

KVM 初探

本文首发于我的公众号 **Linux云计算网络 (id: cloud_dev)**，专注于干货分享，号内有 **10T** 书籍和视频资源，后台回复「**初探**」即可获取，欢迎大家关注，二维码文末可以扫。

CONTENTS	×
1. KVM 与 Qemu 的前世今生	
2. KVM 架构	
3. VMM 管理工具 —— libvirt	

KVM 是业界最为流行的 Hypervisor，全称是 Kernel-based Virtual Machine。它是作为 Linux kernel 中的一个内核模块 `kvm.ko`，也可以看作是一个进程，被内核调度并管理，从 Linux 2.6.20 版本开始被完全正式加入到内核的主干开发和测试中。KVM 主要用于管理 CPU 和内存的虚拟化，IO 设备的虚拟化则是由 Qemu 来完成。为什么会有这样的分工，请继续往下看。

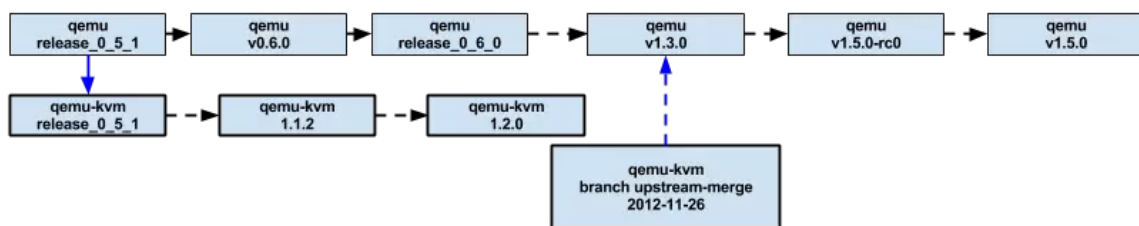
KVM 与 Qemu 的前世今生

Qemu 是一个纯软件实现的开源「模拟」软件，它能够模拟整套虚拟机的实现，包括 CPU、内存、各种 IO 设备、鼠标、键盘、USB、网卡、声卡等等，基本上没有它不能模拟的。有人可能会比较疑惑它跟 KVM 之间到底有何关系，我们可以把它们看成是合作关系，好基友，谁都离不开彼此。

KVM 离不开 Qemu。KVM 实现初期，为了简化开发和代码重用，在 Qemu 的基础上进行了修改，主要是将比较耗性能的 CPU 虚拟化和内存虚拟化部分移到了内核中实现，保留 IO 虚拟化模块在用户空间实现。这样的做法主要是考虑到性能的原因，CPU 和内存虚拟化是非常复杂的虚拟化模块，而且使用非常频繁，如果实现在用户空间的话，用户态和内核态的频繁切换势必会对性能造成很大的影响。那为什么要单独保留 IO 虚拟化在用户空间呢，这个也是权衡之下的结果，首先 IO 设备太多了，其次 IO 虚拟化相对其他两个模块使用不是很频繁，开销会小一些，所以，为了尽可能保持内核的纯净性，才有了这样的分配。

Qemu 离不开 KVM。上面也说了，Qemu 是一个纯软件的实现，运行在用户空间，性能非常低下，所以，从 Qemu 的角度，可以说是 Qemu 使用了 KVM 的虚拟化功能，为自身虚拟机提供加速。

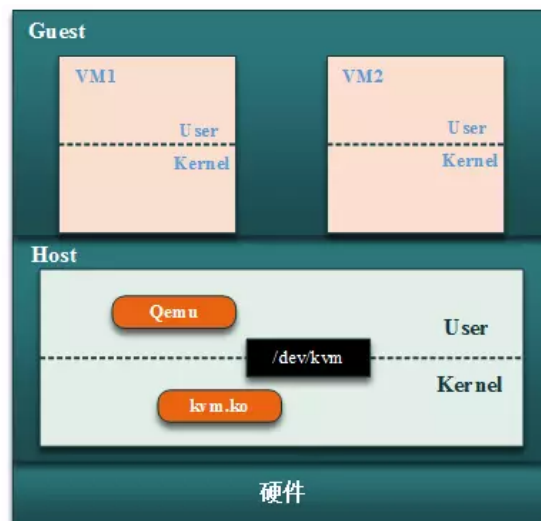
早期两者还没有区分（没有同居），KVM 修改的模块叫 `qemu-kvm`，到 Qemu1.3 版本之后，两者就合二为一了（同居啦），如果我们在用 Qemu 创建虚拟机时，要加载 KVM 模块，需要为其指定参数 `--enable-kvm`。



KVM 架构

KVM 是基于硬件虚拟化（Intel VT 或 AMD-V）实现的一套虚拟化解决方案，通过以上一个与 Qemu 关系的分析，我们基本上知道它在虚拟化领域处在一个什么样的地位。它其实只负责 CPU 和内存的虚拟化，不负责任何设备的模拟，而是提供接口给用户空间的 Qemu 来模拟。这个接口是 `/dev/kvm`，

