

分割广播域时，一般都必须使用到路由器。使用路由器后，可以以路由器上的网络接口（LAN Interface）为单位分割广播域。

但是，通常情况下路由器上不会有太多的网络接口，其数目多在1~4个左右。随着宽带连接的普及，宽带路由器（或者叫IP共享器）变得较为常见，但是需要注意的是，它们上面虽然带着多个（一般为4个左右）连接LAN一侧的网络接口，但那实际上是路由器内置的交换机，并不能分割广播域。

况且使用路由器分割广播域的话，所能分割的个数完全取决于路由器的网络接口个数，使得用户无法自由地根据实际需要分割广播域。

与路由器相比，二层交换机一般带有多个网络接口。因此如果能使用它分割广播域，那么无疑运用上的灵活性会大大提高。

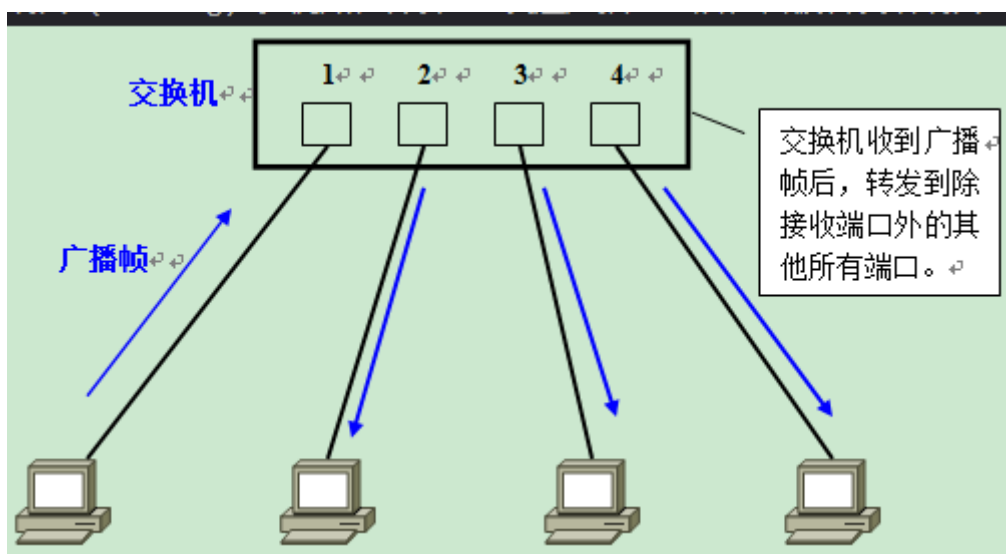
用于在二层交换机上分割广播域的技术，就是VLAN。通过利用VLAN，我们可以自由设计广播域的构成，提高网络设计的自由度。

## 2.实现VLAN的机制

### 2.1 实现VLAN的机制

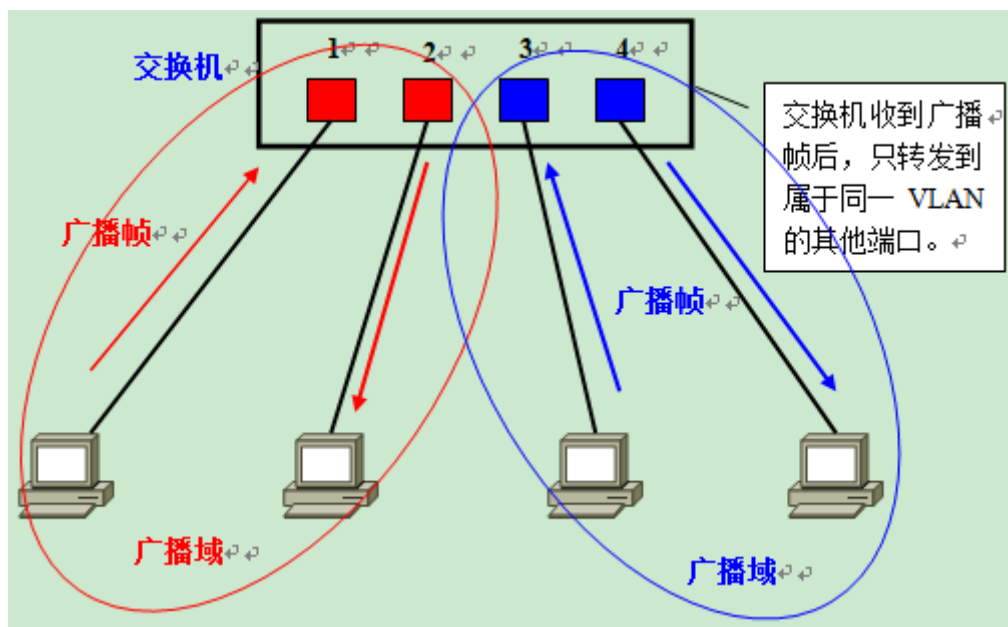
在理解了“为什么需要VLAN”之后，接下来让我们了解一下交换机是如何使用VLAN分割广播域的。

