# 概述

KVM网络优化方案，就是让虚拟机访问物理机网卡的层数更少，中间环节减少了，效率自然提高了，最理想的状态就是虚拟机独占物理网卡，和物理机一样使用物理网卡，这是网卡透传方案，除此外，还有好几种方案。

以往虚拟机数据包到达物理机的路径是：虚拟机🡪QEMU虚拟网卡🡪虚拟化层🡪内核网桥🡪物理网卡

# 全虚拟化网卡

在KVM中，默认情况下网络设备是由QEMU在Linux的用户空间模拟出来并提供给虚拟机使用的。其优点在于通过模拟可以提供给虚拟机各种类型的网卡，但是由于网络I/O过程需要虚拟化引擎的参数，因此产生了大量的vm exit、vm entry，效率低下。

# 半虚拟化网卡

（1）概述

在实际应用中，使用较多的是半虚拟化技术，即virtio技术。Virtio驱动因为改造了虚拟机操作系统，让虚拟机可以直接和虚拟化层通信，从而大大提高了虚拟机的性能。

目前，Linux内核默认集成Virtio驱动，大部分Linux发行版可以直接使用Virtio驱动；Windows虚拟机一直都是使用全虚拟化方案，现在Virtio驱动不断改善，已经可以很好地支持Windows操作系统了，需要注意的是Windows虚拟机需要额外安装Virtio的驱动。

（2）配置

有两种方法配置半虚拟网卡，一种是基于命令行的，一种是基于config.xml配置文件的。

1. 在虚拟机启动命令中加入virtio-net-pci参数
2. 使用libvirt管理的KVM虚拟机，可以修改xml配置文件

# MacVTap技术

MacVTap是跳过内核网桥，虚拟化层直接和物理网卡进行通信。

# vhost-net技术

vhost\_net技术使虚拟机的网络通信绕过用户空间的虚拟化层，可以直接和内核通信，使用vhost\_net，必须使用Virtio半虚拟化网卡。vhost\_net是对Virtio优化，

# passthrough技术

## （1）概述

如果虚拟机对网络的要求比较高，通过PCI passthrough技术将物理网卡直接给虚拟机使用，虚拟机单独使用物理网卡，可以达到几乎和物理网卡一样的性能。

## （2）配置

1）使用lspci查看网卡硬件设备信息（bus:slot:function）

2）使用virsh dumpxml获取xml配置信息

3）编辑虚拟机xml配置文件，加入PCI信息

## （3）应用

PCI passthrough是性能最好的虚拟化解决方案，但是由于是独占网卡，所以对于宿主机的网卡数量有要求。可以使用1~2个网卡透传给网络压力比较大的虚拟机，其余网络压力较小的采用其余虚拟化方案。如果都是作为普通的办公使用，没必要采用这种奢侈的虚拟化方案了。

# SR-IOV技术

## 概述

SR-IOV是比较好的硬件解决方案，使用时需要特定的硬件支持，缺点是还不支持虚拟机的在线迁移。

SR-IOV（Single Root I/O Virtualization）是单根虚拟化，是一个PCI-E设备共享给虚拟机使用的标准，在网卡和显卡中有应用。SR-IOV提供了一种从硬件上染过系统和虚拟化层，并且每个虚拟机都能有单独的内存、终端、DMA流。

## 配置

1. 加载SR-IOV内核模块

modprobe igb

1. 配置文件写入

## （3）应用

# 8、Open vSwitch