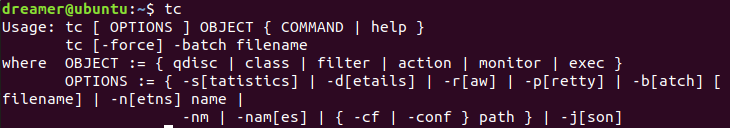
**Linux下TC使用调研**

# 一、TC原理介绍

Linux操作系统中的流量控制器TC（Traffic Control）用于Linux内核的流量控制，主要是通过在输出端口处建立一个队列来实现流量控制。

接收包从输入接口进来后，经过流量限制丢弃不符合规定的数据包，由输入多路分配器进行判断选择：如果接收包的目的是本主机，那么将该包送给上层处理；否则需要进行转发，将接收包交到转发块处理。转发块同时也接收本主机上层产生的包。转发块通过查看路由表，决定所处理包的下一跳。然后，对包进行排列以便将它们传送到输出接口。一般我们只能限制网卡发送的数据包，不能限制网卡接收的数据包，所以我们可以通过改变发送次序来控制传输速率。Linux流量控制主要是在输出接口排列时进行处理和实现的。

通过在ubuntu系统中命令行输入TC命令，可以得知TC命令的使用方法如下图所示：



# 二、TC规则

## 1、流量控制方式

流量控制包括以下几种方式：

1.SHAPING(限制)：当流量被限制，它的传输速率就被控制在某个值以下。限制值可以大大小于有效带宽，这样可以平滑突发数据流量，使网络更为稳定。shaping（限制）只适用于向外的流量。

2.SCHEDULING(调度)：通过调度数据包的传输，可以在带宽范围内，按照优先级分配带宽。SCHEDULING(调度)也只适于向外的流量。

3.POLICING(策略)：SHAPING用于处理向外的流量，而POLICIING(策略)用于处理接收到的数据。

4.DROPPING(丢弃)：如果流量超过某个设定的带宽，就丢弃数据包，不管是向内还是向外。

## 2、流量控制处理对象

流量的处理由三种对象控制，它们是：qdisc(排队规则)、class(类别)和filter(过滤器)。

1.QDISC(排队规则)：QDisc(排队规则)是queueing discipline的简写，它是理解流量控制(traffic control)的基础。无论何时，内核如果需要通过某个网络接口发送数据包，它都需要按照为这个接口配置的qdisc(排队规则)把数据包加入队列。然后，内核会尽可能多地从qdisc里面取出数据包，把它们交给网络适配器驱动模块。最简单的QDisc是pfifo它不对进入的数据包做任何的处理，数据包采用先入先出的方式通过队列。不过，它会保存网络接口一时无法处理的数据包。

QDISC的类别如下：

（1）、不可分类QDisc：

[p|b]fifo：使用最简单的qdisc，纯粹的先进先出。只有一个参数：limit，用来设置队列的长度,pfifo是以数据包的个数为单位；bfifo是以字节数为单位。

pfifo\_fast：在编译内核时，如果打开了高级路由器(Advanced Router)编译选项，pfifo\_fast就是系统的标准QDISC。它的队列包括三个波段(band)。在每个波段里面，使用先进先出规则。而三个波段(band)的优先级也不相同，band 0的优先级最高，band 2的最低。如果band里面有数据包，系统就不会处理band 1里面的数据包，band 1和band 2之间也是一样。数据包是按照服务类型(Type of Service,TOS)被分配多三个波段(band)里面的。

Red：red是Random Early Detection(随机早期探测)的简写。如果使用这种QDISC，当带宽的占用接近于规定的带宽时，系统会随机地丢弃一些数据包。它非常适合高带宽应用。

Sfq：sfq是Stochastic Fairness Queueing的简写。它按照会话(session--对应于每个TCP连接或者UDP流)为流量进行排序，然后循环发送每个会话的数据包。

Tbf：tbf是Token Bucket Filter的简写，适合于把流速降低到某个值。

2．不可分类QDisc的配置

如果没有可分类QDisc，不可分类QDisc只能附属于设备的根。它们的用法如下：

添加一个不可分类的QDisc：

tc qdisc add dev DEV root QDISC QDISC-PARAMETERS

删除一个不可分类QDisc：

tc qdisc del dev DEV root

一个网络接口上如果没有设置QDisc，pfifo\_fast就作为缺省的QDisc。

（2）、CLASSFUL QDISC(分类QDisc)：

CBQ：CBQ是Class Based Queueing(基于类别排队)的缩写。它实现了一个丰富的连接共享类别结构，既有限制(shaping)带宽的能力，也具有带宽优先级管理的能力。带宽限制是通过计算连接的空闲时间完成的。空闲时间的计算标准是数据包离队事件的频率和下层连接(数据链路层)的带宽。

