H3C S5130S-SI[LI]&S5120V2-SI[LI]&S5110V2-SI&S5000V3-EI&S5000E-X&S3100V3-SI 系列以太网交换机

ACL 和 QoS 配置指导

新华三技术有限公司
http://www.h3c.com

资料版本：6W103-20190822
产品版本：Release 612x 系列

前 言

本配置指导主要介绍如何使用 ACL、QoS、数据缓冲区和时间段。

前言部分包含如下内容：

读者对象

本手册主要适用于如下工程师：

- 网络规划人员
- 现场技术支持与维护人员
- 负责网络配置和维护的网络管理员

本书约定

1. 命令行格式约定

格　式	意　义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | ... }	表示从多个选项中仅选取一个。
[x | y | ...]	表示从多个选项中选取一个或者不选。
{ x | y | ... }*	表示从多个选项中至少选取一个。
[x | y | ...]*	表示从多个选项中选取一个、多个或者不选。
&<1-n>	表示符号&前面的参数可以重复输入1～n次。
#	由“#”号开始的行表示为注释行。

2. 图形界面格式约定

格　式	意　义
< >	带尖括号“< >”表示按钮名，如“单击<确定>按钮”。
[ ]	带方括号“[ ]”表示窗口名、菜单名和数据表，如“弹出[新建用户]窗口”。
/	多级菜单用“/”隔开。如[文件/新建/文件夹]多级菜单表示[文件]菜单下的[新建]子菜单下的[文件夹]菜单项。

3. 各类标志

本书还采用各种醒目标志来表示在操作过程中应该特别注意的地方，这些标志的意义如下：

警告	该标志后的注释需给予格外关注，不当的操作可能会对人身造成伤害。
注意	提醒操作中应注意的事项，不当的操作可能会导致数据丢失或者设备损坏。
提示	为确保设备配置成功或者正常工作而需要特别关注的操作或信息。
说明	对操作内容的描述进行必要的补充和说明。
窍门	配置、操作、或使用设备的技巧、小窍门。

4. 图标约定

本书使用的图标及其含义如下：

	该图标及其相关描述文字代表一般网络设备，如路由器、交换机、防火墙等。
	该图标及其相关描述文字代表一般意义下的路由器，以及其他运行了路由协议的设备。
	该图标及其相关描述文字代表二、三层以太网交换机，以及运行了二层协议的设备。
	该图标及其相关描述文字代表无线控制器、无线控制器业务板和有线无线一体化交换机的无线控制引擎设备。
	该图标及其相关描述文字代表无线接入点设备。
	该图标及其相关描述文字代表无线终结单元。
	该图标及其相关描述文字代表无线终结者。
	该图标及其相关描述文字代表无线Mesh设备。
	该图标代表发散的无线射频信号。
	该图标代表点到点的无线射频信号。
	该图标及其相关描述文字代表防火墙、UTM、多业务安全网关、负载均衡等安全设备。
	该图标及其相关描述文字代表防火墙插卡、负载均衡插卡、NetStream插卡、SSL VPN插卡、IPS插卡、ACG插卡等安全插卡。

5. 示例约定

由于设备型号不同、配置不同、版本升级等原因，可能造成本手册中的内容与用户使用的设备显示信息不一致。实际使用中请以设备显示的内容为准。

本手册中出现的端口编号仅作示例，并不代表设备上实际具有此编号的端口，实际使用中请以设备上存在的端口编号为准。

资料意见反馈

如果您在使用过程中发现产品资料的任何问题，可以通过以下方式反馈：

E-mail：info@h3c.com

感谢您的反馈，让我们做得更好！

目 录

1 ACL

1.1 ACL简介

ACL（Access Control List，访问控制列表）是一系列用于识别报文流的规则的集合。这里的规则是指描述报文匹配条件的判断语句，匹配条件可以是报文的源地址、目的地址、端口号等。设备依据 ACL 规则识别出特定的报文，并根据预先设定的策略对其进行处理，最常见的应用就是使用 ACL 进行报文过滤。此外，ACL 还可应用于诸如路由、安全、QoS 等业务中识别报文，对这些报文的具体处理方式由应用 ACL 的业务模块来决定。

1.1.1 ACL的编号和名称

用户在创建ACL时必须为其指定编号或名称，不同的编号对应不同类型的ACL，如 表 1-1 所示；当ACL创建完成后，用户就可以通过指定编号或名称的方式来应用和编辑该ACL。

对于编号相同的基本 ACL 或高级 ACL，必须通过 **ipv6** 关键字进行区分。对于名称相同的 ACL，必须通过 **ipv6** 和 **mac** 关键字进行区分。

1.1.2 ACL的分类

根据规则制订依据的不同，可以将ACL分为如 表 1-1 所示的几种类型。

表1-1 ACL 的分类

ACL 类型	编号范围	适用的 IP 版本	规则制订依据
基本ACL	2000～2999	IPv4	报文的源IPv4地址
		IPv6	报文的源IPv6地址
高级ACL	3000～3999	IPv4	报文的源IPv4地址、目的IPv4地址、报文优先级、IPv4承载的协议类型及特性等三、四层信息
		IPv6	报文的源IPv6地址、目的IPv6地址、报文优先级、IPv6承载的协议类型及特性等三、四层信息
二层ACL	4000～4999	IPv4和IPv6	报文的源MAC地址、目的MAC地址、802.1p优先级、链路层协议类型等二层信息

1.1.3 ACL的规则匹配顺序

当一个 ACL 中包含多条规则时，报文会按照一定的顺序与这些规则进行匹配，一旦匹配上某条规则便结束匹配过程。ACL 的规则匹配顺序有以下两种：

- 配置顺序：按照规则编号由小到大进行匹配。
- 自动排序：按照“深度优先”原则由深到浅进行匹配，各类型ACL的“深度优先”排序法则如 表 1-2 所示。

表1-2 各类型 ACL 的“深度优先”排序法则

ACL 类型	“深度优先”排序法则
IPv4基本ACL	**a** 先比较源 IPv4 地址范围，较小者优先 **b** 如果源 IPv4 地址范围相同，再比较配置的先后次序，先配置者优先
IPv4高级ACL	**c** 先比较协议范围，指定有 IPv4 承载的协议类型者优先 **d** 如果协议范围相同，再比较源 IPv4 地址范围，较小者优先 **e** 如果源 IPv4 地址范围也相同，再比较目的 IPv4 地址范围，较小者优先 **f** 如果目的 IPv4 地址范围也相同，再比较四层端口（即 TCP/UDP 端口）号的覆盖范围，较小者优先 **g** 如果四层端口号的覆盖范围无法比较，再比较配置的先后次序，先配置者优先
IPv6基本ACL	**h** 先比较源 IPv6 地址范围，较小者优先 **i** 如果源 IPv6 地址范围相同，再比较配置的先后次序，先配置者优先
IPv6高级ACL	**j** 先比较协议范围，指定有 IPv6 承载的协议类型者优先 **k** 如果协议范围相同，再比较源 IPv6 地址范围，较小者优先 **l** 如果源 IPv6 地址范围也相同，再比较目的 IPv6 地址范围，较小者优先 **m** 如果目的 IPv6 地址范围也相同，再比较四层端口（即 TCP/UDP 端口）号的覆盖范围，较小者优先 **n** 如果四层端口号的覆盖范围无法比较，再比较配置的先后次序，先配置者优先
二层ACL	**o** 先比较源 MAC 地址范围，较小者优先 **p** 如果源 MAC 地址范围相同，再比较目的 MAC 地址范围，较小者优先 **q** 如果目的 MAC 地址范围也相同，再比较配置的先后次序，先配置者优先

说明

比较 IPv4 地址范围的大小，就是比较 IPv4 地址通配符掩码中“0”位的多少：“0”位越多，范围越小。通配符掩码（又称反向掩码）以点分十进制表示，并以二进制的“0”表示“匹配”，“1”表示“不关心”，这与子网掩码恰好相反，譬如子网掩码 255.255.255.0 对应的通配符掩码就是 0.0.0.255。此外，通配符掩码中的“0”或“1”可以是不连续的，这样可以更加灵活地进行匹配，譬如 0.255.0.255 就是一个合法的通配符掩码。

比较 IPv6 地址范围的大小，就是比较 IPv6 地址前缀的长短：前缀越长，范围越小。

比较 MAC 地址范围的大小，就是比较 MAC 地址掩码中“1”位的多少：“1”位越多，范围越小。

1.1.4 ACL的步长

ACL 中的每条规则都有自己的编号，这个编号在该 ACL 中是唯一的。在创建规则时，可以手工为其指定一个编号，如未手工指定编号，则由系统为其自动分配一个编号。由于规则的编号可能影响规则匹配的顺序，因此当由系统自动分配编号时，为了方便后续在已有规则之前插入新的规则，系统通常会在相邻编号之间留下一定的空间，这个空间的大小（即相邻编号之间的差值）就称为 ACL 的步长。譬如，当步长为 5 时，系统会将编号 0、5、10、15……依次分配给新创建的规则。

系统为规则自动分配编号的方式如下：系统从规则编号的起始值开始，自动分配一个大于现有最大编号的步长最小倍数。譬如原有编号为 0、5、9、10 和 12 的五条规则，步长为 5，此时如果创建一条规则且不指定编号，那么系统将自动为其分配编号 15。

如果步长或规则编号的起始值发生了改变，ACL 内原有全部规则的编号都将自动从规则编号的起始值开始按步长重新排列。譬如，某 ACL 内原有编号为 0、5、9、10 和 15 的五条规则，当修改步长为 2 之后，这些规则的编号将依次变为 0、2、4、6 和 8。

1.1.5 ACL对分片报文的处理

传统报文过滤只对分片报文的首个分片进行匹配过滤，对后续分片一律放行，因此网络攻击者通常会构造后续分片进行流量攻击。为提高网络安全性，ACL 规则缺省会匹配所有非分片报文和分片报文的全部分片，但这样又带来效率低下的问题。为了兼顾网络安全和匹配效率，可将过滤规则配置为仅对后续分片有效。

1.2 ACL配置限制和指导

指定 ACL 编号创建的 ACL，可以通过如下命令进入其视图：

- **acl** [**ipv6**] **number** *acl-number*；
- **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } *acl-number*。

通过 **acl** [**ipv6**] **number** *acl-number* **name** *acl-name* 命令创建的 ACL，可以通过如下命令进入其视图：

- **acl** [**ipv6**] **name** *acl-name*，本命令仅支持进入已创建的基本 ACL 或高级 ACL 视图；
- **acl** [**ipv6**] **number** *acl-number* [**name** *acl-name*]；
- **acl** { [**ipv6**] { **advanced** | **basic** } | **mac**] } **name** *acl-name*。

通过 **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } **name** *acl-name* 命令创建的 ACL，只能通过如下命令进入其视图：

- **acl** [**ipv6**] **name** *acl-name*，本命令仅支持进入已创建的基本 ACL 或高级 ACL 视图；
- **acl** { [**ipv6**] { **advanced** | **basic** } | **mac** } **name** *acl-name*。

如果 ACL 规则的匹配项中包含了除 IP 五元组（源 IP 地址、源端口号、目的 IP 地址、目的端口号、传输层协议）、ICMP 报文的消息类型和消息码信息、日志操作和时间段之外的其它匹配项，则设备转发 ACL 匹配的这类报文时会启用慢转发流程。慢转发时设备会将报文上送控制平面，计算报文相应的表项信息。执行慢转发流程时，设备的转发能力将会有所降低。

1.3 ACL配置任务简介

ACL 配置任务如下

- 配置不同类型的 ACL
  - 配置基本ACL
  - 配置高级ACL
  - 配置二层ACL
- （可选）复制ACL
- （可选）应用ACL进行报文过滤

1.4 配置基本ACL

1.4.1 功能简介

基本 ACL 根据报文的源 IP 地址来制订规则，对报文进行匹配。

1.4.2 配置IPv4 基本ACL

(1) 进入系统视图。

system-view

(2) 创建 IPv4 基本 ACL。

acl basic { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

acl number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]

(3) （可选）配置 ACL 的描述信息。

description *text*

缺省情况下，未配置 ACL 的描述信息。

(4) （可选）配置规则编号的步长。

step *step-value* [**start** *start-value*]

缺省情况下，规则编号的步长为 5，起始值为 0。

(5) 创建规则。

rule [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **logging** | **source** { *source-address source-wildcard* | **any** } | **time-range** *time-range-name*] *

logging 参数是否生效取决于引用该 ACL 的模块是否支持日志记录功能，例如报文过滤支持日志记录功能，如果其引用的 ACL 规则中配置了 **logging** 参数，该参数可以生效。

(6) （可选）为规则配置描述信息。

rule *rule-id* **comment** *text*

缺省情况下，未配置规则的描述信息。

1.4.3 配置IPv6 基本ACL

1. 配置限制和指导

当 ACL 用于 QoS 策略的流分类或用于报文过滤功能时：

- 规则不支持配置 **fragment** 参数。
- 若 QoS 策略或报文过滤功能应用于出方向，规则不支持配置 **routing** 参数。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建 IPv6 基本 ACL。

acl ipv6 basic { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

acl ipv6 number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]

