

Linux认证辅导:Linux内核中流量控制(4)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_644951.htm

4. 基本操作各种流控算法是通过流控操作结构实现，然后这些操作结构登记到内核的流控链表，在使用时可为网卡构造新的流控结构，将该结构和某种流控操作结构挂钩，这样就实现网卡采用某种策略发送数据进行流控，所有操作在用户空间都可通过tc程序设置。 4.1

Qdisc的一些基本操作4.1.1 分配新的流控结构/*

```
net/sched/sch_generic.c */ // 分配新的Qdisc结构, Qdisc的操作结构由函数参数指定
struct Qdisc *qdisc_alloc(struct net_device *dev, struct Qdisc_ops *ops) { void *p. struct Qdisc *sch. unsigned int size. int err = -ENOMEM. /* ensure that the Qdisc and the private data are 32-byte aligned */ // Qdisc空间按32字节对齐 size = QDISC_ALIGN(sizeof(*sch)). //增加私有数据空间 size = ops->padded = (char *) sch - (char *) p. // ----- // |_____||_____|| // ^ ^ ^ // | pad | | // p sch // 初始化链表, 将用于挂接到dev的Qdisc链表 INIT_LIST_HEAD(gt.list). //初始化数据包链表 skb_queue_head_init(gt.q). // Qdisc结构参数 sch->enqueue = ops->enqueue. sch->dequeue = ops->dequeue. sch->dev = dev. //网卡使用计数增加 dev_hold(dev). sch-&dev-&sch->init || ops->queue_lock and BH! */ //调用Qdisc操作函数中的reset函数 void qdisc_reset(struct Qdisc *qdisc) { struct Qdisc_ops *ops = qdisc->ops->reset(qdisc). } /* this is the rcu callback
```

function to clean up a qdisc when there * are no further references to
it *//真正释放Qdisc缓冲区 100Test 下载频道开通，各类考试
题目直接下载。详细请访问 www.100test.com