRF 模式）

control-plane chassis *chassis-number* **slot** *slot-number*

(3) 在控制平面上应用已创建的 QoS 策略。

qos apply policy *policy-name* **inbound**

缺省情况下，未在控制平面上应用 QoS 策略。

2.7.8 基于管理口控制平面应用 QoS 策略

1. 功能简介

管理口控制平面仅针对管理口上送给控制平面的报文。

如果管理口上送给控制平面的报文速率超过其处理能力，报文会得不到正确转发或及时处理，从而影响协议的正常运行。

为了解决此问题，用户可以把 QoS 策略应用在管理口控制平面上，通过对管理口上送给控制平面的报文进行 QoS 限速处理，达到保护管理口正常报文的收发、维护管理口正常处理状态的目的。

预定义的 QoS 策略中通过协议类型或者协议组类型来标识各种上送管理口控制平面的报文类型，用户也可以在流分类视图下通过 **if-match** 命令引用这些协议类型或者协议组类型来进行报文分类，然后根据需要为这些报文重新配置流行为。系统预定义的 QoS 策略信息可以通过 **display qos policy control-plane management pre-defined** 命令查看。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入管理口控制平面视图。

control-plane management

(3) 在管理口控制平面上应用已创建的 QoS 策略。

qos apply policy *policy-name* **inbound**

缺省情况下，未在管理口控制平面上应用 QoS 策略。

2.7.9 基于上线用户应用 QoS 策略

1. 功能简介

用户通过身份认证后，认证服务器会将与用户账户绑定的 User Profile 名称下发给设备，设备可以通过 User Profile 视图下配置 QoS 策略来对上线用户的流量进行管理。User Profile 视图下的 QoS 策略只有在用户成功上线后才生效。

2. 配置限制和指导

一个策略可以应用于多个上线用户。上线用户的每个方向（发送和接收两个方向）只能应用一个策略，如果用户想修改某方向上应用的策略，必须先取消原先的配置，然后再配置新的策略。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 User Profile 视图。

user-profile *profile-name*

(3) 在 User Profile 下应用 QoS 策略。

qos apply policy *policy-name* { **inbound** | **outbound** }

缺省情况下，未在 User Profile 下应用 QoS 策略。

参数	说明
inbound	表示对设备入方向的流量（即上线用户发给设备的流量）应用策略
outbound	表示对设备出方向的流量（即设备发给上线用户的流量）应用策略

2.8 配置接口流速统计时间

1. 功能简介

通过配置接口流速统计时间，我们可以统计经过 QoS 策略流分类后每类报文的发送和丢弃速率。假设流速统计时间为 t（t 默认为 5 分钟），则系统将统计最近 t 时间内每类报文发送和丢弃的平均速率，且每 t/5 分钟刷新一次统计速率。流速统计的结果可以通过命令 **display qos policy interface** 查看。

2. 配置限制和指导

配置接口流速统计时间时需要注意的是：

- ATM PVC 的流速统计时间采用所在 ATM 接口上设置的统计时间。
- 子接口的流速统计时间采用主接口上设置的统计时间。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置接口流速统计时间。

qos flow-interval *interval*

缺省情况下，接口流速统计时间为 5 分钟。

2.9 QoS策略显示和维护

在任意视图下执行 **display** 命令可以显示 QoS 策略的运行情况，通过查看显示信息验证配置的效果。

在用户视图下执行 **reset** 命令可以清除 QoS 策略的统计信息。

表2-1 QoS 策略显示和维护

操作	命令
显示QoS策略的配置信息	（独立运行模式） **display qos policy** { **system-defined** | **user-defined** } [*policy-name* [**classifier** *classifier-name*]] [**slot** *slot-number*] （IRF模式） **display qos policy** { **system-defined** | **user-defined** } [*policy-name* [**classifier** *classifier-name*]] [**chassis** *chassis-number* **slot** *slot-number*]
显示Tunnel接口Hub-Spoke隧道应用QoS策略的配置信息和运行情况	**display qos policy advpn tunnel** *number* [*ipv4-address* | *ipv6-address*] [**outbound**]
显示基于控制平面应用QoS策略的信息	（独立运行模式） **display qos policy control-plane slot** *slot-number* （IRF模式） **display qos policy control-plane chassis** *chassis-number* **slot** *slot-number*
显示管理口控制平面应用的QoS策略信息	**display qos policy control-plane management**
显示系统预定义的管理口控制平面应用QoS策略的信息	**display qos policy control-plane management pre-defined**
显示系统预定义的控制平面应用QoS策略的信息	（独立运行模式） **display qos policy control-plane pre-defined** [**slot** *slot-number*] （IRF模式） **display qos policy control-plane pre-defined** [**chassis** *chassis-number* **slot** *slot-number*]
显示基于全局应用QoS策略的信息	（独立运行模式） **display qos policy global** [**slot** *slot-number*] [**inbound** | **outbound**] （IRF模式） **display qos policy global** [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
显示接口上QoS策略的配置信息和运行情况	（独立运行模式） **display qos policy interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] [**slot** *slot-number*] [**inbound** | **outbound**] （IRF模式） **display qos policy interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
显示L2VPN PW上QoS策略的配置信息和运行情况	**display qos policy l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*] [**outbound**]

操作	命令
显示用户上线后User Profile下应用的QoS策略的信息和运行情况	（独立运行模式） **display qos policy user-profile** [**name** *profile-name*] [**user-id** *user-id*] [**slot** *slot-number*] [**inbound** | **outbound**] （IRF模式） **display qos policy user-profile** [**name** *profile-name*] [**user-id** *user-id*] [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
显示QoS和ACL资源的使用情况	（独立运行模式） **display qos-acl resource** [**slot** *slot-number*] （IRF模式） **display qos-acl resource** [**chassis** *chassis-number* **slot** *slot-number*]
显示流行为的配置信息	（独立运行模式） **display traffic behavior** { **system-defined** | **user-defined** } [*behavior-name*] [**slot** *slot-number*] （IRF模式） **display traffic behavior** { **system-defined** | **user-defined** } [*behavior-name*] [**chassis** *chassis-number* **slot** *slot-number*]
显示类的配置信息	（独立运行模式） **display traffic classifier** { **system-defined** | **user-defined** } [*classifier-name*] [**slot** *slot-number*] （IRF模式） **display traffic classifier** { **system-defined** | **user-defined** } [*classifier-name*] [**chassis** *chassis-number* **slot** *slot-number*]
清除控制平面应用QoS策略的统计信息	（独立运行模式） **reset qos policy control-plane slot** *slot-number* （IRF模式） **reset qos policy control-plane chassis** *chassis-number* **slot** *slot-number*
清除Tunnel接口Hub-Spoke隧道应用QoS策略的统计信息	**reset qos policy advpn tunnel** *number* [*ipv4-address* | *ipv6-address*] [**outbound**]
清除管理口控制平面应用QoS策略的统计信息	**reset qos policy control-plane management**
清除全局应用QoS策略的统计信息	**reset qos policy global** [**inbound** | **outbound**]

3 优先级映射

3.1 优先级映射简介

优先级映射可以将报文携带的优先级字段映射成指定优先级字段值，设备根据映射后的优先级字段，为报文提供有差别的 QoS 服务，从而为全面有效的控制报文的转发调度等级提供依据。

3.1.1 优先级介绍

优先级用于标识报文传输的优先程度，可以分为两类：报文携带优先级和设备调度优先级。

报文携带优先级包括：802.1p 优先级、DSCP 优先级、IP 优先级、EXP 优先级等。这些优先级都是根据公认的标准和协议生成，体现了报文自身的优先等级。相关介绍请参见“10.3 附录 C 各种优先级介绍”。

设备调度优先级是指报文在设备内转发时所使用的优先级，只对当前设备自身有效。设备调度优先级包括以下几种：

- 本地优先级（LP）：设备为报文分配的一种具有本地意义的优先级，每个本地优先级对应一个队列，本地优先级值越大的报文，进入的队列优先级越高，从而能够获得优先的调度。
- 丢弃优先级（DP）：在进行报文丢弃时参考的参数，丢弃优先级值越大的报文越被优先丢弃。
- 用户优先级（UP）：设备对于进入的流量，会自动获取报文的优先级作为后续转发调度的参数，这种报文优先级称为用户优先级。对于不同类型的报文，用户优先级所代表的优先级字段不同。对于二层报文，用户优先级取自 802.1p 优先级；对于三层报文，用户优先级取自 IP 优先级；对于 MPLS 报文，用户优先级取自 EXP。

3.1.2 优先级映射表

设备提供了多张优先级映射表，分别对应不同的优先级映射关系。

通常情况下，设备可以通过查找缺省优先级映射表（10.2 附录 B 缺省优先级映射表）来为报文分配相应的优先级。如果缺省优先级映射表无法满足用户需求，可以根据实际情况对映射表进行修改。

3.1.3 优先级映射配置方式

优先级映射配置方式包括：优先级信任模式方式、端口优先级方式、通过 QoS 策略配置（配置 Primap）方式。

1. 优先级信任模式方式

配置端口的优先级信任模式后，设备将信任报文自身携带的优先级。通过优先级映射表，使用所信任的报文携带优先级进行优先级映射，根据映射关系完成对报文优先级的修改，以及实现报文在设备内部的调度。

2. 端口优先级方式

未配置端口的优先级信任模式时，设备会将端口优先级作为报文自身的优先级。通过优先级映射表，对报文进行映射。用户可以配置端口优先级，通过优先级映射，使不同端口收到的报文进入对应的队列，以此实现对不同端口收到报文的差异化调度。

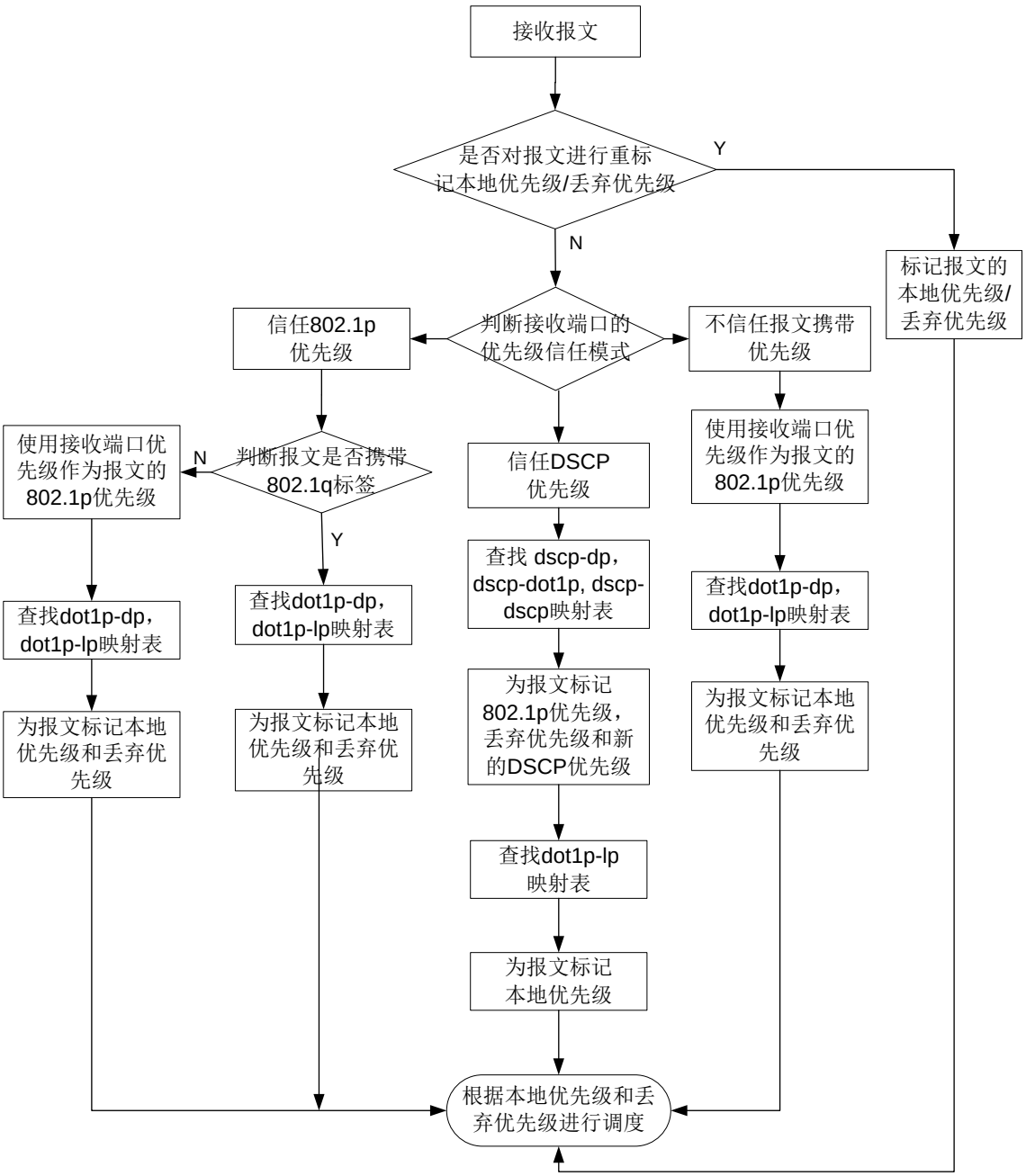
3. QoS 策略配置方式

通过 QoS 策略配置方式，可以对匹配到的报文应用流行为中定义的优先级映射动作，灵活方便的控制报文的优先级映射。

3.1.4 优先级映射过程

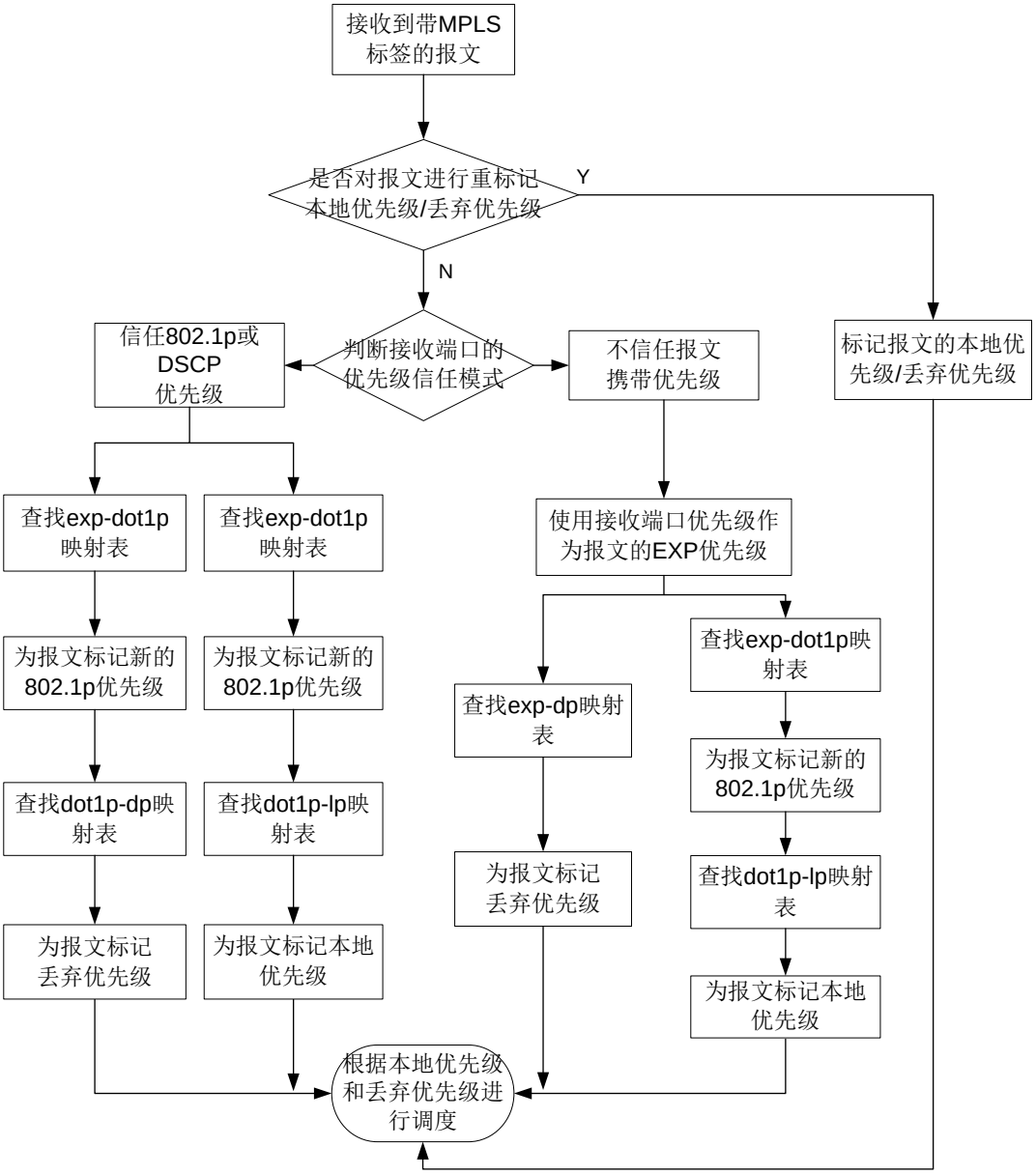
对于接收到的以太网报文，根据优先级信任模式和报文的 802.1Q 标签状态，设备将采用不同的方式为其标记调度优先级。如图 3-1 所示：

图3-1 以太网报文优先级映射过程



对于接收到的 MPLS 报文，根据优先级信任模式和报文的 EXP 优先级状态，设备将采用不同的方式为其标记调度优先级。如图 3-2 所示：

图3-2 MPLS 报文优先级映射过程



说明

关于重标记优先级功能的介绍，请参见重标记。

3.2 优先级映射配置任务简介

优先级映射配置任务如下：

(1) （可选）配置优先级映射表

(2) 配置优先级映射方式。

- 配置优先级信任模式

3.3 配置优先级映射表

(1) 进入系统视图。

system-view

(2) 进入指定的优先级映射表视图。

qos map-table { **dot1p-lp** | **dscp-lp** | **lp-dot1p** }

(3) 配置指定优先级映射表的映射关系。

import *import-value-list* **export** *export-value*

缺省情况下，优先级映射表的映射关系请参见“10.2 附录 B 缺省优先级映射表”。

多次执行本命令，最后一次执行的命令生效。

3.4 配置优先级信任模式

1. 功能简介

配置优先级信任模式后，设备将根据报文自身的优先级，查找优先级映射表，为报文分配优先级参数。

在配置接口上的优先级模式时，用户可以选择下列信任模式：

- **dot1p**：信任报文自带的 802.1p 优先级，以此优先级进行优先级映射。
- **dscp**：信任 IP 报文自带的 DSCP 优先级，以此优先级进行优先级映射。
- **none**：不信任任何优先级。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置优先级信任模式。

qos trust { **dot1p** | **dscp** | **none** }

缺省情况下，不信任报文中的任何优先级，信任端口优先级。

(4) 退回系统视图。

quit

3.5 配置端口优先级

1. 功能简介

按照接收端口的端口优先级，设备通过一一映射为报文分配相应的优先级。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置端口优先级。

qos priority [**dot1p** | **dp** | **dscp** | **exp** | **lp**] *priority-value*

缺省情况下，**lp** 类型优先级的缺省值为 2，**dp** 类型优先级的缺省值为 0，其余类型优先级没有缺省值。

3.6 优先级映射显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后优先级映射的运行情况，通过查看显示信息验证配置的效果。

表3-1 优先级映射显示和维护

操作	命令
显示指定优先级映射表配置情况	**display qos map-table** [**dot1p-lp** | **dscp-lp** | **lp-dot1p**]
显示端口优先级信任模式信息	**display qos trust interface** [*interface-type interface-number*]

