

KVM 的存储虚拟化是通过存储池（Storage Pool）和卷（Volume）来管理的。

Storage Pool 是宿主机上可以看到的一片存储空间，可以是多种类型，后面会详细讨论。Volume 是在 Storage Pool 中划分出的一块空间，宿主机将 Volume 分配给虚拟机，Volume 在虚拟机中看到的就是一块硬盘。

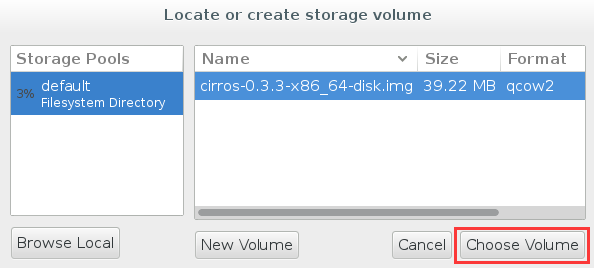
下面我们学习不同类型的 Storage Pool

**目录类型的 Storage Pool**

文件目录是最常用的 Storage Pool 类型。  
KVM 将宿主机目录 /var/lib/libvirt/images/ 作为默认的 Storage Pool。

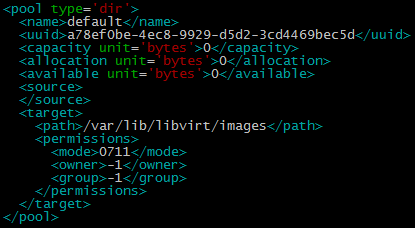
那么 Volume 是什么呢？  
答案就是该目录下面的文件了，一个文件就是一个 Volume。

大家是否还记得我们之前创建第一个虚机 kvm1 的时候，就是将镜像文件 cirros-0.3.3-x8664-disk.img 放到了这个目录下。文件 cirros-0.3.3-x8664-disk.img 也就是Volume，对于 kvm1 来说，就是它的启动磁盘了。



那 KVM 是怎么知道要把 /var/lib/libvirt/images 这个目录当做默认 Storage Pool 的呢？

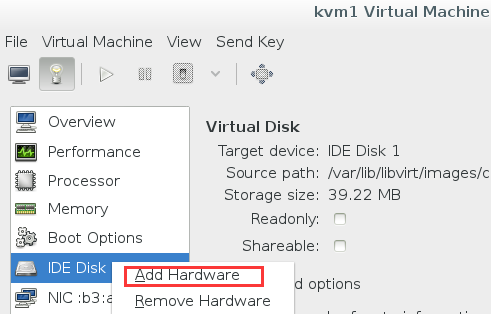
实际上 KVM 所有可以使用的 Storage Pool 都定义在宿主机的 /etc/libvirt/storage 目录下，每个 Pool 一个 xml 文件，默认有一个 default.xml，其内容如下：



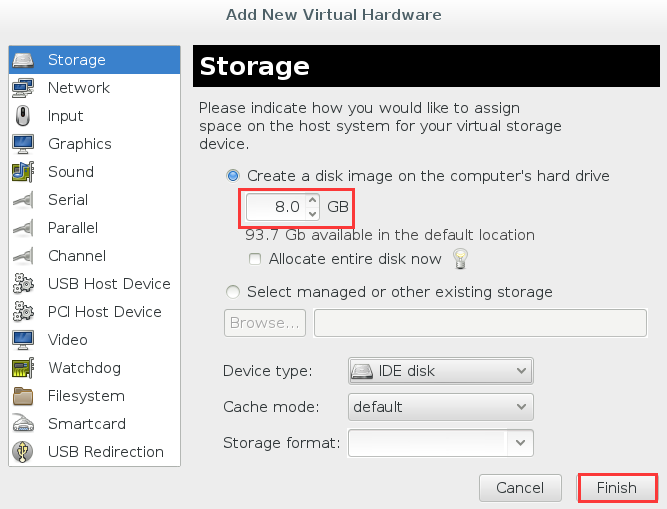
注意：Storage Pool 的类型是 “dir”，目录的路径就是 /var/lib/libvirt/images

下面我们为虚机 kvm1 添加一个新的磁盘，看看有什么变化。

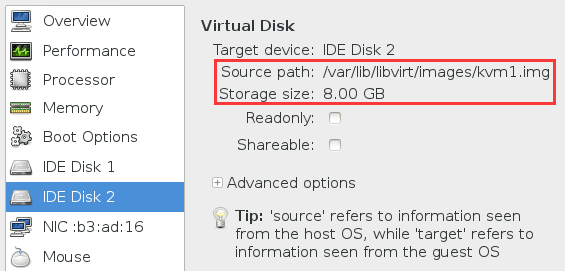
在 virt-manager 中打开 kvm1 的配置页面，右键添加新硬件



在默认 Pool 中创建一个 8G 的卷。



点击 “Finish”，可以看到新磁盘的信息。

http://ipaiban.com/js/ueditor/themes/default/images/spacer.gif

在 /var/lib/libvirt/images/ 下多了一个 8G 的文件 kvm1.img

*root@ubuntu:~# ls -l /var/lib/libvirt/images/          
total 14044*

*-rw-r--r-- 1 root root   14417920 Sep  4 11:24 cirros-0.3.3-x86\_64-disk.img*

*-rw------- 1 root root 8589934592 Sep  4 21:39 kvm1.img*

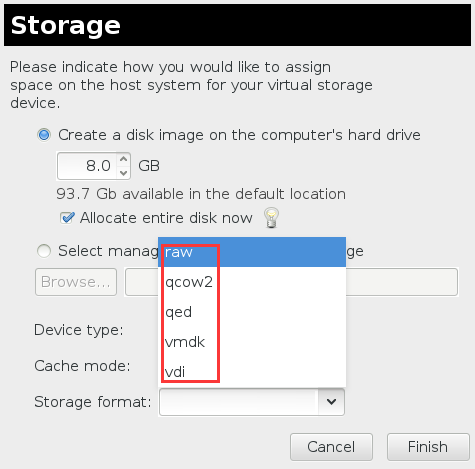
使用文件做 Volume 有很多优点：存储方便、移植性好、可复制、可远程访问。

前面几个优点都很好理解，这里对“可远程访问”多解释一下。

远程访问的意思是镜像文件不一定都放置到宿主机本地文件系统中，也可以存储在通过网络连接的远程文件系统，比如 NFS，或者是分布式文件系统中，比如 GlusterFS。

这样镜像文件就可以在多个宿主机之间共享，便于虚机在不同宿主机之间做 Live Migration；如果是分布式文件系统，多副本的特性还可以保证镜像文件的高可用。

KVM 支持多种 Volume 文件格式，在添加 Volume 时可以选择



raw 是默认格式，即原始磁盘镜像格式，移植性好，性能好，但大小固定，不能节省磁盘空间。

qcow2 是推荐使用的格式，cow 表示 copy on write，能够节省磁盘空间，支持 AES 加密，支持 zlib 压缩，支持多快照，功能很多。

vmdk 是 VMWare 的虚拟磁盘格式，也就是说 VMWare 虚机可以直接在 KVM上 运行。

下一节介绍 LVM 类型的 Storage Pool。