目录

[1. 什么是桥接](#_Toc7661)

[2. 概念](#_Toc12422)

[3. 功能](#_Toc31808)

[4. 扩展](#_Toc497)

[5. 一次局域网内的ping过程](#_Toc7156)

[6. 二层数据包转发过程](#_Toc23447)

[7. 补充](#_Toc1755)

|  |  |  |
| --- | --- | --- |
| 作者 | 李勇君 | |
| 版本信息/修改记录 |  | |
| 版本 | 描述 |
| 2020-05-29 | 1.0 | 初始版本 |
|  |  |  |

1. 什么是桥接

简单来说，桥接就是把一台机器上的若干个网口"连接"起来。

这样，其中一个网口收到的报文会被复制给其他网口并发送出去。

以使得网口之间的报文能够相互转发。

1. 概念

二层

虚拟的网络设备

1. 功能

转发：依赖FDB

学习：生成FDB

1. 扩展

4.1网卡四种工作模式

* 广播模式(BroadCast Model)：接收广播帧
* 多播模式(MultiCast Model)：接收多播帧
* 直接模式(Direct Model)：只接收目的地址是自己 mac 地址的帧
* 混杂模式(Promiscuous Model)：接收流过网卡的所有帧

网卡缺省模式包含广播模式和直接模式

添加到网桥的物理网口，会被设为混杂模式（将物理网口添加到网桥时，内核会做此工作）

处于混杂模式下网卡的IP无效，连mac地址都不关心了

4.2 网卡的五种状态

* #define BR\_STATE\_DISABLED 0 // 什么功能都没有，只有一个逻辑设备。
* #define BR\_STATE\_LISTENING 1 // 可以接收和发送网络传输的BPDU，但不能进行数据帧的转发、不能学习。
* #define BR\_STATE\_LEARNING 2 // 可以接收和发送BPDU，可以学习，但是不能进行数据帧的转发。
* #define BR\_STATE\_FORWARDING 3 // 可以接收和发送BPDU、可以学习、可以进行数据帧的转发。
* #define BR\_STATE\_BLOCKING 4 // 只能接收BPDU，不能发送BPDU，不能学习，不能转发数据帧。

4.3 bridge的MAC地址

* 新创建的bridge，获得一个随机MAC地址
* 添加Port后，指定其中一个Port的MAC作为bridge的MAC
* 如果没有手动指定，br0 会根据 bridge 中 port 的变化，自动选择 port 最小的一个 MAC 作为自身 MAC 地址

4.4 bridge的IP地址

* 给网桥设置一个 IP 地址，这个地址可以作为其下主机的网关。作为网关时，还要有路由和nat功能。
* 给网桥一个地址，这样就能远程管理网桥了。

4.5 bridge 在内核中的位置

源码：/usr/src/linux-source-x.xx.x/net/bridge

相对简单独立的module

make -C /lib/modules/4.15.0-101-generic/build/ M=/usr/src/linux-headers-4.15.0-101-generic/net/bridge modules

rmmod bridge

insmod bridge.ko

4.6 用户空间操作bridge

命令：brctl

添加网桥(br0)