3.7 优先级映射典型配置举例

3.7.1 优先级信任模式和端口优先级配置举例

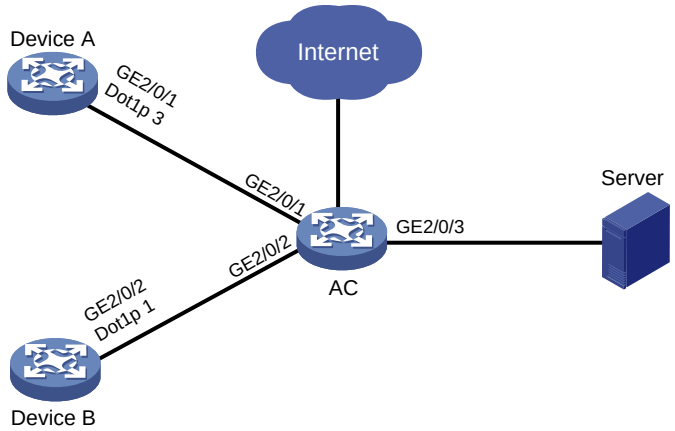
1. 组网需求

如图 3-3 所示，Device A、Device B、Server 均与 Device C 相连。通过配置优先级信任模式和端口优先级实现如下需求：

如果 Device C 在接口 GigabitEthernet2/0/3 的出方向发生拥塞，则优先让 Device A 访问 Server。

2. 组网图

图3-3 优先级信任模式和端口优先级配置组网图



3. 配置步骤

(1) 方法一

在接口 GigabitEthernet2/0/1 和 GigabitEthernet2/0/2 上分别配置优先级信任模式为 **dot1p**。

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 2/0/1
[DeviceC-GigabitEthernet2/0/1] qos trust dot1p
[DeviceC-GigabitEthernet2/0/1] quit
[DeviceC] interface gigabitethernet 2/0/2
[DeviceC-GigabitEthernet2/0/2] qos trust dot1p
[DeviceC-GigabitEthernet2/0/2] quit
```

(2) 方法二

在接口 GigabitEthernet2/0/1 和 GigabitEthernet2/0/2 上分别配置端口优先级，GigabitEthernet2/0/1 上配置的端口优先级值要高于 GigabitEthernet2/0/2 上配置的端口优先级值。（同时保证在接口 GigabitEthernet2/0/1 和 GigabitEthernet2/0/2 上没有配置信任模式。）

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 2/0/1
[DeviceC-GigabitEthernet2/0/1] qos priority 3
[DeviceC-GigabitEthernet2/0/1] quit
[DeviceC] interface gigabitethernet 2/0/2
[DeviceC-GigabitEthernet2/0/2] qos priority 1
[DeviceC-GigabitEthernet2/0/2] quit
```

3.7.2 优先级映射表和重标记配置举例

1. 组网需求

公司企业网通过 Device 实现各部门之间的互连。网络环境描述如下：

- 市场部门通过端口 GigabitEthernet2/0/1 接入 Device，标记市场部门发出的报文的 802.1p 优先级为 3；
- 研发部门通过端口 GigabitEthernet2/0/2 接入 Device，标记研发部门发出的报文的 802.1p 优先级为 4；
- 管理部门通过端口 GigabitEthernet2/0/3 接入 Device，标记管理部门发出的报文的 802.1p 优先级为 5。

实现如下需求：

访问公共服务器的时候，研发部门 > 管理部门 > 市场部门。

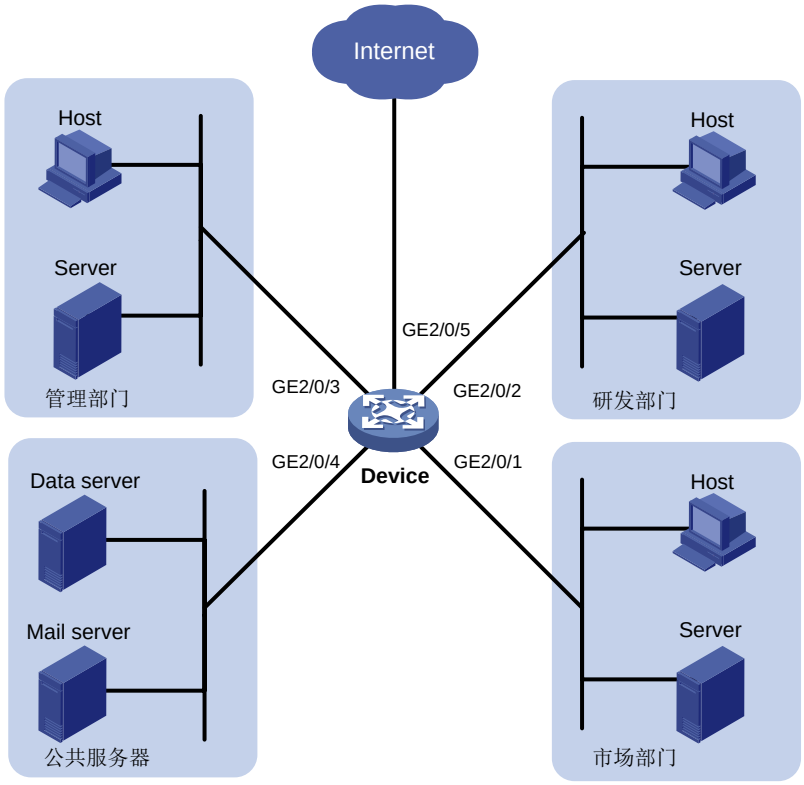
- 通过优先级映射将研发部门发出的报文放入出队列 6 中，优先进行处理；
- 通过优先级映射将管理部门发出的报文放入出队列 4 中，次优先进行处理；
- 通过优先级映射将市场部门发出的报文放入出队列 2 中，最后进行处理。

访问 Internet 的时候，管理部门 > 市场部门 > 研发部门。

- 重标记管理部门发出的报文本地优先级为 6，优先进行处理；
- 重标记市场部门发出的报文的本地优先级为 4，次优先进行处理；
- 重标记研发部门发出的报文的本地优先级为 2，最后进行处理。

2. 组网图

图3-4 优先级映射表和重标记配置组网图



3. 配置步骤

(1) 配置端口的端口优先级

配置端口 GigabitEthernet2/0/1 的端口优先级为 3。

```
<Device> system-view
[Device] interface gigabitethernet 2/0/1
[Device-GigabitEthernet2/0/1] qos priority 3
[Device-GigabitEthernet2/0/1] quit
```

配置端口 GigabitEthernet2/0/2 的端口优先级为 4。

```
[Device] interface gigabitethernet 2/0/2
[Device-GigabitEthernet2/0/2] qos priority 4
[Device-GigabitEthernet2/0/2] quit
```

配置端口 GigabitEthernet2/0/3 的端口优先级为 5。

```
[Device] interface gigabitethernet 2/0/3
[Device-GigabitEthernet2/0/3] qos priority 5
[Device-GigabitEthernet2/0/3] quit
```

(2) 配置优先级映射表

配置 802.1p 优先级到本地优先级映射表，将 802.1p 优先级 3、4、5 对应的本地优先级配置为 2、6、4。保证访问服务器的优先级为研发部门（6）>管理部门（4）>市场部门（2）。

```
[Device] qos map-table dot1p-lp
[Device-maptbl-dot1p-lp] import 3 export 2
```

```
[Device-maptbl-dot1p-lp] import 4 export 6
[Device-maptbl-dot1p-lp] import 5 export 4
[Device-maptbl-dot1p-lp] quit
```

(3) 配置重标记

将管理、市场、研发部门发出的 HTTP 报文的 802.1p 优先级分别重标记为 4、5、3，使其能根据前面配置的映射表分别映射到本地优先级 6、4、2。

创建 ACL 3000，用来匹配 HTTP 报文。

```
[Device] acl advanced 3000
[Device-acl-adv-3000] rule permit tcp destination-port eq 80
[Device-acl-adv-3000] quit
# 创建流分类，匹配 ACL 3000。
[Device] traffic classifier http
[Device-classifier-http] if-match acl 3000
[Device-classifier-http] quit
```

配置管理部门的重标记策略并应用到接口 GigabitEthernet2/0/3 的入方向。

```
[Device] traffic behavior admin
[Device-behavior-admin] remark dot1p 4
[Device-behavior-admin] quit
[Device] qos policy admin
[Device-qospolicy-admin] classifier http behavior admin
[Device-qospolicy-admin] quit
[Device] interface gigabitethernet 2/0/3
[Device-GigabitEthernet2/0/3] qos apply policy admin inbound
```

配置市场部门的重标记策略并应用到接口 GigabitEthernet2/0/1 的入方向。

```
[Device] traffic behavior market
[Device-behavior-market] remark dot1p 5
[Device-behavior-market] quit
[Device] qos policy market
[Device-qospolicy-market] classifier http behavior market
[Device-qospolicy-market] quit
[Device] interface gigabitethernet 2/0/1
[Device-GigabitEthernet2/0/1] qos apply policy market inbound
```

配置研发部门的重标记策略并应用到接口 GigabitEthernet2/0/2 的入方向。

```
[Device] traffic behavior rd
[Device-behavior-rd] remark dot1p 3
[Device-behavior-rd] quit
[Device] qos policy rd
[Device-qospolicy-rd] classifier http behavior rd
[Device-qospolicy-rd] quit
[Device] interface gigabitethernet 2/0/2
[Device-GigabitEthernet1/0/2] qos apply policy rd inbound
```

4 流量监管、流量整形和限速

4.1 流量监管、流量整形和限速简介

如果不限制用户发送的流量，那么大量用户不断突发的数据只会使网络更拥挤。为了使有限的网络资源能够更好地发挥效用，更好地为更多的用户服务，必须对用户的流量加以限制。流量监管、流量整形和限速可以实现流量的速率限制功能，而要实现此功能就必须对通过设备的流量进行度量。一般采用令牌桶（Token Bucket）对流量进行度量。

4.1.1 流量评估与令牌桶

1. 令牌桶

令牌桶可以看作是一个存放一定数量令牌的容器。系统按设定的速度向桶中放置令牌，当桶中令牌满时，多出的令牌溢出，桶中令牌不再增加。

2. 用令牌桶评估流量

在用令牌桶评估流量规格时，是以令牌桶中的令牌数量是否足够满足报文的转发为依据的。如果桶中存在足够的令牌可以用来转发报文，称流量遵守或符合这个规格，否则称为不符合或超标。

评估流量时令牌桶的参数包括：

- 平均速率：向桶中放置令牌的速率，即允许的流的平均速度。通常配置为 CIR。
- 突发尺寸：令牌桶的容量，即每次突发所允许的最大的流量尺寸。通常配置为 CBS，突发尺寸必须大于最大报文长度。

每到达一个报文就进行一次评估。每次评估，如果桶中有足够的令牌可供使用，则说明流量控制在允许的范围内，此时要从桶中取走满足报文的转发的令牌；否则说明已经耗费太多令牌，流量超标了。

3. 复杂评估

为了评估更复杂的情况，实施更灵活的调控策略，可以使用两个令牌桶（分别称为 C 桶和 E 桶）对流量进行评估。主要有如下三种算法。

(1) 单速率单桶双色算法

- CIR：表示向 C 桶中投放令牌的速率，即 C 桶允许传输或转发报文的平均速率；
- CBS：表示 C 桶的容量，即 C 桶瞬间能够通过的承诺突发流量。

每次评估时，依据下面的情况，可以分别实施不同的流控策略：

- 如果 C 桶有足够的令牌，报文被标记为 green，即绿色报文；
- 如果 C 桶令牌不足，报文被标记为 red，即红色报文。

(2) 单速率双桶三色算法

- CIR：表示向 C 桶中投放令牌的速率，即 C 桶允许传输或转发报文的平均速率；
- CBS：表示 C 桶的容量，即 C 桶瞬间能够通过的承诺突发流量；
- EBS：表示 E 桶的容量的增量，即 E 桶瞬间能够通过的超出突发流量，取值不为 0。E 桶的容量等于 CBS 与 EBS 的和。

每次评估时，依据下面的情况，可以分别实施不同的流控策略：

- 如果 C 桶有足够的令牌，报文被标记为 green，即绿色报文；
- 如果 C 桶令牌不足，但 E 桶有足够的令牌，报文被标记为 yellow，即黄色报文；
- 如果 C 桶和 E 桶都没有足够的令牌，报文被标记为 red，即红色报文。

(3) 双速率双桶三色算法

- CIR：表示向 C 桶中投放令牌的速率，即 C 桶允许传输或转发报文的平均速率；
- CBS：表示 C 桶的容量，即 C 桶瞬间能够通过的承诺突发流量；
- PIR：表示向 E 桶中投放令牌的速率，即 E 桶允许传输或转发报文的最大速率；
- EBS：表示 E 桶的容量，即 E 桶瞬间能够通过的超出突发流量。

每次评估时，依据下面的情况，可以分别实施不同的流控策略：

- 如果 C 桶有足够的令牌，报文被标记为 green，即绿色报文；
- 如果 C 桶令牌不足，但 E 桶有足够的令牌，报文被标记为 yellow，即黄色报文；
- 如果 C 桶和 E 桶都没有足够的令牌，报文被标记为 red，即红色报文。

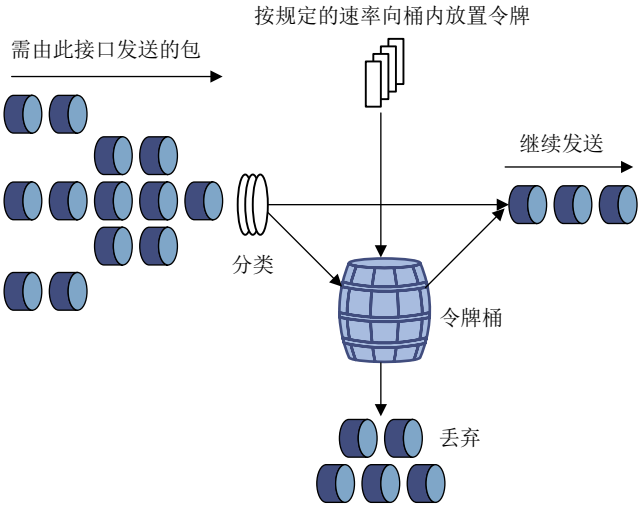
4.1.2 流量监管

说明

流量监管支持应用在入方向和出方向，即对设备接收或发出的流量生效。

流量监管就是对流量进行控制，通过监督进入网络的流量速率，对超出部分的流量进行“惩罚”，使进入的流量被限制在一个合理的范围之内，以保护网络资源和运营商的利益。例如可以限制 HTTP 报文不能占用 50%以上的网络带宽。如果发现流量超出规格，流量监管可以选择丢弃超规格报文，或重新配置超规格报文的优先级。

图4-1 TP 示意图



流量监管广泛的用于监管进入 Internet 服务提供商 ISP 的网络流量。流量监管还包括对所监管流量的流分类服务，并依据不同的评估结果，实施预先设定好的监管动作。这些动作可以是：

- 转发：比如对评估结果为“符合”的报文继续转发。
- 丢弃：比如对评估结果为“不符合”的报文进行丢弃。
- 改变优先级并转发：比如对评估结果为“符合”的报文，将其优先级进行重标记后再进行转发。
- 改变优先级并进入下一级监管：比如对评估结果为“符合”的报文，将其优先级进行重标记后再进入下一级的监管。
- 进入下一级的监管：流量监管可以进行分级，每级关注和监管更具体的目标。

4.1.3 流量整形

说明

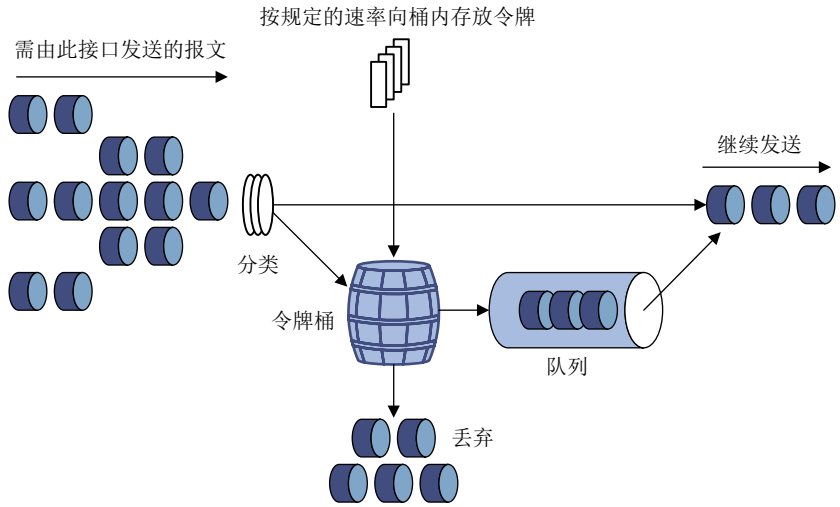
流量整形仅对设备发出的流量生效。

流量整形是一种主动调整流量输出速率的措施。一个典型应用是基于下游网络节点的流量监管指标来控制本地流量的输出。

流量整形与流量监管的主要区别在于：

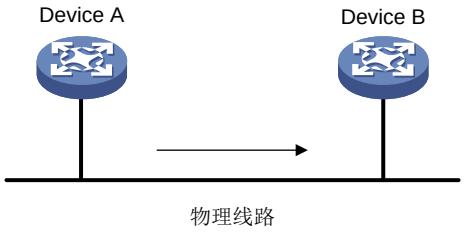
- 流量整形对流量监管中需要丢弃的报文进行缓存——通常是将它们放入缓冲区或队列内，如图 4-2 所示。当令牌桶有足够的令牌时，再均匀的向外发送这些被缓存的报文。
- 流量整形可能会增加延迟，而流量监管几乎不引入额外的延迟。

图4-2 流量整形示意图



例如，在图 4-3 所示的应用中，设备 Device A 向 Device B 发送报文。Device B 要对 Device A 发送来的报文进行流量监管，对超出规格的流量直接丢弃。

图4-3 流量整形的应用



为了减少报文的无谓丢失，可以在 Device A 的出口对报文进行流量整形处理。将超出流量整形特性的报文缓存在 Device A 中。当可以继续发送下一批报文时，流量整形再从缓冲队列中取出报文进行发送。这样，发向 Device B 的报文将都符合 Device B 的流量规定。

4.1.4 限速

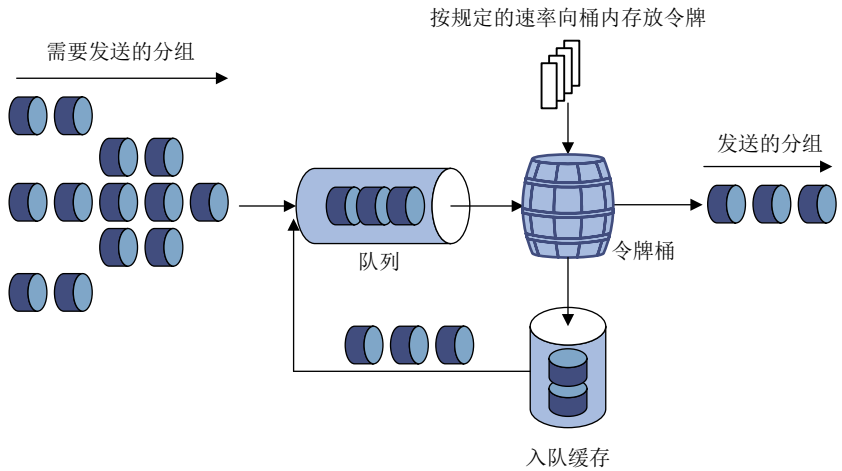
说明

限速支持应用在入方向和出方向，即对设备接收或发出的流量生效。

利用限速可以在一个接口或 PW 上限制发送报文（除紧急报文）的总速率。

限速也是采用令牌桶进行流量控制。假如在设备的某个接口上配置了限速，所有经由该接口发送的报文首先要经过限速的令牌桶进行处理。如果令牌桶中有足够的令牌，则报文可以发送；否则，报文将进入 QoS 队列进行拥塞管理。这样，就可以对该接口的报文流量进行控制。

图4-4 限速处理过程示意图



由于采用了令牌桶控制流量，当令牌桶中存有令牌时，可以允许报文的突发性传输；当令牌桶中没有令牌时，报文必须等到桶中生成了新的令牌后才可以继续发送。这就限制了报文的流量不能大于令牌生成的速度，达到了限制流量，同时允许突发流量通过的目的。

与流量监管相比，限速能够限制所有报文。当用户只要求对所有报文限速时，使用限速比较简单。

4.2 配置流量监管

4.2.1 流量监管配置方式介绍

可以通过MQC方式和非MQC方式配置流量监管，其中非MQC方式配置流量监管时分为以下几种：

- 基于CAR列表的流量监管配置。
- 基于ACL的流量监管配置。
- 适配所有流的流量监管配置。
- 基于上线用户的流量监管配置。

如果接口上同时采用了MQC方式和非MQC方式配置了流量监管，那么只有前者会生效。

4.2.2 配置流量监管（MQC方式）

1. 配置限制和指导

设备支持基于接口、PVC、控制平面、管理口控制平面和上线用户应用QoS策略配置流量监管。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 定义类。

a. 创建类，并进入类视图。

traffic classifier *classifier-name* [**operator** { **and** | **or** }]

b. 定义匹配数据包的规则。

if-match [**not**] *match-criteria*

缺省情况下，未定义匹配数据包的规则。

具体规则的介绍，请参见“QoS命令”中的**if-match**命令。

c. 退回系统视图。

quit

(3) 定义流行为。

a. 创建一个流行为并进入流行为视图。

traffic behavior *behavior-name*

b. 配置流量监管动作。

（绝对值配置方式）

car cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

car cir *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

（百分比配置方式）

car cir percent *cir-percent* [**cbs** *cbs-time* [**ebs** *ebs-time*]] [**green** *action* | **red** *action* | **yellow** *action*] *

car cir percent *cir-percent* [**cbs** *cbs-time*] **pir percent** *pir-percent* [**ebs** *ebs-time*] [**green** *action* | **red** *action* | **yellow** *action*] *

