第二篇　容器网络技术原理■ 第5章 网络命名空间通信■ 第6章 容器网络■ 第7章 Kubernetes网络本篇首先结合网络命名空间、虚拟化网络设备原理，讲解跨网络命名空间通信的解决方案，然后结合Docker和Kubernetes进行实践，并对实践过程中的关键技术进行细致分析。本篇包含以下内容。·第5章结合第4章介绍的虚拟网络设备，介绍三种跨网络命名空间通信解决方案，并以实例化的方式进行功能验证。·第6章以Docker引擎为例，介绍三种跨网络命名空间通信解决方案在容器技术中的应用，实现容器网络的互通互联。·第7章结合实际应用讲解Kubernetes网络通信原理，包括Pod网络、Service网络和Ingress网络。

第5章　网络命名空间通信第4章讲述了网络命名空间实现原理，本章将讲述网络命名空间在虚拟化网络中的应用。用户可以将网络命名空间理解为独立的网络运行环境，每个网络命名空间有自己的设备、路由表、iptables规则等，不同的网络命名空间之间完全隔离。用户将应用部署在独立的网络命名空间中以独享网络运行环境，从而避免应用之间的网络配置冲突。那么，不同网络命名空间之间如何通信呢？对此，业界有多种解决方案，而本章介绍常用的三种。·“网桥+虚拟网卡对”方案：在宿主机中创建一个网桥设备，为每个网络命名空间创建一个虚拟网卡对设备，虚拟网卡对设备的一端位于网络命名空间中，另一端连接宿主机的网桥。不同网络命名空间的应用通过宿主机的网桥相互通信，同时可以通过网桥访问宿主机以外的网络。·macvlan方案：为每个网络命名空间创建一个macvlan虚接口设备，不同的网络命名空间通过二层交换网络直接通信。·ipvlan方案：为每个网络命名空间创建一个ipvlan虚接口设备，不同的网络命名空间通过二层/三层交换网络直接通信。

5.1 “网桥+虚拟网卡对”方案本节将结合Linux内置的网桥和虚拟网卡对设备，实现跨网络命名空间通信功能（Linux网桥的实现原理请参考4.2.1节，虚拟网卡对的实现原理请参考4.2.2节）​。回顾一下两种虚拟网络设备的特性。·网桥(bridge)：相当于虚拟的二层交换机，网桥将不同的网络设备连接到同一个二层网络，用户可以给网桥配置IP地址，并将此IP地址作为网络命名空间网络的网关。·虚拟网卡对(veth pair)：相当于连接网络命名空间和网桥的“网线”​，每个虚拟网卡对设备对应了两个网络接口设备，两个设备之间就像有一条管道直连，向一个端口写入数据后，就可以从另一个端口读取写入的数据。使用“网桥+虚拟网卡对”方案的一般过程如下。①创建网络命名空间，此网络命名空间与宿主机网络隔离。②在宿主机上创建网桥，此网桥运行在宿主机的默认网络命名空间中。③创建虚拟网卡对，将虚拟网卡对的一个端点置于网络命名空间内，另一个端点连接网桥，从而连通网桥和命名空间。“网桥+虚拟网卡对”这一网络命名空间通信方案的组网参考图5-1。