$ brctl addbr br0

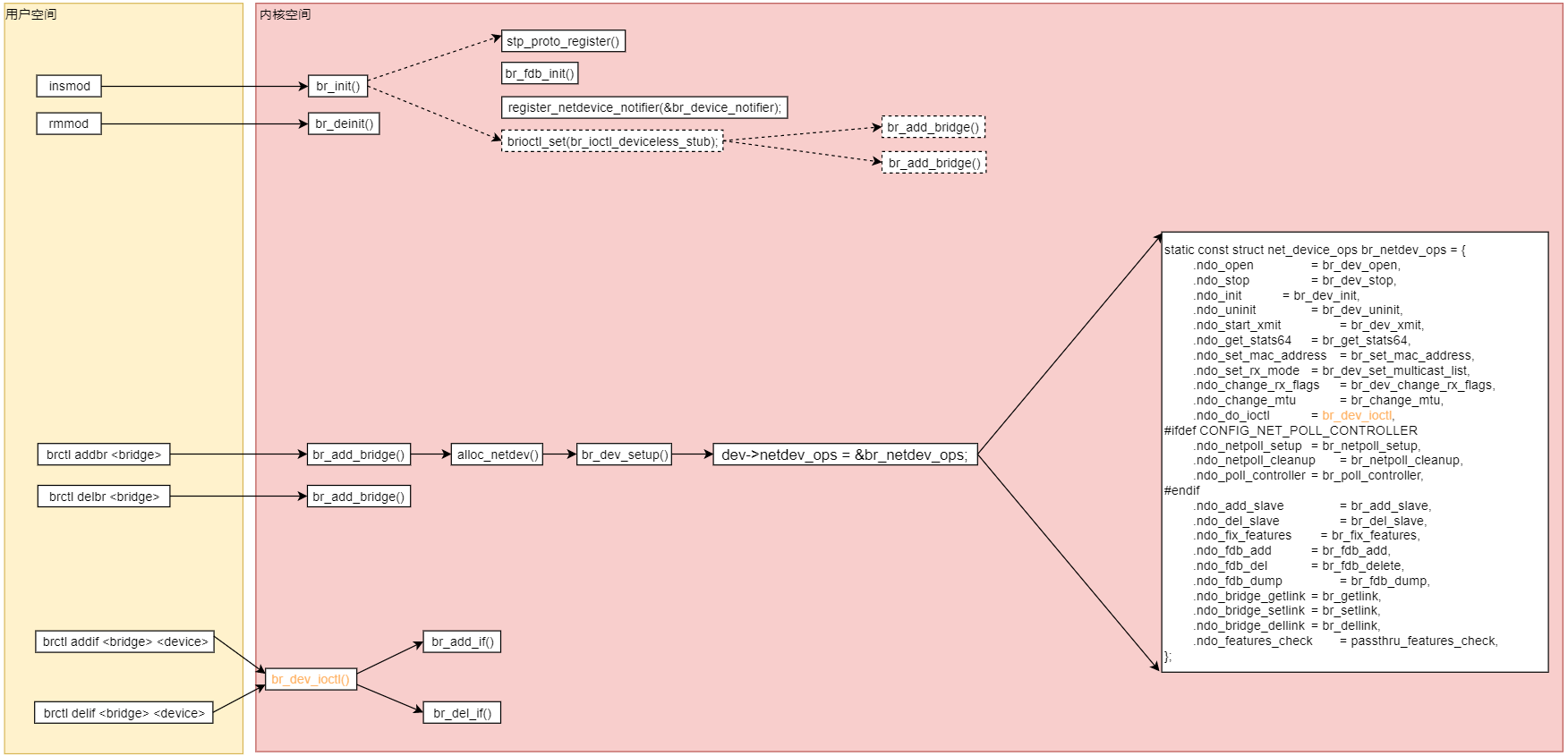
将物理网口加入网桥(br0)