缺省情况下，未配置流量监管动作。

c. 退回系统视图。

quit

(4) 定义策略。

a. 创建策略并进入策略视图。

qos policy *policy-name*

b. 在策略中为类指定采用的流行为。

classifier *classifier-name* **behavior** *behavior-name*

缺省情况下，未指定类对应的流行为。

c. 退回系统视图。

quit

(5) 应用 QoS 策略。

具体配置请参见“2.7 应用策略”

缺省情况下，未应用 QoS 策略。

4.2.3 配置基于 CAR 列表的流量监管

(1) 进入系统视图。

system-view

(2) 创建 CAR 列表并配置匹配规则。

qos carl *carl-index* { **dscp** *dscp-list* | **mac** *mac-address* | **mpls-exp** *mpls-exp-value* | **precedence** *precedence-value* | { **destination-ip-address** | **source-ip-address** } { **range** *start-ip-address* **to** *end-ip-address* | **subnet** *ip-address mask-length* } [**per-address** [**shared-bandwidth**]] }

(3) 进入接口视图。

interface *interface-type interface-number*

(4) 在接口上配置基于 CAR 列表的 CAR 策略。

（绝对值配置方式）

qos car { **inbound** | **outbound** } **carl** *carl-index* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

qos car { **inbound** | **outbound** } **carl** *carl-index* **cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

（百分比配置方式）

qos car { **inbound** | **outbound** } **carl** *carl-index* **percent cir** *cir-percent* [**cbs** *cbs-time* [**ebs** *ebs-time*]] [**green** *action* | **red** *action* | **yellow** *action*] *

qos car { **inbound** | **outbound** } **carl** *carl-index* **percent cir** *cir-percent* [**cbs** *cbs-time*] **pir** *pir-percent* [**ebs** *ebs-time*] [**green** *action* | **red** *action* | **yellow** *action*] *

缺省情况下，接口上未应用 CAR 策略。

4.2.4 配置基于 CAR 列表的动态流量监管

1. 功能简介

动态流量监管功能指的是设备能够动态调整流量的可用带宽，提高接口的带宽利用率。

当符合以下配置时，设备将启动接口上的动态流量监管功能：

- 配置了对网段内逐 IP 地址流量进行限速的 CAR 列表（通过 **qos carl** 命令创建 CAR 列表，并指定 **per-address** 参数）。
- 在接口同方向上仅应用了一个 CAR 策略（即引用上一步创建的 CAR 列表），且指定了最大承诺信息速率（配置 **qos car { inbound | outbound } carl** 命令并指定 **max-cir** 参数）。

接口流量可以分为两部分：不受流量监管限制的流量和受到流量监管限制的流量。当接口上启动动态流量监管功能后，将根据 **qos car bandwidth-refresh-interval** 命令配置的带宽刷新时间定时刷新每 IP 地址流的允许 CIR 值：

- 如果当前接口流量带宽未超过带宽最大使用门限值，且刷新后的允许 CIR 值小于最大 CIR 值（由 **qos car** 命令的 **max-cir** 参数指定），则增加允许 CIR 值。
- 如果当前接口流量带宽已达到带宽最大使用门限值，或刷新后的允许 CIR 值超过最大 CIR 值（由 **qos car** 命令的 **max-cir** 参数指定），则允许 CIR 值将不再增加。
- 如果接口下有新 IP 地址流加入，新 IP 地址流将使用与其他流相同的允许 CIR 值。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建 CAR 列表。

qos carl *carl-index* { *destination-ip-address* | *source-ip-address* } { **object-group** *object-group-name* | **range** *start-ip-address* **to** *end-ip-address* | **subnet** *ip-address mask-length* } **per-address** [**shared-bandwidth**] [**time-range** *time-range-name*]

(3) （可选）配置动态流量监管的带宽刷新时间。

qos car bandwidth-refresh-interval *interval*

缺省情况下，动态流量监管的带宽刷新时间为 600 秒。

(4) 进入接口视图。

interface *interface-type interface-number*

(5) （可选）设置流量的带宽最大使用率门限。

qos car bandwidth-utilization-threshold *high-percent*

缺省情况下，流量的带宽最大使用率门限为 80%。

(6) 在接口上应用 CAR 列表进行流量监管。

qos car { **inbound** | **outbound** } **carl** *carl-index* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] **max-cir** *max-committed-information-rate* [**green** *action* | **red** *action* | **yellow** *action*] *

缺省情况下，未在接口上应用 CAR 列表进行流量监管。

4.2.5 配置基于 ACL 的流量监管

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 在接口上配置基于 ACL 规则的 CAR 策略。

（绝对值配置方式）

qos car { **inbound** | **outbound** } **acl** [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

qos car { **inbound** | **outbound** } **acl** [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

缺省情况下，接口上未应用 CAR 策略。

4.2.6 配置适配所有流的流量监管

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 在接口应用 CAR 策略。

（绝对值配置方式）

qos car { **inbound** | **outbound** } **any cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**green** *action* | **red** *action* | **yellow** *action*] *

qos car { **inbound** | **outbound** } **any cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**green** *action* | **red** *action* | **yellow** *action*] *

缺省情况下，接口上没有应用 CAR 策略。

4.2.7 配置基于上线用户的流量监管

1. 功能简介

用户通过身份认证后，认证服务器会将与用户账户绑定的 User Profile 名称下发给设备，设备可以通过 User Profile 视图下配置 CAR 策略来对上线用户进行流量监管：当用户数据流量符合承诺速率时，允许数据包通过；用户数据流量不符合承诺速率时，丢弃数据包，只要用户上线，认证服务器会自动下发相应的 User Profile，当用户下线，系统会自动取消相应的配置，不需要再进行手工调整。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入 User Profile 视图。

user-profile *profile-name*

(3) 在 User Profile 下应用 CAR 策略。

qos car { **inbound** | **outbound** } **any cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]]

qos car { **inbound** | **outbound** } **any cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*]

缺省情况下，在 User Profile 下没有应用 CAR 策略。

4.3 配置流量整形

4.3.1 流量整形配置方式介绍

可以通过 MQC 方式和非 MQC 方式配置流量整形，其中非 MQC 方式配置流量整形时分为以下几种：

- 基于 ACL 的流量整形配置。
- 适配所有流的流量整形配置。

如果接口上同时采用了 MQC 方式和非 MQC 方式配置了流量整形，那么只有前者会生效。

4.3.2 配置流量整形（MQC 方式）

1. 配置限制和指导

设备支持基于接口、PVC、VLAN、全局、控制平面和上线用户应用 QoS 策略配置流量整形。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 定义类。

a. 创建类，并进入类视图。

traffic classifier *classifier-name* [**operator** { **and** | **or** }]

b. 定义匹配数据包的规则。

if-match [**not**] *match-criteria*

缺省情况下，未定义匹配数据包的规则。

具体规则的介绍，请参见“QoS 命令”中的 **if-match** 命令。

c. 退回系统视图。

quit

(3) 定义流行为。

a. 创建一个流行为并进入流行为视图。

traffic behavior *behavior-name*

b. 配置流量整形动作。

（绝对值配置方式）

gts cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**queue-length** *queue-length*]

（百分比配置方式）

gts percent cir *cir-percent* [**cbs** *cbs-time* [**ebs** *ebs-time*]] [**queue-length** *queue-length*]

缺省情况下，未配置流量整形动作。

c. 退回系统视图。

quit

(4) 定义策略。

a. 创建策略并进入策略视图。

qos policy *policy-name*

b. 在策略中为类指定采用的流行为。

classifier *classifier-name* **behavior** *behavior-name*

缺省情况下，未指定类对应的流行为。

c. 退回系统视图。

quit

(5) 应用 QoS 策略。

具体配置请参见“2.7 应用策略”

缺省情况下，未应用 QoS 策略。

4.3.3 配置基于 ACL 的流量整形

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置基于 ACL 的流量整形。

（绝对值配置方式）

qos gts acl [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**queue-length** *queue-length*]

qos gts acl [**ipv6**] *acl-number* **cir** *committed-information-rate* [**cbs** *committed-burst-size*] **pir** *peak-information-rate* [**ebs** *excess-burst-size*] [**queue-length** *queue-length*]

缺省情况下，接口上未配置流量整形。

4.3.4 配置适配所有流的流量整形

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置适配所有流的流量整形。

qos gts any cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]] [**queue-length** *queue-length*]

缺省情况下，接口上未配置流量整形。

4.4 配置限速

4.4.1 配置接口限速

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置接口限速。

（绝对值配置方式）

qos lr outbound cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]]

缺省情况下，接口上未配置接口限速。

4.4.2 配置 PW 限速

(1) 进入系统视图。

system-view

(2) 进入 PW 视图。请选择其中一项进行配置。

- 请依次执行以下命令进入交叉连接 PW 视图。

  xconnect-group *group-name*

  connection *connection-name*

peer *ip-address* **pw-id** *pw-id* [**in-label** *label-value* **out-label** *label-value*] [**pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI LDP PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling ldp

  peer *ip-address* [**pw-id** *pw-id*] [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI 静态 PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling static

  peer *ip-address* [**pw-id** *pw-id*] [**in-label** *label-value* **out-label** *label-value* [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *]

(3) 配置 PW 限速。

qos lr outbound cir *committed-information-rate* [**cbs** *committed-burst-size* [**ebs** *excess-burst-size*]]

缺省情况下，未配置 PW 限速。

4.5 流量监管、流量整形和限速显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后流量监管、流量整形和接口限速的运行情况，通过查看显示信息验证配置的效果。

表4-1 流量监管、流量整形和限速显示和维护

操作	命令
显示接口的流量监管配置情况和统计信息	**display qos car interface** [*interface-type interface-number*]
显示CAR列表	（独立运行模式） **display qos carl** [*carl-index*] [**slot** *slot-number*] （IRF模式） **display qos carl** [*carl-index*] [**chassis** *chassis-number* **slot** *slot-number*]
显示接口的流量整形配置情况和统计信息	**display qos gts interface** [*interface-type interface-number*]
显示限速配置情况和统计信息	**display qos lr** { **interface** [*interface-type interface-number*] | **l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*] }
显示QoS和ACL资源的使用情况（本命令的详细介绍，请参见“ACL和QoS命令参考”中的“ACL”）	（独立运行模式） **display qos-acl resource** [**slot** *slot-number*] （IRF模式） **display qos-acl resource** [**chassis** *chassis-number* **slot** *slot-number*]

操作	命令
显示流量监管的相关配置信息	（独立运行模式） **display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*] （IRF模式） **display traffic behavior user-defined** [*behavior-name*] [**chassis** *chassis-number* **slot** *slot-number*]

4.6 流量监管、流量整形和限速典型配置举例

4.6.1 流量监管典型配置举例

1. 配置需求

- 设备 Device A 通过接口 GigabitEthernet2/0/3 和设备 Device B 的接口 GigabitEthernet2/0/1 互连
- Server、Host A、Host B 可经由 Device A 和 Device B 访问 Internet
- Server、Host A 与 Device A 的 GigabitEthernet2/0/1 接口在同一网段
- Host B 与 Device A 的 GigabitEthernet2/0/2 接口在同一网段

要求在设备 Device A 上对接口 GigabitEthernet2/0/1 接收到的源自 Server 和 Host A 的报文流分别实施流量控制如下：

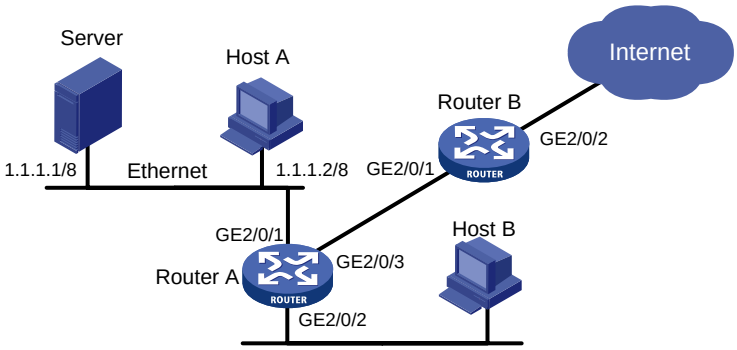
- 来自 Server 的报文流量约束为 10240kbps，流量小于 10240kbps 时可以正常发送，流量超过 10240kbps 时则将违规报文的优先级设置为 0 后进行发送；
- 来自 Host A 的报文流量约束为 2560kbps，流量小于 2560kbps 时可以正常发送，流量超过 2560kbps 时则丢弃违规报文；

对设备 Device B 的 GigabitEthernet2/0/1 和 GigabitEthernet2/0/2 接口收发报文有如下要求：

- Device B 的 GigabitEthernet2/0/1 接口接收报文的总流量限制为 20480kbps，如果超过流量限制则将违规报文丢弃；
- 经由 Device B 的 GigabitEthernet2/0/2 接口进入 Internet 的报文流量限制为 10240kbps，如果超过流量限制则将违规报文丢弃。

2. 组网图

图4-5 流量监管配置组网图



3. 配置步骤

(1) 配置设备 Device A

配置 ACL 规则列表，分别匹配来源于 Server 和 Host A 的报文流。

```
<DeviceA> system-view
[DeviceA] acl basic 2001
[DeviceA-acl-ipv4-basic-2001] rule permit source 1.1.1.1 0
[DeviceA-acl-ipv4-basic-2001] quit
[DeviceA] acl basic 2002
[DeviceA-acl-ipv4-basic-2002] rule permit source 1.1.1.2 0
[DeviceA-acl-ipv4-basic-2002] quit
```

创建流分类 server，匹配 Server 发出的报文流。

```
[DeviceA] traffic classifier server
[DeviceA-classifier-server] if-match acl 2001
[DeviceA-classifier-server] quit
```

创建流分类 host，匹配 Host 发出的报文流。

```
[DeviceA] traffic classifier host
[DeviceA-classifier-host] if-match acl 2002
[DeviceA-classifier-host] quit
```

创建流行为 server，动作为流量监管，cir 为 10240kbps，对超出限制的报文（红色报文）将其 DSCP 优先级设置为 0 后发送。

```
[DeviceA] traffic behavior server
[DeviceA-behavior-server] car cir 10240 red remark-dscp-pass 0
[DeviceA-behavior-server] quit
```

创建流行为 host，动作为流量监管，cir 为 2560kbps，由于默认对红色报文的处理方式就是丢弃，因此无需配置。

```
[DeviceA] traffic behavior host
[DeviceA-behavior-host] car cir 2560
[DeviceA-behavior-host] quit
```

创建 QoS 策略，命名为 car，将流分类 server 和流行为 server 进行关联；将流分类 host 和流行为 host 进行关联。

```
[DeviceA] qos policy car
```

```
[DeviceA-qospolicy-car] classifier server behavior server
[DeviceA-qospolicy-car] classifier host behavior host
[DeviceA-qospolicy-car] quit
```

将 QoS 策略 car 应用到接口 GigabitEthernet2/0/1 的入方向上。

```
[DeviceA] interface gigabitethernet 2/0/1
[DeviceA-GigabitEthernet2/0/1] qos apply policy car inbound
```

(2) 配置设备 Device B

配置高级 ACL3001，匹配 HTTP 报文。

```
<DeviceB> system-view
[DeviceB] acl advanced 3001
[DeviceB-acl-adv-3001] rule permit tcp destination-port eq 80
[DeviceB-acl-adv-3001] quit
```

创建流分类 http，匹配 ACL 3001。

```
[DeviceB] traffic classifier http
[DeviceB-classifier-http] if-match acl 3001
[DeviceB-classifier-http] quit
```

创建流分类 class，匹配所有报文。

```
[DeviceB] traffic classifier class
[DeviceB-classifier-class] if-match any
[DeviceB-classifier-class] quit
```

创建流行为 car_inbound，动作为流量监管，cir 为 20480kbps，由于默认对红色报文的处理方式就是丢弃，因此无需配置。

```
[DeviceB] traffic behavior car_inbound
[DeviceB-behavior-car_inbound] car cir 20480
[DeviceB-behavior-car_inbound] quit
```

创建流行为 car_outbound，动作为流量监管，cir 为 10240kbps。

```
[DeviceB] traffic behavior car_outbound
[DeviceB-behavior-car_outbound] car cir 10240
[DeviceB-behavior-car_outbound] quit
```

创建 QoS 策略，命名为 car_inbound，将流分类 class 和流行为 car_inbound 进行关联。

```
[DeviceB] qos policy car_inbound
[DeviceB-qospolicy-car_inbound] classifier class behavior car_inbound
[DeviceB-qospolicy-car_inbound] quit
```

创建 QoS 策略，命名为 car_outbound，将流分类 http 和流行为 car_outbound 进行关联。

```
[DeviceB] qos policy car_outbound
[DeviceB-qospolicy-car_outbound] classifier http behavior car_outbound
[DeviceB-qospolicy-car_outbound] quit
```

将 QoS 策略 car_inbound 应用到接口 GigabitEthernet2/0/1 的入方向上。

```
[DeviceB] interface gigabitethernet 2/0/1
[DeviceB-GigabitEthernet2/0/1] qos apply policy car_inbound inbound
```

将 QoS 策略 car_outbound 应用到接口 GigabitEthernet2/0/2 的出方向上。

```
[DeviceB] interface gigabitethernet 2/0/2
[DeviceB-GigabitEthernet2/0/2] qos apply policy car_outbound outbound
```

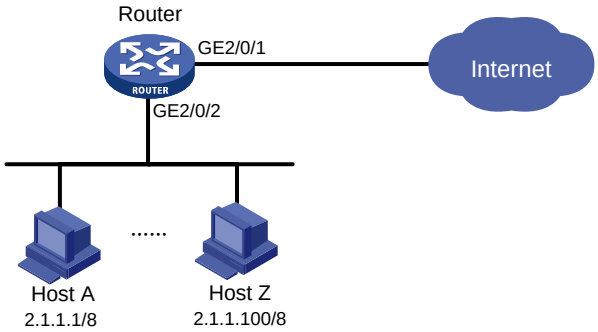
4.6.2 IP 限速配置举例

1. 配置需求

要求在设备 Device 上对接口 GigabitEthernet2/0/2 接收到的报文流进行限速：对 HostA～HostZ（源地址属于 IP 地址段 2.1.1.1～2.1.1.100）进行 IP 限速，逐 IP 地址流量限速 5kbps，网段内各 IP 地址的流量共享剩余带宽。

2. 组网图

图4-6 IP 限速配置组网图



3. 配置步骤

在接口 GigabitEthernet2/0/2 上对源地址属于 IP 地址段 2.1.1.1～2.1.1.100 内所有 PC 进行限速，网段内各 IP 地址的流量共享剩余带宽。

```
<Device> system-view
[Device] qos carl 1 source-ip-address range 2.1.1.1 to 2.1.1.100 per-address shared-bandwidth
[Device] interface gigabitethernet 2/0/2
[Device-GigabitEthernet2/0/2] qos car inbound carl 1 cir 500 cbs 1875 ebs 0 green pass red discard
[Device-GigabitEthernet2/0/2] quit
```

5 拥塞管理

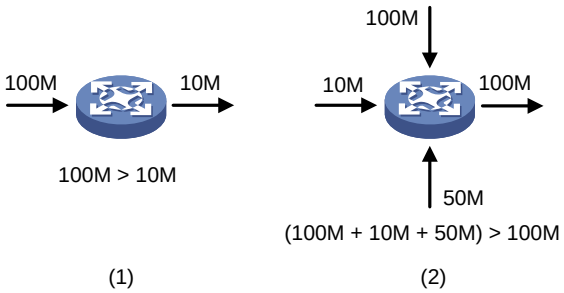
5.1 拥塞管理简介

5.1.1 拥塞的产生、影响和对策

所谓拥塞，是指当前供给资源相对于正常转发处理需要资源的不足，从而导致服务质量下降的一种现象。

在复杂的 Internet 分组交换环境下，拥塞极为常见。以图 5-1 中的两种情况为例：

图5-1 流量拥塞示意图



拥塞有可能会引发一系列的负面影响：

- 拥塞增加了报文传输的延迟和抖动，可能会引起报文重传，从而导致更多的拥塞产生。
- 拥塞使网络的有效吞吐率降低，造成网络资源的利用率降低。
- 拥塞加剧会耗费大量的网络资源（特别是存储资源），不合理的资源分配甚至可能导致系统陷入资源死锁而崩溃。