(3) （可选）配置 ACL 的描述信息。

description *text*

缺省情况下，未配置 ACL 的描述信息。

(4) （可选）配置规则编号的步长。

step *step-value* [**start** *start-value*]

缺省情况下，规则编号的步长为 5，起始值为 0。

(5) 创建规则。

rule [*rule-id*] { **deny** | **permit** } [**counting** | **fragment** | **logging** | **routing** [**type** *routing-type*] | **source** { *source-address source-prefix* | *source-address/source-prefix* | **any** } | **time-range** *time-range-name*] *

logging 参数是否生效取决于引用该 ACL 的模块是否支持日志记录功能，例如报文过滤支持日志记录功能，如果其引用的 ACL 规则中配置了 **logging** 参数，该参数可以生效。

(6) （可选）为规则配置描述信息。

rule *rule-id* **comment** *text*

缺省情况下，未配置规则的描述信息。

1.5 配置高级ACL

1.5.1 功能简介

高级 ACL 可根据报文的源地址、目的地址、报文优先级、承载的协议类型及特性（如 TCP/UDP 的源端口和目的端口、TCP 报文标识、ICMP 或 ICMPv6 协议的消息类型和消息码等），对报文进行匹配。用户可利用高级 ACL 制订比基本 ACL 更准确、丰富、灵活的规则。

1.5.2 配置IPv4 高级ACL

(1) 进入系统视图。

system-view

(2) 创建 IPv4 高级 ACL。

acl advanced { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

acl number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]

(3) （可选）配置 ACL 的描述信息。

description *text*

缺省情况下，未配置 ACL 的描述信息。

(4) （可选）配置规则编号的步长。

step *step-value* [**start** *start-value*]

缺省情况下，规则编号的步长为 5，起始值为 0。

(5) 创建规则。

rule [*rule-id*] { **deny** | **permit** } *protocol* [{ { **ack** *ack-value* | **fin** *fin-value* | **psh** *psh-value* | **rst** *rst-value* | **syn** *syn-value* | **urg** *urg-value* } * | **established** } | **counting** | **destination** { *dest-address dest-wildcard* | **any** } | **destination-port** *operator port1* [*port2*] | { **dscp** *dscp* | { **precedence** *precedence* | **tos** *tos* } * } | **fragment** | **icmp-type** { *icmp-type* [*icmp-code*] | *icmp-message* } | **logging** | **source** { *source-address source-wildcard* | **any** } | **source-port** *operator port1* [*port2*] | **time-range** *time-range-name*] *

logging 参数是否生效取决于引用该 ACL 的模块是否支持日志记录功能，例如报文过滤支持日志记录功能，如果其引用的 ACL 规则中配置了 **logging** 参数，该参数可以生效。

(6) （可选）为规则配置描述信息。

rule *rule-id* **comment** *text*

缺省情况下，未配置规则的描述信息。

1.5.3 配置IPv6 高级ACL

1. 配置限制和指导

当 ACL 用于 QoS 策略的流分类或报文过滤功能时：

- 规则不支持配置 **fragment** 参数。
- 如果 QoS 策略或报文过滤功能应用于出方向，规则不支持配置 **routing**、**hop-by-hop**、**flow-label** 参数。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建 IPv6 高级 ACL。

acl ipv6 advanced { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

acl ipv6 number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]

(3) （可选）配置 ACL 的描述信息。

description *text*

缺省情况下，未配置 ACL 的描述信息。

(4) （可选）配置规则编号的步长。

step *step-value* [**start** *start-value*]

缺省情况下，规则编号的步长为 5，起始值为 0。

(5) 创建规则。

rule [*rule-id*] { **deny** | **permit** } *protocol* [{ { **ack** *ack-value* | **fin** *fin-value* | **psh** *psh-value* | **rst** *rst-value* | **syn** *syn-value* | **urg** *urg-value* } * | **established** } | **counting** | **destination** { *dest-address dest-prefix* | *dest-address/dest-prefix* | **any** } | **destination-port** *operator port1*

[*port2*] | **dscp** *dscp* | **flow-label** *flow-label-value* | **fragment** | **icmp6-type** { *icmp6-type icmp6-code* | *icmp6-message* } | **logging** | **routing** [**type** *routing-type*] | **hop-by-hop** [**type** *hop-type*] | **source** { *source-address source-prefix* | *source-address/source-prefix* | **any** } | **source-port** *operator port1* [*port2*] | **time-range** *time-range-name*] *

logging 参数是否生效取决于引用该 ACL 的模块是否支持日志记录功能，例如报文过滤支持日志记录功能，如果其引用的 ACL 规则中配置了 **logging** 参数，该参数可以生效。

(6) （可选）为规则配置描述信息。

rule *rule-id* **comment** *text*

缺省情况下，未配置规则的描述信息。

1.6 配置二层ACL

1. 功能简介

二层 ACL 可根据报文的源 MAC 地址、目的 MAC 地址、802.1p 优先级、链路层协议类型、报文的封装类型等二层信息来制订规则，对报文进行匹配。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 创建二层 ACL。

acl mac { *acl-number* | **name** *acl-name* } [**match-order** { **auto** | **config** }]

acl number *acl-number* [**name** *acl-name*] [**match-order** { **auto** | **config** }]

(3) （可选）配置 ACL 的描述信息。

description *text*

缺省情况下，未配置 ACL 的描述信息。

(4) （可选）配置规则编号的步长。

step *step-value* [**start** *start-value*]

缺省情况下，规则编号的步长为 5，起始值为 0。

(5) 创建规则。

rule [*rule-id*] { **deny** | **permit** } [**cos** *dot1p* | **counting** | **dest-mac** *dest-address dest-mask* | { **lsap** *lsap-type lsap-type-mask* | **type** *protocol-type protocol-type-mask* } | **source-mac** *source-address source-mask* | **time-range** *time-range-name*] *

(6) （可选）为规则配置描述信息。

rule *rule-id* **comment** *text*

缺省情况下，未配置规则的描述信息。

1.7 复制ACL

1. 功能简介

用户可通过复制一个已存在的 ACL（即源 ACL），来生成一个新的同类型 ACL（即目的 ACL）。除了 ACL 的编号和名称不同外，目的 ACL 与源 ACL 完全相同。

2. 配置限制和指导

目的 ACL 要与源 ACL 的类型相同，且目的 ACL 必须不存在，否则将导致复制 ACL 失败。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 复制并生成一个新的 ACL。

acl [**ipv6** | **mac**] **copy** { *source-acl-number* | **name** *source-acl-name* } **to** { *dest-acl-number* | **name** *dest-acl-name* }

1.8 应用ACL进行报文过滤

1.8.1 功能简介

ACL 最基本的应用就是进行报文过滤。例如，将 ACL 规则应用到指定接口的入或出方向上，从而对该接口收到或发出的报文进行过滤。

1.8.2 在接口上应用ACL进行报文过滤

1. 配置限制和指导

本节中的“接口”指的是二层以太网接口和 VLAN 接口。

一个接口在一个方向上最多可应用 3 个 ACL 进行报文过滤，包括一个 IPv4 ACL、一个 IPv6 ACL 以及一个二层 ACL。

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 ACL 进行报文过滤。

packet-filter [**ipv6** | **mac**] { *acl-number* | **name** *acl-name* } { **inbound** | **outbound** } [**hardware-count**]

缺省情况下，未配置接口的报文过滤。

1.8.3 配置报文过滤日志信息或告警信息的生成与发送周期

1. 功能简介

报文过滤日志或告警信息的生成与发送周期起始于报文过滤中 ACL 匹配数据流的第一个数据包，报文过滤日志或告警信息包括周期内被匹配的报文数量以及所使用的 ACL 规则。在一个周期内：

- 对于规则匹配数据流的第一个数据包，设备会立即生成报文过滤日志或告警信息；
- 对于规则匹配数据流的其他数据包，设备将在周期结束后生成报文过滤日志或告警信息。

设备生成的报文过滤日志将发送给信息中心，有关信息中心的详细介绍，请参见“网络管理和监控配置指导”中的“信息中心”。

设备生成的告警信息将发送给 SNMP，有关 SNMP 的详细介绍，请参见“网络管理和监控配置指导”中的“SNMP”。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置报文过滤日志信息或告警信息的生成与发送周期。

acl { **logging** | **trap** } **interval** *interval*

缺省情况下，报文过滤日志信息或告警信息的生成与发送周期为 0 分钟，即不记录报文过滤的日志和告警信息。

1.8.4 配置报文过滤的缺省动作

1. 功能简介

系统缺省的报文过滤动作为 Permit，即允许未匹配上 ACL 规则的报文通过。通过本配置可更改报文过滤的缺省动作为 Deny，即禁止未匹配上 ACL 规则的报文通过。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置报文过滤的缺省动作为 Deny。

packet-filter default deny

缺省情况下，报文过滤的缺省动作为 Permit，即允许未匹配上 ACL 规则的报文通过。

1.9 ACL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 ACL 配置后的运行情况，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 ACL 的统计信息。

表1-3 ACL 显示和维护

配置	命令
显示ACL的配置和运行情况	**display acl** [**ipv6** | **mac**] { *acl-number* | **all** | **name** *acl-name* }

配置	命令
显示ACL在报文过滤中的应用情况	**display packet-filter interface** [*interface-type interface-number*] [**inbound** | **outbound**] [**slot** *slot-number*]
显示ACL在报文过滤中应用的统计信息	**display packet-filter statistics interface** *interface-type interface-number* { **inbound** | **outbound** } [[**ipv6** | **mac**] { *acl-number* | **name** *acl-name* }] [**brief**]
显示ACL在报文过滤中应用的累加统计信息	**display packet-filter statistics sum** { **inbound** | **outbound** } [**ipv6** | **mac**] { *acl-number* | **name** *acl-name* } [**brief**]
显示ACL在报文过滤中的详细应用情况	**display packet-filter verbose interface** *interface-type interface-number* { **inbound** | **outbound** } [[**ipv6** | **mac**] { *acl-number* | **name** *acl-name* }] [**slot** *slot-number*]
显示QoS和ACL资源的使用情况	**display qos-acl resource** [**slot** *slot-number*]
清除ACL在报文过滤中应用的统计信息	**reset packet-filter statistics interface** [*interface-type interface-number*] { **inbound** | **outbound** } [[**ipv6** | **mac**] { *acl-number* | **name** *acl-name* }]

1.10 ACL典型配置举例

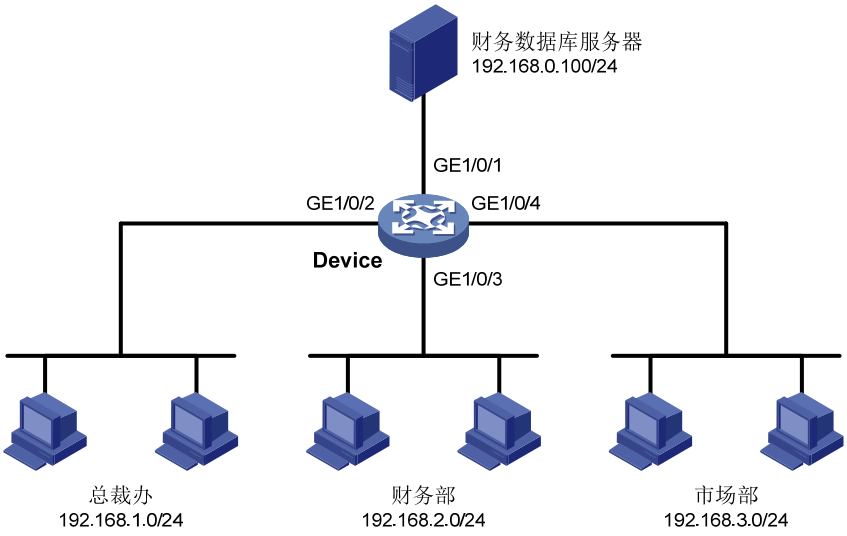
1.10.1 在接口上应用包过滤的ACL配置举例

1. 组网需求

- 某公司内的各部门之间通过 Device 实现互连，该公司的工作时间为每周工作日的 8 点到 18 点。
- 通过配置，允许总裁办在任意时间、财务部在工作时间访问财务数据库服务器，禁止其它部门在任何时间、财务部在非工作时间访问该服务器。

2. 组网图

图1-1 ACL 典型配置组网图



3. 配置步骤

创建名为 work 的时间段，其时间范围为每周工作日的 8 点到 18 点。

```
<Device> system-view
[Device] time-range work 08:00 to 18:00 working-day
```

创建 IPv4 高级 ACL 3000，并制订如下规则：允许总裁办在任意时间、财务部在工作时间访问财务数据库服务器，禁止其它部门在任何时间、财务部在非工作时间访问该服务器。

```
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule permit ip source 192.168.1.0 0.0.0.255 destination
192.168.0.100 0
[Device-acl-ipv4-adv-3000] rule permit ip source 192.168.2.0 0.0.0.255 destination
192.168.0.100 0 time-range work
[Device-acl-ipv4-adv-3000] rule deny ip source any destination 192.168.0.100 0
[Device-acl-ipv4-adv-3000] quit
```