首先，在一台未设置任何VLAN的二层交换机上，任何广播帧都会被转发给除接收端口外的所有其他端口（Flooding）。例如，计算机A发送广播信息后，会被转发给端口2、3、4。



这时，如果在交换机上生成红、蓝两个VLAN；同时设置端口1、2属于红色VLAN、端口3、4属于蓝色VLAN。再从A发出广播帧的话，交换机就只会把它转发给同属于一个VLAN的其他端口——也就是同属于红色VLAN的端口2，不会再转发给属于蓝色VLAN的端口。

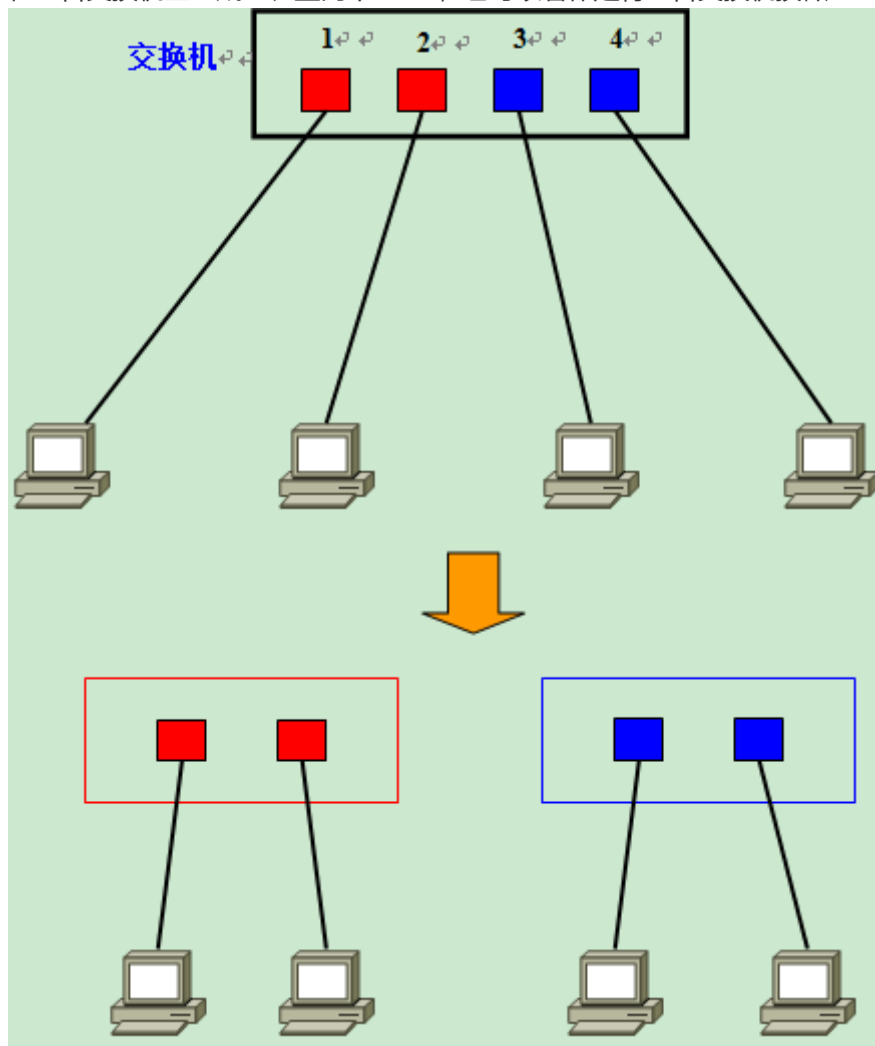
同样，C发送广播信息时，只会被转发给其他属于蓝色VLAN的端口，不会被转发给属于红色VLAN的端口。



就这样，VLAN通过限制广播帧转发的范围分割了广播域。上图中为了便于说明，以红、蓝两色识别不同的VLAN，在实际使用中则是用“VLAN ID”来区分的。

## 2.2 直观地描述VLAN

如果要更为直观地描述VLAN的话，我们可以把它理解为将一台交换机在逻辑上分割成了数台交换机。在一台交换机上生成红、蓝两个VLAN，也可以看作是将一台交换机换做一红一蓝两台虚拟的交换机。