HTB：HTB是Hierarchy Token Bucket的缩写。通过在实践基础上的改进，它实现了一个丰富的连接共享类别体系。使用HTB可以很容易地保证每个类别的带宽，虽然它也允许特定的类可以突破带宽上限，占用别的类的带宽。HTB可以通过TBF(Token Bucket Filter)实现带宽限制，也能够划分类别的优先级。

PRIO：PRIO QDisc不能限制带宽，因为属于不同类别的数据包是顺序离队的。使用PRIO QDisc可以很容易对流量进行优先级管理，只有属于高优先级类别的数据包全部发送完毕，才会发送属于低优先级类别的数据包。为了方便管理，需要使用iptables或者ipchains处理数据包的服务类型(Type Of Service,ToS)。

CLASS(类)：某些QDisc(排队规则)可以包含一些类别，不同的类别中可以包含更深入的QDisc(排队规则)，通过这些细分的QDisc还可以为进入的队列的数据包排队。通过设置各种类别数据包的离队次序，QDisc可以为设置网络数据流量的优先级。

FILTER(过滤器)：Filter(过滤器)用于为数据包分类，决定它们按照何种QDisc进入队列。无论何时数据包进入一个划分子类的类别中，都需要进行分类。分类的方法可以有多种，使用fileter(过滤器)就是其中之一。使用filter(过滤器)分类时，内核会调用附属于这个类(class)的所有过滤器，直到返回一个判决。如果没有判决返回，就作进一步的处理，而处理方式和QDISC有关。需要注意的是，filter(过滤器)是在QDisc内部，它们不能作为主体。

## 3、操作原理

类(Class)组成一个树，每个类都只有一个父类，而一个类可以有多个子类。某些QDisc(例如：CBQ和HTB)允许在运行时动态添加类，而其它的QDisc(例如：PRIO)不允许动态建立类。允许动态添加类的QDisc可以有零个或者多个子类，由它们为数据包排队。此外，每个类都有一个叶子QDisc，默认情况下，这个叶子QDisc使用pfifo的方式排队，我们也可以使用其它类型的QDisc代替这个默认的QDisc。而且，这个叶子叶子QDisc有可以分类，不过每个子类只能有一个叶子QDisc。 当一个数据包进入一个分类QDisc，它会被归入某个子类。我们可以使用以下三种方式为数据包归类，不过不是所有的QDisc都能够使用这三种方式。

tc过滤器(tc filter)：如果过滤器附属于一个类，相关的指令就会对它们进行查询。过滤器能够匹配数据包头所有的域，也可以匹配由ipchains或者iptables做的标记。

服务类型(Type of Service)：某些QDisc有基于服务类型（Type of Service,ToS）的内置的规则为数据包分类。

skb->priority：用户空间的应用程序可以使用SO\_PRIORITY选项在skb->priority域设置一个类的ID。

树的每个节点都可以有自己的过滤器，但是高层的过滤器也可以直接用于其子类。

如果数据包没有被成功归类，就会被排到这个类的叶子QDisc的队中。相关细节在各个QDisc的手册页中。

## 4、命名规则

所有的QDisc、类和过滤器都有ID。ID可以手工设置，也可以有内核自动分配。ID由一个主序列号和一个从序列号组成，两个数字用一个冒号分开。

QDISC：一个QDisc会被分配一个主序列号，叫做句柄(handle)，然后把从序列号作为类的命名空间。句柄采用象10:一样的表达方式。习惯上，需要为有子类的QDisc显式地分配一个句柄。

类：在同一个QDisc里面的类分享这个QDisc的主序列号，但是每个类都有自己的从序列号，叫做类识别符(classid)。类识别符只与父QDisc有关，和父类无关。类的命名习惯和QDisc的相同。

过滤器：过滤器的ID有三部分，只有在对过滤器进行散列组织才会用到。详情请参考tc-filters手册页。

## 5、单位

tc命令的所有参数都可以使用浮点数，可能会涉及到以下计数单位。

1．带宽或者流速单位：

kbps 千字节／秒

mbps 兆字节／秒

kbit KBits／秒

mbit MBits／秒

bps或者一个无单位数字 字节数／秒

2．数据的数量单位：

kb或者k 千字节

mb或者m 兆字节

mbit 兆bit

kbit 千bit

b或者一个无单位数字 字节数

3．时间的计量单位：

s、sec或者secs 秒

ms、msec或者msecs 分钟

us、usec、usecs或者一个无单位数字 微秒

# 三、TC命令

tc可以使用以下命令对QDisc、类和过滤器进行操作：

add

在一个节点里加入一个QDisc、类或者过滤器。添加时，需要传递一个祖先作为参数，传递参数时既可以使用ID也可以直接传递设备的根。如果要建立一个QDisc或者过滤器，可以使用句柄(handle)来命名；如果要建立一个类，可以使用类识别符(classid)来命名。