应用 IPv4 高级 ACL 3000 对接口 GigabitEthernet1/0/1 出方向上的报文进行过滤。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] packet-filter 3000 outbound
[Device-GigabitEthernet1/0/1] quit
```

4. 验证配置

配置完成后，在各部门的 PC（假设均为 Windows XP 操作系统）上可以使用 **ping** 命令检验配置效果，在 Device 上可以使用 **display acl** 命令查看 ACL 的配置和运行情况。例如在工作时间：

在财务部的 PC 上检查到财务数据库服务器是否可达。

```
C:\> ping 192.168.0.100

Pinging 192.168.0.100 with 32 bytes of data:

Reply from 192.168.0.100: bytes=32 time=1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
```

```
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255
Reply from 192.168.0.100: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

由此可见，财务部的 PC 能够在工作时间访问财务数据库服务器。

在市场部的 PC 上检查财务数据库服务器是否可达。

```
C:\> ping 192.168.0.100

Pinging 192.168.0.100 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.0.100:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

由此可见，市场部的 PC 不能在工作时间访问财务数据库服务器。

查看 IPv4 高级 ACL 3000 的配置和运行情况。

```
[Device] display acl 3000
Advanced IPv4 ACL 3000, 3 rules,
ACL's step is 5
 rule 0 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.100 0
 rule 5 permit ip source 192.168.2.0 0.0.0.255 destination 192.168.0.100 0 time-range work
(Active)
 rule 10 deny ip destination 192.168.0.100 0
```

由此可见，由于目前是工作时间，因此规则 5 是生效的；且由于之前使用了 **ping** 命令的缘故，规则 5 和规则 10 分别被匹配了 4 次。

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

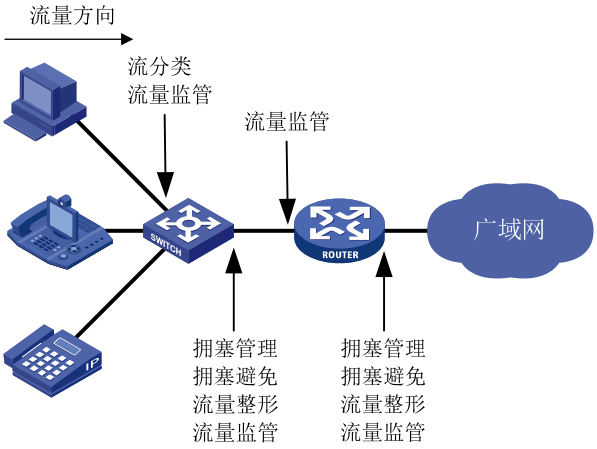
DiffServ 是一个多服务模型，它可以满足不同的 QoS 需求。与 IntServ 不同，它不需要通知网络为每个业务预留资源。区分服务实现简单，扩展性较好。

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。

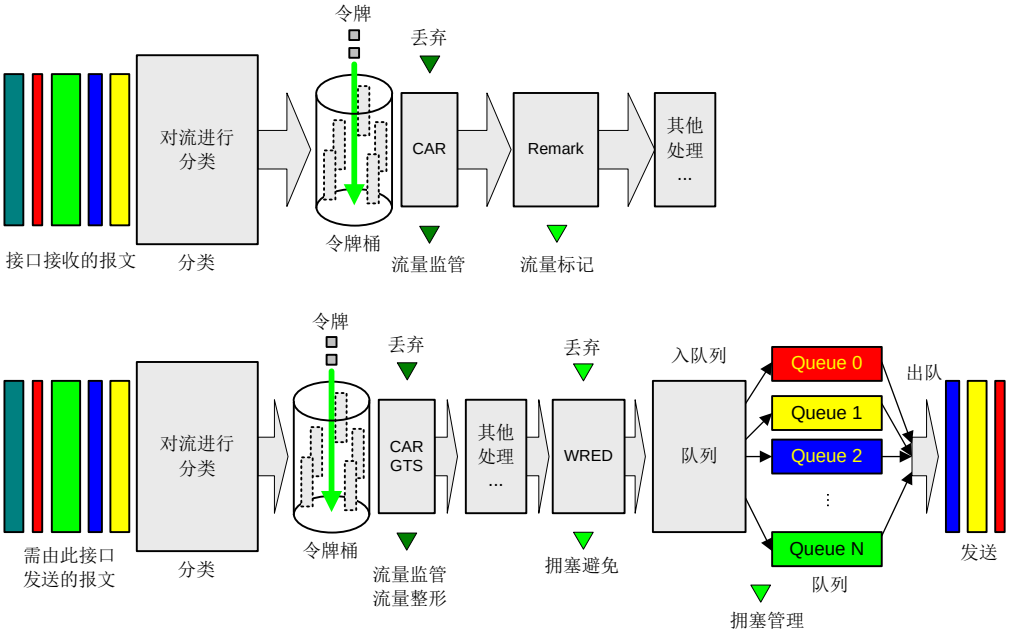
1.3 QoS技术在设备中的处理顺序

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。

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) 应用策略

- 基于接口应用QoS策略
- 基于VLAN应用QoS策略
- 基于全局应用QoS策略
- 基于上线用户应用QoS策略

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) （可选）配置类的描述信息。

description *text*

缺省情况下，未配置类的描述信息。

(4) 定义匹配数据包的规则。

if-match *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) 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 应用策略

2.6.1 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，QoS 策略对通过接口接收或发送的流量生效。
- 基于 VLAN 应用 QoS 策略，QoS 策略对通过同一个 VLAN 内所有接口接收或发送的流量生效。
- 基于全局应用 QoS 策略，QoS 策略对所有流量生效。
- 基于上线用户应用 QoS 策略，QoS 策略对通过上线用户接收或发送的流量生效。

2.6.2 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）。

2.6.3 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：

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

- 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。
- 本节中的“接口”指的是二层以太网接口。

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** }

缺省情况下，未在接口上应用 QoS 策略。

2.6.4 基于VLAN应用QoS策略

1. 功能简介

基于 VLAN 应用 QoS 策略可以对属于某个 VLAN 内的所有接口上的流量进行管理。

2. 配置限制和指导

基于 VLAN 应用的 QoS 策略时需要注意的是：

- 不能应用在动态 VLAN 上。
- 基于 VLAN 应用 QoS 策略时，该 QoS 策略会被所有成员设备上的 VLAN 应用，如果某个成员设备 QACL 资源不足，将导致 QoS 策略应用失败。此时需要先执行 **undo qos vlan-policy vlan** 命令删除基于 VLAN 应用的 QoS 策略，待预留足够资源后，再将 QoS 策略应用到该 VLAN 上。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 在指定 VLAN 上应用已创建的 QoS 策略。

qos vlan-policy *policy-name* **vlan** *vlan-id-list* { **inbound** | **outbound** }

缺省情况下，未在指定 VLAN 上应用 QoS 策略。

2.6.5 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 策略会被所有成员设备应用，如果某个成员设备 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** }

缺省情况下，未在全局应用 QoS 策略。

2.6.6 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帐户绑定的 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	表示对设备接收上线用户的流量（即上线用户发送的流量）应用策略
outbound	表示对设备发送给上线用户的流量（即上线用户接收的流量）应用策略

2.7 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	**display qos policy user-defined** [*policy-name* [**classifier** *classifier-name*]] [**slot** *slot-number*]

操作	命令
显示基于全局应用QoS策略的信息	**display qos policy global** [**slot** *slot-number*] [**inbound** | **outbound**]
显示接口上QoS策略的配置信息和运行情况	**display qos policy interface** [*interface-type interface-number*] [**inbound** | **outbound**]
显示用户上线后User Profile下应用的QoS策略的信息和运行情况	**display qos policy user-profile** [**name** *profile-name*] [**user-id** *user-id*] [**slot** *slot-number*] [**inbound** | **outbound**]
显示基于VLAN应用QoS策略的信息	**display qos vlan-policy** { **name** *policy-name* | **vlan** [*vlan-id*] } [**slot** *slot-number*] [**inbound** | **outbound**]
显示QoS和ACL资源的使用情况	**display qos-acl resource** [**slot** *slot-number*]
显示流行为的配置信息	**display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*]
显示类的配置信息	**display traffic classifier user-defined** [*classifier-name*] [**slot** *slot-number*]
清除全局应用QoS策略的统计信息	**reset qos policy global** [**inbound** | **outbound**]
清除VLAN应用QoS策略的统计信息	**reset qos vlan-policy** [**vlan** *vlan-id*] [**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“12.3 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。

设备仅支持以本地优先级（LP）作为设备调度优先级。

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（12.2 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。

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。

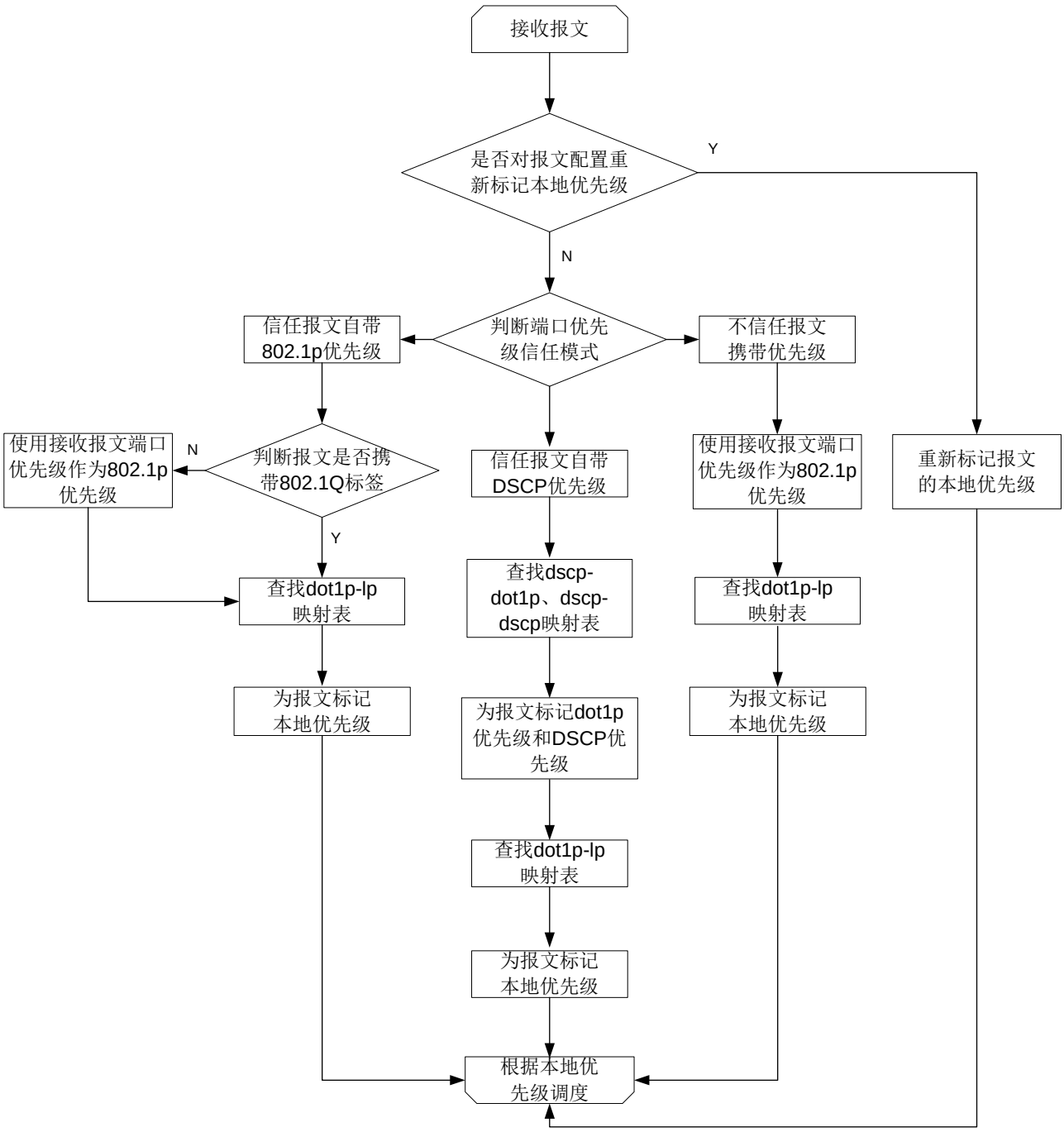
2. 端口优先级方式

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。

3.1.4 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



如果通过 QoS 策略同时配置重新标记报文的 dot1p 和 DSCP 优先级，设备仅以 QoS 策略修改后的 dot1p 优先级作为 dot1p-lp 优先级映射表的输入值，根据 dot1p-lp 优先级映射表得到报文的本地优先级。

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
- 配置端口优先级

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-dot1p | dscp-dscp }

(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“12.2 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。

2. 配置限制和指导

本节中的“接口”指的是二层以太网接口。

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) 配置优先级信任模式。

qos trust { **dot1p** | **dscp** }

设备不信任报文携带的优先级，会使用端口优先级作为报文的 802.1p 优先级进行优先级映射。

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 *priority-value*

缺省情况下，端口优先级为 0。

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-dot1p** | **dscp-dscp**]
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

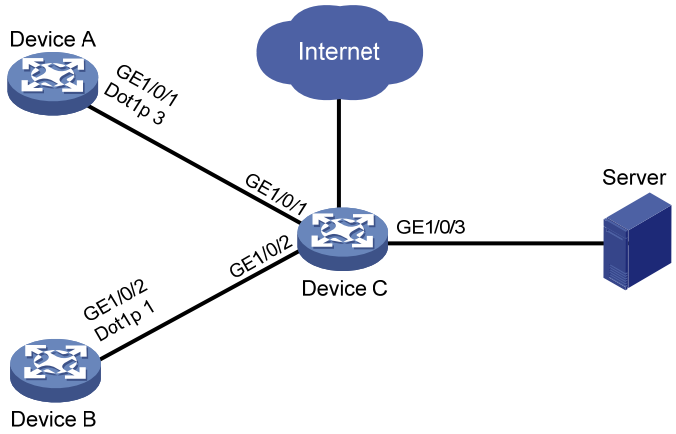
Device A 和 Device B 通过 Device C 实现互连。网络环境描述如下：

- Device A 通过端口 GigabitEthernet1/0/1 接入 Device C，向 Device C 发送 dot1p 值为 3 的报文；
- Device B 通过端口 GigabitEthernet1/0/2 接入 Device C，向 Device C 发送 dot1p 值为 1 的报文。

要求通过配置实现如下需求：如果 Device C 在接口 GigabitEthernet1/0/3 的出方向发生拥塞，则优先让 Device A 访问 Server。

2. 组网图

图3-2 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 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 上分别配置优先级信任模式为 **dot1p**。

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] qos trust dot1p
[DeviceC-GigabitEthernet1/0/1] quit
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] qos trust dot1p
[DeviceC-GigabitEthernet1/0/2] quit
```