在红、蓝两个VLAN之外生成新的VLAN时，可以想象成又添加了新的交换机。

但是，VLAN生成的逻辑上的交换机是互不相通的。因此，在交换机上设置VLAN后，如果未做其他处理，VLAN间是无法通信的。

明明接在同一台交换机上，但却偏偏无法通信——这个事实也许让人难以接受。但它既是VLAN方便易用的特征，又是使VLAN令人难以理解的原因。


Bridge（桥）是 Linux 上用来做 TCP/IP 二层协议交换的设备，与现实世界中的交换机功能相似。Bridge 设备实例可以和 Linux 上其他网络设备实例连接，既 attach 一个从设备，类似于在现实世界中的交换机和一个用户终端之间连接一根网线。当有数据到达时，Bridge 会根据报文中的 MAC 信息进行广播、转发、丢弃处理。

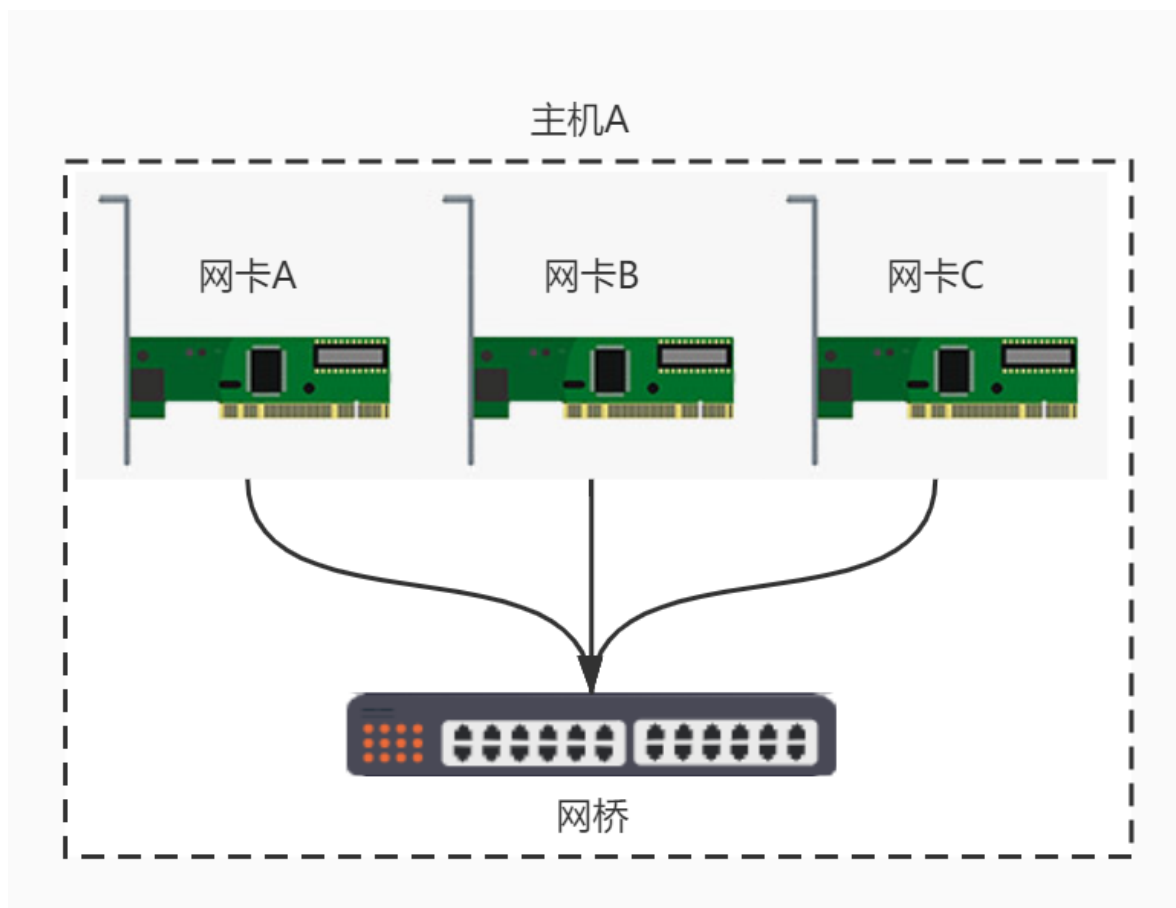
