

使用linux下的TC进行服务器流量控制Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022__E4_BD_BF_E7_94_A8linu_c103_644685.htm 公司一台服务器，网络环境太高，那台服务器和源服务器连接下载，就跑到400M-500M，为了控制一下，所以研究了一下TC.来做流量控制。给他控制到小点，不要让这一台占了所有的网络。TC很是强大啊，很多所谓的硬件路由器，都是基于这个做的。TC介绍 在linux中，TC有二种控制方法CBQ和HTB.HTB是设计用来替换CBQ的。它是一个层次式的过滤框架。TC包括三个基本的构成块：队列规定qdisc（queueing discipline）、类（class）和分类器（Classifiers）队列（queueing discipline）：用来实现控制网络的收发速度。通过队列，linux可以将网络数据包缓存起来，然后根据用户的设置，在尽量不中断连接（如TCP）的前提下平滑网络流量。需要注意的是，linux对接收队列的控制不够好，所以我们一般只用发送队列，即“控发不控收”。它封装了其他两个主要TC组件（类和分类器）。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。队列规则包括FIFO（先进先出），RED（随机早期探测），SFQ（随机公平队列）和令牌桶（Token Bucket），类基队列（CBQ），CBQ是一种超级队列，即它能够包含其它队列（甚至其它CBQ）

。 class用来表示控制策略。很显然，很多时候，我们很可能要对不同的IP实行不同的流量控制策略，这时候我们就得用不同的class来表示不同的控制策略了。 filter用来将用户划入到具体的控制策略中（即不同的class中）。比如，现在，我们想对xxa，xxb两个IP实行不同的控制策略（A，B），这时，我们可用filter将xxa划入到控制策略A，将xxb划入到控制策略B，filter划分的标志位可用u32打标功能或IPtables的 set-mark（大多使用iptables来做标记）功能来实现。目前，TC可以使用的过滤器有：fwmark分类器，u32分类器，基于路由的分类器和RSVP分类器（分别用于IPV6、IPV4）等；其中，fwmark分类器允许我们使用Linux netfilter 代码选择流量，而u32分类器允许我们选择基于 ANY 头的流量。需要注意的是，filter（过滤器）是在QDisc内部，它们不能作为主体。数据包->TC（class）->TC（queue）应用 假设eth0是服务器的外网网络接口。1）首先在eth0的qdiscA，qdiscA控制通过本机到外网的速度，因此是用来控制服务器流出速度的 #tc qdisc add dev eth1 root handle 1 : htb default 1 添加 设置接口 最上层 句柄（做标记用） 标记 默认使用1的class 解释如下：无论是队列，还是class和filter都有ID之类的标志符，一般都有parent（父，上层的），注意ID具有接口本地性，不同的网络接口可以有相同的ID.对于这里因为qdisc在顶部，所以parent无，用‘root’字样来标识，ID用1：来标志‘default 91’表示当某个ip流不满足任何已设定的filter规则时，将自动归入class 1中。更详细的指令规则说明请参考手册。2）然后在qdisc下建立两个class，来指定eth0控制通过本机到外网的速度 #tc class add dev eth0 parent 1 : 0 classid 1 : 30 htb rate 2mbit ceil 4mbit prio

2 注：以上就是我们控制输出服务器的速度，为2M，最大可以到4M rate：是一个类保证得到的带宽值。如果有不只一个类，请保证所有子类总和是小于或等于父类。prio：用来指示借用带宽时的竞争力，prio越小，优先级越高，竞争力越强。ceil：ceil是一个类最大能得到的带宽值。

3) 接着针对不同的应用在各root class下设置不同的类，。示例如下。（如果只有一个类，这个就没有必要了）

```
#tc class add dev eth0 parent 1 : 30 classid 1 : 31 htbrate 0.5mbit ceil 2mbit prio 3
```

同时为了不使一个会话永占带宽，在节点（即本文的各应用结点）添加随即公平队列sfq。（多IP）

```
#tc qdisc add dev eth0 parent 1 : 31 handle 31 : sfq perturb 10 4
```

接着添加过滤器。

```
#tc filter add dev eth0 parent 1 : protocol ipprio 31 handle 31 fw flowid 1 : 31 5
```

用iptables打标，也可以使用u32之类

```
#iptables -t mangle -I FORWARD -i ! eth1 -p tcp sport 80 -s xxx.xxx.xxx.xxx j MARK set-mark 31 100
```

Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com