(2) 方法二

在接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 上分别配置端口优先级，GigabitEthernet1/0/1 上配置的端口优先级值要高于 GigabitEthernet1/0/2 上配置的端口优先级值。（同时保证在接口 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 上没有配置信任模式。）

```
<DeviceC> system-view
[DeviceC] interface gigabitethernet 1/0/1
[DeviceC-GigabitEthernet1/0/1] qos priority 3
[DeviceC-GigabitEthernet1/0/1] quit
[DeviceC] interface gigabitethernet 1/0/2
[DeviceC-GigabitEthernet1/0/2] qos priority 1
[DeviceC-GigabitEthernet1/0/2] 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 GigabitEthernet1/0/1 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 GigabitEthernet1/0/2 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 GigabitEthernet1/0/3 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。

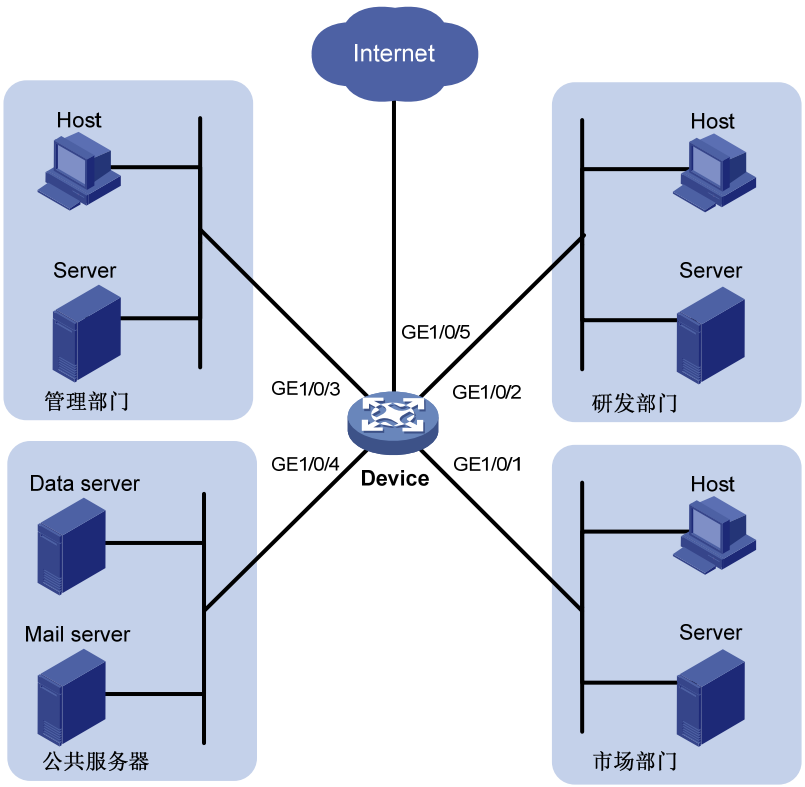
- 通过优先级映射将研发部门发出的报文放入出队列 6 中，优先进行处理；
- 通过优先级映射将管理部门发出的报文放入出队列 4 中，次优先进行处理；
- 通过优先级映射将市场部门发出的报文放入出队列 2 中，最后进行处理。

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3-3 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 GigabitEthernet1/0/1 的端口优先级为 3。

```
<Device> system-view
[Device] interface gigabitethernet 1/0/1
```

```
[Device-GigabitEthernet1/0/1] qos priority 3
[Device-GigabitEthernet1/0/1] quit
```

配置端口 GigabitEthernet1/0/2 的端口优先级为 4。

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] qos priority 4
[Device-GigabitEthernet1/0/2] quit
```

配置端口 GigabitEthernet1/0/3 的端口优先级为 5。

```
[Device] interface gigabitethernet 1/0/3
[Device-GigabitEthernet1/0/3] qos priority 5
[Device-GigabitEthernet1/0/3] 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 GigabitEthernet1/0/3 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 1/0/3
[Device-GigabitEthernet1/0/3] qos apply policy admin inbound
```

配置市场部门的重标记策略并应用到接口 GigabitEthernet1/0/1 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 1/0/1
```