$ brctl addif br0 enp3s0

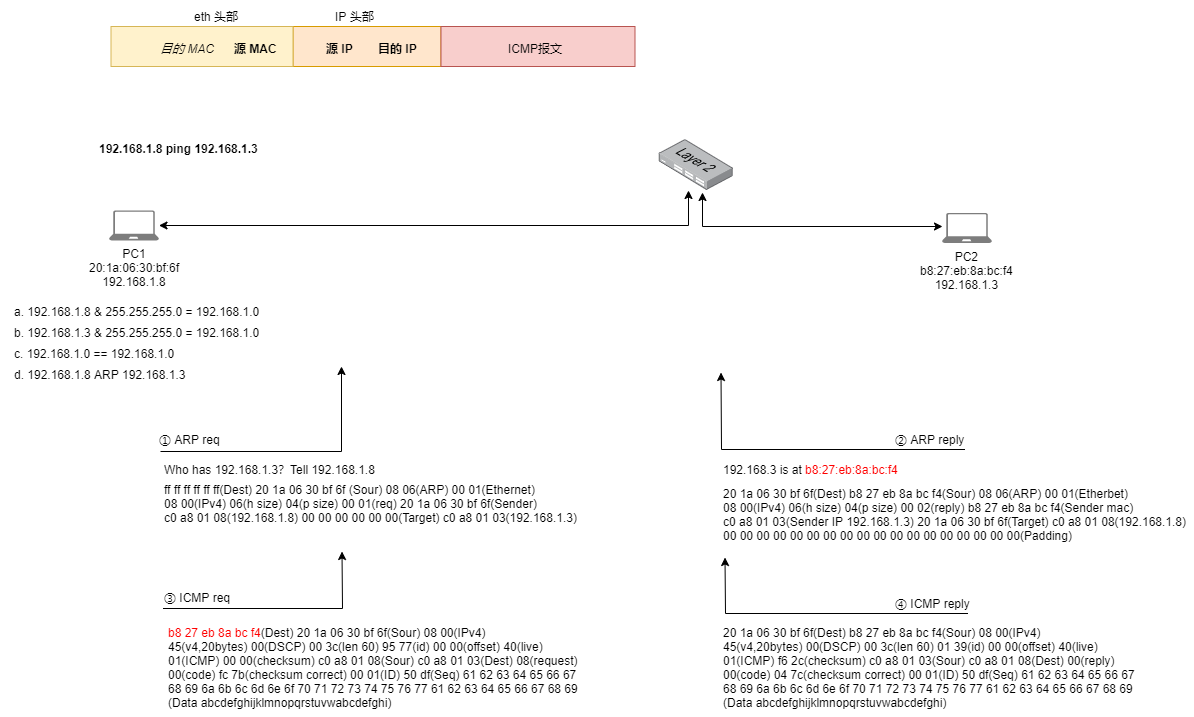
$ brctl addif br0 enp4s0

linux内核支持网口的桥接（目前只支持以太网接口）。？？？不理解

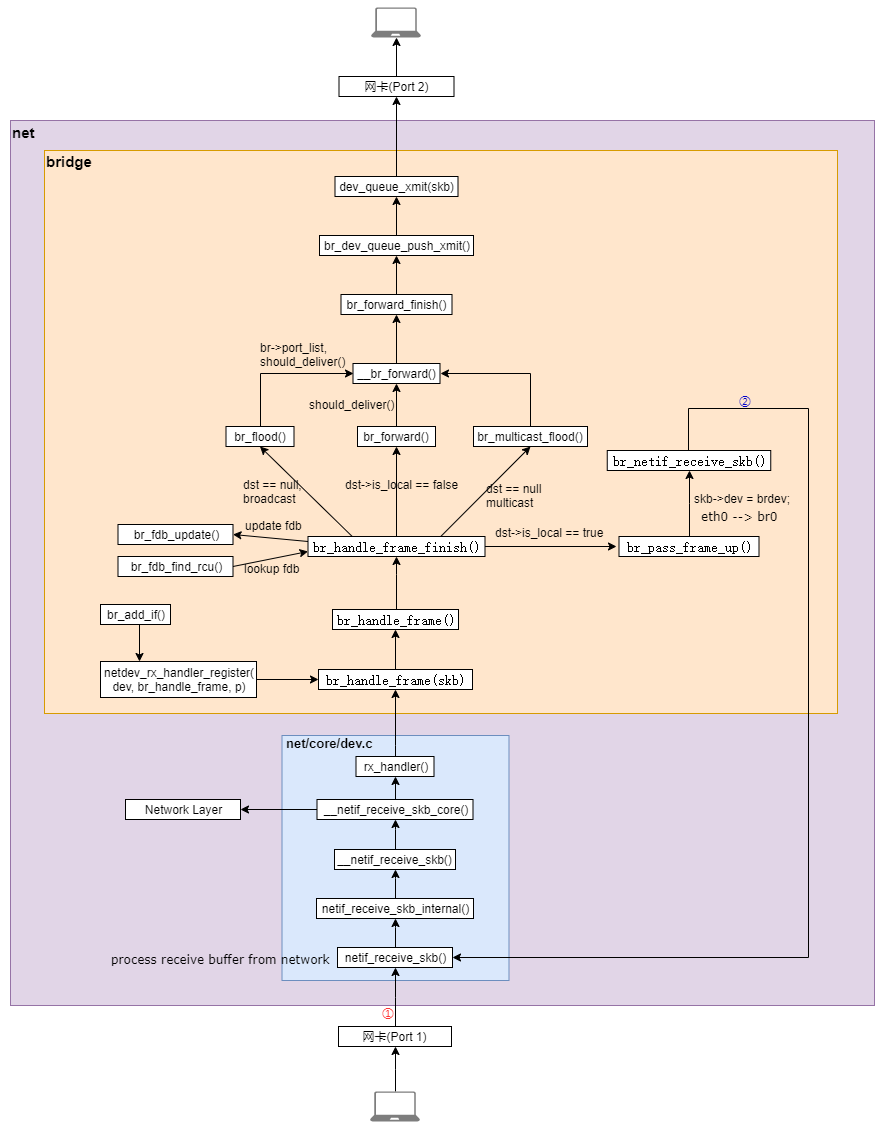
$ sudo brctl addif br0 enp3s0  
$ sudo brctl addif br0 wlp5s0  
can't add wlp5s0 to bridge br0: Operation not supported  
​

