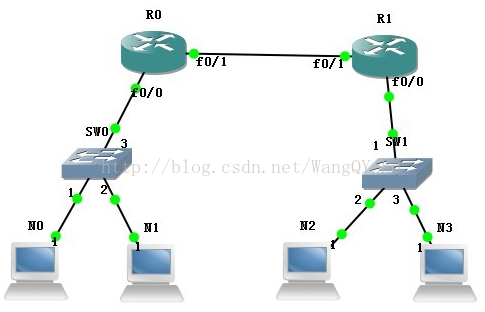
* 标签：
* [编辑](http://write.blog.csdn.net/postedit/69747974)
* 删除



情景一：同一广播域内，两台主机通信过程。

    我们知道两主机要通信传送数据时，就要把应用数据封装成IP包（因为我们的网络大多都是TCP/IP的以太网了），然后再交给下一层数据链路层继续封装成帧；之后根据MAC地址才能把数据从一台主机，准确无误的传送到另一台主机。

    如图：当NO要和N1通信时，假如N0知道N1的IP但却不知道它的MAC地址，那NO就会发送一个ARP的广播请求（里面源IP是NO 目标IP是N1  源MAC是N0  目标MAC是12个F）给同一广播域中的所有成员，当交换机SW0从自己的1接口上收到这个广播包，然后它会读取这个帧的源MAC地址和目标MAC地址，由于交换机SW0刚启动加电时，它的MAC表为空的。所以它会把NO的MAC地址与之相对应的接口1放到一张表里，这张表就是MAC地址表。然后他再从别的接口广播这个数据帧，当别的主机收到这个广播时，查看目标IP不是自己的，就会丢弃此包。如果N1接收到这个数据帧，它检查目标IP和这个的IP是一样的，就会回应这个ARP请求，把自己的IP和MAC封装成源IP和源MAC，N0的IP和N0的MAC地址为目标IP与目标MAC，并记录NO的MAC与IP，放进自己的ARP缓存表中。此时，这个应答包经过交换机SWO时，它又会检查源MAC 、 目标MAC，把N1的MAC和自己接口2放进MAC地址表中，再查看自己的MAC地址表，发现存在目标MAC与自己的1接口对应（由于刚开始有记录过N0的MAC），那它就会直接把这个应答包从接口1送出去了。主机N0收到这个包后发现目标MAC是自己，就会处理这个包。并把N1的MAC与IP放进自己的ARP缓存表中。这时主机N0就知道N1的MAC地址了，以后要发送数据，就直接把N1的IP与MAC封装进帧中进行点对点的发送了。

情景二：跨路由的数据传输过程。

    当NO要和N2通信时，此时NO会检查N2的IP地址跟自己是否处在同一网段，图上得知，两主机肯定不会是同一网段的。因为N2和自己处在不同网段，所以，N0会把数据包发给它的网关，也就是R0上的F0/0接口了。源IP和源MAC地址是N0自己的，目标IP是N2的，目标MAC是R0上接口F0/0的（如果N0不知道F0/0的MAC,就会跟情景一相似，发个ARP广播来得到F0/0的MAC地址）。当这个数据包到达R0时，路由器R0会查看目标IP的是否是自己，由于目标不是自己，所以，会查看自己的路由表，找出到达N2网段的路由；如果没有相关条目，就直接丢弃。当查看路由表后发现到达N2网段的出接口是F0/1。于是，把数据包转到F0/1接口上，再由接口F0/1传给R1。这个过程，数据包的源IP是N0 源MAC是F0/1 目标IP是N2 目标MAC是R1的F0/1接口IP 。

    当R1收到这个数据包后，同样也要检查包的目标IP是否是自己，它会主动查找自己的路由表，发现目标IP跟自己F0/0接口处在同一网段，于是就把包传到F0/0接口上去发给N2 （假如R1上的ARP缓存表中没有N2的MAC，则接口F0/0会发送一个ARP广播给跟它相连的广播域中；这个ARP广播包的源IP是接口F0/0的IP 源MAC也是F0/0的MAC  目标IP是N2 目标MAC为12个F）,假如N2的MAC地址已经在R1的ARP缓存中了，那就会直接把数据包封装成：源IP为N0 源MAC为R1的F0/0 目标IP为N2 目标MAC为N2了。

到了这里，包的跨路由传输就会结束了，当包到达N2，做反向操作即可把包发给N0了。

总结：同一广播域中，包的源、目标IP;源、目标MAC是真实的两台主机上的IP与MAC地址。

跨路由中，包的源IP与目标IP始终不会发生变化，源和目标MAC根据所经过的路由接口不同而发生相应变化。