```
[Device-GigabitEthernet1/0/1] qos apply policy market inbound
```

配置研发部门的重标记策略并应用到接口 GigabitEthernet1/0/2 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 1/0/2
[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，可以配置两个令牌桶（分别称为 C 桶和 E 桶）。以流量监管为例，分为单速率单桶双色算法、单速率双桶三色算法和双速率双桶三色算法。

(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：

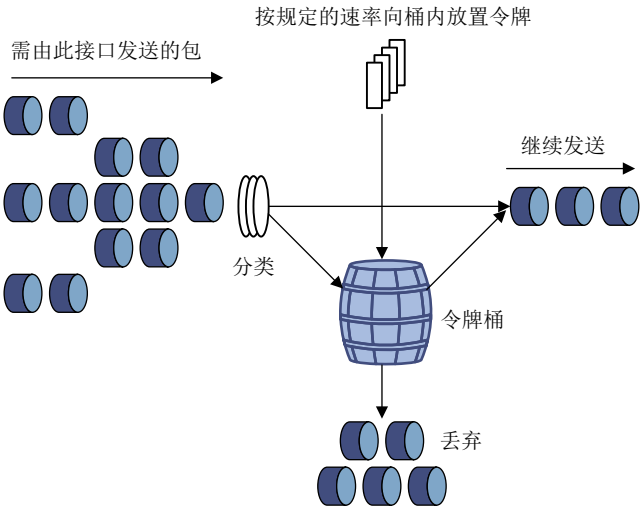
- 如果 C 桶有足够的令牌，报文被标记为 green，即绿色报文；
- 如果 C 桶令牌不足，但 E 桶有足够的令牌，报文被标记为 yellow，即黄色报文；
- 如果 C 桶和 E 桶都没有足够的令牌，报文被标记为 red，即红色报文。

4.1.2 流量监管

流量监管分为入和出两个方向，为了方便描述，下文以出方向为例。

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 报文不能占用超过 50%的网络带宽。如果发现某个连接的流量超标，流量监管可以选择丢弃报文，或重新配置报文的优先级。

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。

4.1.3 流量整形

说明

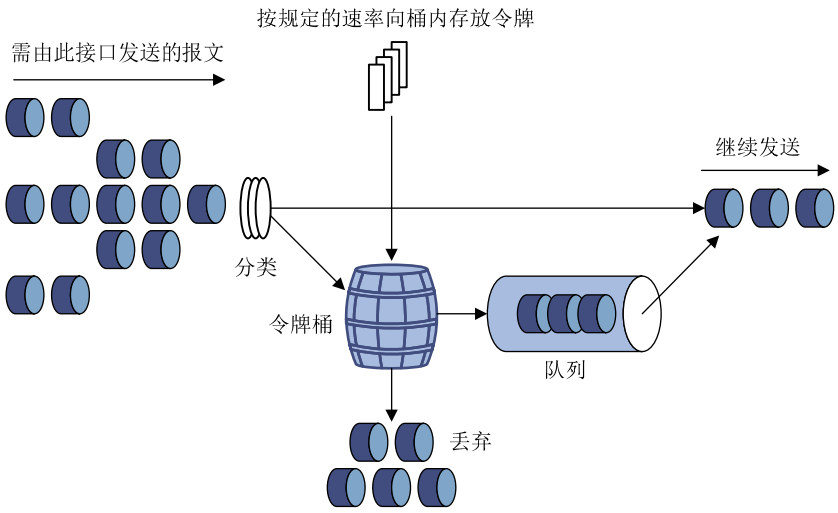
流量整形目前只支持出方向。

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：

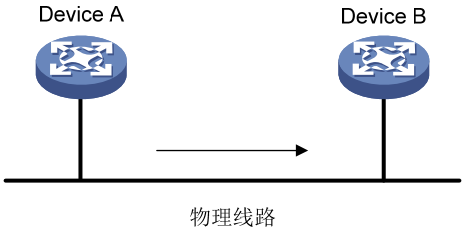
- 流量整形对流量监管中需要丢弃的报文进行缓存——通常是将它们放入缓冲区或队列内，如图 4-2 所示。当令牌桶有足够的令牌时，再均匀的向外发送这些被缓存的报文。
- 流量整形可能会增加延迟，而流量监管几乎不引入额外的延迟。

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。

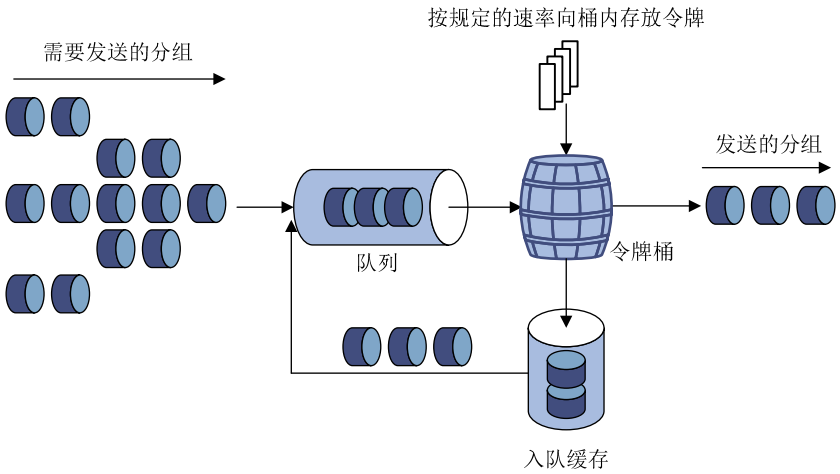
4.1.4 限速

限速分为入和出两个方向，为了方便描述，下文以出方向为例。

利用限速可以在一个接口上限制发送报文（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 流量监管、流量整形和限速配置限制和指导

相邻的数据帧之间存有一定的间隙，即帧间隙。帧间隙主要有如下作用：

- 便于设备区分不同的数据帧。
- 设备收到数据帧后有一定的时间处理当前数据帧并预接收下一数据帧。

流量监管、流量整形和限速中配置的承诺信息速率为剔除帧间隙后的单位时间的流量大小，所以流量监管、流量整形和限速实际生效的数值要略大于配置的承诺信息速率的数值。

4.3 配置流量监管

1. 配置限制和指导

设备支持基于接口、VLAN、全局和上线用户应用 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 *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。

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*] *

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“2.6 应用策略”

缺省情况下，未应用 QoS 策略。

4.4 配置流量整形

1. 配置限制和指导

本节中的“接口”指的是二层以太网接口。

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) 配置基于队列的流量整形。

qos gts queue *queue-id* **cir** *committed-information-rate* [**cbs** *committed-burst-size*]

undo qos gts queue *queue-id*

缺省情况下，接口上未配置流量整形。

4.5 配置限速

1. 配置限制和指导

本节中的“接口”指的是二层以太网接口。

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) 配置接口限速。

qos lr { **inbound** | **outbound** } **cir** *committed-information-rate* [**cbs** *committed-burst-size*]

缺省情况下，接口上未配置接口限速。

4.6 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
显示接口的流量整形配置情况	**display qos gts interface** [*interface-type interface-number*]
显示限速配置情况	**display qos lr interface** [*interface-type interface-number*]

操作	命令
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”）	**display qos-acl resource** [**slot** *slot-number*]
显示流量监管的相关配置信息	**display traffic behavior user-defined** [*behavior-name*] [**slot** *slot-number*]

4.7 流量监管、流量整形和限速典型配置举例

4.7.1 流量监管与流量整形典型配置举例

1. 配置需求

- 设备 Device A 通过接口 GigabitEthernet1/0/3 和设备 Device B 的接口 GigabitEthernet1/0/1 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 GigabitEthernet1/0/1 接口在同一网段
- Host B 与 Device A 的 GigabitEthernet1/0/2 接口在同一网段

要求在设备 Device A 上对接口 GigabitEthernet1/0/1 接收到的源自 Server 和 Host A 的报文流分别实施流量控制如下：

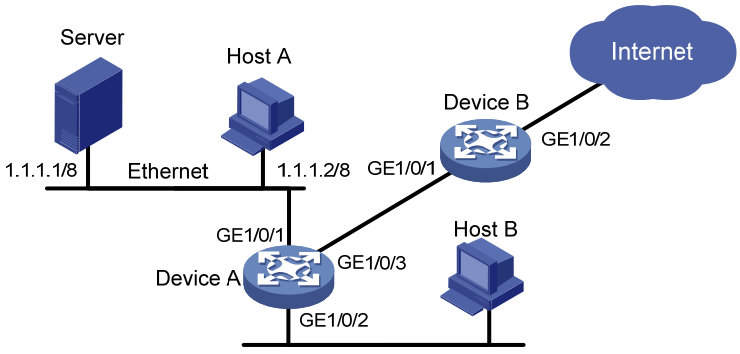
- 来自 Server 的报文流量约束为 10240kbps，流量小于 10240kbps 时可以正常发送，流量超过 10240kbps 时则将违规报文的优先级设置为 0 后进行发送；
- 来自 Host A 的报文流量约束为 2560kbps，流量小于 2560kbps 时可以正常发送，流量超过 2560kbps 时则丢弃违规报文；

对设备 Device B 的 GigabitEthernet1/0/1 和 GigabitEthernet1/0/2 接口收发报文有如下要求：

- Device B 的 GigabitEthernet1/0/1 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 GigabitEthernet1/0/2 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、流量整形配置组网图



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] 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 GigabitEthernet1/0/1 的入方向上。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] 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 GigabitEthernet1/0/1 的入方向上。
```
[DeviceB] interface gigabitethernet 1/0/1
[DeviceB-GigabitEthernet1/0/1] qos apply policy car_inbound inbound
```
将 QoS 策略 car_outbound 应用到接口 GigabitEthernet1/0/2 的出方向上。
```
[DeviceB] interface gigabitethernet 1/0/2
[DeviceB-GigabitEthernet1/0/2] qos apply policy car_outbound outbound
```

5 拥塞管理

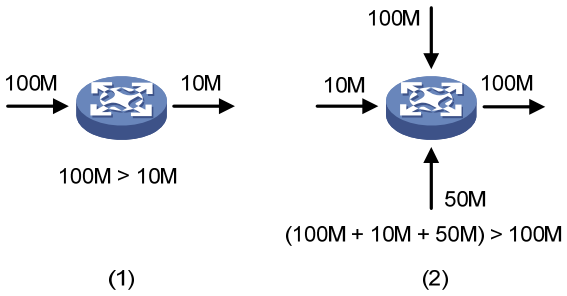
5.1 拥塞管理简介

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。以下图中的两种情况为例：

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。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。

5.1.2 设备支持的拥塞管理方法

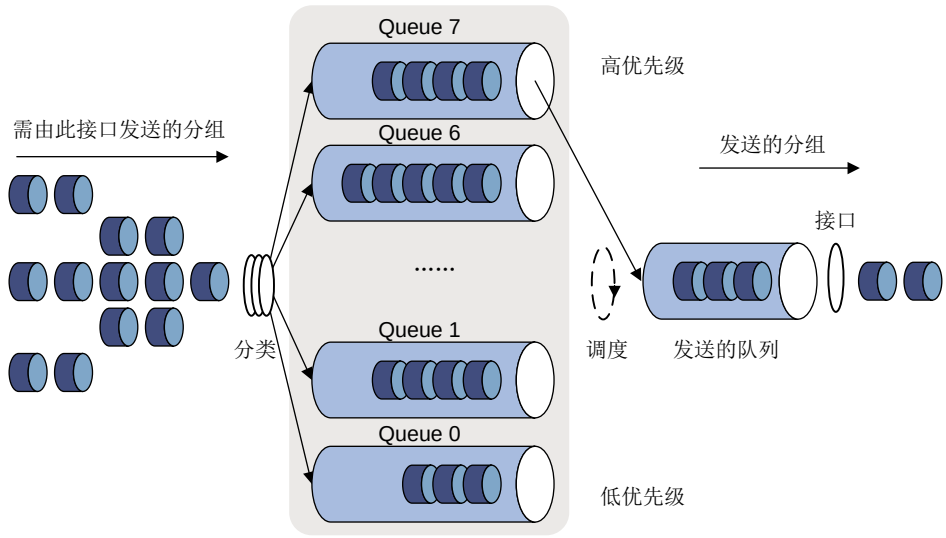
对于拥塞管理，一般采用队列技术，使用一个队列算法对流量进行分类，之后用某种优先级别算法将这些流量发送出去。

目前设备支持如下几种队列：

- SP 队列
- WRR 队列

1. SP队列

图5-2 SP 队列示意图



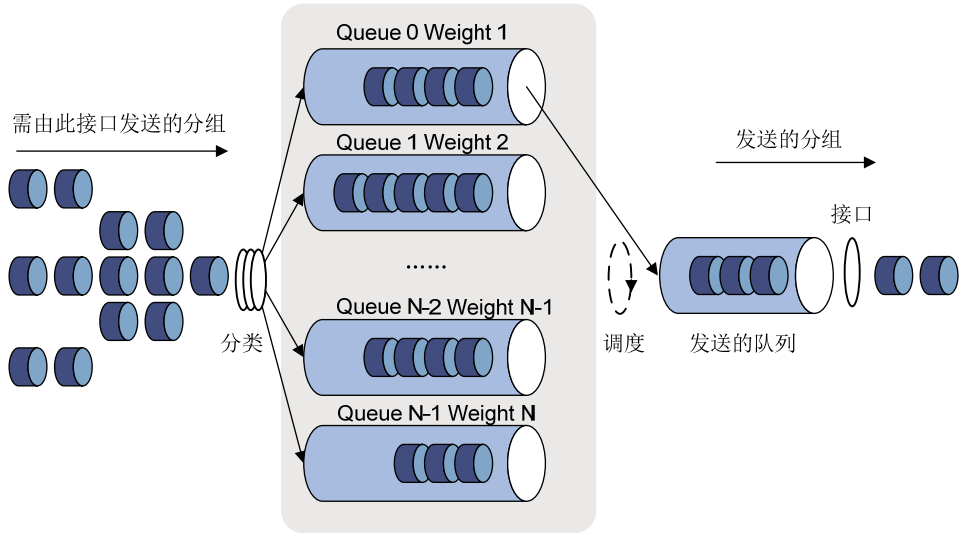
SP队列是针对关键业务类型应用设计的。关键业务有一个重要的特点，即在拥塞发生时要求优先获得服务以减小响应的延迟。以 图 5-2 为例，优先队列将端口的 8 个输出队列分成 8 类，依次为 7、6、5、4、3、2、1、0 队列，它们的优先级依次降低。

在队列调度时，SP 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。

SP 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。

2. WRR队列

图5-3 WRR 队列示意图



WRR 队列在队列之间进行轮流调度，保证每个队列都得到一定的服务时间。以端口有 8 个输出队列为例，WRR 可为每个队列配置一个加权值（依次为 w7、w6、w5、w4、w3、w2、w1、w0），加权值表示获取资源的比重。如一个 100Mbps 的端口，配置它的 WRR 队列的加权值为 50、50、30、30、10、10、10、10（依次对应 w7、w6、w5、w4、w3、w2、w1、w0），这样可以保证最低优先级队列至少获得 5Mbps 的带宽，解决了采用 SP 调度时低优先级队列中的报文可能长时间得不到服务的问题。

WRR 队列还有一个优点是，虽然多个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，这样带宽资源可以得到充分的利用。

WRR 队列分为：

- 基本 WRR 队列：基本 WRR 队列包含多个队列，用户可以定制各个队列的权重，WRR 按用户设定的参数进行加权轮询调度。
- 分组 WRR 队列：所有队列全部采用 WRR 调度，用户可以根据需要将输出队列划分为 WRR 优先级队列组 1 和 WRR 优先级队列组 2。进行队列调度时，设备首先在 WRR 优先级队列组 1 中进行轮询调度；优先级队列组 1 中没有报文发送时，设备才在优先级队列组 2 中进行轮询调度。当前设备仅支持 WRR 优先级队列组 1。

在分组 WRR 队列中，也可以配置队列加入 SP 分组，采用严格优先级调度算法。调度时先调度 SP 组，然后调度其他 WRR 优先组。

5.2 拥塞管理配置任务简介

拥塞管理配置任务如下：

- 配置接口队列

  请选择以下一项任务进行配置：
  - 配置SP队列
  - 配置WRR队列
  - 配置SP+WRR队列
- 配置队列调度策略

5.3 配置接口队列

5.3.1 接口队列配置限制和指导

本节中的“接口”指的是二层以太网接口。

display qos queue interface 命令显示信息中的 Queue ID、Queue name、Group、weight 字段下的信息组成了一个队列调度模板。

设备最多支持 8 个队列调度模板，除接口缺省队列调度模板、IRF 物理端口队列调度模板和设备 CPU 的预定义队列调度模板外，如果多个端口需要使用同一个自定义的队列调度模板，则至少需要存在一个剩余自定义队列调度模板。

当设备上自定义队列调度模板用尽后，可通过配置队列调度策略的方式配置接口的拥塞管理，关于队列调度策略的配置的配置方式，请查看 5.4 配置队列调度策略。

5.3.2 配置SP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 SP 队列。

qos sp

缺省情况下，接口采用 WRR 调度算法，各队列按照每次轮询可发送的字节数进行计算。

5.3.3 配置WRR队列

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 开启 WRR 队列。

qos wrr weight

接口采用 WRR 调度算法，各队列按照每次轮询可发送的报文个数进行计算。

(4) 配置分组 WRR 队列的参数。

qos wrr *queue-id* **group 1 weight** *schedule-value*

所有队列都处于 WRR 调度组 1 中，调度权重从队列 0 到 7 分别为 1、2、3、4、5、9、13、15，各队列按照每次轮询可发送的报文个数进行计算。

5.3.4 配置SP+WRR队列

(1) 进入系统视图。

system-view

(2) 进入接口视图。

interface *interface-type interface-number*

(3) 开启 WRR 队列。

qos wrr weight

缺省情况下，接口采用 WRR 调度算法，各队列按照每次轮询可发送的报文个数进行计算。

(4) 配置队列加入 SP 组，采用严格优先级调度算法。

qos wrr *queue-id* **group sp**

当接口使用 WRR 队列时，所有队列均处于 WRR 调度组 1 中。

(5) 配置队列加入 WRR 调度组。

qos wrr *queue-id* **group 1 weight** *schedule-value*

当接口使用 WRR 队列时，所有队列都处于 WRR 调度组 1 中，调度权重从队列 0 到 7 分别为 1、2、3、4、5、9、13、15，各队列按照每次轮询可发送的报文个数进行计算。

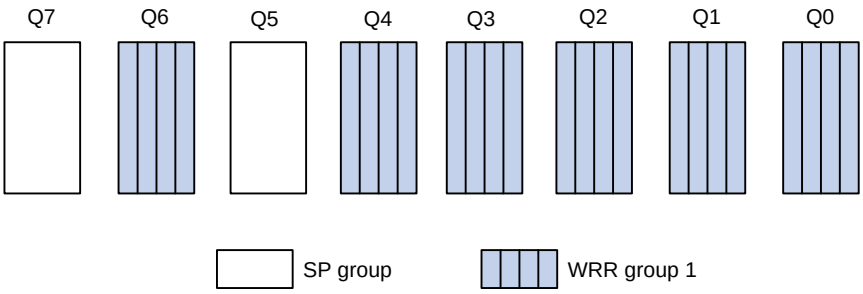
5.4 配置队列调度策略

5.4.1 队列调度策略简介

队列调度策略配置是在一个策略中配置各个队列的调度参数，最后通过在接口应用该策略来实现拥塞管理功能。

队列调度策略中的队列支持两种调度方式：SP和WRR。在一个队列调度策略中支持SP和WRR的混合配置。具体调度方式，可参见 5.1.2 设备支持的拥塞管理方法中介绍的内容。以SP和WRR分组混合配置为例，调度关系如 图 5-4 所示。

图5-4 SP 和 WRR 混合配置图



- 设备优先调度 SP 组队列中的报文。
- 队列 7（即图中的 Q7，下同）在 SP 组中优先级最高，该队列的报文优先发送。
- 队列 5 在 SP 组中优先级次之，队列 7 为空时发送本队列的报文。
- 队列 6、4、3、2、1、0 之间按照权重轮询调度，在队列 7、5 为空时调度 WRR 分组 1。

5.4.2 配置限制和指导

在配置队列调度策略时需要注意的是：

- 队列调度策略中队列的调度参数支持动态修改，从而方便修改已经应用的队列调度策略。
- 本节中的“接口”指的是二层以太网接口。

5.4.3 创建队列调度策略

(1) 进入系统视图。

system-view

(2) 创建队列调度策略，并进入相应的队列调度策略视图。

qos qmprofile *profile-name*

(3) （可选）配置队列调度参数。请选择其中一项进行配置。

- 配置严格优先级调度：

  queue *queue-id* **sp**

- 配置加权轮询调度：

  queue *queue-id* **wrr group** *group-id* { **weight** | **byte-count** } *schedule-value*

缺省情况下，队列调度策略的内容是所有队列均采用 SP 方式调度。

5.4.4 应用队列调度策略

(1) 进入系统视图。

system-view

(2) 进入已创建的队列调度策略视图。

qos qmprofile *profile-name*

(3) 请依次执行以下命令在接口出方向上应用队列调度策略。

interface *interface-type interface-number*

qos apply qmprofile *profile-name*

缺省情况下，接口上未应用队列调度策略。

5.4.5 队列调度策略典型配置举例

1. 配置需求

接口 GigabitEthernet1/0/1 的队列调度方式如下：

- 队列 7 优先级最高，该队列报文优先发送。
- 队列 0～6 之间按照权重轮询调度，属于 WRR 分组，使用报文个数作为调度权重，分别为 2、1、2、4、6、8、10，在队列 7 为空时调度 WRR 分组。

2. 配置步骤

进入系统视图。

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

创建队列调度策略 qm1。