在分组交换以及多用户业务并存的复杂环境下，拥塞又是不可避免的，因此必须采用适当的方法来解决拥塞。

拥塞管理的中心内容就是当拥塞发生时如何制定一个资源的调度策略，以决定报文转发的处理次序。

5.1.2 设备支持的拥塞管理方法

对于拥塞管理，一般采用队列技术，使用一个队列算法对流量进行分类，之后用某种优先级别算法将这些流量发送出去。每种队列算法都是用以解决特定的网络流量问题，并对带宽资源的分配、延迟、抖动等有着十分重要的影响。

拥塞管理的处理包括队列的创建、报文的分类、将报文送入不同的队列、队列调度等。

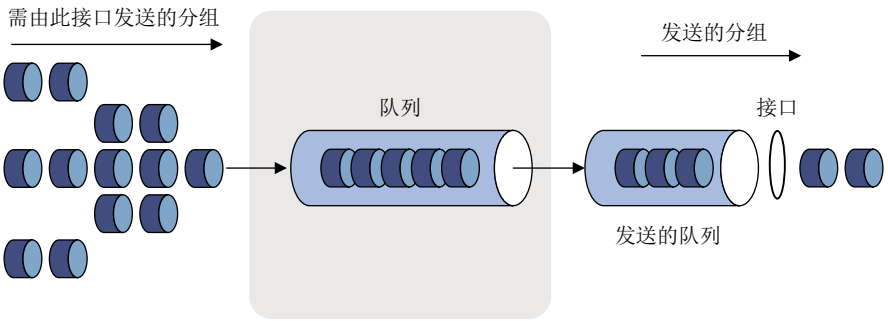
目前，设备支持如下几种队列：

- FIFO 队列
- PQ 队列
- CQ 队列
- WFQ 队列

- CBQ 队列
- RTP 优先队列

5.1.3 FIFO 队列

图5-2 先入先出队列示意图



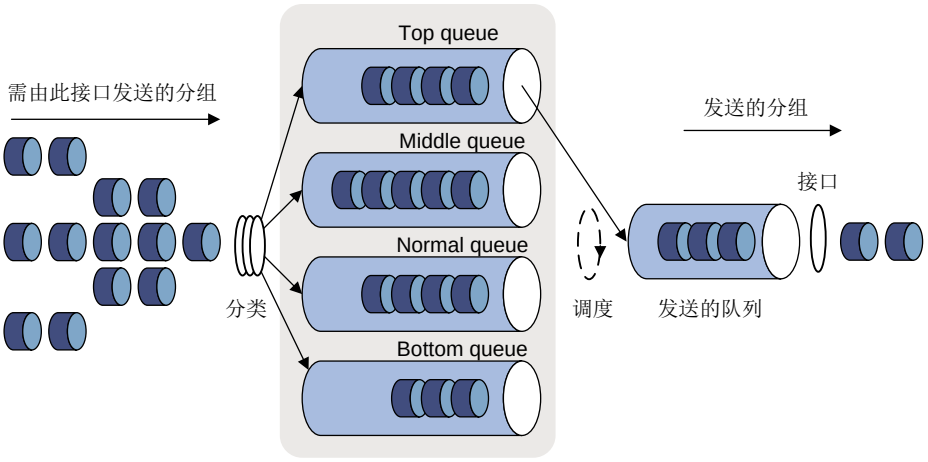
如图 5-2 所示，FIFO 按照时间到达的先后决定分组的转发次序，先进的先出，后进的后出，不需要进行流分类和队列调度，FIFO 关心的只是队列的长度，队列的长度对延迟和丢包率的影响。用户的业务流在某个设备能够获得的资源取决于分组的到达时机及当时的负载情况。Best-Effort 报文转发方式采用的就是 FIFO 的排队策略。

如果设备的每个端口只有一个基于 FIFO 的输入或输出队列，那么恶性的应用可能会占用所有的网络资源，严重影响关键业务数据的传送。所以还需要配置一些其他的队列调度机制与 FIFO 配合对流量进行调度和拥塞控制。

每个队列内部报文的发送次序缺省是 FIFO。

5.1.4 PQ 队列

图5-3 PQ 队列示意图



PQ 队列是针对关键业务应用设计的。关键业务有一个重要的特点，即在拥塞发生时要求优先获得服务以减小响应的延迟。PQ 可以根据网络协议（比如 IP、IPX）、数据流入接口、报文长度、源地址/目的地址等灵活地为数据流指定优先次序。如图 5-3 所示，PQ 的 4 个队列分别为高优先队列（top）、

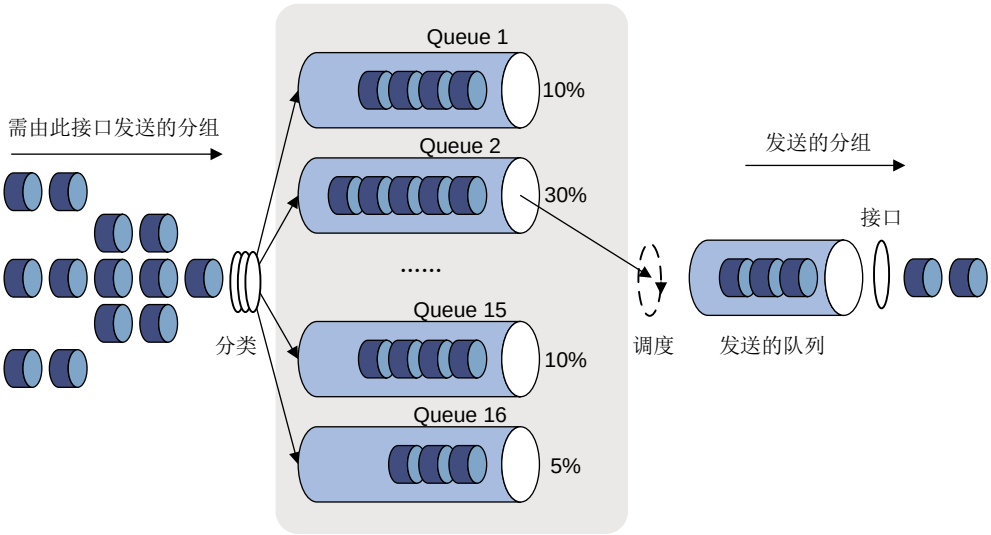
中优先队列（middle）、正常优先队列（normal）和低优先队列（bottom）。所有报文将被分成 4 类，并按所属类别进入 4 个队列中的一个。缺省情况下，数据流进入 normal 队列。每个队列内部又遵循 FIFO 原则。

在队列调度时，PQ 严格按照优先级从高到低的次序，优先发送较高优先级队列中的分组，当较高优先级队列为空时，再发送较低优先级队列中的分组。这样，将关键业务的分组放入较高优先级的队列，将非关键业务的分组放入较低优先级的队列，可以保证关键业务的分组被优先传送，非关键业务的分组在处理关键业务数据的空闲间隙被传送。

PQ 的缺点是：拥塞发生时，如果较高优先级队列中长时间有分组存在，那么低优先级队列中的报文将一直得不到服务。

5.1.5 CQ 队列

图5-4 CQ 队列示意图



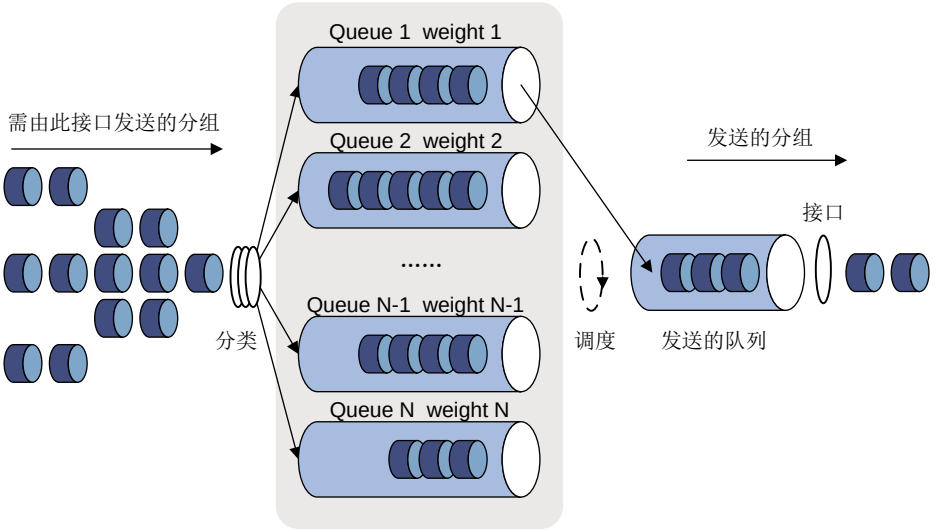
CQ 队列包含 16 个队列，1 到 16 号队列是用户队列，如图 5-4 所示。用户可以配置流分类的规则，指定 16 个队列占用接口或 PVC 带宽的比例关系。

在队列调度时，采用轮询的方式按照预先配置的轮询字节数依次从 1 到 16 号用户队列中取出一定数量的分组发送出去。这样，就可以使不同业务的分组获得不同的带宽，既可以保证关键业务能获得较多的带宽，又不至于使非关键业务得不到带宽。每个队列所占的带宽比例为：该队列的轮询字节数/所有队列的轮询字节数之和。缺省情况下，数据流进入 1 号队列。

CQ 队列的另一个优点是：可根据业务的繁忙程度分配带宽，适用于对带宽有特殊需求的应用。虽然 16 个用户队列的调度是轮询进行的，但对每个队列不是固定地分配服务时间片——如果某个队列为空，那么马上换到下一个队列调度。因此，当没有某些类别的报文时，CQ 调度机制能自动增加现存类别的报文可占的带宽。

5.1.6 WFQ 队列

图5-5 WFQ 队列示意图



在介绍加权公平队列前，先要理解 FQ 队列。FQ 队列是为了公平地分享网络资源，尽可能使所有流的延迟和抖动达到最优而推出的。它照顾了各方面的利益，主要表现在：

- 不同的队列获得公平的调度机会，从总体上均衡各个流的延迟。
- 短报文和长报文获得公平的调度：如果不同队列间同时存在多个长报文和短报文等待发送，应当顾及短报文的利益，让短报文优先获得调度，从而在总体上减少各个流的报文间的抖动。

与 FQ 相比，WFQ 在计算报文调度次序时增加了优先权方面的考虑。从统计上，WFQ 使高优先权的报文获得优先调度的机会多于低优先权的报文。WFQ 能够按流的“会话”信息（协议类型、源和目的 TCP 或 UDP 端口号、源和目的 IP 地址、ToS 域中的优先级位等）自动进行流分类，并且尽可能多地提供队列，以将每个流均匀地放入不同队列中，从而在总体上均衡各个流的延迟。在出队的时候，WFQ 按流的优先级来分配每个流应占有出口的带宽。优先级的数值越小，所得的带宽越少。优先级的数值越大，所得的带宽越多。

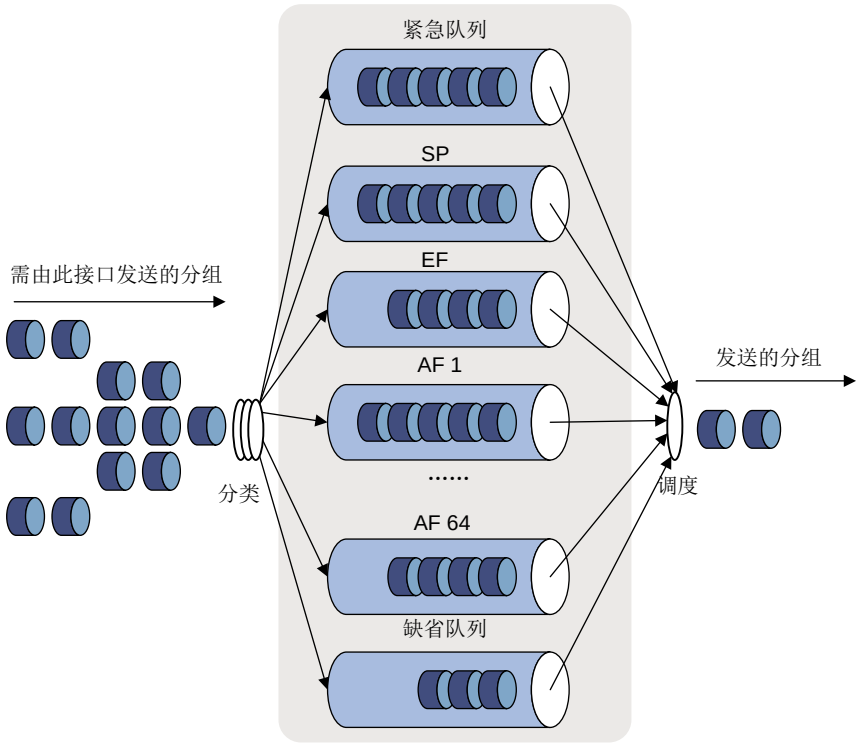
例如：接口中当前共有 5 个流，它们的优先级分别为 0、1、2、3、4，则带宽总配额为所有（流的优先级＋1）的和，即 1＋2＋3＋4＋5＝15。

每个流所占带宽比例为：（自己的优先级数＋1）/（所有（流的优先级＋1）的和）。即每个流可得的带宽分别为：1/15，2/15，3/15，4/15，5/15。

由于 WFQ 在拥塞发生时能均衡各个流的延迟和抖动，所以 WFQ 在一些特殊场合得到了有效的应用。比如在使用 RSVP 协议的保证型业务中，通常就是采用 WFQ 作为调度策略；在流量整形中，也采用 WFQ 调度缓存的报文。

5.1.7 CBQ 队列

图5-6 基于类的队列示意图



CBQ 是对 WFQ 功能的扩展，为用户提供了定义类的支持。在网络拥塞时，CBQ 根据用户定义的类规则对报文进行匹配，并使其进入相应的队列，在入队列之前必须进行拥塞避免机制和带宽限制的检查。在报文出队列时，加权公平调度每个类对应的队列中的报文。

CBQ 包括以下队列：

- 紧急队列：CBQ 提供一个紧急队列，紧急报文入该队列，该队列采用 FIFO 调度，没有带宽限制。
- SP：即严格优先级队列。SP 队列是针对关键业务类型应用设计的。关键业务有一个重要的特点，即在拥塞发生时要求优先获得服务以减小响应的延迟。通过引入 SP 队列，CBQ 可以提供不受带宽检查限制的严格优先服务。最多支持 64 个 SP 队列。
- LLQ：即 EF 队列。如果 CBQ 加权公平对待所有类的队列，实时业务报文（包括语音与视频业务，对延迟比较敏感）就可能得不到及时发送。为此引入一个 EF 队列，为实时业务报文提供严格优先发送服务。LLQ 将严格优先队列机制与 CBQ 结合起来使用，用户在定义类时可以指定其享受严格优先服务，这样的类称作优先类。所有优先类的报文将进入同一个优先队列，在入队列之前需对各类报文进行带宽限制的检查。报文出队列时，将首先发送优先队列中的报文，直到发送完后才发送其他类对应的队列的报文。为了不让其他队列中的报文延迟时间过长，在使用 LLQ 时将会为每个优先类指定可用最大带宽，该带宽值用于拥塞发生时监管流量。如果拥塞未发生，优先类允许使用超过分配的带宽。如果拥塞发生，优先类超过分配带宽的数据包将被丢弃。最多支持 64 个 EF 队列。
- BQ：即 AF 队列。为 AF 业务提供严格、精确的带宽保证，并且保证各类 AF 业务之间按一定的比例关系进行队列调度。最多支持 64 个 AF 队列。

- 缺省队列：一个 WFQ 队列，用来支撑 BE 业务，使用接口剩余带宽进行发送。

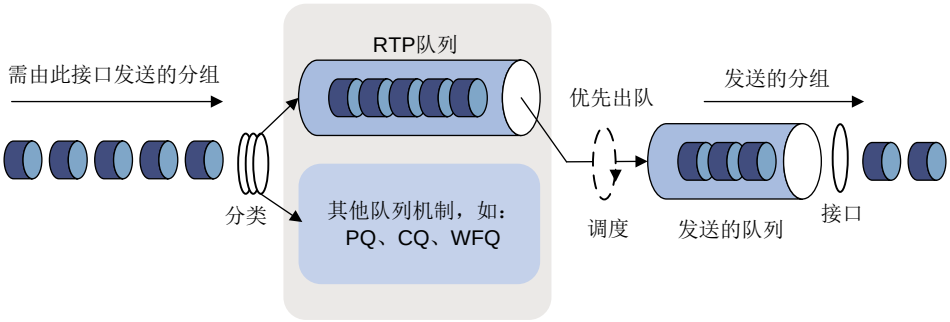
系统在为报文匹配规则时，规则如下：

- 先匹配优先类，然后再匹配其他类；
- 对多个优先类，按照配置顺序逐一匹配；
- 对其他类，也是按照配置顺序逐一匹配；
- 对类中多个规则，按照配置顺序逐一匹配。

5.1.8 RTP 优先队列

RTP 优先队列是一种保证实时业务（包括语音与视频业务）服务质量的简单的队列技术。其原理就是将承载语音或视频的 RTP 报文送入高优先级队列，使其得到优先发送，保证时延和抖动降低为最低限度，从而保证了语音或视频这种对时延敏感业务的服务质量。

图5-7 RTP 优先队列示意图



如图 5-7 所示，RTP 优先队列将 RTP 报文送入一个具有较高优先级的队列。RTP 报文是端口号在一定范围内为偶数的UDP报文，端口号的范围可以配置。RTP优先队列可以同其他队列（包括FIFO、PQ、CQ 和 WFQ）结合使用，而它的优先级是最高的。

5.1.9 拥塞管理技术的对比

设备上提供了以上拥塞管理技术，突破了传统 IP 设备的单一 FIFO 拥塞管理策略，提供了强大的 QoS 能力，使得 IP 设备可以满足不同业务所要求的不同服务质量的要求。为了用户更好地利用拥塞管理技术，现对各种队列技术做一比较。

表5-1 拥塞管理技术对比

类型	队列数	优点	缺点
FIFO	1	• 不需要配置，易于使用 • 处理简单，延迟小	• 所有的报文均进入一个“先进先出”的队列，发送报文所占用的带宽、延迟时间、丢失的概率均由报文到达队列的先后顺序决定 • 对不匹配的数据源（即没有流控机制的流，如 UDP 报文发送）无约束力，不匹配的数据源会造成匹配的数据源（如 TCP 报文发送）带宽受损失 • 对时间敏感的实时应用（如 VoIP）的延迟得不到保证

类型	队列数	优点	缺点
PQ	4	可对优先级高的业务提供绝对的优先，对时间敏感的实时应用（如VoIP）的延迟可以得到保证	• 需配置，处理速度慢 • 如果较高优先级队列中总有报文存在，那么低优先级队列中的报文将一直得不到服务
CQ	16	• 可对不同业务的报文按带宽比例分配带宽 • 当没有某些类别的报文时，能自动增加现存类别的报文可占的带宽	需配置，处理速度慢
WFQ	可配置	• 配置容易 • 可以保护配合（交互）的数据源（如 TCP 报文发送）的带宽 • 可以减小抖动 • 可以减小数据量小的交互式应用的延迟 • 可以为不同优先级的流分配不同的带宽 • 当流的数目减少时，能自动增加现存流可占的带宽	处理速度比FIFO要慢
CBQ	可配置	• 可以对数据根据灵活、多样的分类规则进行划分，分别为 EF（加速转发）、AF（确保转发）、BE（尽力转发）业务提供不同的队列调度机制 • 可以为 AF 业务提供严格、精确的带宽保证，并且保证各类 AF 业务之间根据权值按一定的比例关系进行队列调度 • 可以为 EF 业务提供绝对优先的队列调度，确保实时数据的时延满足要求；同时通过对高优先级数据流量的限制，克服了 PQ 的低优先级队列可能得不到服务的弊病 • 对于尽力转发的缺省类数据，提供 WFQ 队列调度	系统开销比较大
RTP	1	• 保证了实时业务优先处理 • 在入队前进行流量监管的处理，避免出现其他队列得不到处理的情况	适用范围较窄，一般仅适用于对时延敏感的业务（如语音和视频业务）


