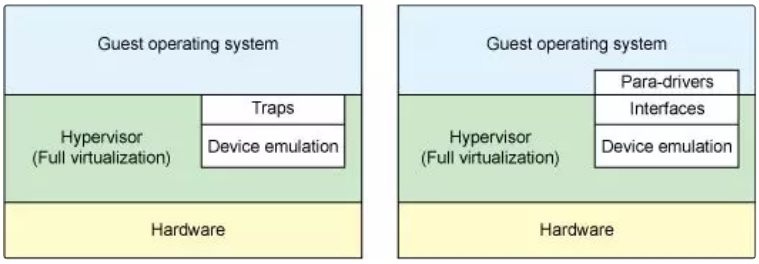
# virtio 简介





## 什么是 virtio

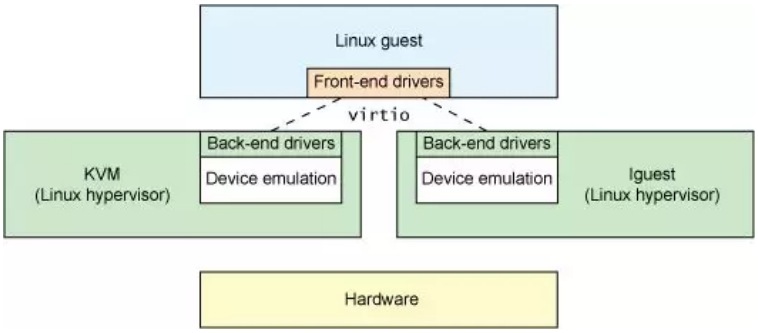
virtio 是一种 I/O 半虚拟化解决方案，是一套通用 I/O 设备虚拟化的程序，是对半虚拟化 Hypervisor 中的一组通用 I/O 设备的抽象。提供了一套上层应用与各 Hypervisor 虚拟化设备（KVM，Xen，VMware等）之间的通信框架和编程接口，减少跨平台所带来的兼容性问题，提高驱动程序开发效率。




## 为什么是 virtio

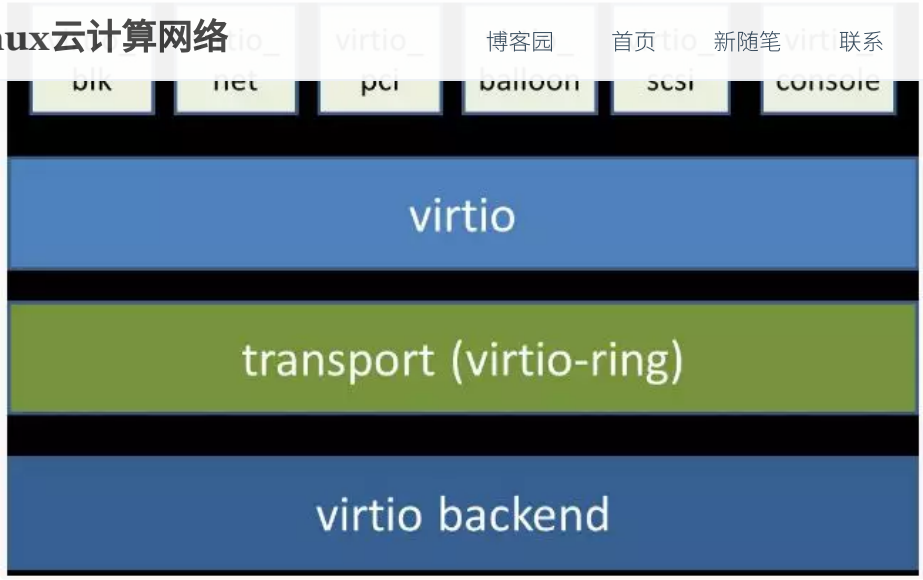
在完全虚拟化的解决方案中，guest VM 要使用底层 host 资源，需要 Hypervisor 来截获所有的请求指令，然后模拟出这些指令的行为，这样势必会带来很多性能上的开销。半虚拟化通过底层硬件辅助的方式，将部分没必要虚拟化的指令通过硬件来完成，Hypervisor 只负责完成部分指令的虚拟化，要做到这点，需要 guest 来配合，guest 完成不同设备的前端驱动程序，Hypervisor 配合 guest 完成相应的后端驱动程序，这样两者之间通过某种交互机制就可以实现高效的虚拟化过程。

由于不同 guest 前端设备其工作逻辑大同小异（如块设备、网络设备、PCI设备、balloon驱动等），单独为每个设备定义一套接口实属没有必要，而且还要考虑扩平台的兼容性问题，另外，不同后端 Hypervisor 的实现方式也大同小异（如KVM、Xen等），这个时候，就需要一套通用框架和标准接口（协议）来完成两者之间的交互过程，virtio 就是这样一套标准，它极大地解决了这些不通用的问题。




## virtio 的架构

从总体上看，virtio 可以分为四层，包括前端 guest 中各种驱动程序模块，后端 Hypervisor （实现在Qemu上）上的处理程序模块，中间用于前后端通信的 virtio 层和 virtio-ring 层，virtio 这一层实现的是虚拟队列接口，算是前后端通信的桥梁，而 virtio-ring 则是该桥梁的具体实现，它实现了两个环形缓冲区，分别用于保存前端驱动程序和后端处理程序执行的信息。



