此文章已于 14:12:17 2017/1/24 重新发布到 MADHEX

linux接收数据包的处理流程

1. 硬件收到数据，驱动取出数据，通过netif\_rx把数据放到queue->input\_pkt\_queue中，

并产生NET\_RX\_SOFTIRQ软中断。

2. 在启动内核时，通过open\_softirq(NET\_RX\_SOFTIRQ, net\_rx\_action);

注册了软中断回调函数net\_rx\_action。

3. net\_rx\_action判断是否queue->input\_pkt\_queue是否有数据，如果有再通过process\_backlog

调用netif\_receive\_skb处理。

4. netif\_receive\_skb通过ptype\_base判断数据类型，再调用对应的fun(arp\_rcv|ip\_rcv)

5. ip\_rcv做IP头检查，再调用ip\_route\_input设置数据包的路由。

调用dst\_input

如果是本地包则调用ip\_local\_deliver

如果是转发包则调用ip\_forward

6. 在ip\_local\_deliver中，

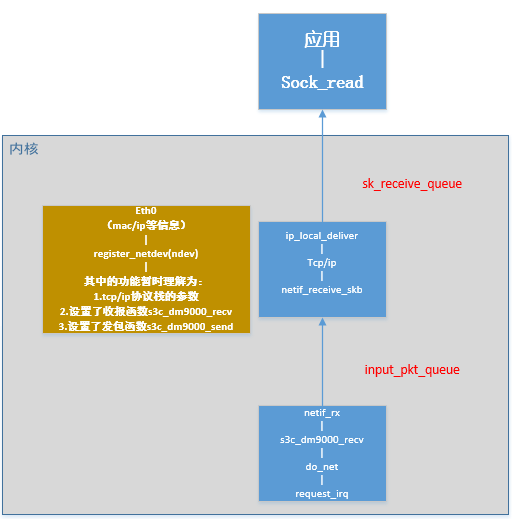
例如，如果是ping包，则调用icmp\_rcv把数据放到sk->sk\_receive\_queue

7. socket 通过中断到内核的

sys\_socketcall->Sys\_read->Sock\_read->sock\_recvmsg->inet\_recvmsg->

udp\_recvmsg->skb\_recv\_datagram从sk->sk\_receive\_queue把数据取出来。

再来个图让你更清晰：收包示例



这里值得总结一下。

当IP报文从软中断上来的时候，

则必须从ptype\_base[]数组中找到对应函数，目前内核中与IP有关的只有3个，

一个是ip\_packet\_type，另一个是arp\_packet\_type，最后是rarp\_packet\_type。

如果是ip报文，当执行到ip\_local\_deliver\_finish时，

则应该从inet\_protos[]数组中找到对应函数，里面包含ICMP、UDP、TCP等上层协议的处理函数。

这2个数组极易造成混淆，因为都有一个回调函数。在经历这么多函数的研究，读者可能都有点晕了。

一个比较好的记忆方法是：对于前一个，其回调函数是func，处理packet，后一个回调函数是handler，处理协议。

以上为收包流程，收包的时候，利用中断直接定位到s3c\_dm9000\_recv

但大家还需思考下发包的时候是如果定位到s3c\_dm9000\_send函数的呢？