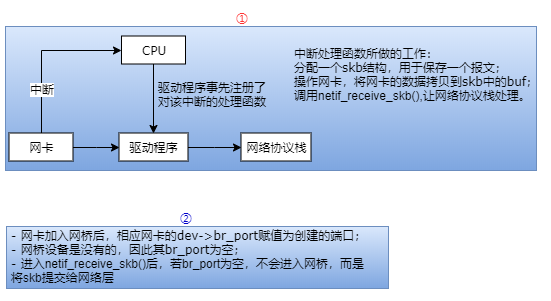


1. 一次局域网内的ping过程



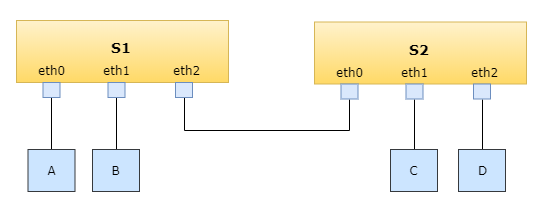
1. 二层数据包转发过程



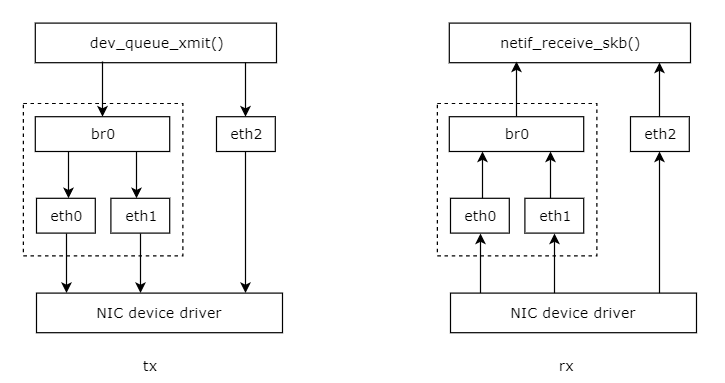


1. 补充

7.1超时策略

需要定时忘记之前的记录。 

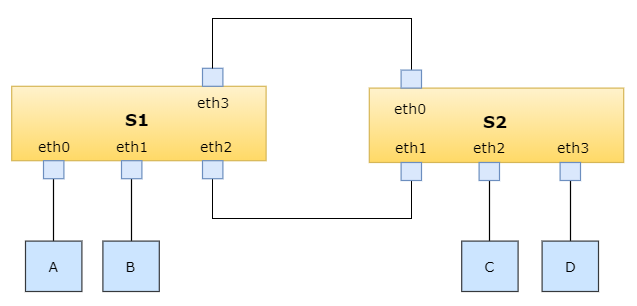
7.2 bridge模型



网桥设备br0绑定了eth0和eth1。对于网络协议栈的上层来说，只看得到br0，因为桥接是在数据链路层实现的，上层不需要关心桥接的细节。于是协议栈上层需要发送的报文被送到br0，网桥设备的处理代码再来判断报文该被转发到eth0或是eth1，或者两者皆是;

反过来，从eth0或从eth1接收到的报文被提交给网桥的处理代码，在这里会判断报文该转发、丢弃、或提交到协议栈上层。  
而有时候eth0、eth1也可能会作为报文的源地址或目的地址，直接参与报文的发送与接收（从而绕过网桥）。???

* 1. 生成树协议



冗余链路的必要性

在复杂网络中，需要对链路做一定的冗余，以便当其中一条链路或者某台交换机出现故障时，整个网络还能保持通畅。

冗余链路的弊端

有可能形成环路，造成网络风暴。

假设两台交换机都刚刚启动，C向B发送一个报文，S2由于地址表为空，所以要对其他口进行广播，S1会从eth2和eth3口都受到此广播报文，并且同时会向其他口转发，由此形成环路。并且非环路接口也会因此一直收到广播报文，造成网络风暴。

解决办法——生成树协议

环形🡪树形

a)先找树根

规定，只有树根节点才能向外发送BPUD报文，用来协调其它交换机

上电后，都认为自己是树根，向各个网口发BPUD报文，BPUD报文包含root\_id，由网络管理员指定，越小优先级越高，一般是性能较好的交换机。相同时，再比较MAC地址。

当收到比自己优先级更高的BPUD时，停止自己的BPUD发送，转而转发更高优先级的BPUD报文。

b)确定唯一上行端口

到树根开销最小的那个网口

c)确定要被阻塞的下行端口