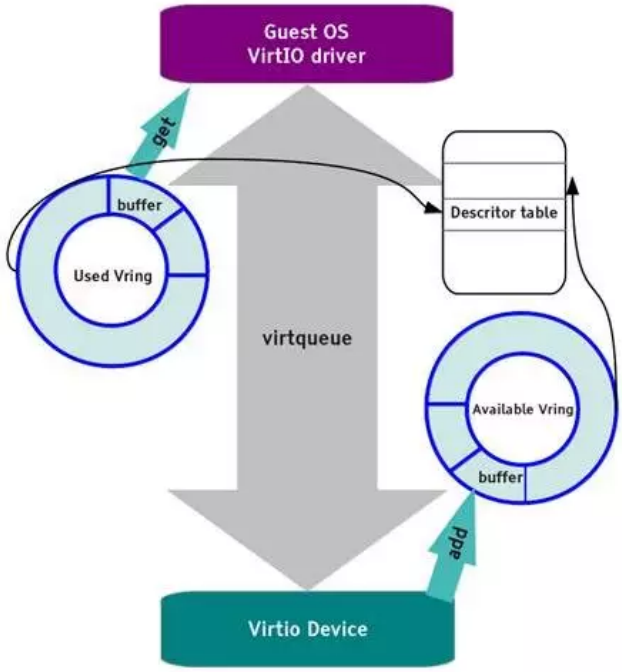

CONTENTS



严格来说，virtio 和 virtio-ring 可以看做是一层，virtio-ring 实现了 virtio 的具体通信机制和数据流程。或者这么理解可能更好，virtio 层属于控制层，负责前后端之间的通知机制（kick，notify）和控制流程，而 virtio-vring 则负责具体数据流转发。

## virtio 数据流交互机制

vring 主要通过两个环形缓冲区来完成数据流的转发，如下图所示。

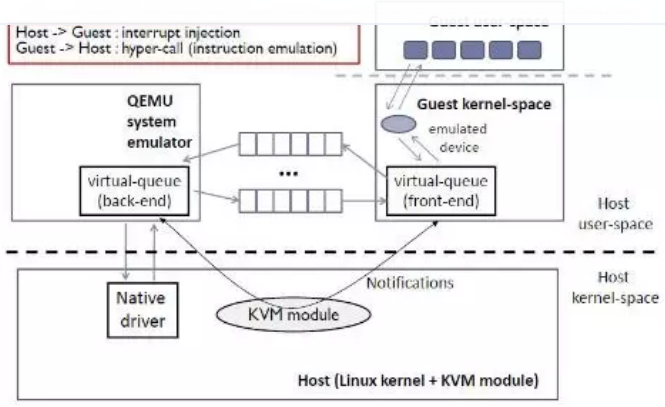



vring 包含三个部分，描述符数组 desc，可用的 available ring 和使用过的 used ring。

desc 用于存储一些关联的描述符，每个描述符记录一个对 buffer 的描述，available ring 则用于 guest 端表示当前有哪些描述符是可用的，而 used ring 则表示 host 端哪些描述符已经被使用。

Virtio 使用 virtqueue 来实现 I/O 机制，每个 virtqueue 就是一个承载大量数据的队列，具体使用多少个队列取决于需求，例如，virtio 网络驱动程序（virtio-net）使用两个队列（一个用于接受，另一个用于发送），而 virtio 块驱动程序（virtio-blk）仅使用一个队列。

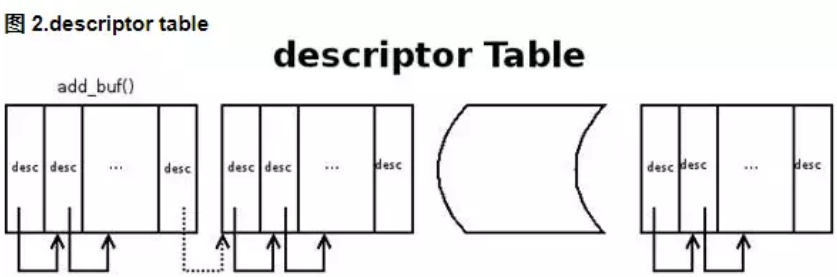
具体的，假设 guest 要向 host 发送数据，首先，guest 通过函数 virtqueue_add_buf 将存有数据的 buffer 添加到 virtqueue 中，然后调用 virtqueue_kick 函数，virtqueue_kick 调用 virtqueue_notify 函数，通过写入寄存器的方式来通知到 host。host 调用 virtqueue_get_buf 来 virtqueue 中收到的数据。






存放数据的 buffer 是一种分散-聚集的数组，由 desc 结构来承载，如下是一种常用的 desc 的结构：

图 2.descriptor table




当 guest 向 virtqueue 中写数据时，实际上是向 desc 结构指向的 buffer 中填充数据，完了会更新 available ring，然后再通知 host。

当 host 收到接收数据的通知时，首先从 desc 指向的 buffer 中找到 available ring 中添加的 buffer，映射内存，同时更新 used ring，并通知 guest 接收数据完毕。

## 总结：

virtio 是 guest 与 host 之间通信的润滑剂，提供了一套通用框架和标准接口或协议来完成两者之间的交互过程，极大地解决了各种驱动程序和不同虚拟化解决方案之间的适配问题。

virtio 抽象了一套 vring 接口来完成 guest 和 host 之间的数据收发过程，结构新颖，接口清晰。

---



作者：公众号「Linux云计算网络」，专注于Linux、云计算、网络领域技术干货分享

出处：https://www.cnblogs.com/bakari/p/8309638.html



分类：云计算，虚拟化
标签：虚拟化，云计算，KVM

« 上一篇： 那些年追过的……写过的技术博客
» 下一篇： DPDK NFV 性能提升

posted @ 2018-01-18 13:12 CloudDeveloper 阅读(1



CONTENTS