```
[Sysname] qos qmprofile qm1
[Sysname-qmprofile-qm1]
```

配置队列 7 为 SP 队列。

```
[Sysname-qmprofile-qm1] queue 7 sp
```

配置队列 0～6 属于 WRR 分组 1，使用报文个数作为调度权重，分别为 2、1、2、4、6、8、10。

```
[Sysname-qmprofile-qm1] queue 0 wrr group 1 weight 2
[Sysname-qmprofile-qm1] queue 1 wrr group 1 weight 1
[Sysname-qmprofile-qm1] queue 2 wrr group 1 weight 2
[Sysname-qmprofile-qm1] queue 3 wrr group 1 weight 4
[Sysname-qmprofile-qm1] queue 4 wrr group 1 weight 6
[Sysname-qmprofile-qm1] queue 5 wrr group 1 weight 8
[Sysname-qmprofile-qm1] queue 6 wrr group 1 weight 10
[Sysname-qmprofile-qm1] quit
```

把队列调度策略 qm1 应用到接口 GigabitEthernet1/0/1 上。

```
[Sysname] interface gigabitethernet 1/0/1
[Sysname-GigabitEthernet1/0/1] qos apply qmprofile qm1
```

配置完成后，接口 GigabitEthernet1/0/1 按指定方式进行队列调度。

5.5 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** 命令可以显示配置后队列的运行情况，通过查看显示信息验证配置的效果。

表5-1 拥塞管理的显示和维护

操作	命令
显示队列调度策略的配置信息	**display qos qmprofile configuration** [*profile-name*] [**slot** *slot-number*]
显示接口的队列调度策略应用信息	**display qos qmprofile interface** [*interface-type interface-number*]
显示端口队列出方向的统计信息	**display qos queue-statistics interface outbound**
显示SP队列	**display qos queue sp interface** [*interface-type interface-number*]
显示WRR队列的配置	**display qos queue wrr interface** [*interface-type interface-number*]

6 流量过滤

6.1 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。

6.2 流量过滤配置限制和指导

设备支持基于接口、VLAN、全局和上线用户应用 QoS 策略配置流量过滤。

6.3 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 *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“2.6 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。

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

6.4 流量过滤典型配置举例

6.4.1 流量过滤基本组网配置举例

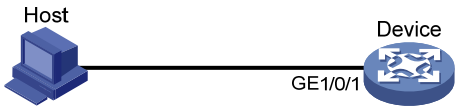
1. 组网需求

Host 通过接口 GigabitEthernet1/0/1 接入设备 Device。

配置流量过滤功能，对接口 GigabitEthernet1/0/1 接收的源端口号不等于 21 的 TCP 报文进行丢弃。

2. 组网图

图6-1 流量过滤基本组网图



3. 配置步骤

定义高级 ACL 3000，匹配源端口号不等于 21 的数据流。

```
<Device> system-view
[Device] acl advanced 3000
[Device-acl-ipv4-adv-3000] rule 0 permit tcp source-port neq 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 GigabitEthernet1/0/1 的入方向上。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] qos apply policy policy inbound
```

7 重标记

7.1 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 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。

重标记可以和优先级映射功能配合使用，具体请参见“3 优先级映射”。

7.2 配置重标记

1. 配置限制和指导

设备支持基于接口、VLAN、全局和上线用户应用 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 *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“2.6 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。

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

7.3 重标记典型配置举例

7.3.1 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：

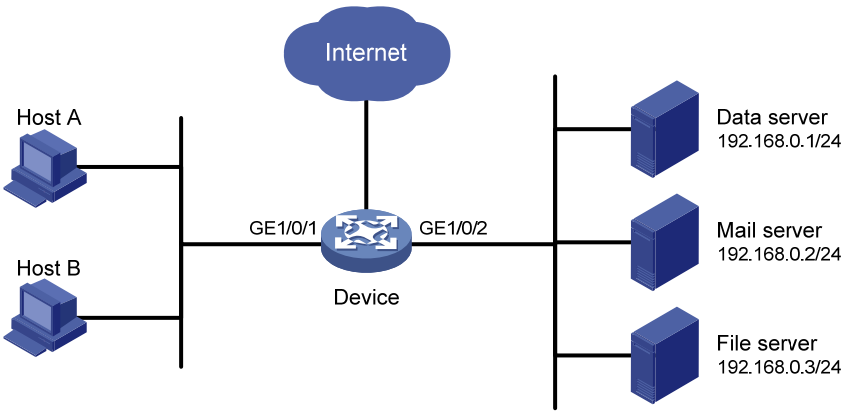
- Host A 和 Host B 通过端口 GigabitEthernet1/0/1 接入 Device；
- 数据库服务器、邮件服务器和文件服务器通过端口 GigabitEthernet1/0/2 接入 Device。

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7-1 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 GigabitEthernet1/0/1 上。

```
[Device] interface gigabitethernet 1/0/1
[Device-GigabitEthernet1/0/1] qos apply policy policy_server inbound
[Device-GigabitEthernet1/0/1] quit
```

8 Nest

8.1 Nest简介

Nest 用来为符合流分类的流添加一层 VLAN Tag，使携带该 VLAN Tag 的报文通过对应 VLAN。例如，为从用户网络进入运营商网络的 VLAN 报文添加外层 VLAN Tag，使其携带运营商网络分配的 VLAN Tag 穿越运营商网络。

8.2 Nest配置限制和指导

设备支持基于接口、VLAN、全局和上线用户入方向应用 QoS 策略配置 Nest。

如果该接口已使能 QinQ 功能，且 QoS 策略中配置了匹配 VLAN Tag VLAN ID 的规则，则该接口必须允许 VLAN ID 匹配的报文带 Tag 通过，才能保证 QinQ 功能和 Nest 同时生效，否则 Nest 将不会生效。

8.3 配置Nest

(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 *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. 配置添加报文的外层 VLAN Tag 的动作。

nest top-most vlan *vlan-id*

缺省情况下，未配置添加报文外层 VLAN Tag 动作。

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“2.6 应用策略”

缺省情况下，未应用 QoS 策略。

(6) （可选）显示 Nest 的相关配置信息。

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

8.4 Nest典型配置举例

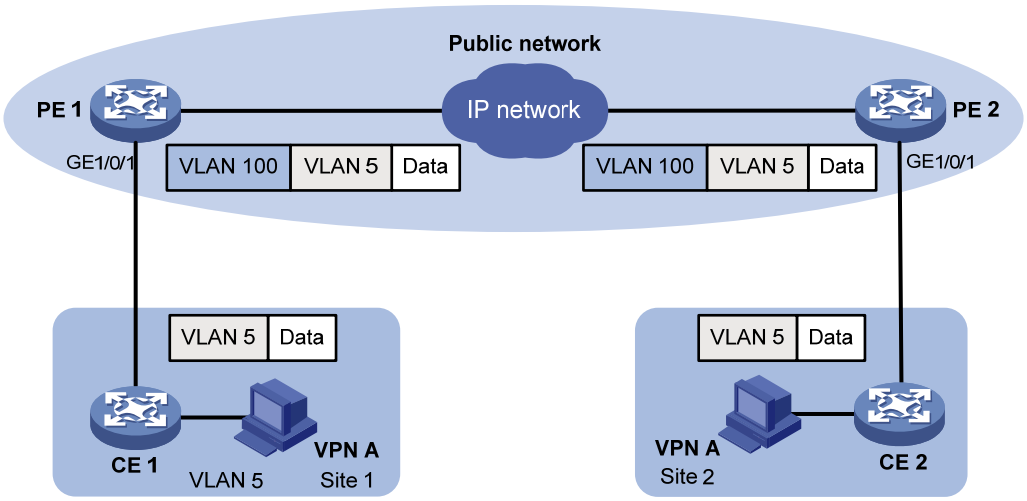
8.4.1 Nest基本功能配置举例

1. 组网需求

- VPN A 中的 Site 1 和 Site 2 是某公司的两个分支机构，利用 VLAN 5 承载业务。由于分处不同地域，这两个分支机构采用了服务提供商（SP）所提供的 VPN 接入服务，SP 将 VLAN 100 分配给这两个分支机构使用。
- 该公司希望其下属的这两个分支机构可以跨越 SP 的网络实现互通。

2. 组网图

图8-1 Nest 基本功能组网图



3. 配置步骤

(1) 配置 PE 1

定义类 test 的匹配规则为：匹配从 GigabitEthernet1/0/1 收到的 VLAN ID 值为 5 的报文。

```
<PE1> system-view
```

```
[PE1] traffic classifier test
[PE1-classifier-test] if-match service-vlan-id 5
[PE1-classifier-test] quit
```

在流行为 test 上配置如下动作：添加 VLAN ID 为 100 的外层 VLAN Tag。

```
[PE1] traffic behavior test
[PE1-behavior-test] nest top-most vlan 100
[PE1-behavior-test] quit
```

在策略 test 中为类 test 指定采用流行为 test。

```
[PE1] qos policy test
[PE1-qospolicy-test] classifier test behavior test
[PE1-qospolicy-test] quit
```

配置下行端口 GigabitEthernet1/0/1 为 Hybrid 端口且允许 VLAN 100 的报文不携带 VLAN Tag 通过。

```
[PE1] interface gigabitethernet 1/0/1
[PE1-GigabitEthernet1/0/1] port link-type hybrid
[PE1-GigabitEthernet1/0/1] port hybrid vlan 100 untagged
```

在下行端口 GigabitEthernet1/0/1 的入方向上应用上行策略 test。

```
[PE1-GigabitEthernet1/0/1] qos apply policy test inbound
[PE1-GigabitEthernet1/0/1] quit
```

配置上行端口 GigabitEthernet1/0/2 为 Trunk 端口且允许 VLAN 100 通过。

```
[PE1] interface gigabitethernet 1/0/2
[PE1-GigabitEthernet1/0/2] port link-type trunk
[PE1-GigabitEthernet1/0/2] port trunk permit vlan 100
[PE1-GigabitEthernet1/0/2] quit
```

(2) 配置 PE 2

PE 2 的配置与 PE 1 完全一致，这里不再赘述。

9 流量重定向

9.1 流量重定向简介

流量重定向就是将符合流分类的流重定向到其他地方进行处理。

目前支持的流量重定向包括以下几种：

- 重定向到 CPU：对于需要 CPU 处理的报文，可以通过配置上送给 CPU。
- 重定向到接口：对于收到需要由某个接口处理的报文时，可以通过配置重定向到此接口。

9.2 流量重定向配置限制和指导

配置流量重定向时需要注意的是：

- 设备支持基于接口、VLAN、全局和上线用户的入方向应用 QoS 策略配置流量重定向。
- 在配置重定向动作时，同一个流行为中，多次配置重定向命令，最后一次配置的生效。

9.3 配置流量重定向

(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 *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. 配置流量重定向动作。

redirect { **cpu** | **interface** *interface-type interface-number* }

缺省情况下，未配置流量重定向动作。

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“2.6 应用策略”

缺省情况下，未应用 QoS 策略。

(6) （可选）显示流量重定向的相关配置信息。

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

9.4 流量重定向典型配置举例

9.4.1 重定向至接口配置举例

1. 组网需求

网络环境描述如下：

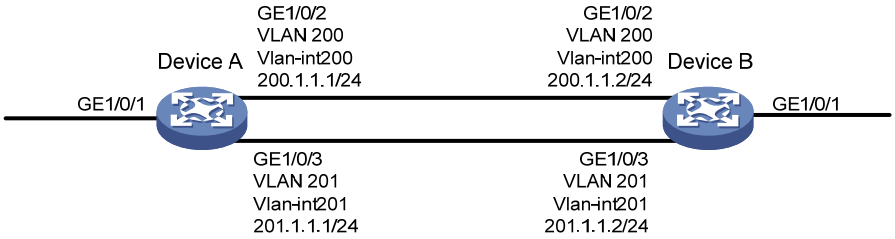
- Device A 通过两条链路与 Device B 连接，同时 Device A 和 Device B 各自连接其他的设备；
- Device A 上的端口 GigabitEthernet1/0/2 的链路类型为 Trunk 类型，且允许 VLAN200 和 VLAN201 的报文通过；
- Device A 上的端口 GigabitEthernet1/0/2 和 Device B 上的端口 GigabitEthernet1/0/2 属于 VLAN 200；
- Device A 上的端口 GigabitEthernet1/0/3 和 Device B 上的端口 GigabitEthernet1/0/3 属于 VLAN 201；
- Device A 上接口 Vlan-interface200 的 IP 地址为 200.1.1.1/24，接口 Vlan-interface201 的 IP 地址为 201.1.1.1/24；
- Device B 上接口 Vlan-interface200 的 IP 地址为 200.1.1.2/24，接口 Vlan-interface201 的 IP 地址为 201.1.1.2/24。

配置重定向至接口，满足如下需求：

- 将 Device A 的端口 GigabitEthernet1/0/1 接收到的源 IP 地址为 2.1.1.1 的报文转发至 GigabitEthernet1/0/2；
- 将 Device A 的端口 GigabitEthernet1/0/1 接收到的源 IP 地址为 2.1.1.2 的报文转发至 GigabitEthernet1/0/3；
- 对于 Device A 的端口 GigabitEthernet1/0/1 接收到的其它报文，按照查找路由表的方式进行转发。

2. 组网图

图9-1 重定向至接口配置组网图



3. 配置步骤

定义基本 ACL 2000，对源 IP 地址为 2.1.1.1 的报文进行分类。