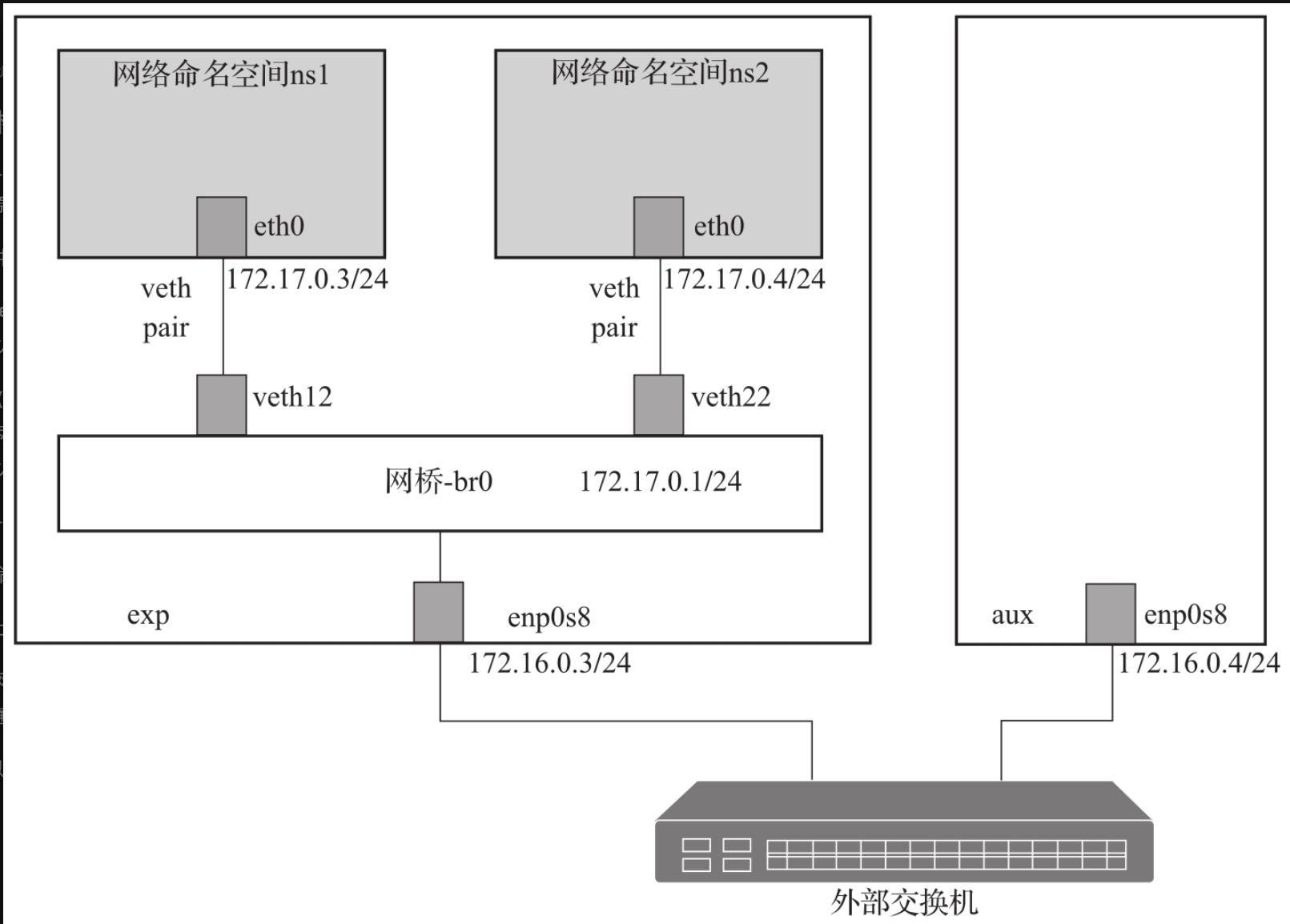
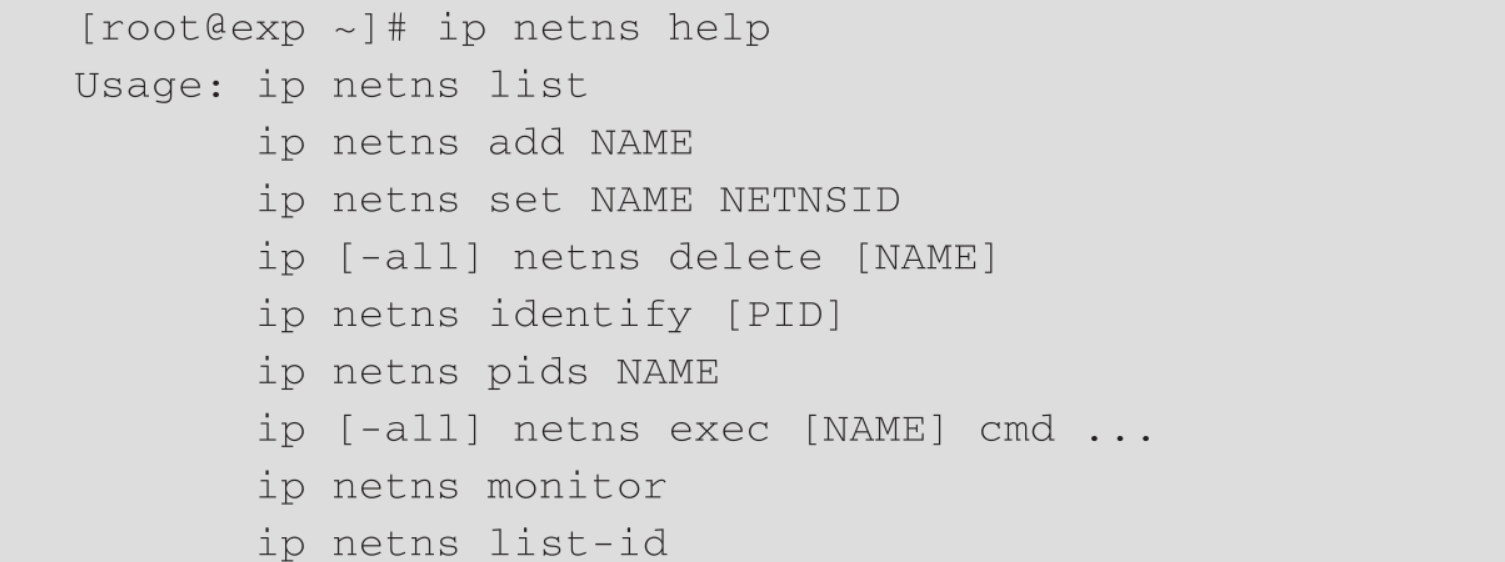
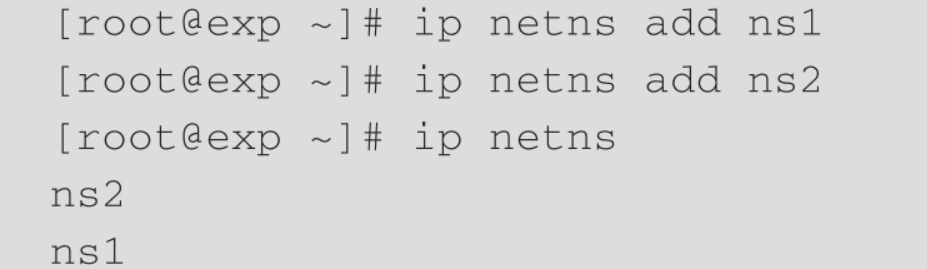


