

Linux认证辅导:Linux内核中流量控制(1)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\\_Linux\\_E8\\_AE\\_A4\\_E8\\_AF\\_c103\\_644954.htm](https://www.100test.com/kao_ti2020/644/2021_2022_Linux_E8_AE_A4_E8_AF_c103_644954.htm)

1. 前言linux内核中提供了流量控制的相关处理功能，相关代码在net/sched目录下；而应用层上的控制是通过iproute2软件包中的tc来实现，tc和 sched的关系就好象iptables和netfilter的关系一样，一个是用户层接口，一个是具体实现，关于tc的使用方法可详将Linux Advanced Routing HOWTO，本文主要分析内核中的具体实现。流控包括几个部分：流控算法，通常在net/sched/sch\_\*.c中实现，缺省的是FIFO，是比较典型的黑盒模式，对外只看到入队和出对两个操作。流控结构的操作处理。和用户空间的控制接口，是通过rtnetlink实现的。以下内核代码版本为2.6.19.2。

2. 控制入口  
2.1 控制入口linux流控功能反映为网卡设备的属性，表明是网络最底层的处理部分，已经和上层的网络协议栈无关了：/\*

```
include/linux/netdevice.h */ struct net_device { ..... /* * Cache line mostly used on queue transmit path (qdisc) */ /* device queue lock */ spinlock_t queue_lock ____cacheline_aligned_in_smp. //这是发送数据时的队列处理 struct Qdisc *qdisc. //网卡停止时保存网卡活动时的队列处理方法 struct Qdisc *qdisc_sleeping. //网卡处理的数据队列链表 struct list_head qdisc_list. //最大队列长度 unsigned long tx_queue_len. /* Max frames per queue allowed */ /* Partially transmitted GSO packet. */ struct sk_buff *gso_skb. /* ingress path synchronizer */ //输入流控锁 spinlock_t ingress_lock. //这是对于接收数据时的队列处理 struct Qdisc *qdisc_ingress.
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问