remove

删除有某个句柄(handle)指定的QDisc，根QDisc(root)也可以删除。被删除QDisc上的所有子类以及附属于各个类的过滤器都会被自动删除。

change

以替代的方式修改某些条目。除了句柄(handle)和祖先不能修改以外，change命令的语法和add命令相同。换句话说，change命令不能一定节点的位置。

replace

对一个现有节点进行近于原子操作的删除／添加。如果节点不存在，这个命令就会建立节点。

link

只适用于DQisc，替代一个现有的节点。

例：

tc qdisc [ add | change | replace | link ] dev DEV [ parent qdisc-id | root ] [ handle qdisc-id ] qdisc [ qdisc specific parameters ]

tc class [ add | change | replace ] dev DEV parent qdisc-id [ classid class-id ] qdisc [ qdisc specific parameters ]

tc filter [ add | change | replace ] dev DEV [ parent qdisc-id | root ] protocol protocol prio priority filtertype [ filtertype specific parameters ] flowid flow-id

tc [-s | -d ] qdisc show [ dev DEV ]

tc [-s | -d ] class show dev DEV tc filter show dev DEV

# 四、具体操作

Linux流量控制主要分为建立队列、建立分类和建立过滤器三个方面。

1、基本实现步骤为：

1.针对网络物理设备（如以太网卡eth0）绑定一个队列QDisc；2.在该队列上建立分类class；3.为每一分类建立一个基于路由的过滤器filter；4.最后与过滤器相配合，建立特定的路由表。

2、环境模拟实例:

流量控制器上的以太网卡(eth0) 的IP地址为192.168.1.66，在其上建立一个CBQ队列。假设包的平均大小为1000字节，包间隔发送单元的大小为8字节，可接收冲突的发送最长包数目为20字节。

假如有三种类型的流量需要控制：

1.是发往主机1的，其IP地址为192.168.1.24。其流量带宽控制在8Mbit，优先级为2；

2.是发往主机2的，其IP地址为192.168.1.30。其流量带宽控制在1Mbit，优先级为1；

3.是发往子网1的，其子网号为192.168.1.0，子网掩码为255.255.255.0。流量带宽控制 在1Mbit，优先级为6。

1. 建立队列

一般情况下，针对一个网卡只需建立一个队列。将一个cbq队列绑定到网络物理设备eth0上，其编号为1:0；网络物理设备eth0的实际带宽为10 Mbit，包的平均大小为1000字节；包间隔发送单元的大小为8字节，最小传输包大小为64字节。

对应的输入命令行为：

·tc qdisc add dev eth0 root handle 1: cbq bandwidth 10Mbit avpkt 1000 cell 8 mpu 64

2. 建立分类

分类建立在队列之上。一般情况下，针对一个队列需建立一个根分类，然后再在其上建立子分类。对于分类，按其分类的编号顺序起作用，编号小的优先；一旦符合某个分类匹配规则，通过该分类发送数据包，则其后的分类不再起作用。

1）创建根分类1:1；分配带宽为10Mbit，优先级别为8。

输入命令行为：

·tc class add dev eth0 parent 1:0 classid 1:1 cbq bandwidth 10Mbit rate 10Mbit maxburst 20 allot 1514 prio 8 avpkt 1000 cell 8 weight 1Mbit

该队列的最大可用带宽为10Mbit，实际分配的带宽为10Mbit，可接收冲突的发送最长包数目为20字节；最大传输单元加MAC头的大小为1514字节，优先级别为8，包的平均大小为1000字节，包间隔发送单元的大小为8字节，相应于实际带宽的加权速率为1Mbit。

2）创建分类1:2，其父分类为1:1，分配带宽为8Mbit，优先级别为2。

输入命令行为：

·tc class add dev eth0 parent 1:1 classid 1:2 cbq bandwidth 10Mbit rate 8Mbit maxburst 20 allot 1514 prio 2 avpkt 1000 cell 8 weight 800Kbit split 1:0 bounded

该队列的最大可用带宽为10Mbit，实际分配的带宽为 8Mbit，可接收冲突的发送最长包数目为20字节；最大传输单元加MAC头的大小为1514字节，优先级别为1，包的平均大小为1000字节，包间隔发送单元的大小为8字节，相应于实际带宽的加权速率为800Kbit，分类的分离点为1:0，且不可借用未使用带宽。

3）创建分类1:3，其父分类为1:1，分配带宽为1Mbit，优先级别为1。

输入命令行为：

·tc class add dev eth0 parent 1:1 classid 1:3 cbq bandwidth 10Mbit rate 1Mbit maxburst 20 allot 1514 prio 1 avpkt 1000 cell 8 weight 100Kbit split 1:0