## bridge常用场景

现在的Linux 网桥可以看做是（三层的）虚拟交换机，功能和物理交换机一样，最常用的功能是链接虚拟机和容器--为虚拟机和容器提供一个虚拟交换机。

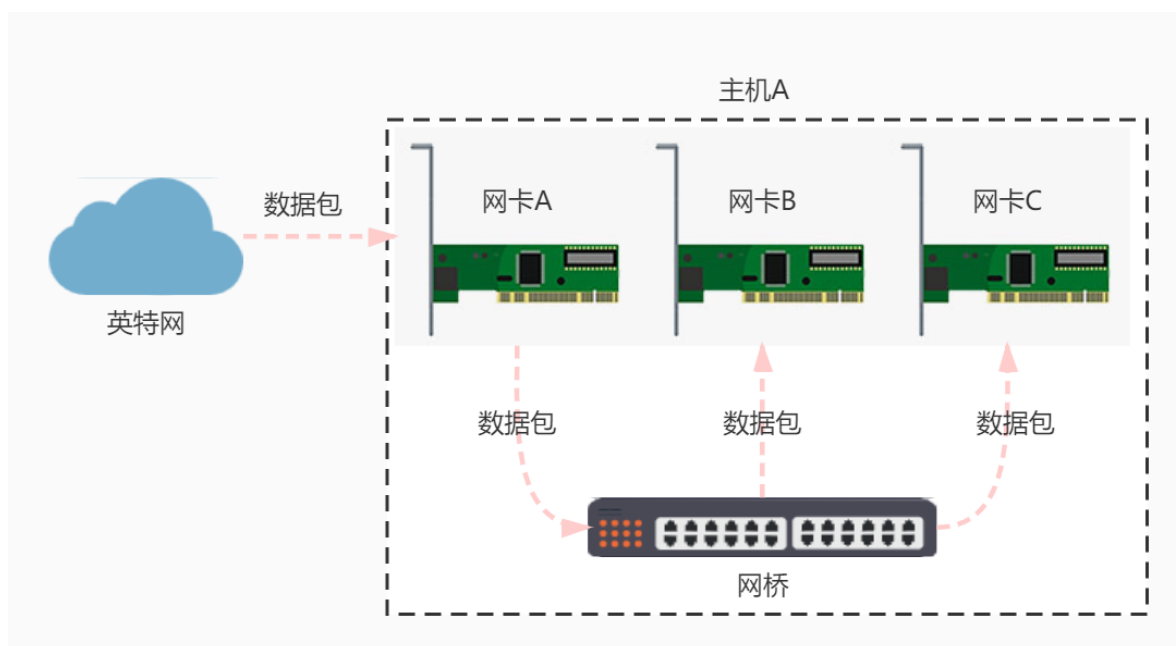
创建一个bridge后（br0），可以把其他的网络设备（比如eth0）attach到br0上，eth0称作br0的从设备。需要注意的是，eth0 attach到br0上，不是对应的将eth0插接到“交换机”br0上，而是eth0编程了br0的一个端口（网线插口）。那什么时候才是插入网线呢？

通常的网桥是二层设备，不需要有IP地址。但是linux网桥是虚拟网络设备，是有ip和mac的（br设备的MAC地址是它所有从设备中最小的MAC地址）。从设备（eth0）被attach到br上之后，它的IP及MAC都不再可用了（退化为一个端口了）且它们被设置为接收任何包（工作在链路层，且是混杂模式，不需要ip），最终由bridge设备来决定数据包的去向：接收到本机、转发、丢弃。