图5-1 “网桥+虚拟网卡对”方案组网下面对方案中的组网情况进行说明。·在exp主机上创建网桥br0，此网桥连接该主机的两个网络命名空间。网桥设备使用的网段为172.17.0.0/24，两个网络命名空间中的设备亦使用此网段。·在exp主机上创建两个网络命名空间，分别命名为ns1和ns2。·创建两个虚拟网卡对，分别是：veth11/veth12，其中veth11置入网络命名空间ns1中，veth12连接到网桥br0上；veth21/veth22，其中veth21置入网络命名空间ns2中，veth22连接到网桥br0上。·主机exp与测试机aux物理网卡通过外部交换机二层直连。

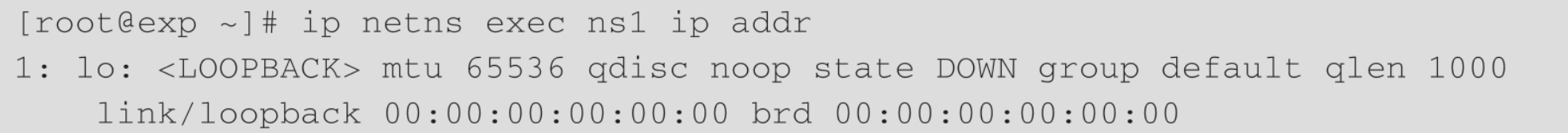
5.1.1 主机内跨网络命名空间通信本节主要验证位于同一个主机中的两个网络命名空间之间的网络通信功能。1 创建网络命名空间网络命名空间管理命令由iproute软件包提供，对应的命令为“ip netns”​，其帮助信息如下：



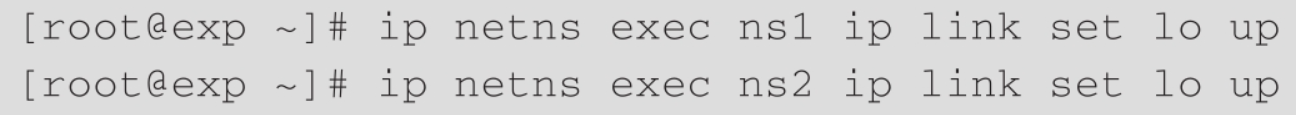
接下来使用“ip netns add”命令创建两个网络命名空间，名字分别是ns1和ns2。使用不带任何参数的“ip netns”命令显示当前系统中存在的网络命名空间列表：



使用“ip netns exec<netNS><command>”命令进入对应的命名空间内并执行命令。例如，进入ns1命名空间执行“ip addr”命令查看网络地址配置，结果如下：



新创建的命名空间中只有一个环回设备，并且状态为DOWN。Linux系统中，只有使能环回设备后，才能通过ping命令测试该命名空间的网络连通性。通过下面的命令使能两个命名空间中的环回设备：



至此，网络命名空间创建完成。