说明

如果流量突发较大，可以通过增加队列长度的方法来改善队列调度的准确率。

5.2 配置先进先出队列的长度

FIFO 是接口缺省使用的队列调度机制，可以通过配置命令改变其队列长度。

5.2.1 配置接口先进先出队列的长度

1. 配置限制和指导

在子接口上配置 FIFO 队列时，接口上需要开启接口限速功能以保证队列功能生效。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置先进先出队列的长度。

qos fifo queue-length *queue-length*

缺省情况下，FIFO 队列的长度为 75。

如果流量突发较大，可以通过增加队列长度的方法来改善队列调度的准确率。

5.2.2 配置 PVC 先进先出队列的长度

(1) 进入系统视图。

system-view

(2) 请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(3) 配置先进先出队列的长度。

qos fifo queue-length *queue-length*

缺省情况下，FIFO 队列的长度为 75。

如果流量突发较大，可以通过增加队列长度的方法来改善队列调度的准确率。

5.2.3 配置 PW 先进先出队列的长度

(1) 进入系统视图。

system-view

(2) 进入 PW 视图。请选择其中一项进行配置。

- 请依次执行以下命令进入交叉连接 PW 视图。

  xconnect-group *group-name*

  connection *connection-name*

  peer *ip-address* **pw-id** *pw-id* [**in-label** *label-value* **out-label** *label-value*] [**pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI LDP PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling ldp

peer *ip-address* [**pw-id** *pw-id*] [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI 静态 PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling static

  peer *ip-address* [**pw-id** *pw-id*] [**in-label** *label-value* **out-label** *label-value* [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *]

(3) 配置先进先出队列的长度。

qos fifo queue-length *queue-length*

缺省情况下，FIFO 队列的长度为 75。

如果流量突发较大，可以通过增加队列长度的方法来改善队列调度的准确率。

5.3 配置优先队列

5.3.1 功能简介

可以给一个优先列表定义多条规则，然后把该组规则应用在某接口或 PVC 上。在进行流分类时，数据流按照配置顺序进行匹配，如果匹配上某规则，则进入相应的队列，匹配结束；如果分组不与任何规则匹配，则进入缺省队列。

5.3.2 配置限制和指导

配置优先队列时需要注意的是：

- 若指定的接口是 Tunnel 接口、子接口、三层聚合接口、HDLC 捆绑接口，或是封装了 PPPoE、PPPoA、PPPoEoA、PPPoFR、MPoFR（FR 接口未开启帧中继流量整形功能）协议的 VT、Dialer 接口，则接口需要开启接口限速功能以保证队列生效。关于接口限速的详细介绍，请参见“4.4.1 配置接口限速”。
- 将一组优先列表应用到接口或 PVC 上。对于同一个接口或 PVC，若优先队列的应用命令的重复使用，则最新的配置生效。

5.3.3 配置接口的优先级队列

(1) 进入系统视图。

system-view

(2) 配置优先列表。请至少选择其中一项进行配置。

- 配置基于协议的分类规则优先列表。

  qos pql *pql-index* **protocol** { **ip** | **ipv6** } [*queue-key key-value*] **queue** { **bottom** | **middle** | **normal** | **top** }

- 配置基于接口的分类规则优先列表。

  qos pql *pql-index* **inbound-interface** *interface-type interface-number* **queue** { **bottom** | **middle** | **normal** | **top** }

- 配置基于本地优先级的分类规则优先列表。

  qos pql *pql-index* **local-precedence** *local-precedence-list* **queue { bottom | middle | normal | top }**

- 配置基于 MPLS EXP 优先级的分类规则优先列表。

  qos pql *pql-index* **protocol mpls exp** *exp-list* **queue** { **bottom** | **middle** | **normal** | **top** }

(3) （可选）配置缺省队列。

qos pql *pql-index* **default-queue** { **bottom | middle** | **normal** | **top** }

缺省情况下，缺省队列为 **normal**。

本配置用来指明不匹配规则的数据包的入队队列。

(4) （可选）配置队列长度。

qos pql *pql-index* **queue { bottom | middle** | **normal** | **top** } **queue-length** *queue-length*

缺省情况下，高优先队列的缺省长度值为 20，中优先队列的缺省长度值为 40，正常优先队列的缺省长度值为 60，低优先队列的缺省长度值为 80。

(5) 进入接口视图。

interface *interface-type interface-number*

(6) 应用优先列表。

qos pq pql *pql-index*

缺省情况下，接口使用 FIFO 队列。

5.3.4 配置 PVC 的优先队列

(1) 进入系统视图。

system-view

(2) 配置优先列表。请至少选择其中一项进行配置。

- 配置基于协议的分类规则优先列表。

  qos pql *pql-index* **protocol** { **ip** | **ipv6** } [*queue-key key-value*] **queue { bottom | middle** | **normal** | **top** }

- 配置基于接口的分类规则优先列表。

  qos pql *pql-index* **inbound-interface** *interface-type interface-number* **queue** { **bottom | middle** | **normal** | **top** }

- 配置基于本地优先级的分类规则优先列表。

  qos pql *pql-index* **local-precedence** *local-precedence-list* **queue { bottom | middle** | **normal** | **top** }

- 配置基于 MPLS EXP 优先级的分类规则优先列表。

  qos pql *pql-index* **protocol mpls exp** *exp-list* **queue** { **bottom** | **middle** | **normal** | **top** }

(3) （可选）配置缺省队列。

qos pql *pql-index* **default-queue** { **bottom | middle** | **normal** | **top** }

缺省情况下，缺省队列为 **normal**。

本配置用来指明不匹配规则的数据包的入队队列。

(4) （可选）配置队列长度。

qos pql *pql-index* **queue { bottom | middle | normal | top } queue-length** *queue-length*

缺省情况下，高优先队列的缺省长度值为 20，中优先队列的缺省长度值为 40，正常优先队列的缺省长度值为 60，低优先队列的缺省长度值为 80。

(5) 请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(6) 应用优先列表。

qos pq pql *pql-index*

缺省情况下，PVC 使用 FIFO 队列。

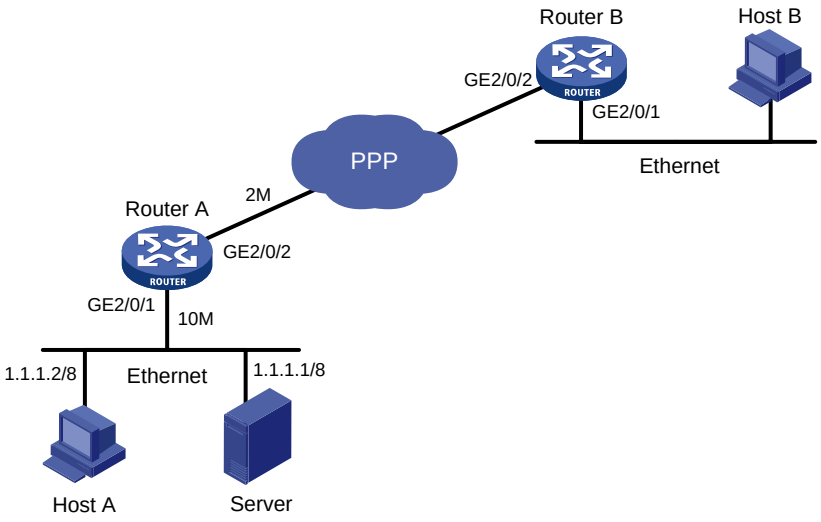
5.3.5 优先队列典型配置举例

1. 配置需求

如图所示，Server 和 Host A 通过 Device A 向 Host B 发送数据（其中 Server 发送关键业务数据，Host A 发送非关键业务数据）时，由于 Device A 入接口 GigabitEthernet2/0/1 的速率大于出接口 GigabitEthernet2/0/2 的速率，在 GigabitEthernet2/0/2 接口处可能发生拥塞，导致丢包。要求在网络拥塞时保证 Server 发送的关键业务数据得到优先处理。

2. 组网图

图5-8 优先队列配置组网图



3. 配置步骤

配置 ACL 规则列表，分别匹配来源于 Server 和 Host A 的报文。

```
[DeviceA] acl basic 2001
```

```
[DeviceA-acl-ipv4-basic-2001] rule permit source 1.1.1.1 0
[DeviceA] acl basic 2002
[DeviceA-acl-ipv4-basic-2002] rule permit source 1.1.1.2 0
```

配置优先队列规则组，使得网络拥塞发生时，源自 Server 的报文能够进入 PQ 的 top 队列缓存，源自 Host A 的报文能够进入 bottom 队列缓存，并且设定 top 队列的最大队列长度为 50、bottom 队列的最大队列长度为 100。

```
[DeviceA] qos pql 1 protocol ip acl 2001 queue top
[DeviceA] qos pql 1 protocol ip acl 2002 queue bottom
[DeviceA] qos pql 1 queue top queue-length 50
[DeviceA] qos pql 1 queue bottom queue-length 100
```

在接口 GigabitEthernet2/0/2 上启用优先队列规则组 1。

```
[DeviceA] interface gigabitethernet 2/0/2
[DeviceA-GigabitEthernet2/0/2] qos pq pql 1
```

5.4 配置定制队列

5.4.1 功能简介

定制列表共可分为 16 个组（1～16），每个组指明了什么样的分组进入什么样的队列、各队列的长度和每次轮询各队列所能连续发送的字节数等信息。

5.4.2 配置限制和指导

配置定制队列时需要注意的是：

- 对于同一个接口或 PVC，若定制队列的应用命令的重复使用，则最新的配置生效。

5.4.3 配置接口的定制队列

(1) 进入系统视图。

system-view

(2) 配置定制列表。请至少选择其中一项进行配置。

- 配置基于协议的分类规则定制列表。

  qos cql *cql-index* **protocol** { **ip** | **ipv6** } [*queue-key key-value*] **queue** *queue-id*

- 配置基于接口的分类规则定制列表。

  qos cql *cql-index* **inbound-interface** *interface-type interface-number* **queue** *queue-id*

- 配置基于本地优先级的分类规则定制列表。

  qos cql *cql-index* **local-precedence** *local-precedence-list* **queue** *queue-id*

- 配置基于 MPLS EXP 优先级的分类规则定制列表。

  qos cql *cql-index* **protocol mpls exp** *exp-list* **queue** *queue-id*

(3) （可选）配置缺省队列。

qos cql *cql-index* **default-queue** *queue-id*

缺省情况下，缺省队列号为 1。

本配置指明不匹配规则的数据包的入队队列。

(4) （可选）配置队列长度。

qos cql *cql-index* **queue** *queue-id* **queue-length** *queue-length*

缺省情况下，队列长度值是 20。

(5) （可选）配置各队列每次轮询所发送数据包的字节数。

qos cql *cql-index* **queue** *queue-id* **serving** *byte-count*

缺省情况下，发送数据包的字节数为 1500。

(6) 进入接口视图。

interface *interface-type interface-number*

(7) 应用定制列表。

qos cq cql *cql-index*

缺省情况下，接口使用 FIFO 队列。

5.4.4 配置 PVC 的定制队列

(1) 进入系统视图。

system-view

(2) 配置定制列表。请至少选择其中一项进行配置。

- 配置基于协议的分类规则定制列表。

  qos cql *cql-index* **protocol** { **ip** | **ipv6** } [*queue-key key-value*] **queue** *queue-id*

- 配置基于接口的分类规则定制列表。

  qos cql *cql-index* **inbound-interface** *interface-type interface-number* **queue** *queue-id*

- 配置基于本地优先级的分类规则定制列表。

  qos cql *cql-index* **local-precedence** *local-precedence-list* **queue** *queue-id*

- 配置基于 MPLS EXP 优先级的分类规则定制列表。

  qos cql *cql-index* **protocol mpls exp** *exp-list* **queue** *queue-id*

(3) （可选）配置缺省队列。

qos cql *cql-index* **default-queue** *queue-id*

缺省情况下，缺省队列号为 1。

本配置指明不匹配规则的数据包的入队队列。

(4) （可选）配置队列长度。

qos cql *cql-index* **queue** *queue-id* **queue-length** *queue-length*

缺省情况下，队列长度值是 20。

(5) （可选）配置各队列每次轮询所发送数据包的字节数。

qos cql *cql-index* **queue** *queue-id* **serving** *byte-count*

缺省情况下，发送数据包的字节数为 1500。

(6) 请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(7) 应用定制列表。

qos cq cql *cql-index*

缺省情况下，PVC 使用 FIFO 队列。

5.5 配置加权公平队列

5.5.1 配置限制和指导

配置加权公平队列时需要注意的是：

- 若指定的接口为Tunnel接口、子接口、三层聚合接口、HDLC捆绑接口，或是封装了PPPoE、PPPoA、PPPoEoA、PPPoFR、MPoFR（FR 接口未开启帧中继流量整形功能）协议的 VT、Dialer 接口，则接口需要开启接口限速功能以保证队列生效。关于接口限速的详细介绍，请参见“4.4.1 配置接口限速”。
- 当接口或 PVC 没有使用 WFQ 策略时，使用 **qos wfq** 命令可以使接口或 PVC 使用 WFQ 策略，同时指定 WFQ 的参数。如果接口或 PVC 已经使用了 WFQ 策略，使用 **qos wfq** 命令可以修改 WFQ 的参数。

5.5.2 配置接口的加权公平队列

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置加权队列。

qos wfq [**dscp** | **precedence**] [**queue-length** *max-queue-length* | **queue-number** *total-queue-number*] *

缺省情况下，接口上未配置 WFQ 队列。

5.5.3 配置 PVC 的加权公平队列

(1) 进入系统视图。

system-view

(2) 请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(3) 配置加权队列。

qos wfq [**dscp** | **precedence**] [**queue-length** *max-queue-length* | **queue-number** *total-queue-number*] *

缺省情况下，PVC 上未配置 WFQ 队列。

5.5.4 配置 PW 加权公平队列

(1) 进入系统视图。

system-view

(2) 进入 PW 视图。请选择其中一项进行配置。

- 请依次执行以下命令进入交叉连接 PW 视图。

  xconnect-group *group-name*

  connection *connection-name*

  peer *ip-address* **pw-id** *pw-id* [**in-label** *label-value* **out-label** *label-value*] [**pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI LDP PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling ldp

  peer *ip-address* [**pw-id** *pw-id*] [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *

- 请依次执行以下命令进入 VSI 静态 PW 视图。

  vsi *vsi-name* [**hub-spoke**]

  pwsignaling static

  peer *ip-address* [**pw-id** *pw-id*] [**in-label** *label-value* **out-label** *label-value* [**hub** | **no-split-horizon** | **pw-class** *class-name* | **tunnel-policy** *tunnel-policy-name*] *]

(3) 配置加权队列。

qos wfq [**dscp** | **precedence**] [**queue-length** *max-queue-length* | **queue-number** *total-queue-number*] *

缺省情况下，PW 上未配置 WFQ 队列。

5.6 配置RTP优先队列

5.6.1 配置指导和限制

配置 RTP 优先队列时：

- 若指定的接口为Tunnel接口、子接口、三层聚合接口、HDLC捆绑接口，或是封装了PPPoE、PPPoA、PPPoEoA、PPPoFR、MPoFR（FR 接口未开启帧中继流量整形功能）协议的 VT、Dialer 接口，则接口需要开启接口限速功能以保证队列生效。关于接口限速的详细介绍，请参见“4.4.1 配置接口限速”。
- RTP 和 CBQ 互斥，不能结合使用。

5.6.2 配置接口的 RTP 优先队列

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置 RTP 队列。

qos rtpq start-port *first-rtp-port-number* **end-port** *last-rtp-port-number* **bandwidth** *bandwidth* [**cbs** *cbs*]

缺省情况下，接口上未开启 RTP 队列特性。

5.6.3 配置 PVC 的 RTP 优先队列

(1) 进入系统视图。

system-view

(2) 进入 PVC 视图。请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(3) 配置 RTP 队列。

qos rtpq start-port *first-rtp-port-number* **end-port** *last-rtp-port-number* **bandwidth** *bandwidth* [**cbs** *cbs*]

缺省情况下，PVC 上未开启 RTP 队列特性。

5.7 配置报文信息预提取功能

1. 功能简介

对于 Tunnel 接口，如果到达对应物理接口的 IP 数据报文已经进行了处理，比如，Tunnel 接口进行了 GRE 封装，此时 QoS 处理的是 GRE 封装后的 IP 数据报文，QoS 无法识别出原始报文的 IP 数据，无法基于原始报文信息对报文进行分类。

开启报文信息预提取功能后，系统在逻辑接口获取原始报文的 IP 数据，并在物理接口应用此 IP 数据，可以基于原始报文信息进行分类，从而进行各种 QoS 处理。

Tunnel 接口的具体内容请参见“三层技术-IP 业务配置指导”中的“隧道”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface tunnel *number* [**mode** { **ds-lite-aftr** | **evi** [**ipv6**] | **gre** [**ipv6**] | **ipv4-ipv4** | **ipv6** | **ipv6-ipv4** [**6to4** | **auto-tunnel** | **isatap**] | **mpls-te** }]

(3) 开启报文信息预提取功能。

qos pre-classify

缺省情况下，报文信息预提取功能处于关闭状态。

5.8 开启QoS队列增强功能

1. 配置限制和指导

本命令仅对 RT-FIP-680 单板生效，并且配置后仅对单板上的固定端口生效，对接口模块上的端口不生效。开启 QoS 队列增强功能，可以提升 RT-FIP-680 单板上的固定端口处理 QoS 队列调度的性能。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 开启 QoS 队列增强功能。

（独立运行模式）

qos-queue enhanced slot *slot-number*

（IRF 模式）

qos-queue enhanced chassis *chassis-number* **slot** *slot-number*

缺省情况下，QoS 队列增强未开启。

5.9 拥塞管理显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示拥塞管理各种队列的运行情况，通过查看显示信息验证配置的效果。

表5-2 拥塞管理的显示和维护

操作	命令
显示定制列表的内容	**display qos cql** [*cql-index*]
显示指定策略中指定类及与类关联的流行为的配置信息	**display qos policy** { **system-defined** | **user-defined** } [*policy-name* [**classifier** *classifier-name*]]
显示接口或PVC上策略的配置信息和运行情况	（独立运行模式） **display qos policy interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] [**slot** *slot-number*] [**inbound** | **outbound**] （IRF模式） **display qos policy interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] [**chassis** *chassis-number* **slot** *slot-number*] [**inbound** | **outbound**]
显示优先列表的内容	**display qos pql** [*pql-index*]
显示接口、PVC或PW上基于类的队列配置信息和运行情况	**display qos queue cbq** { **interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] | **l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*] }
显示接口、PVC定制队列配置信息和运行情况	**display qos queue cq interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]]

操作	命令
显示接口、PVC或PW上先进先出队列配置信息和运行情况	**display qos queue fifo** { **interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] | **l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*] }
显示接口或PVC上所有队列配置情况和统计信息	**display qos queue interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]]
显示PW上所有队列配置情况和统计信息	**display qos queue l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*]
显示接口、PVC优先级队列配置信息和运行情况	**display qos queue pq interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]]
显示接口、PVC实时传输协议队列配置信息和运行情况	**display qos queue rtpq interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]]
显示接口、PVC或PW上加权公平队列的配置统计信息	**display qos queue wfq** { **interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]] | **l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*] }
显示设备配置的流行为信息	**display traffic behavior** { **system-defined** | **user-defined** } [*behavior-name*]
显示设备配置的类信息	**display traffic classifier** { **system-defined** | **user-defined** } [*classifier-name*]
清除PW下QoS的统计信息	**reset qos statistics l2vpn-pw** [**peer** *ip-address* **pw-id** *pw-id*]

6 拥塞避免

6.1 拥塞避免简介

拥塞避免是一种流量控制机制，它通过监视网络资源（如队列或内存缓冲区）的使用情况，在拥塞产生或有加剧的趋势时主动丢弃报文，通过调整网络的流量来避免网络过载。设备在丢弃报文时，需要与源端的流量控制动作（比如 TCP 流量控制）相配合，调整网络的流量到一个合理的负载状态。丢包策略和源端的流量控制相结合，可以使网络的吞吐量和利用效率最大化，并且使报文丢弃和延迟最小化。

6.1.1 传统的丢包策略

传统的丢包策略采用尾部丢弃（Tail-Drop）的方法。当队列的长度达到最大值后，所有新到来的报文都将被丢弃。

这种丢弃策略会引发 TCP 全局同步现象：当队列同时丢弃多个 TCP 连接的报文时，将造成多个 TCP 连接同时进入拥塞避免和慢启动状态以降低并调整流量，而后又会在某个时间同时出现流量高峰。如此反复，使网络流量忽大忽小，网络不停震荡。

6.1.2 RED 与 WRED

为避免 TCP 全局同步现象，可使用 RED 或 WRED。

RED 和 WRED 通过随机丢弃报文避免了 TCP 的全局同步现象，使得当某个 TCP 连接的报文被丢弃、开始减速发送的时候，其他的 TCP 连接仍然有较高的发送速度。这样，无论什么时候，总有 TCP 连接在进行较快的发送，提高了线路带宽的利用率。