Linux 的  网桥 是一种虚拟设备（使用软件实现），可以将 Linux 内部多个网络接口连接起来，如下图所示：

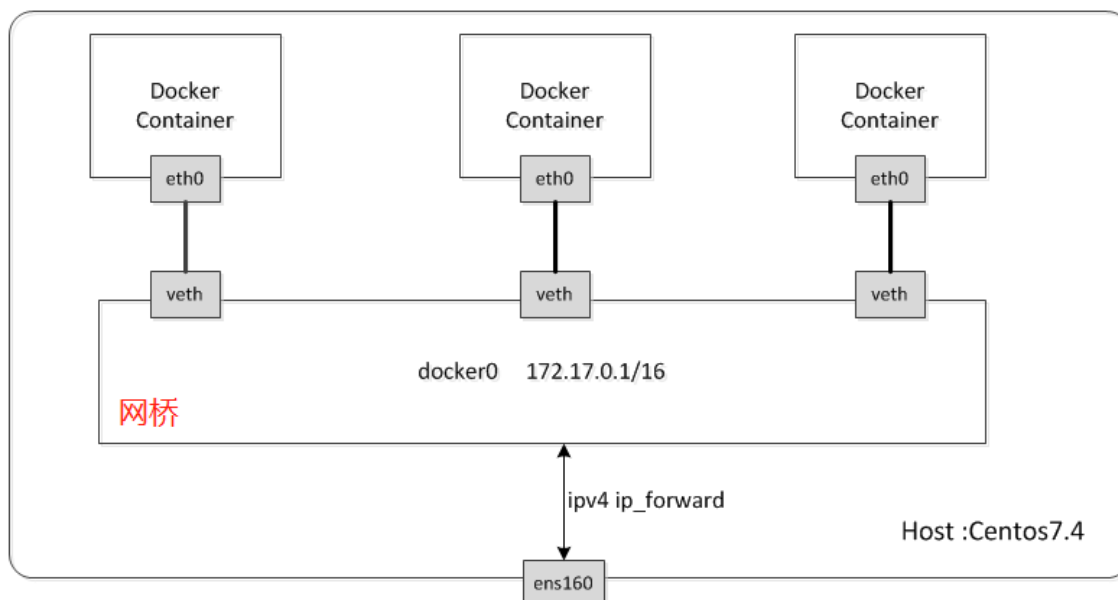


而将网络接口连接起来的结果就是，一个网络接口接收到网络数据包后，会复制到其他网络接口中，如下图所示：



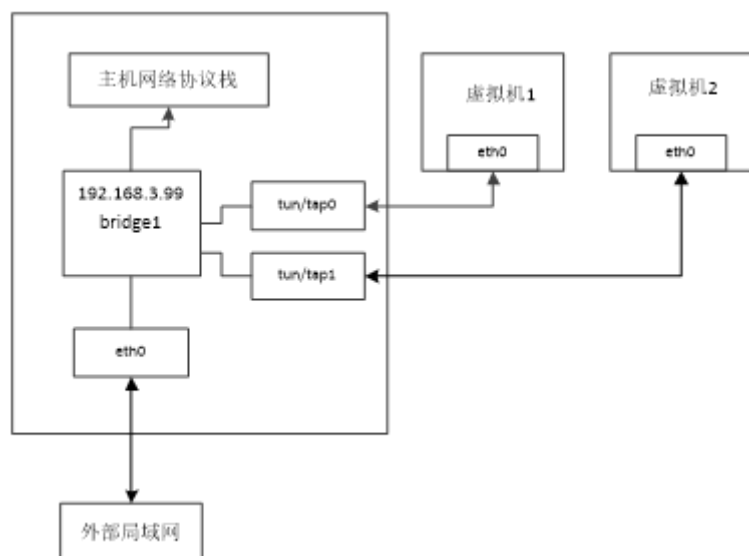
如上图所示，当网络接口A接收到数据包后，网桥 会将数据包复制并且发送给连接到 网桥 的其他网络接口（如上图中的网卡B和网卡C）。

Docker 就是使用 网桥 来进行容器间通讯的，我们来看看 Docker 是怎么利用 网桥 来进行容器间通讯的，原理如下图：



Docker 在启动时，会创建一个名为 `docker0` 的网桥，并且把其 IP 地址设置为 `172.17.0.1/16`（私有 IP 地址）。然后使用虚拟设备对 `veth-pair` 来将容器与网桥连接起来，如上图所示。而对于 `172.17.0.0/16` 网段的数据包，Docker 会定义一条 `iptables NAT` 的规则来将这些数据包的 IP 地址转换成公网 IP 地址，然后通过真实网络接口（如上图的 `ens160` 接口）发送出去。

目前，虚拟网桥最主要的作用是为虚拟机提供一种网络链接的方式。



如图所示，是最常用的链接方式。主机上创建tap设备（理解为虚拟网卡），`bridge1`设备attach `eth0`和`tap0`、`tap1`，此时`bridge1`可以视作交换机，`eth0`\`tap0`\`tap1`都是这个交换机的网口。虚拟机通过tap网口链接，虚拟机和主机、外部局域网主机是同一个局域网内的机器，可以设置成同一个网段的IP。

这里比较别扭的就是主机的`eth0`，退化成网口，`bridge1`接替它成为主机的上网设备。

桥接是连接两个不同的网段使用的，vlan是把网络隔离成两个不同的网段

vlan+bridge在功能层面完整模拟交换机，实现软件二层交换