```
<DeviceA> system-view
[DeviceA] acl basic 2000
[DeviceA-acl-ipv4-basic-2000] rule permit source 2.1.1.1 0
[DeviceA-acl-ipv4-basic-2000] quit
```

定义基本 ACL 2001，对源 IP 地址为 2.1.1.2 的报文进行分类。

```
[DeviceA] acl basic 2001
[DeviceA-acl-ipv4-basic-2001] rule permit source 2.1.1.2 0
[DeviceA-acl-ipv4-basic-2001] quit
```

定义类 classifier_1，匹配基本 ACL 2000。

```
[DeviceA] traffic classifier classifier_1
[DeviceA-classifier-classifier_1] if-match acl 2000
[DeviceA-classifier-classifier_1] quit
```

定义类 classifier_2，匹配基本 ACL 2001。

```
[DeviceA] traffic classifier classifier_2
[DeviceA-classifier-classifier_2] if-match acl 2001
[DeviceA-classifier-classifier_2] quit
```

定义流行为 behavior_1，动作为重定向至 GigabitEthernet1/0/2。

```
[DeviceA] traffic behavior behavior_1
[DeviceA-behavior-behavior_1] redirect interface gigabitethernet 1/0/2
[DeviceA-behavior-behavior_1] quit
```

定义流行为 behavior_2，动作为重定向至 GigabitEthernet1/0/3。

```
[DeviceA] traffic behavior behavior_2
[DeviceA-behavior-behavior_2] redirect interface gigabitethernet 1/0/3
[DeviceA-behavior-behavior_2] quit
```

定义策略 policy，为类 classifier_1 指定流行为 behavior_1，为类 classifier_2 指定流行为 behavior_2。

```
[DeviceA] qos policy policy
[DeviceA-qospolicy-policy] classifier classifier_1 behavior behavior_1
[DeviceA-qospolicy-policy] classifier classifier_2 behavior behavior_2
[DeviceA-qospolicy-policy] quit
```

将策略 policy 应用到端口 GigabitEthernet1/0/1 的入方向上。

```
[DeviceA] interface gigabitethernet 1/0/1
[DeviceA-GigabitEthernet1/0/1] qos apply policy policy inbound
```

10 全局CAR

10.1 全局CAR简介

全局 CAR 是在全局创建的一种策略，所有应用该策略的数据流将共同接受全局 CAR 的监管。全局 CAR 分为聚合 CAR 和分层 CAR。

10.1.1 聚合CAR

聚合 CAR 是指能够对多个业务流使用同一个 CAR 进行流量监管，即如果多个端口应用同一聚合 CAR，则这多个端口的流量之和必须在此聚合 CAR 设定的流量监管范围之内。

10.1.2 分层CAR

分层 CAR 是一种更灵活的流量监管策略，用户可以在为每个流单独配置 CAR 动作（或聚合 CAR）的基础上，再通过分层 CAR 对多个流的流量总和进行限制。

分层 CAR 与普通 CAR（或聚合 CAR）的结合应用有两种模式：

- and：在该模式下，对于多条数据流应用同一个分层 CAR，必须每条流满足各自的普通 CAR（或聚合 CAR）配置，同时各流量之和又满足分层 CAR 的配置，流量才能正常通过。and 模式适用于严格限制流量带宽的环境，分层 CAR 的限速配置通常小于各流量自身 CAR 的限速值之和。例如对于 Internet 流量，可以使用普通 CAR 将数据流 1 和数据流 2 各自限速为 240kbps，再使用分层 CAR 限制总流量为 320kbps。当不存在数据流 1 时，数据流 2 可以用达到自身限速上限的速率访问 Internet，如果存在数据流 1，则两个数据流不能超过各自限速且总速率不能超过 320kbps。
- or：在该模式下，对于多条数据流应用同一个分层 CAR，只要每条流满足各自的普通 CAR（或聚合 CAR）配置或者各流量之和满足分层 CAR 配置，流量即可正常通过。or 模式适用于保证高优先级业务带宽的环境，分层 CAR 的限速值通常等于或大于各流量自身的限速值之和。例如对于视频流量，使用普通 CAR 将数据流 1 和数据流 2 各自限速 240kbps，再使用分层 CAR 限制总流量为 560kbps，则当数据流 1 的流量不足 240kbps 时，即使数据流 2 的流量达到了 320kbps，仍然可以正常通过。

10.2 全局CAR配置限制和指导

当前设备仅支持聚合 CAR。

当前设备支持基于接口、VLAN、全局和上线用户入方向应用 QoS 策略配置聚合 CAR。

10.3 配置聚合CAR

(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 *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) 配置聚合 CAR。

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

qos car *car-name* **aggregative 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，未配置聚合 CAR。

(4) 定义流行为。

a. 进入流行为视图。

traffic behavior *behavior-name*

b. 在流行为中应用聚合 CAR 动作。

car name *car-name*

缺省情况下，流行为中未应用聚合 CAR 动作。

(5) 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

(6) 应用 QoS 策略。

具体配置请参见“2.6 应用策略”

缺省情况下，未应用 QoS 策略。

10.4 全局CAR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全局 CAR 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** 命令可以清除全局 CAR 统计信息。

表10-1 全局 CAR 显示和维护

操作	命令
显示全局CAR的配置和统计信息	**display qos car name** [*car-name*]
清除全局CAR的统计信息	**reset qos car name** [*car-name*]

11 流量统计

11.1 流量统计简介

流量统计就是通过与类关联，对符合匹配规则的流进行统计。例如，可以统计从某个源 IP 地址发送的报文，然后管理员对统计信息进行分析，根据分析情况采取相应的措施。

11.2 流量统计配置限制和指导

设备支持基于接口、VLAN、全局和上线用户应用 QoS 策略配置流量统计。

11.3 配置流量统计

(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 *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. 为流行为配置流量统计动作。

accounting { **byte** | **packet** }

缺省情况下，未配置流量统计动作。

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“2.6 应用策略”

缺省情况下，未应用 QoS 策略。

(6) （可选）显示流量统计的相关配置信息。

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

11.4 流量统计典型配置举例

11.4.1 流量统计基本组网配置举例

1. 组网需求

用户网络描述如下：Host 通过接口 GigabitEthernet1/0/1 接入设备 Device。

配置流量统计功能，对接口 GigabitEthernet1/0/1 接收的源 IP 地址为 1.1.1.1/24 的报文进行统计。

2. 组网图

图11-1 流量统计基本组网图

3. 配置步骤

定义基本 ACL 2000，对源 IP 地址为 1.1.1.1 的报文进行分类。

```
<Device> system-view
[Device] acl basic 2000
[Device-acl-ipv4-basic-2000] rule permit source 1.1.1.1 0
[Device-acl-ipv4-basic-2000] quit
```

定义类 classifier_1，匹配基本 ACL 2000。

```
[Device] traffic classifier classifier_1
[Device-classifier-classifier_1] if-match acl 2000
[Device-classifier-classifier_1] quit
```

定义流行为 behavior_1，动作为流量统计。

```
[Device] traffic behavior behavior_1
[Device-behavior-behavior_1] accounting packet
[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 GigabitEthernet1/0/1 的入方向上。

```
[Device] interface gigabitethernet 1/0/1
```

```
[Device-GigabitEthernet1/0/1] qos apply policy policy inbound
[Device-GigabitEthernet1/0/1] quit
```

查看配置后流量统计的情况。

```
[Device] display qos policy interface gigabitethernet 1/0/1
Interface: GigabitEthernet1/0/1
  Direction: Inbound
  Policy: policy
   Classifier: classifier_1
     Operator: AND
     Rule(s) :
      If-match acl 2000
     Behavior: behavior_1
      Accounting enable:
        28529 (Packets)
```

12 附录

12.1 附录 A 缩略语表

表12-1 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
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
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
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
LLQ	Low Latency Queuing	低时延队列
LR	Line Rate	限速
LSP	Label Switched Path	标签交换路径

缩略语	英文全名	中文解释
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
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

12.2 附录 B 缺省优先级映射表

说明

dscp-dscp 映射表的缺省映射关系为：映射输出值等于输入值。

表12-2 dot1p-lp 缺省映射关系

映射输入索引	dot1p-lp 映射
dot1p	lp
0	0

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

表12-3 dscp-dot1p 缺省映射关系

映射输入索引	dscp-dot1p 映射
dscp	dot1p
0～7	0
8～15	1
16～23	2
24～31	3
32～39	4
40～47	5
48～55	6
56～63	7

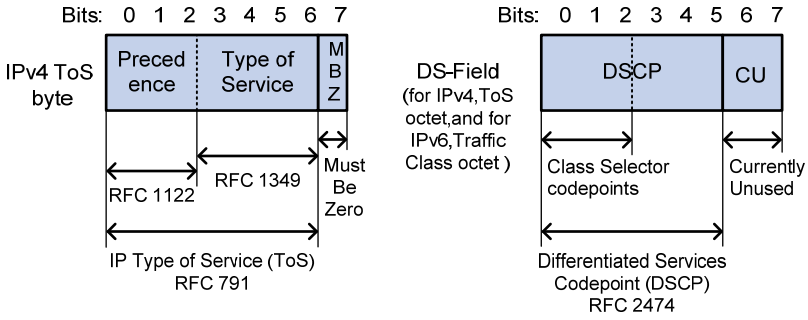
表12-4 端口优先级和 LP 映射关系

端口优先级	LP
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7

12.3 附录 C 各种优先级介绍

12.3.1 IP优先级和DSCP优先级

图12-1 ToS 和 DS 域



如 图 12-1 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12-5 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12-6 DSCP 优先级说明

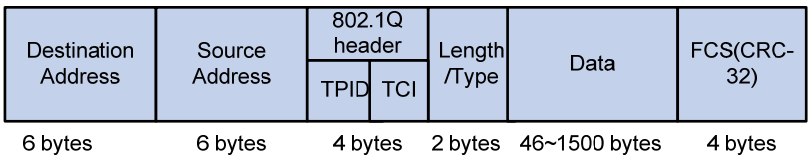
DSCP 优先级（十进制）	DSCP 优先级（二进制）	关键字
46	101110	ef
10	001010	af11
12	001100	af12
14	001110	af13
18	010010	af21
20	010100	af22
22	010110	af23

DSCP 优先级（十进制）	DSCP 优先级（二进制）	关键字
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）

12.3.2 802.1p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12-2 带有 802.1Q 标签头的以太网帧



如 图 12-2 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 12-3 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12-3 802.1Q 标签头

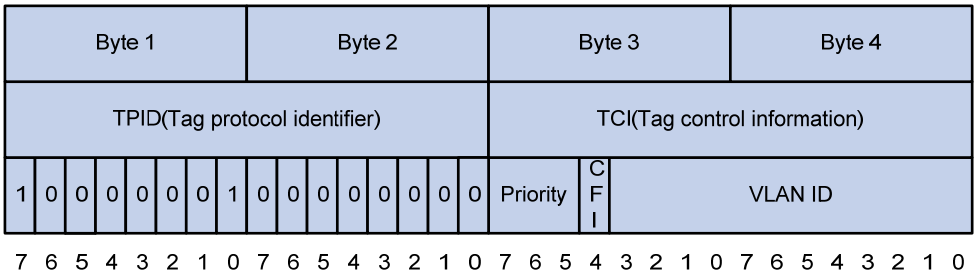


表12-7 802.1p 优先级说明

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

目 录

1 数据缓冲区

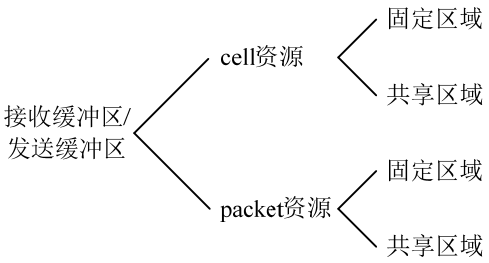
1.1 数据缓冲区简介

1.1.1 数据缓冲区分类

数据缓冲区用来临时存储报文，以免报文丢失。如 图 1-1 所示，数据缓冲区分为接收缓冲区和发送缓冲区：

- 接收缓冲区：用来缓存接收的数据。当设备的 CPU 繁忙时，端口不能立即将收到的报文交给 CPU 处理，会将数据暂时存储到接收缓冲区；
- 发送缓冲区：用来缓存发送的数据。当网络拥塞时，端口不能立即发送数据，为防止数据丢失，会将数据暂时存储到发送缓冲区。