在 RED 类算法中，为每个队列都设定上限和下限，对队列中的报文进行如下处理：

- 当队列的长度小于下限时，不丢弃报文；
- 当队列的长度超过上限时，丢弃所有到来的报文；
- 当队列的长度在上限和下限之间时，开始随机丢弃到来的报文。队列越长，丢弃概率越高，但有一个最大丢弃概率。

直接采用队列的长度和上限、下限比较并进行丢弃，将会对突发性的数据流造成不公正的待遇，不利于数据流的传输。WRED 采用平均队列和设置的队列上限、下限比较来确定丢弃的概率。

队列平均长度既反映了队列的变化趋势，又对队列长度的突发变化不敏感，避免了对突发性数据流的不公正待遇。

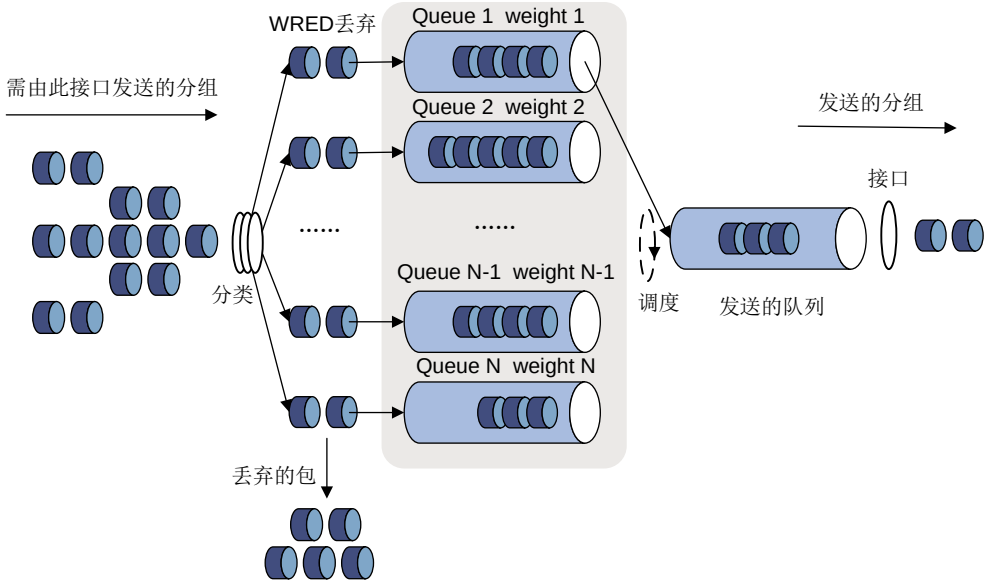
当队列机制采用 WFQ 时，可以为不同优先级的报文设定计算队列平均长度时的指数、上限、下限、丢弃概率，从而对不同优先级的报文提供不同的丢弃特性。

当队列机制采用 FIFO、PQ、CQ 时，可以为每个队列设定计算队列平均长度时的指数、上限、下限、丢弃概率，为不同类别的报文提供不同的丢弃特性。

6.1.3 WRED 和队列机制的关系

WRED 和队列机制的关系如下图所示。

图6-1 WRED 和队列机制关系示意图



当 WRED 和 WFQ 配合使用时，可以实现基于流的 WRED。在进行分类的时候，不同的流有自己的队列，对于流量小的流，由于其队列长度总是比较小，所以丢弃的概率将比较小。而流量大的流将会有较大的队列长度，从而丢弃较多的报文，保护了流量较小的流的利益。

6.1.4 拥塞通知

WRED 采用的丢弃报文的动作虽然缓解了拥塞对网络的影响，但将报文从发送端转发到被丢弃位置之间所消耗的网络资源已经被浪费了。因此，在拥塞发生时，如果能将网络的拥塞状况告知发送端，使其主动降低发送速率或减小报文窗口大小，便可以更高效的利用网络资源。

RFC 2481 定义了一种端到端的拥塞通知机制，称为 ECN 功能。ECN 功能利用 IP 报文头中的 DS 域来标记报文传输路径上的拥塞状态。支持该功能的终端设备可以通过报文内容判断出传输路径上发生了拥塞，从而调整报文的发送方式，避免拥塞加剧。ECN 功能对 IP 报文头中 DS 域的最后两个比特位（称为 ECN 域）进行了如下定义：

- 比特位 6 用于标识发送端设备是否支持 ECN 功能，称为 ECT 位（ECN-Capable Transport）
- 比特位 7 用于标识报文在传输路径上是否经历过拥塞，称为 CE 位（Congestion Experienced）

说明

- 关于 DS 域的介绍，请参见“10.3.1 IP 优先级和 DSCP 优先级”。
- 在实际应用中，设备将 ECT 位为 1、CE 位为 0 的报文，以及 ECT 位为 0，CE 位为 1 的报文都识别为由支持 ECN 功能的终端发出的报文。

在设备上开启 ECN 功能后，拥塞管理功能将按如下方式对报文进行处理：

- 如果队列长度小于下限，不丢弃报文，也不对 ECN 域进行识别和标记。
- 如果队列长度在上限和下限之间，当设备根据丢弃概率计算出需要丢弃某个报文时，将检查该报文的 ECN 域。如果 ECN 域显示该报文由支持 ECN 的终端发出，设备会将报文的 ECT

位和 CE 位都标记为 1，然后转发该报文；如果 ECN 域显示报文传输路径中已经经历过拥塞（即 ECT 和 CE 位都为 1），则设备直接转发该报文，不对 ECN 域进行重新标记；如果 ECT 位和 CE 位都为 0，设备会将该报文丢弃。

- 如果队列长度超过上限，无论报文是否由支持 ECN 的终端发出，都将会被设备丢弃。

6.1.5 WRED 的配置方式

配置 WRED 参数方式：在接口或 PVC 上配置 WRED 的各种参数，并开启 WRED。

6.1.6 WRED 的参数说明

在进行 WRED 配置时，需要事先确定如下参数：

- 队列上限和下限：当队列平均长度小于下限时，不丢弃报文。当队列平均长度在上限和下限之间时，设备随机丢弃报文，队列越长，丢弃概率越高。当队列平均长度超过上限时，丢弃所有到来的报文。
- 丢弃优先级：在进行报文丢弃时参考的参数，0 对应绿色报文、1 对应黄色报文、2 对应红色报文，红色报文将被优先丢弃。
- 计算平均队列长度的指数：指数越大，计算平均队列长度时对队列的实时变化越不敏感。计算队列平均长度的公式为：平均队列长度=（以前的平均队列长度×（$1-1/2^n$））＋（当前队列长度×（$1/2^n$））。其中 n 表示指数。
- 计算丢弃概率的分母：配置接口和 PVC 的 WRED 参数时，在计算丢弃概率的公式中作为分母。取值越大，计算出的丢弃概率越小。
- 丢弃概率：配置接口应用 WRED 表时，使用百分数的形式表示丢弃报文的概率，取值越大，报文被丢弃的机率越大。

6.2 配置接口的WRED参数

6.2.1 配置限制和指导

qos wred enable 命令在设备接口配置时需要先在接口上应用 WFQ 队列。

6.2.2 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 开启 WRED。

qos wred [**dscp** | **ip-precedence**] **enable**

缺省情况下，队列丢弃方法为尾丢弃。

(4) （可选）配置计算平均队列长度的指数。

qos wred weighting-constant *exponent*

缺省情况下，WRED 计算平均队列长度的指数为 9。

(5) （可选）配置各优先级对应的参数。

qos wred { **ip-precedence** *ip-precedence* | **dscp** *dscp-value* } **low-limit** *low-limit* **high-limit** *high-limit* **discard-probability** *discard-prob*

缺省情况下，下限缺省值为 10，上限缺省值为 30，丢弃概率的分母缺省值为 10。

6.2.1 配置接口 WRED 参数典型配置举例

1. 组网需求

- 在接口 GigabitEthernet2/0/1 上配置基于 IP 优先级的 WRED。
- 配置 IP 优先级为 3 的报文的队列下限为 20、上限为 40、丢弃概率分母为 15。
- 配置计算平均队列长度的指数为 6。

2. 配置步骤

进入系统视图。

```
<Sysname> system-view
```

进入接口视图。

```
[Sysname] interface gigabitethernet 2/0/1
```

开启基于 IP 优先级的 WRED。

```
[Sysname-GigabitEthernet2/0/1] qos wred ip-precedence enable
```

配置优先级为 3 的报文的队列下限为 20，上限为 40，丢弃概率分母为 15。

```
[Sysname-GigabitEthernet2/0/1] qos wred ip-precedence 3 low-limit 20 high-limit 40
discard-probability 15
```

配置计算平均队列长度的指数为 6。

```
[Sysname-GigabitEthernet2/0/1] qos wred weighting-constant 6
```

6.3 配置PVC的WRED参数

1. 配置限制和指导

qos wred enable 命令在设备接口配置时需要先在接口上应用 WFQ 队列。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 请依次执行以下命令进入 PVC 视图。

interface atm *interface-number*

pvc *vpi/vci*

(3) 开启 WRED。

qos wred [**dscp** | **ip-precedence**] **enable**

缺省情况下，队列丢弃方法为尾丢弃。

(4) （可选）配置计算平均队列长度的指数。

qos wred weighting-constant *exponent*

缺省情况下，WRED 计算平均队列长度的指数为 9。

(5) （可选）配置各优先级对应的参数。

qos wred { **ip-precedence** *ip-precedence* | **dscp** *dscp-value* } **low-limit** *low-limit* **high-limit** *high-limit* **discard-probability** *discard-prob*

缺省情况下，下限缺省值为 10，上限缺省值为 30，丢弃概率的分母缺省值为 10。

6.4 拥塞避免显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以显示配置后 WRED 的运行情况，通过查看显示信息验证配置的效果。

表6-1 拥塞避免显示和维护

操作	命令
显示接口或PVC的WRED配置情况和统计信息	**display qos wred interface** [*interface-type interface-number* [**pvc** { *pvc-name* | *vpi/vci* }]]

7 流量过滤

7.1 流量过滤简介

流量过滤是指对符合流分类的流进行过滤的动作。例如，可以根据网络的实际情况禁止从某个源 IP 地址发送的报文通过。

7.2 流量过滤配置限制和指导

设备支持基于接口、PVC、控制平面和上线用户应用 QoS 策略配置流量过滤。

7.3 配置流量过滤

(1) 进入系统视图。

system-view

(2) 定义类。

a. 创建一个类，并进入类视图。

traffic classifier *classifier-name* [**operator** { **and** | **or** }]

b. 定义匹配数据包的规则。

if-match [**not**] *match-criteria*

缺省情况下，未定义匹配数据包的规则。

具体规则的介绍，请参见“QoS 命令”中的 **if-match** 命令。

c. 退回系统视图。

quit

(3) 定义流行为。

a. 创建一个流行为，并进入流行为视图。

traffic behavior *behavior-name*

b. 配置流量过滤动作。

filter { **deny** | **permit** }

缺省情况下，未配置流量过滤动作。

如果配置了 **filter deny** 命令，则在该流行为视图下配置的其他流行为（除流量统计外）都不会生效。

c. 退回系统视图。

quit

(4) 定义策略。

a. 创建策略并进入策略视图。

qos policy *policy-name*

b. 在策略中为类指定采用的流行为。

classifier *classifier-name* **behavior** *behavior-name*

缺省情况下，未指定类对应的流行为。

c. 退回系统视图。

quit

(5) 应用 QoS 策略。

具体配置请参见“2.7 应用策略”

缺省情况下，未应用 QoS 策略。

(6) （可选）显示流量过滤的相关配置信息。

（独立运行模式）

display traffic behavior user-defined [*behavior-name*] [**slot** *slot-number*]

（IRF 模式）

display traffic behavior user-defined [*behavior-name*] [**chassis** *chassis-number* **slot** *slot-number*]

7.4 流量过滤典型配置举例

7.4.1 流量过滤基本组网配置举例

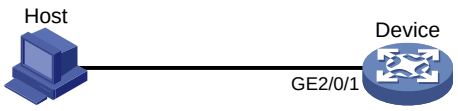
1. 组网需求

Host 通过接口 GigabitEthernet2/0/1 接入设备 Device。

配置流量过滤功能，对接口 GigabitEthernet2/0/1 接收的目的端口号等于 21 的 TCP 报文进行丢弃。

2. 组网图

图7-1 流量过滤基本组网图



3. 配置步骤

定义高级 ACL 3000，匹配目的端口号等于 21 的数据流。

```
<Device> system-view
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule 0 permit tcp destination-port eq 21
[Device-acl-ipv4-adv-3000] quit
```

定义类 classifier_1，匹配高级 ACL 3000。

```
[Device] traffic classifier classifier_1
[Device-classifier-classifier_1] if-match acl 3000
[Device-classifier-classifier_1] quit
```

定义流行为 behavior_1，动作为流量过滤（**deny**），对数据包进行丢弃。

```
[Device] traffic behavior behavior_1
[Device-behavior-behavior_1] filter deny
```

```
[Device-behavior-behavior_1] quit
```

定义策略 policy，为类 classifier_1 指定流行为 behavior_1。

```
[Device] qos policy policy
[Device-qospolicy-policy] classifier classifier_1 behavior behavior_1
[Device-qospolicy-policy] quit
```

将策略 policy 应用到端口 GigabitEthernet2/0/1 的入方向上。

```
[Device] interface gigabitethernet 2/0/1
[Device-GigabitEthernet2/0/1] qos apply policy policy inbound
```

8 重标记

8.1 重标记简介

重标记是将报文的优先级或者标志位进行设置，重新定义报文的优先级等。例如，对于 IP 报文来说，可以利用重标记对 IP 报文中的 IP 优先级或 DSCP 值进行重新设置，控制 IP 报文的转发。

重标记动作的配置，可以通过与类关联，将原来报文的优先级或标志位重新进行标记。

8.2 配置重标记

1. 配置限制和指导

设备支持基于接口、PVC、控制平面和上线用户应用 QoS 策略配置重标记。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 定义类。

a. 创建一个类，并进入类视图。

traffic classifier *classifier-name* [**operator** { **and** | **or** }]

b. 定义匹配数据包的规则。

if-match [**not**] *match-criteria*

缺省情况下，未定义匹配数据包的规则。

具体规则的介绍，请参见“QoS 命令”中的 **if-match** 命令。

c. 退回系统视图。

quit

(3) 定义流行为

a. 创建一个流行为，并进入流行为视图。

traffic behavior *behavior-name*

b. 重新标记报文的动作。

具体重标记动作的介绍，请查看“QoS 命令”中的 **remark** 命令。

c. 退回系统视图。

quit

(4) 定义策略。

a. 创建一个策略，并进入策略视图。

qos policy *policy-name*

b. 在策略中为类指定采用的流行为。

classifier *classifier-name* **behavior** *behavior-name*

缺省情况下，未指定类对应的流行为。

c. 退回系统视图。

quit

(5) 应用 QoS 策略。

具体配置请参见“2.7 应用策略”

缺省情况下，未应用 QoS 策略。

(6) （可选）显示重标记的相关配置信息。

（独立运行模式）

display traffic behavior user-defined [*behavior-name*] [**slot** *slot-number*]

（IRF 模式）

display traffic behavior user-defined [*behavior-name*] [**chassis** *chassis-number* **slot** *slot-number*]

8.3 重标记典型配置举例

8.3.1 重标记基本组网配置举例

1. 组网需求

公司企业网通过 Device 实现互连。网络环境描述如下：

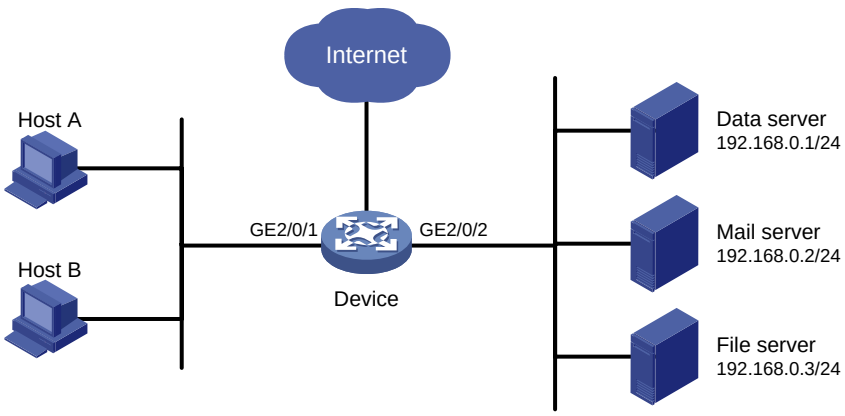
- Host A 和 Host B 通过端口 GigabitEthernet2/0/1 接入 Device；
- 数据库服务器、邮件服务器和文件服务器通过端口 GigabitEthernet2/0/2 接入 Device。

通过配置重标记功能，Device 上实现如下需求：

- 优先处理 Host A 和 Host B 访问数据库服务器的报文；
- 其次处理 Host A 和 Host B 访问邮件服务器的报文；
- 最后处理 Host A 和 Host B 访问文件服务器的报文。

2. 组网图

图8-1 重标记基本组网图



3. 配置步骤

定义高级 ACL 3000，对目的 IP 地址为 192.168.0.1 的报文进行分类。

```
<Device> system-view
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip destination 192.168.0.1 0
[Device-acl-ipv4-adv-3000] quit
```

定义高级 ACL 3001，对目的 IP 地址为 192.168.0.2 的报文进行分类。

```
[Device] acl advanced 3001
[Device-acl-ipv4-adv-3001] rule permit ip destination 192.168.0.2 0
[Device-acl-ipv4-adv-3001] quit
```

定义高级 ACL 3002，对目的 IP 地址为 192.168.0.3 的报文进行分类。

```
[Device] acl advanced 3002
[Device-acl-ipv4-adv-3002] rule permit ip destination 192.168.0.3 0
[Device-acl-ipv4-adv-3002] quit
```

定义类 classifier_dbserver，匹配高级 ACL 3000。

```
[Device] traffic classifier classifier_dbserver
[Device-classifier-classifier_dbserver] if-match acl 3000
[Device-classifier-classifier_dbserver] quit
```

定义类 classifier_mserver，匹配高级 ACL 3001。

```
[Device] traffic classifier classifier_mserver
[Device-classifier-classifier_mserver] if-match acl 3001
[Device-classifier-classifier_mserver] quit
```

定义类 classifier_fserver，匹配高级 ACL 3002。

```
[Device] traffic classifier classifier_fserver
[Device-classifier-classifier_fserver] if-match acl 3002
[Device-classifier-classifier_fserver] quit
```

定义流行为 behavior_dbserver，动作为重标记报文的本地优先级为 4。

```
[Device] traffic behavior behavior_dbserver
[Device-behavior-behavior_dbserver] remark local-precedence 4
[Device-behavior-behavior_dbserver] quit
```

定义流行为 behavior_mserver，动作为重标记报文的本地优先级为 3。

```
[Device] traffic behavior behavior_mserver
[Device-behavior-behavior_mserver] remark local-precedence 3
[Device-behavior-behavior_mserver] quit
```

定义流行为 behavior_fserver，动作为重标记报文的本地优先级为 2。

```
[Device] traffic behavior behavior_fserver
[Device-behavior-behavior_fserver] remark local-precedence 2
[Device-behavior-behavior_fserver] quit
```

定义策略 policy_server，为类指定流行为。

```
[Device] qos policy policy_server
[Device-qospolicy-policy_server] classifier classifier_dbserver behavior behavior_dbserver
[Device-qospolicy-policy_server] classifier classifier_mserver behavior behavior_mserver
[Device-qospolicy-policy_server] classifier classifier_fserver behavior behavior_fserver
[Device-qospolicy-policy_server] quit
```

将策略 policy_server 应用到端口 GigabitEthernet2/0/1 上。

```
[Device] interface gigabitethernet 2/0/1
[Device-GigabitEthernet2/0/1] qos apply policy policy_server inbound
[Device-GigabitEthernet2/0/1] quit
```

9 QPPB

9.1 QPPB简介

QPPB（QoS Policy Propagation Through the Border Gateway Protocol，通过 BGP 传播 QoS 策略）技术是一项通过 BGP 路由策略部署 QoS 的技术，通过基于 BGP 路由的团体列表、AS-Paths list 和 ACL、Prefix list 等属性进行路由分类，对不同的分类应用不同的 QoS 策略。