该队列的最大可用带宽为10Mbit，实际分配的带宽为 1Mbit，可接收冲突的发送最长包数目为20字节；最大传输单元加MAC头的大小为1514字节，优先级别为2，包的平均大小为1000字节，包间隔发送单元的大小为8字节，相应于实际带宽的加权速率为100Kbit，分类的分离点为1:0。

4）创建分类1:4，其父分类为1:1，分配带宽为1Mbit，优先级别为6。

输入命令行为：

·tc class add dev eth0 parent 1:1 classid 1:4 cbq bandwidth 10Mbit rate 1Mbit maxburst 20 allot 1514 prio 6 avpkt 1000 cell 8 weight 100Kbit split 1:0

该队列的最大可用带宽为10Mbit，实际分配的带宽为1Mbit，可接收冲突的发送最长包数目为20字节；最大传输单元加MAC头的大小为1514字节，优先级别为6，包的平均大小为1000字节，包间隔发送单元的大小为8字节，相应于实际带宽的加权速率为100Kbit，分类的分离点为1:0。

3. 建立过滤器

过滤器主要服务于分类。一般只需针对根分类提供一个过滤器，然后为每个子分类提供路由映射。

1） 应用路由分类器到cbq队列的根，父分类编号为1:0；过滤协议为ip，优先级别为100，过滤器为基于路由表。

·tc filter add dev eth0 parent 1:0 protocol ip prio 100 route

2） 建立路由映射分类1:2, 1:3, 1:4

·tc filter add dev eth0 parent 1:0 protocol ip prio 100 route to 2 flowid 1:2

·tc filter add dev eth0 parent 1:0 protocol ip prio 100 route to 3 flowid 1:3

·tc filter add dev eth0 parent 1:0 protocol ip prio 100 route to 4 flowid 1:4

4.建立路由

该路由是与前面所建立的路由映射一一对应。

1）发往主机192.168.1.24的数据包通过分类2转发(分类2的速率8Mbit)

·ip route add 192.168.1.24 dev eth0 via 192.168.1.66 realm 2

2）发往主机192.168.1.30的数据包通过分类3转发(分类3的速率1Mbit)

·ip route add 192.168.1.30 dev eth0 via 192.168.1.66 realm 3

3）发往子网192.168.1.0/24的数据包通过分类4转发(分类4的速率1Mbit)

·ip route add 192.168.1.0/24 dev eth0 via 192.168.1.66 realm 4

注：一般对于流量控制器所直接连接的网段建议使用IP主机地址流量控制限制，不要使用子网流量控制限制。如一定需要对直连子网使用子网流量控制限制，则在建立该子网的路由映射前，需将原先由系统建立的路由删除，才可完成相应步骤。

5.维护

主要包括对队列、分类、过滤器和路由的增添、修改和删除。增添动作一般依照"队列->分类->过滤器->路由"的顺序进行；修改动作则没有什么要求；删除则依照"路由->过滤器->分类->队列"的顺序进行。

1）队列的维护

一般对于一台流量控制器来说，出厂时针对每个以太网卡均已配置好一个队列了，通常情况下对队列无需进行增添、修改和删除动作了。

2）分类的维护

增添：增添动作通过tc class add命令实现，如前面所示

修改：修改动作通过tc class change命令实现，如下所示：

·tc class change dev eth0 parent 1:1 classid 1:2 cbq bandwidth 10Mbit rate 7Mbit maxburst 20 allot 1514 prio 2 avpkt 1000 cell 8 weight 700Kbit split 1:0 bounded

对于bounded命令应慎用，一旦添加后就进行修改，只可通过删除后再添加来实现。

删除：删除动作只在该分类没有工作前才可进行，一旦通过该分类发送过数据，则无法删除它了。因此，需要通过shell文件方式来修改，通过重新启动来完成删除动作。

3）过滤器的维护

增添：增添动作通过tc filter add命令实现，如前面所示。

修改：修改动作通过tc filter change命令实现，如下所示：

·tc filter change dev eth0 parent 1:0 protocol ip prio 100 route to 10 flowid 1:8

删除：删除动作通过tc filter del命令实现，如下所示：

·tc filter del dev eth0 parent 1:0 protocol ip prio 100 route to 10

4）与过滤器一一映射路由的维护

增添：增添动作通过ip route add命令实现，如前面所示。

修改：修改动作通过ip route change命令实现，如下所示：

·ip route change 192.168.1.30 dev eth0 via 192.168.1.66 realm 8

删除：删除动作通过ip route del命令实现，如下所示：

·ip route del 192.168.1.30 dev eth0 via 192.168.1.66 realm 8

·ip route del 192.168.1.0/24 dev eth0 via 192.168.1.66 realm 4