图1-1 数据缓冲区划分示意图



1.1.2 cell资源和packet资源

发送数据缓冲区和接收数据缓冲区在缓存数据时，都会同时用到两类资源：

- cell 资源：用来存储数据包的内容，端口会根据报文的实际大小占用相应大小的 cell 资源。比如一个 cell 资源是 208 字节，当发送的报文是 128 字节时，端口会给它分配一个 cell 资源，当发送的报文是 300 字节时，端口会给它分配两个 cell 资源。
- packet 资源：用来存储报文的指针，指针指明报文在 cell 资源中的存放位置。设备每发送/接收一个数据包，无论该数据包的长度是多少，均占用 1 个 packet 资源。

当端口接收/发送报文时，既使用相当于报文长度的 cell 资源，同时也使用相当于报文数量的 packet 资源。

1.1.3 固定区域和共享区域

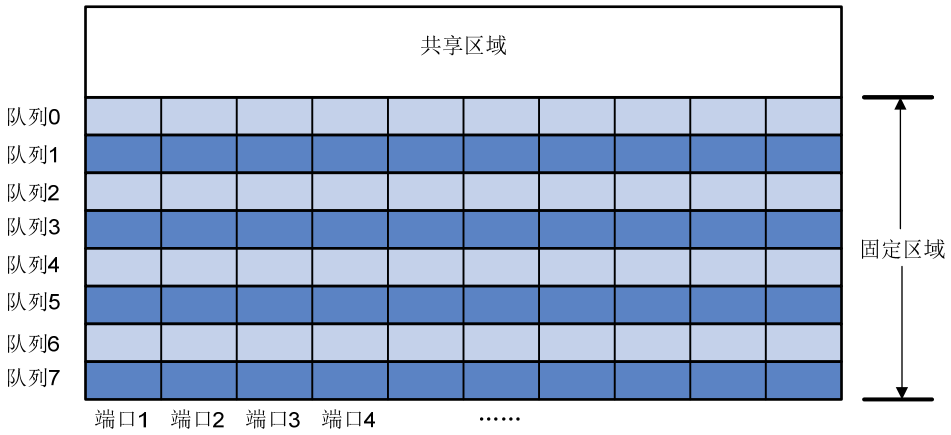
cell 资源和 packet 资源又分别分为共享区域和固定区域。

- 固定区域是按队列划分的，每个队列又按端口均分，如 图 1-2 所示。如果设备CPU繁忙或网络发生拥塞，设备在接收或发送报文时，会根据一定的策略将报文分发到相应的队列。如果该端口的该队列缓冲区满，则放到共享区域中的相应队列；如果共享区域中该队列满，则将报文丢弃。在固定缓冲区中，系统会根据用户的配置给队列预留指定大小的空间，即便该队

列没有报文存储需求，其他队列也不能抢占。给队列预留的空间均分给每个端口的，即使某端口的某队列没有报文存储需求，其他端口也不能抢占。

- 共享缓冲区只按队列划分，不再按端口均分，如 图 1-2 所示。系统会根据用户配置以及实际需要收发报文的数量决定每个队列实际可占用的缓冲区的大小。如果某个队列没有报文存储需求，则其他队列会抢占该队列的配额。对于某个队列的缓冲区，所有端口接收或发送的报文采用抢占的方式，先到先得，如果资源耗尽，则后到达的报文将被丢弃。

图1-2 固定区域和共享区域划分示意图



1.2 数据缓冲区配置限制和指导

用户可以使用以下两种方式配置数据缓冲区：

- 配置 Burst 功能自动分配缓冲区
- 手工配置数据缓冲区

开启数据缓冲区自动分配功能后，如下缓冲区的比例将发生变化：

- 每个队列最多可使用缓冲区中固定区域比例。
- 每个队列最多可使用缓冲区中共享区域的比例。
- 缓冲区中共享区域所占比例。

开启前后的具体变化情况，可通过 **display buffer** 命令查看。

需要注意的是，以上两种数据缓冲区的配置方式不能同时使用，如果已经使用某一种方式进行了配置，则必须先取消该方式的配置之后，才能使用另外一种方式进行配置。否则，配置失败。

数据缓冲区的配置比较复杂，而且对设备的转发功能有重要的影响，建议用户不要轻易修改数据缓冲区的缺省参数。在需要较大的缓存空间时，建议使用 Burst 功能来自动分配缓冲区。

1.3 数据缓冲区配置任务简介

数据缓冲区配置任务如下：

- 配置Burst功能自动分配缓冲区
- 手工分配数据缓冲区

1.4 配置Burst功能自动分配缓冲区

1. 功能简介

在下列情况下，Burst 功能可以提供更好的报文缓存功能和流量转发性能：

- 广播或者组播报文流量密集，瞬间突发大流量的网络环境中。
- 报文从高速链路进入设备，由低速链路转发出去。
- 报文从相同速率的多个端口同时进入设备，由一个相同速率的端口转发出去。

开启数据缓冲区自动分配功能前后，设备缓冲区的分配情况会有较大的变化，可以通过 **display buffer** 命令查看开启前后设备数据缓冲区的分配情况。

2. 配置步骤

(1) 进入系统视图。

system-view

(2) 开启 Burst 功能。

burst-mode enable

缺省情况下，未开启 Burst 功能。

1.5 手工分配数据缓冲区

1. 功能简介

设备上整个数据缓冲区的大小是固定的，用户可以手工配置共享区域的大小，其余部分将自动成为固定区域。

缺省情况下，所有队列均分共享区域/固定区域。用户可以手工调整指定队列最多可使用的共享区域/固定区域的大小，其它未配置的队列最多可使用的共享区域/固定区域的大小仍遵循缺省值。

2. 配置限制和指导

对于 Release 6126P13 及以上版本，在组播视频流网络场景中，可以通过如下配置缓解视频播放卡顿问题：配置队列最多可使用的 cell 共享区域的比例、cell 缓冲区共享区域的比例、队列最多可使用的 packet 共享区域的比例及 packet 缓冲区共享区域的比例均为 100。

上述配置与 Burst 功能冲突，请先通过 **undo burst-mode enable** 命令关闭 Burst 功能。

3. 配置步骤

(1) 进入系统视图。

system-view

(2) 配置数据缓冲区分配规则。请至少选择其中一项进行配置。

- 配置数据缓冲区中共享区域的比例。

  buffer egress [**slot** *slot-number*] { **cell** | **packet** } **total-shared ratio** *ratio*

  缺省情况下，cell 缓冲区中共享区域所占比例为 52%；packet 缓冲区中共享区域所占比例为 50%。

- 配置队列最多可使用的共享区域的比例。

buffer egress [**slot** *slot-number*] { **cell** | **packet** } [**queue** *queue-id*] **shared ratio** *ratio*

缺省情况下，每个端口最多可使用的 packet 资源中共享区域的比例为 10%；每个端口最多可使用的 cell 资源中共享区域的比例为 10%。

各队列最多可使用的共享区域的大小将根据 **buffer queue shared** 配置，以及实际需要收发报文的数量决定。

- 配置指定队列最多可使用的固定区域的比例。

  buffer egress [**slot** *slot-number*] { **cell** | **packet** } **queue** *queue-id* **guaranteed ratio** *ratio*

  缺省情况下，每个队列最多可使用的 cell 资源中固定区域比例为 12%；每个队列最多可使用的 packet 资源中固定区域的比例为 12%。

  所有队列所配置的固定区域大小之和，不应超过可配置的总固定区域大小，否则配置失败。

(3) 应用数据缓冲区分配规则。

buffer apply

执行本命令后，所配置的数据缓冲区分配规则才会生效。如果需要修改数据缓冲区分配规则，需要先取消应用，再修改分配规则并重新应用。

1.6 数据缓冲区显示和维护

在完成上述配置后，在任意视图下执行 **display** 命令可以查看数据缓冲区的手工配置效果。

表1-1 设备管理显示和维护

操作	命令
显示数据缓冲区的大小	**display buffer** [**slot** *slot-number*] [**queue** [*queue-id*]]

目 录

1 时间段

1.1 时间段简介

时间段（Time Range）定义了一个时间范围。用户通过创建一个时间段并在某业务中将其引用，就可使该业务在此时间段定义的时间范围内生效。

譬如，当一个 ACL 规则只需在某个特定时间范围内生效时，就可以先配置好这个时间段，然后在配置该 ACL 规则时引用此时间段，这样该 ACL 规则就只能在该时间段定义的时间范围内生效。

在一个时间段中，可以使用以下两种方式定义时间范围：

- 周期时间段：表示以一周为周期（如每周一的 8 至 12 点）循环生效的时间段。
- 绝对时间段：表示在指定时间范围内（如 2015 年 1 月 1 日 8 点至 2015 年 1 月 3 日 18 点）生效的时间段。

当一个时间段内包含有多个周期时间段和绝对时间段时，系统将先分别取各周期时间段的并集和各绝对时间段的并集，再取这两个并集的交集作为该时间段最终生效的时间范围。

1.2 时间段配置限制和指导

如果一个业务所引用的时间段尚未配置或已被删除，该业务将不会生效。

用户最多可创建 1024 个不同名称的时间段。一个时间段内最多可以包含 32 个周期时间段和 12 个绝对时间段。

1.3 配置时间段

(1) 进入系统视图。

system-view

(2) 创建时间段。

time-range *time-range-name* { *start-time* **to** *end-time days* [**from** *time1 date1*] [**to** *time2 date2*] | **from** *time1 date1* [**to** *time2 date2*] | **to** *time2 date2* }

如果指定的时间段已经创建，则本命令可以修改时间段的时间范围。

1.4 时间段显示和维护

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

表1-1 时间段显示和维护

配置	命令
显示时间段的配置和状态信息	**display time-range** { *time-range-name* | **all** }

1.5 时间段典型配置举例

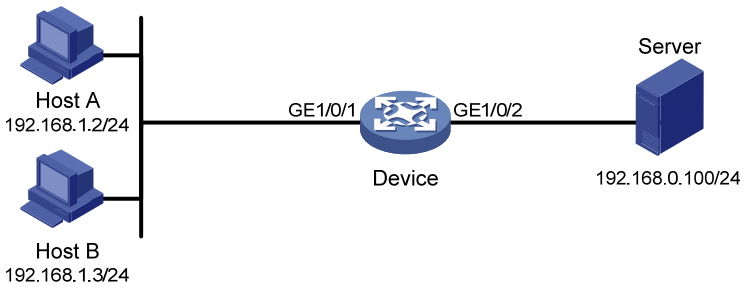
1.5.1 时间段基本组网配置举例

1. 组网需求

要求通过在 Device 上配置 ACL 规则，实现在 2015 年 6 月到 2015 年 12 月之间每周工作日的 8 点到 18 点只允许 Host A 访问 Server。

2. 组网图

图1-1 时间段典型配置组网图



3. 配置步骤

创建名为 work 的时间段，其时间范围为 2015 年 6 月到 2015 年 12 月之间每周工作日的 8 点到 18 点。

```
<Device> system-view
[Device] time-range work 08:00 to 18:00 working-day from 00:00 6/1/2015 to 24:00 12/31/2015
```

创建 IPv4 基本 ACL 2001，并制订如下规则：在名为 work 的时间段内只允许来自 192.168.1.2/32 的报文通过、禁止来自其它 IP 地址的报文通过。

```
[Device] acl basic 2001
[Device-acl-ipv4-basic-2001] rule permit source 192.168.1.2 0 time-range work
[Device-acl-ipv4-basic-2001] rule deny source any time-range work
[Device-acl-ipv4-basic-2001] quit
```

应用 IPv4 基本 ACL 2001 对接口 GigabitEthernet1/0/2 出方向上的报文进行过滤。

```
[Device] interface gigabitethernet 1/0/2
[Device-GigabitEthernet1/0/2] packet-filter 2001 outbound
[Device-GigabitEthernet1/0/2] quit
```

4. 验证配置

配置完成后，在 Device 上可以使用 **display time-range** 命令查看时间段的配置和状态信息：

显示所有时间段的配置和状态信息。

```
[Device] display time-range all
Current time is 13:58:35 6/19/2015 Friday

Time-range : work (Active)
 08:00 to 18:00 working-day
 from 00:00 6/1/2015 to 00:00 1/1/2012
```

由此可见，时间段 work 已经生效。