9.1.1 适用场景

在部署大型复杂网络时，需要执行大量的复杂流分类。如果网络结构不稳定时，配置修改的工作量非常大甚至难以实施。此时可以通过部署 QPPB 减少配置修改的工作量。

应用 QPPB 技术后，BGP 路由发送者通过设置 BGP 属性预先对路由进行分类，在网络拓扑结构发生变化时只需要修改路由发送者上的路由策略就可以改变分类规则。

QPPB 技术适用于如下应用场景：

- 基于目的地址或源地址进行流分类。
- 基于 IBGP 和 EBGP，在同一个自治系统内部或者不同的自治系统之间进行流分类。

9.1.2 QPPB 工作原理

QPPB 技术通过对 BGP 传播的路由属性设置 IP 优先级和 QoS 本地 ID 值，针对具有相同 IP 优先级和 QoS 本地 ID 值的 BGP 路由，应用 QoS 策略，从而实现 QoS 保障。QPPB 工作原理为：

(1) 路由发送者根据路由策略为 BGP 路由设置路由属性。

(2) 当路由接收者收到路由后，根据路由属性为不同的路由设置 IP 优先级和 QoS 本地 ID 值，并将 IP 优先级和 QoS 本地 ID 值添加到路由表中。

(3) 路由接收者接收到报文后，根据报文的源或目的地址查找路由表，获取路由表中的 IP 优先级和 QoS 本地值。

(4) 路由接收者根据 IP 优先级和 QoS 本地 ID 值对报文进行分类和执行分类对应的流行为动作。

9.2 QPPB配置任务简介

QPPB 配置任务如下：

(1) 配置发送端

a. 配置 BGP 基本功能

b. （可选）配置路由策略

(2) 配置接收端

a. 配置 BGP 基本功能

b. 配置路由策略

c. 配置接口的 QPPB 功能并应用 QoS 策略

9.3 配置发送端

路由发送端作为 BGP 路由的发送方，需要根据路由策略设置路由的属性。

9.3.1 配置 BGP 基本功能

具体配置请参见“三层技术-IP 路由配置指导”中的“BGP”。

9.3.2 配置路由策略

根据路由策略对不同的路由信息进行分类，并设置不同路由属性，具体配置请参见“三层技术-IP 路由配置指导”中的“路由策略”。

9.4 配置接收端

9.4.1 配置 BGP 基本功能

具体配置请参见“三层技术-IP 路由配置指导”中的“BGP”。

9.4.2 配置路由策略

根据路由策略匹配发送方设置的路由属性，并对该路由设置 IP 优先级或 QoS 本地 ID 值，具体配置请参见“三层技术-IP 路由配置指导”中的“路由策略”。

9.4.3 配置接口的 QPPB 功能并应用 QoS 策略

1. 配置限制和指导

QoS 策略使用路由策略中设置的 IP 优先级和 QoS 本地 ID 值进行分类，并且为类指定流行为时必须指定 **mode qppb-manipulation** 关键字，否则 QPPB 功能无法生效。关于 QoS 策略的具体配置请参见“2 QoS 策略”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 配置 QPPB 功能。

bgp-policy { **destination** | **source** } { **ip-prec-map** | **ip-qos-map** } *

缺省情况下，未配置 QPPB 功能。

本命令只在流量的入方向生效。

(4) 在接口上应用已创建的 QoS 策略。

qos apply policy *policy-name* { **inbound** | **outbound** }

缺省情况下，未在接口上应用 QoS 策略。

9.5 QPPB典型配置举例

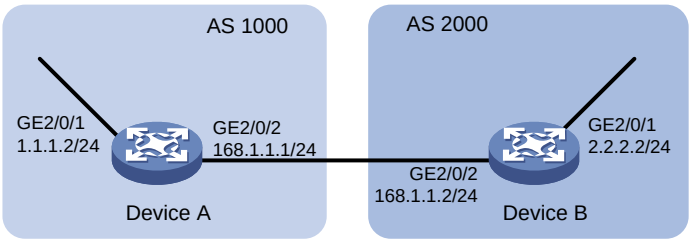
9.5.1 QPPB 在 IPv4 网络中的配置举例

1. 组网需求

如图 9-1 所示，所有设备均运行 BGP 协议。Device B 接收路由，根据路由策略对报文进行 IP 优先级和 QoS 本地 ID 值的设置，并结合 QoS 策略进行 512000kbps 的限速。

2. 组网图

图9-1 QPPB 路由 IPv4 应用配置举例组网图



3. 配置步骤

(1) 配置各接口的 IP 地址（略）

(2) 配置 Device A

配置 BGP 连接。

```
<DeviceA> system-view
[DeviceA] bgp 1000
[DeviceA-bgp] peer 168.1.1.2 as-number 2000
[DeviceA-bgp] peer 168.1.1.2 connect-interface gigabitethernet 2/0/2
[DeviceA-bgp] address-family ipv4
[DeviceA-bgp-ipv4] import-route direct
[DeviceA-bgp-ipv4] peer 168.1.1.2 enable
[DeviceA-bgp-ipv4] quit
[DeviceA-bgp] quit
```

(3) 配置 Device B

配置 BGP 连接。

```
<DeviceB> system-view
[DeviceB] bgp 2000
[DeviceB-bgp] peer 168.1.1.1 as-number 1000
[DeviceB-bgp] peer 168.1.1.1 connect-interface gigabitethernet 2/0/2
[DeviceB-bgp] address-family ipv4
[DeviceB-bgp-ipv4] peer 168.1.1.1 enable
[DeviceB-bgp-ipv4] peer 168.1.1.1 route-policy qppb import
[DeviceB-bgp-ipv4] quit
[DeviceB-bgp] quit
```

配置路由策略。

```
[DeviceB] route-policy qppb permit node 0
[DeviceB-route-policy-qppb-0] apply ip-precedence 1
```

```
[DeviceB-route-policy-qppb-0] apply qos-local-id 3
[DeviceB-route-policy-qppb-0] quit
```

接口开启 QPPB 能力。

```
[DeviceB] interface gigabitethernet 2/0/2
[DeviceB-GigabitEthernet2/0/2] bgp-policy source ip-prec-map ip-qos-map
[DeviceB-GigabitEthernet2/0/2] quit
```

配置 QoS 策略。

```
[DeviceB] traffic classifier qppb
[DeviceB-classifier-qppb] if-match ip-precedence 1
[DeviceB-classifier-qppb] if-match qos-local-id 3
[DeviceB-classifier-qppb] quit
[DeviceB] traffic behavior qppb
[DeviceB-behavior-qppb] car cir 512000 green pass red discard
[DeviceB-behavior-qppb] quit
[DeviceB] qos policy qppb
[DeviceB-qospolicy-qppb] classifier qppb behavior qppb mode qppb-manipulation
[DeviceB-qospolicy-qppb] quit
```

接口应用 QoS 策略。

```
[DeviceB] interface gigabitethernet 2/0/2
[DeviceB-GigabitEthernet2/0/2] qos apply policy qppb inbound
[DeviceB-GigabitEthernet2/0/2] quit
```

4. 验证配置

查看 Device B 相关路由是否生效。

```
[DeviceB] display bgp routing-table ipv4 1.1.1.0
 BGP local router ID: 168.1.1.2
 Local AS number: 2000
 Paths:   1 available, 1 best
 BGP routing table information of 168.1.1.0/24:
 From            : 168.1.1.1 (168.1.1.1)
 Rely nexthop    : 168.1.1.1
 Original nexthop: 168.1.1.1
 Out interface   : GigabitEthernet2/0/2
 Route age       : 00h30m12s
 OutLabel        : NULL
 RxPathID        : 0x0
 TxPathID        : 0x0
 AS-path         : 1000
 Origin          : incomplete
 Attribute value : MED 0, pref-val 0
 State           : valid, external, best
 IP precedence   : 1
 QoS local ID    : 3
 Traffic index   : N/A
 Tunnel policy   : NULL
 Rely tunnel IDs : N/A
```

查看 Device B 的接口 GigabitEthernet2/0/2 上 QoS 策略的配置信息和运行情况。

```
[DeviceB] display qos policy interface gigabitethernet 2/0/2
```

```
Interface: GigabitEthernet2/0/2
  Direction: Inbound
  Policy: qppb
   Classifier: default-class
         Mode: qppb-manipulation
     Matched : 51 (Packets) 4022 (Bytes)
     5-minute statistics:
      Forwarded: 0/28 (pps/bps)
      Dropped  : 0/0 (pps/bps)
     Operator: AND
     Rule(s) :
      If-match any
     Behavior: be
      -none-
   Classifier: qppb
         Mode: qppb-manipulation
     Matched : 0 (Packets) 0 (Bytes)
     5-minute statistics:
      Forwarded: 0/0 (pps/bps)
      Dropped  : 0/0 (pps/bps)
     Operator: AND
     Rule(s) :
      If-match ip-precedence 1
      If-match qos-local-id 3
     Behavior: qppb
      Committed Access Rate:
        CIR 512000 (kbps), CBS 32000000 (Bytes), EBS 0 (Bytes)
        Green action  : pass
        Yellow action : pass
        Red action    : discard
        Green packets : 0 (Packets) 0 (Bytes)
        Yellow packets: 0 (Packets) 0 (Bytes)
        Red packets   : 0 (Packets) 0 (Bytes)
```

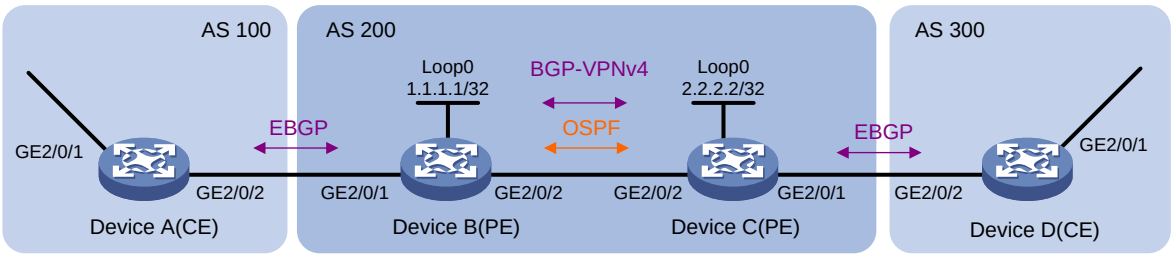
9.5.2 QPPB 在 MPLS L3VPN 中的配置举例

1. 组网需求

如图 9-2 所示，所有设备均运行 BGP 路由协议。Device C 接收路由，进行 QoS 本地 ID 值的设置，并结合 QoS 策略进行双向 200000kbps 的限速。

2. 组网图

图9-2 QPPB 在 MPLS L3VPN 中的配置举例组网图



设备	接口	IP地址	设备	接口	IP地址
Device A	GE2/0/1	192.168.1.2/24	Device B	GE2/0/1	167.1.1.2/24
	GE2/0/2	167.1.1.1/24		GE2/0/2	168.1.1.2/24
Device C	GE2/0/1	169.1.1.2/24	Device D	GE2/0/2	169.1.1.1/24
	GE2/0/2	168.1.1.1/24		GE2/0/1	192.168.3.2/24

3. 配置步骤

(1) 配置各接口的 IP 地址（略）

(2) 配置 Device A

配置 BGP 连接。

```
<DeviceA> system-view
[DeviceA] bgp 100
[DeviceA-bgp] peer 167.1.1.2 as-number 200
[DeviceA-bgp] peer 167.1.1.2 connect-interface gigabitethernet 2/0/2
[DeviceA-bgp] address-family ipv4
[DeviceA-bgp-ipv4] import-route direct
[DeviceA-bgp-ipv4] peer 167.1.1.2 enable
[DeviceA-bgp-ipv4] quit
[DeviceA-bgp] quit
```

(3) 配置 Device B

配置 VPN 实例。

```
<DeviceB> system-view
[DeviceB] ip vpn-instance vpn1
[DeviceB-vpn-instance-vpn1] route-distinguisher 200:1
[DeviceB-vpn-instance-vpn1] vpn-target 200:1 export-extcommunity
[DeviceB-vpn-instance-vpn1] vpn-target 200:1 import-extcommunity
[DeviceB-vpn-instance-vpn1] quit
```

配置 BGP 连接。

```
[DeviceB] router id 1.1.1.1
[DeviceB] bgp 200
[DeviceB-bgp] peer 2.2.2.2 as-number 200
[DeviceB-bgp] peer 2.2.2.2 connect-interface loopback 0
[DeviceB-bgp] ip vpn-instance vpn1
```

```
[DeviceB-bgp-vpn1] peer 167.1.1.1 as-number 100
[DeviceB-bgp-vpn1] address-family ipv4
[DeviceB-bgp-ipv4-vpn1] peer 167.1.1.1 enable
[DeviceB-bgp-ipv4-vpn1] quit
[DeviceB-bgp] address-family vpnv4
[DeviceB-bgp-vpnv4] peer 2.2.2.2 enable
[DeviceB-bgp-vpnv4] quit
[DeviceB-bgp] quit
```

配置 MPLS。

```
[DeviceB] mpls lsr-id 1.1.1.1
[DeviceB] mpls ldp
[DeviceB-mpls-ldp] quit
```

配置 OSPF。

```
[DeviceB] ospf
[DeviceB-ospf-1] area 0
[DeviceB-ospf-1-area-0.0.0.0] network 1.1.1.1 0.0.0.0
[DeviceB-ospf-1-area-0.0.0.0] network 168.1.1.0 0.0.0.255
[DeviceB-ospf-1-area-0.0.0.0] quit
[DeviceB-ospf-1] quit
```

接口 GigabitEthernet2/0/1 绑定 VPN。

```
[DeviceB] interface gigabitethernet 2/0/1
[DeviceB-GigabitEthernet2/0/1] ip binding vpn-instance vpn1
[DeviceB-GigabitEthernet2/0/1] ip address 167.1.1.2 24
[DeviceB-GigabitEthernet2/0/1] quit
```

接口 GigabitEthernet2/0/2 开启 MPLS。

```
[DeviceB] interface gigabitethernet 2/0/2
[DeviceB-GigabitEthernet2/0/2] mpls enable
[DeviceB-GigabitEthernet2/0/2] mpls ldp enable
[DeviceB-GigabitEthernet2/0/2] quit
```

(4) 配置 Device C

配置 VPN 实例。

```
<DeviceC> system-view
[DeviceC] ip vpn-instance vpn1
[DeviceC-vpn-instance-vpn1] route-distinguisher 200:1
[DeviceC-vpn-instance-vpn1] vpn-target 200:1 export-extcommunity
[DeviceC-vpn-instance-vpn1] vpn-target 200:1 import-extcommunity
[DeviceC-vpn-instance-vpn1] quit
```

配置 BGP 连接。

```
[DeviceC] router id 2.2.2.2
[DeviceC] bgp 200
[DeviceC-bgp] peer 1.1.1.1 as-number 200
[DeviceC-bgp] peer 1.1.1.1 connect-interface loopback 0
[DeviceC-bgp] ip vpn-instance vpn1
[DeviceC-bgp-vpn1] peer 169.1.1.1 as-number 300
[DeviceC-bgp-vpn1] address-family ipv4
[DeviceC-bgp-ipv4-vpn1] peer 169.1.1.1 enable
[DeviceC-bgp-ipv4-vpn1] peer 169.1.1.1 route-policy qppb import
```

```
[DeviceC-bgp-ipv4-vpn1] quit
[DeviceC-bgp-vpn1] quit
[DeviceC-bgp] address-family vpnv4
[DeviceC-bgp-vpnv4] peer 1.1.1.1 enable
[DeviceC-bgp-vpnv4] peer 1.1.1.1 route-policy qppb import
[DeviceC-bgp-vpnv4] quit
[DeviceC-bgp] quit
```

配置路由策略。

```
[DeviceC] route-policy qppb permit node 0
[DeviceC-route-policy-qppb-0] apply qos-local-id 3
[DeviceC-route-policy-qppb-0] quit
```

配置 MPLS。

```
[DeviceC] mpls lsr-id 2.2.2.2
[DeviceC] mpls ldp
[DeviceC-mpls-ldp] quit
```

配置 OSPF。

```
[DeviceC] ospf
[DeviceC-ospf-1] area 0
[DeviceC-ospf-1-area-0.0.0.0] network 2.2.2.2 0.0.0.0
[DeviceC-ospf-1-area-0.0.0.0] network 168.1.1.0 0.0.0.255
[DeviceC-ospf-1-area-0.0.0.0] quit
[DeviceC-ospf-1] quit
```

配置 QoS 策略。

```
[DeviceC] traffic classifier qppb
[DeviceC-classifier-qppb] if-match qos-local-id 3
[DeviceC-classifier-qppb] quit
[DeviceC] traffic behavior qppb
[DeviceC-behavior-qppb] car cir 200000 green pass red discard
[DeviceC-behavior-qppb] quit
[DeviceC] qos policy qppb
[DeviceC-qospolicy-qppb] classifier qppb behavior qppb mode qppb-manipulation
[DeviceC-qospolicy-qppb] quit
```

接口 GigabitEthernet2/0/2 开启 MPLS。

```
[DeviceC] interface gigabitethernet 2/0/2
[DeviceC-GigabitEthernet2/0/2] mpls enable
[DeviceC-GigabitEthernet2/0/2] mpls ldp enable
```

接口开启 QPPB 能力。

```
[DeviceC-GigabitEthernet2/0/2] bgp-policy source ip-qos-map
[DeviceC-GigabitEthernet2/0/2] quit
[DeviceC] interface gigabitethernet 2/0/1
[DeviceC-GigabitEthernet2/0/1] bgp-policy source ip-qos-map
[DeviceC-GigabitEthernet2/0/1] quit
```

接口 GigabitEthernet2/0/1 绑定 VPN。

```
[DeviceC] interface gigabitethernet 2/0/1
[DeviceC-GigabitEthernet2/0/1] ip binding vpn-instance vpn1
[DeviceC-GigabitEthernet2/0/1] ip address 169.1.1.2 24
```

接口 GigabitEthernet2/0/1 入方向应用 QoS 策略。

```
[DeviceC-GigabitEthernet2/0/1] qos apply policy qppb inbound
[DeviceC-GigabitEthernet2/0/1] quit
```

接口 GigabitEthernet2/0/2 入方向应用 QoS 策略。

```
[DeviceC] interface gigabitethernet 2/0/2
[DeviceC-GigabitEthernet2/0/2] qos apply policy qppb inbound
```

(5) 配置 Device D

配置 BGP 连接。

```
<DeviceD> system-view
[DeviceD] bgp 300
[DeviceD-bgp] peer 169.1.1.2 as-number 200
[DeviceD-bgp] peer 169.1.1.2 connect-interface gigabitethernet 2/0/2
[DeviceD-bgp] address-family ipv4
[DeviceD-bgp-ipv4] peer 169.1.1.2 enable
[DeviceD-bgp-ipv4] import-route direct
[DeviceD-bgp-ipv4] quit
```

4. 验证配置

查看 Device A 相关路由是否生效。

```
[DeviceA] display ip routing-table

Destinations : 18        Routes : 18

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
0.0.0.0/32          Direct 0    0            127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0            127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0            127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0            127.0.0.1       InLoop0
127.255.255.255/32  Direct 0    0            127.0.0.1       InLoop0
167.1.1.0/24        Direct 0    0            167.1.1.1       GE2/0/2
167.1.1.0/32        Direct 0    0            167.1.1.1       GE2/0/2
167.1.1.1/32        Direct 0    0            127.0.0.1       InLoop0
167.1.1.255/32      Direct 0    0            167.1.1.1       GE2/0/2
169.1.1.0/24        BGP    255  0            167.1.1.2       GE2/0/2
192.168.1.0/24      Direct 0    0            192.168.1.2     GE2/0/1
192.168.1.0/32      Direct 0    0            192.168.1.2     GE2/0/1
192.168.1.2/32      Direct 0    0            127.0.0.1       InLoop0
192.168.1.255/32    Direct 0    0            192.168.1.2     GE2/0/1
192.168.3.0/24      BGP    255  0            167.1.1.2       GE2/0/2
224.0.0.0/4         Direct 0    0            0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0            0.0.0.0         NULL0
255.255.255.255/32  Direct 0    0            127.0.0.1       InLoop0
```