Qemu 通过 `/dev/kvm` 接口设置一个虚拟机的地址空间，然后向它提供模拟好的 I/O 设备，并将相关的设备回显操作映射到宿主机，完成整个 I/O 设备的虚拟化操作。



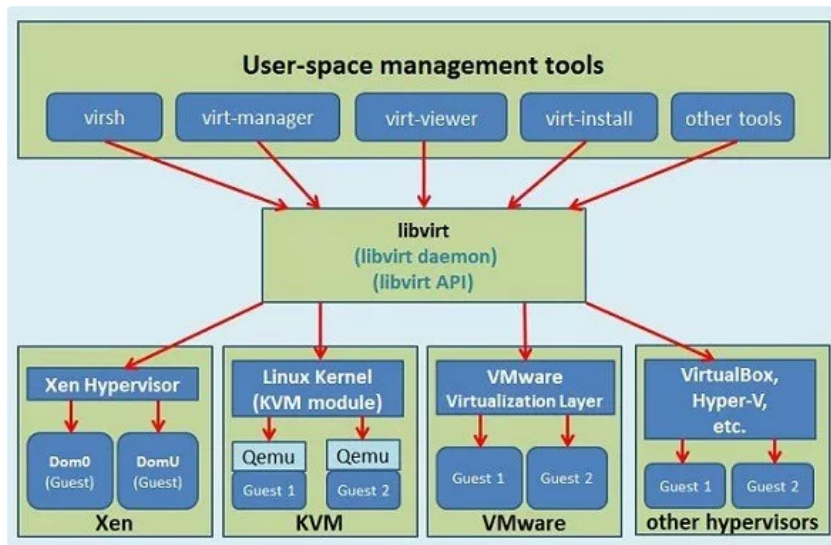


我就拿创建虚拟机举个例子，虚拟机本质上是宿主机的一个进程，包括用户态数据结构和内核态数据结构，用户态部分由 Qemu 创建并初始化，内核态部分则由 KVM 来完成，完成后会返回一个文件句柄来代表所创建的虚拟机，针对该文件句柄的 ioctl 调用就可以对虚拟机进行相应的管理，比如建立虚拟机地址空间和宿主机地址空间的映射关系，创建多个线程（虚拟处理器，vCPU）来供虚拟机使用等。对于创建出的 vCPU，也会生成相应的文件句柄，同样，对 vCPU 的文件句柄的 ioctl 调用就可以对 vCPU 进行管理。

关于这块的具体细节，后面会有文章来专门讨论。

VMM 管理工具 —— libvirt

目前，虚拟化这个领域可以说是百花齐放，针对不同的场景提出了很多的虚拟化解决方案，KVM、Xen、VMware、VirtualBox、Hyper-V 等等，具体的这些方案有什么特点，可以看前文「虚拟化技术总览」。这么多方案势必有很多通用的模块，不同之处可能在于，与不同硬件厂商的适配上，为了支持更多厂商，以及应用更多的领域，有很多 IaaS 解决方案需要融合多种虚拟化技术。这个时候如果有一个平台类的管理工具就会非常方便，libvirt 就是这样一个工具。



libvirt 除了能够支持多种虚拟化方案之外，还支持 OpenVZ、LXC 等容器虚拟化系统。它提供一套完善的虚拟机管理工具，支持 GUI 和命令行的形式，如 virsh、virt-install、virt-manager。由于它的通用性和易管理，很多云计算框架平台都在底层使用 libvirt 的 API 来管理虚拟机，比如 OpenStack、OpenNebula、Eucalyptus 等。这个工具我们仅仅提一下，有兴趣的可以装个玩玩。

下面给出 KVM 和 Qemu 的 git 路径，有兴趣的可以把源码下载下来研究下。

```
kvm.git:  
git clone git://git.kernel.org/pub/scm/virt/kvm/kvm.git  
qemu.git (包括了 kvm) :  
git clone git://git.qemu-project.org/qemu.git
```

Copy

我的博客即将入驻“云栖社区”，诚邀技术同仁一同入驻。

我的公众号「Linux云计算网络」(id: cloud_dev)，号内有 10T 书籍和视频资源，后台回复「1024」即可领取，分享的内容包括但不限于 Linux、网络、云计算虚拟化、容器 Docker、OpenStack、Kubernetes、工具、SDN、OVS、DPDK、Go、Python、C/C++ 编程技术等内容，欢迎大家关注。

Linux云计算网络

云计算 | 网络 | Linux | 干货

获取学习大礼包后台
回复“1024”

加群交流后台回复“加群”





本站使用「署名 4.0 国际」创作共享协议，转载请在文章明显位置注明作者及出处。

分类: 云计算, 虚拟化

标签: 虚拟化, 云计算, KVM

CONTENTS



1. KVM 与 Qemu 的前世今生
2. KVM 架构
3. VMM 管理工具 —— libvirt

推荐 1

赞赏

收藏

反对 0

« 上一篇: OVS 总体架构、源码结构及数据流程全面解析

» 下一篇: 那些年追过的.....写过的技术博客

posted @ 2018-01-01 15:08 CloudDeveloper 阅读(1888) 评论(0) 编辑 收藏

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