查看 Device B 相关路由是否生效。

```
[DeviceB] display ip routing-table

Destinations : 14        Routes : 14
```

```
Destination/Mask    Proto  Pre  Cost         NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
1.1.1.1/32          Direct 0    0           127.0.0.1       InLoop0
2.2.2.2/32          OSPF   10   1           168.1.1.1       GE2/0/2
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
127.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
168.1.1.0/24        Direct 0    0           168.1.1.2       GE2/0/2
168.1.1.0/32        Direct 0    0           168.1.1.2       GE2/0/2
168.1.1.2/32        Direct 0    0           127.0.0.1       InLoop0
168.1.1.255/32      Direct 0    0           168.1.1.2       GE2/0/2
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
255.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
[DeviceB] display ip routing-table vpn-instance vpn1

Destinations : 16        Routes : 16

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
127.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
167.1.1.0/24        Direct 0    0           167.1.1.2       GE2/0/1
167.1.1.0/32        Direct 0    0           167.1.1.2       GE2/0/1
167.1.1.2/32        Direct 0    0           127.0.0.1       InLoop0
167.1.1.255/32      Direct 0    0           167.1.1.2       GE2/0/1
169.1.1.0/24        BGP    255  0           2.2.2.2         GE2/0/2
192.168.1.0/24      BGP    255  0           167.1.1.1       GE2/0/1
192.168.2.0/24      BGP    255  0           167.1.1.1       GE2/0/1
192.168.3.0/24      BGP    255  0           2.2.2.2         GE2/0/2
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
255.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
# 查看 Device C 相关路由是否生效。
[DeviceC] display ip routing-table

Destinations : 14        Routes : 14

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
1.1.1.1/32          OSPF   10   1           168.1.1.2       GE2/0/2
2.2.2.2/32          Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
```

```
127.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
168.1.1.0/24        Direct 0    0           168.1.1.1       GE2/0/2
168.1.1.0/32        Direct 0    0           168.1.1.1       GE2/0/2
168.1.1.1/32        Direct 0    0           127.0.0.1       InLoop0
168.1.1.255/32      Direct 0    0           168.1.1.1       GE2/0/2
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
255.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
[DeviceC] display ip routing-table vpn-instance vpn1

Destinations : 16        Routes : 16

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
127.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
167.1.1.0/24        BGP    255  0           1.1.1.1         GE2/0/2
169.1.1.0/24        Direct 0    0           169.1.1.2       GE2/0/1
169.1.1.0/32        Direct 0    0           169.1.1.2       GE2/0/1
169.1.1.2/32        Direct 0    0           127.0.0.1       InLoop0
169.1.1.255/32      Direct 0    0           169.1.1.2       GE2/0/1
192.168.1.0/24      BGP    255  0           1.1.1.1         GE2/0/2
192.168.2.0/24      BGP    255  0           169.1.1.1       GE2/0/1
192.168.3.0/24      BGP    255  0           169.1.1.1       GE2/0/1
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
255.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
```

查看 Device D 相关路由是否生效。

```
[DeviceD] display ip routing-table

Destinations : 18        Routes : 18

Destination/Mask    Proto  Pre  Cost         NextHop         Interface
0.0.0.0/32          Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/8         Direct 0    0           127.0.0.1       InLoop0
127.0.0.0/32        Direct 0    0           127.0.0.1       InLoop0
127.0.0.1/32        Direct 0    0           127.0.0.1       InLoop0
127.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
167.1.1.0/24        BGP    255  0           169.1.1.2       GE2/0/2
169.1.1.0/24        Direct 0    0           169.1.1.1       GE2/0/2
169.1.1.0/32        Direct 0    0           169.1.1.1       GE2/0/2
169.1.1.1/32        Direct 0    0           127.0.0.1       InLoop0
169.1.1.255/32      Direct 0    0           169.1.1.1       GE2/0/2
192.168.1.0/24      BGP    255  0           169.1.1.2       GE2/0/2
192.168.3.0/24      Direct 0    0           192.168.3.2     GE2/0/1
192.168.3.0/32      Direct 0    0           192.168.3.2     GE2/0/1
```

```
192.168.3.2/32      Direct 0    0           127.0.0.1       InLoop0
192.168.3.255/32    Direct 0    0           192.168.3.2     GE2/0/1
224.0.0.0/4         Direct 0    0           0.0.0.0         NULL0
224.0.0.0/24        Direct 0    0           0.0.0.0         NULL0
255.255.255.255/32  Direct 0    0           127.0.0.1       InLoop0
```

查看 Device C 的接口入方向上 QoS 策略的配置信息和运行情况。

```
[DeviceC] display qos policy interface inbound
Interface: GigabitEthernet2/0/1
  Direction: Inbound
  Policy: qppb
   Classifier: default-class
         Mode: qppb-manipulation
     Matched : 312 (Packets) 18916 (Bytes)
     5-minute statistics:
      Forwarded: 0/24 (pps/bps)
      Dropped  : 0/0 (pps/bps)
     Operator: AND
     Rule(s) :
      If-match any
     Behavior: be
      -none-
   Classifier: qppb
         Mode: qppb-manipulation
     Matched : 0 (Packets) 0 (Bytes)
     5-minute statistics:
      Forwarded: 0/0 (pps/bps)
      Dropped  : 0/0 (pps/bps)
     Operator: AND
     Rule(s) :
      If-match qos-local-id 3
     Behavior: qppb
      Committed Access Rate:
        CIR 200000 (kbps), CBS 1250000 (Bytes), EBS 0 (Bytes)
        Green action  : pass
        Yellow action : pass
        Red action    : discard
        Green packets : 0 (Packets) 0 (Bytes)
        Yellow packets: 0 (Packets) 0 (Bytes)
        Red packets   : 0 (Packets) 0 (Bytes)

Interface: GigabitEthernet2/0/2
  Direction: Inbound
  Policy: qppb
   Classifier: default-class
         Mode: qppb-manipulation
     Matched : 311 (Packets) 23243 (Bytes)
     5-minute statistics:
      Forwarded: 0/24 (pps/bps)
```

```
    Dropped   : 0/0 (pps/bps)
  Operator: AND
  Rule(s) :
    If-match any
  Behavior: be
    -none-
Classifier: qppb
      Mode: qppb-manipulation
  Matched : 0 (Packets) 0 (Bytes)
  5-minute statistics:
    Forwarded: 0/0 (pps/bps)
    Dropped   : 0/0 (pps/bps)
  Operator: AND
  Rule(s) :
    If-match qos-local-id 3
  Behavior: qppb
    Committed Access Rate:
      CIR 200000 (kbps), CBS 12500480 (Bytes), EBS 0 (Bytes)
      Green action  : pass
      Yellow action : pass
      Red action    : discard
      Green packets : 0 (Packets) 0 (Bytes)
      Yellow packets: 0 (Packets) 0 (Bytes)
      Red packets   : 0 (Packets) 0 (Bytes)
```

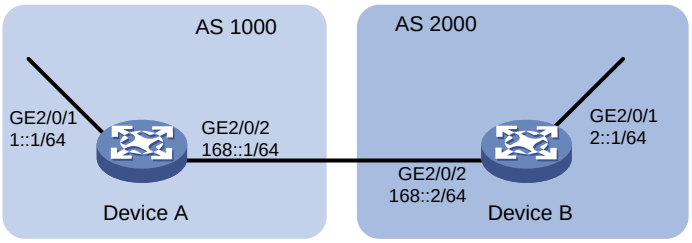
9.5.3 QPPB 在 IPv6 网络中的配置举例

1. 组网需求

如表 9-1 所示，所有设备均运行 BGP 协议。Device B 接收路由，进行 IP 优先级设置，并结合 QoS 策略进行 512000kbps 的限速。

2. 组网图

表9-1 QPPB 在 IPv6 网络中的配置举例组网图



3. 配置步骤

(1) 配置各接口的 IPv6 地址（略）

(2) 配置 Device A

配置 BGP

```
<DeviceA> system-view
```

```
[DeviceA] bgp 1000
[DeviceA] peer 168::2 as-number 2000
[DeviceA] peer 168::2 connect-interface gigabitethernet 2/0/2
[DeviceA-bgp] address-family ipv6
[DeviceA-bgp-ipv6] peer 168::2 enable
[DeviceA-bgp-ipv6] import-route direct
[DeviceA-bgp-ipv6] quit
[DeviceA-bgp] quit
```

(3) 配置 Device B

配置 BGP

```
<DeviceB> system-view
[DeviceB] bgp 2000
[DeviceB] peer 168::1 as-number 1000
[DeviceB] peer 168::1 connect-interface gigabitethernet 2/0/2
[DeviceB-bgp] address-family ipv6
[DeviceB-bgp-ipv6] peer 168::1 enable
[DeviceB-bgp-ipv6] peer 168::1 route-policy qppb import
[DeviceB-bgp-ipv6] quit
[DeviceB-bgp] quit
```

配置路由策略

```
[DeviceB] route-policy qppb permit node 0
[DeviceB-route-policy-qppb-0] apply ip-precedence 4
[DeviceB-route-policy-qppb-0] apply qos-local-id 3
[DeviceB-route-policy-qppb-0] quit
```

接口开启 QPPB 能力

```
[DeviceB] interface gigabitethernet 2/0/2
[DeviceB-GigabitEthernet2/0/2] bgp-policy source ip-prec-map ip-qos-map
```

配置 QoS 策略。

```
[DeviceB] traffic classifier qppb
[DeviceB-classifier-qppb] if-match ip-precedence 4
[DeviceB-classifier-qppb] if-match qos-local-id 3
[DeviceB-classifier-qppb] quit
[DeviceB] traffic behavior qppb
[DeviceB-behavior-qppb] car cir 512000 red discard
[DeviceB-behavior-qppb] quit
[DeviceB] qos policy qppb
[DeviceB-qospolicy-qppb] classifier qppb behavior qppb mode qppb-manipulation
[DeviceB-qospolicy-qppb] quit
```

接口应用 QoS 策略。

```
[DeviceB] interface gigabitethernet 2/0/2
[DeviceB-GigabitEthernet2/0/2] qos apply policy qppb inbound
[DeviceB-GigabitEthernet2/0/2] quit
```

4. 验证配置

查看 Device A 相关路由是否生效。

```
[DeviceA] display bgp routing-table ipv6 2:: 64
 BGP local router ID: 0.0.0.0
```

```
 Local AS number: 1000
 Paths:   1 available, 1 best
 BGP routing table information of 168::/64:
 Imported route.
 Original nexthop: ::
 Out interface   : GigabitEthernet2/0/2
 Route age       : 00h17m18s
 OutLabel        : NULL
 RxPathID        : 0x0
 TxPathID        : 0x0
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, pref-val 32768
 State           : valid, local, best
 IP precedence   : 4
 QoS local ID    : 3
 Traffic index   : N/A
 Tunnel policy   : NULL
 Rely tunnel IDs : N/A
```

查看 Device B 相关路由是否生效。

```
[DeviceB] display bgp routing-table ipv6 1:: 64
 BGP local router ID: 0.0.0.0
 Local AS number: 2000
 Paths:   1 available, 1 best
 BGP routing table information of 168::/64:
 Imported route.
 Original nexthop: ::
 Out interface   : GigabitEthernet2/0/2
 Route age       : 00h05m17s
 OutLabel        : NULL
 RxPathID        : 0x0
 TxPathID        : 0x0
 AS-path         : (null)
 Origin          : incomplete
 Attribute value : MED 0, pref-val 32768
 State           : valid, local, best
 IP precedence   : N/A
 QoS local ID    : N/A
 Traffic index   : N/A
 Tunnel policy   : NULL
 Rely tunnel IDs : N/A
```

查看 Device B 的接口 GigabitEthernet2/0/2 上 QoS 策略的配置信息和运行情况。

```
[DeviceC] display qos policy interface gigabitethernet 2/0/2
Interface: GigabitEthernet2/0/2
  Direction: Inbound
  Policy: qppb
   Classifier: default-class
         Mode: qppb-manipulation
```

```
  Matched : 0 (Packets) 0 (Bytes)
  5-minute statistics:
   Forwarded: 0/0 (pps/bps)
   Dropped  : 0/0 (pps/bps)
  Operator: AND
  Rule(s) :
   If-match any
  Behavior: be
   -none-
Classifier: qppb
      Mode: qppb-manipulation
  Matched : 0 (Packets) 0 (Bytes)
  5-minute statistics:
   Forwarded: 0/0 (pps/bps)
   Dropped  : 0/0 (pps/bps)
  Operator: AND
  Rule(s) :
   If-match ip-precedence 4
   If-match qos-local-id 3
  Behavior: qppb
   Committed Access Rate:
     CIR 512000 (kbps), CBS 32000000 (Bytes), EBS 0 (Bytes)
     Green action  : pass
     Yellow action : pass
     Red action    : discard
     Green packets : 0 (Packets) 0 (Bytes)
     Yellow packets: 0 (Packets) 0 (Bytes)
     Red packets   : 0 (Packets) 0 (Bytes)
```

10 附录

10.1 附录 A 缩略语表

表10-1 附录 A 缩略语表

缩略语	英文全名	中文解释
AF	Assured Forwarding	确保转发
BE	Best Effort	尽力转发
BQ	Bandwidth Queuing	带宽队列
CAR	Committed Access Rate	承诺访问速率
CBQ	Class Based Queuing	基于类的队列
CBS	Committed Burst Size	承诺突发尺寸
CBWFQ	Class Based Weighted Fair Queuing	基于类的加权公平队列
CE	Customer Edge	用户边缘设备
CIR	Committed Information Rate	承诺信息速率
CQ	Custom Queuing	定制队列
DAR	Deeper Application Recognition	深度应用识别
DCBX	Data Center Bridging Exchange Protocol	数据中心桥能力交换协议
DiffServ	Differentiated Service	区分服务
DoS	Denial of Service	拒绝服务
DSCP	Differentiated Services Code Point	区分服务编码点
EACL	Enhanced ACL	增强型ACL
EBS	Excess Burst Size	超出突发尺寸
ECN	Explicit Congestion Notification	显示拥塞通知
EF	Expedited Forwarding	加速转发
FEC	Forwarding Equivalance Class	转发等价类
FIFO	First in First out	先入先出
FQ	Fair Queuing	公平队列
GMB	Guaranteed Minimum Bandwidth	最小带宽保证队列
GTS	Generic Traffic Shaping	通用流量整形
IntServ	Integrated Service	综合服务
ISP	Internet Service Provider	互联网服务提供商
LFI	Link Fragmentation and Interleaving	链路分片与交叉

缩略语	英文全名	中文解释
LLQ	Low Latency Queuing	低时延队列
LR	Line Rate	限速
LSP	Label Switched Path	标签交换路径
MPLS	Multiprotocol Label Switching	多协议标签交换
P2P	Peer-to-Peer	对等
PE	Provider Edge	服务提供商网络边缘
PHB	Per-hop Behavior	单中继段行为
PIR	Peak Information Rate	峰值信息速率
PQ	Priority Queuing	优先队列
PW	Pseudowire	伪线
QoS	Quality of Service	服务质量
QPPB	QoS Policy Propagation Through the Border Gateway Protocol	通过BGP传播QoS策略
RED	Random Early Detection	随机早期检测
RSVP	Resource Reservation Protocol	资源预留协议
RTP	Real-time Transport Protocol	实时传输协议
SLA	Service Level Agreement	服务水平协议
SP	Strict Priority	严格优先级队列
TE	Traffic Engineering	流量工程
ToS	Type of Service	服务类型
TP	Traffic Policing	流量监管
TS	Traffic Shaping	流量整形
VoIP	Voice over IP	在IP网络上传送语音
VPN	Virtual Private Network	虚拟专用网络
VSI	Virtual Station Interface	虚拟服务器接口
WFQ	Weighted Fair Queuing	加权公平队列
WRED	Weighted Random Early Detection	加权随机早期检测
WRR	Weighted Round Robin	加权轮询队列

10.2 附录 B 缺省优先级映射表

表10-2

表10-3 dot1p-lp 缺省映射关系

映射输入索引	dot1p-lp 映射
dot1p	lp
0	2
1	0
2	1
3	3
4	4
5	5
6	6
7	7

表10-4 dscp-lp 缺省映射关系

映射输入索引	dscp-lp 映射
dscp	lp
0～7	0
8～15	1
16～23	2
24～31	3
32～39	4
40～47	5
48～55	6
56～63	7

表10-5 lp-dot1p 缺省映射关系

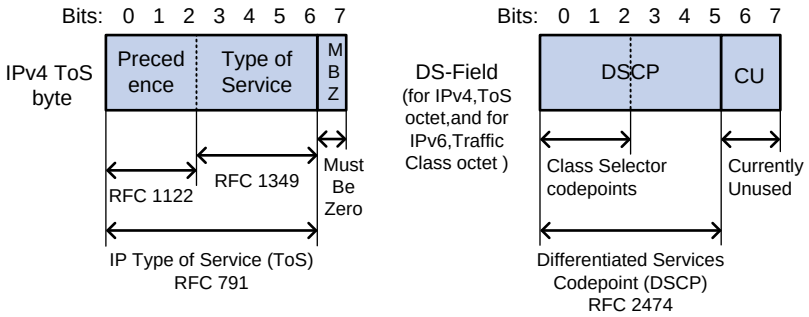
映射输入索引	lp-dot1p 映射
lp	dot1p
0	1
1	2
2	0
3	3
4	4

映射输入索引	Ip-dot1p 映射
5	5
6	6
7	7

10.3 附录 C 各种优先级介绍

10.3.1 IP 优先级和 DSCP 优先级

图10-1 ToS 和 DS 域



如图 10-1 所示，IP 报文头的 ToS 字段有 8 个 bit，其中前 3 个 bit 表示的就是 IP 优先级，取值范围为 0～7。RFC 2474 中，重新定义了 IP 报文头部的 ToS 域，称之为 DS（Differentiated Services，差分服务）域，其中 DSCP 优先级用该域的前 6 位（0～5 位）表示，取值范围为 0～63，后 2 位（6、7 位）是保留位。

表10-6 IP 优先级说明

IP 优先级（十进制）	IP 优先级（二进制）	关键字
0	000	routine
1	001	priority
2	010	immediate
3	011	flash
4	100	flash-override
5	101	critical
6	110	internet
7	111	network

表10-7 DSCP 优先级说明

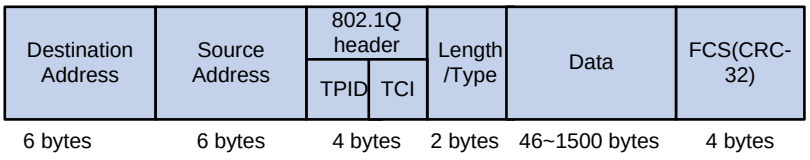
DSCP 优先级（十进制）	DSCP 优先级（二进制）	关键字
46	101110	ef

DSCP 优先级（十进制）	DSCP 优先级（二进制）	关键字
10	001010	af11
12	001100	af12
14	001110	af13
18	010010	af21
20	010100	af22
22	010110	af23
26	011010	af31
28	011100	af32
30	011110	af33
34	100010	af41
36	100100	af42
38	100110	af43
8	001000	cs1
16	010000	cs2
24	011000	cs3
32	100000	cs4
40	101000	cs5
48	110000	cs6
56	111000	cs7
0	000000	be（default）

10.3.2 802.1p 优先级

802.1p 优先级位于二层报文头部，适用于不需要分析三层报头，而需要在二层环境下保证 QoS 的场合。

图10-2 带有 802.1Q 标签头的以太网帧



如图 10-2 所示，4 个字节的 802.1Q 标签头包含了 2 个字节的 TPID（Tag Protocol Identifier，标签协议标识符）和 2 个字节的 TCI（Tag Control Information，标签控制信息），TPID 取值为 0x8100，图 10-3 显示了 802.1Q 标签头的详细内容，Priority 字段就是 802.1p 优先级。之所以称此优先级为 802.1p 优先级，是因为有关这些优先级的应用是在 802.1p 规范中被详细定义的。

图10-3 802.1Q 标签头

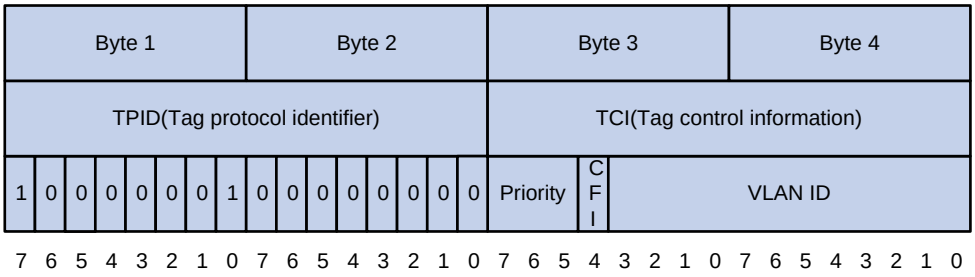


表10-8 802.1p 优先级说明

802.1p 优先级（十进制）	802.1p 优先级（二进制）	关键字
0	000	best-effort
1	001	background
2	010	spare
3	011	excellent-effort
4	100	controlled-load
5	101	video
6	110	voice
7	111	network-management

10.3.3 EXP 优先级

EXP 优先级位于 MPLS 标签内，用于标记 MPLS QoS。

图10-4 MPLS 标签的封装结构

在图 10-4 中，Exp 字段就是 EXP 优先级，长度为 3 比特，取值范围为 0～7。