

---

## 第二章 KVM 虚拟机克隆和快照

### 本节所讲内容：

- 2.1 KVM 虚拟机克隆方法
- 2.2 虚拟机常用镜像格式对比
- 2.3 KVM 虚拟机快照功能使用方法
- 2.4 virsh 命令常见用法
- 2.5 KVM 常用镜像格式转换

### 2.1 KVM 虚拟机克隆方法

#### 2.1.1 克隆 kvm 虚拟机

克隆前，centos7.0 需要提前关机。

语法：virt-clone -o 原虚拟机 -n 新虚拟机 -f 新虚拟机镜像存放路径

选项：-o old -n new

虚拟机镜像：就是整个虚拟机文件。不是操作系统光盘镜像 rhel6.5.iso

比如：KVM 虚拟机的镜像文件

```
[root@xuegod63 ~]# mount /dev/sdb1 /var/lib/libvirt/images/
```

```
[root@xuegod63 ~]# cd /var/lib/libvirt/images/
```

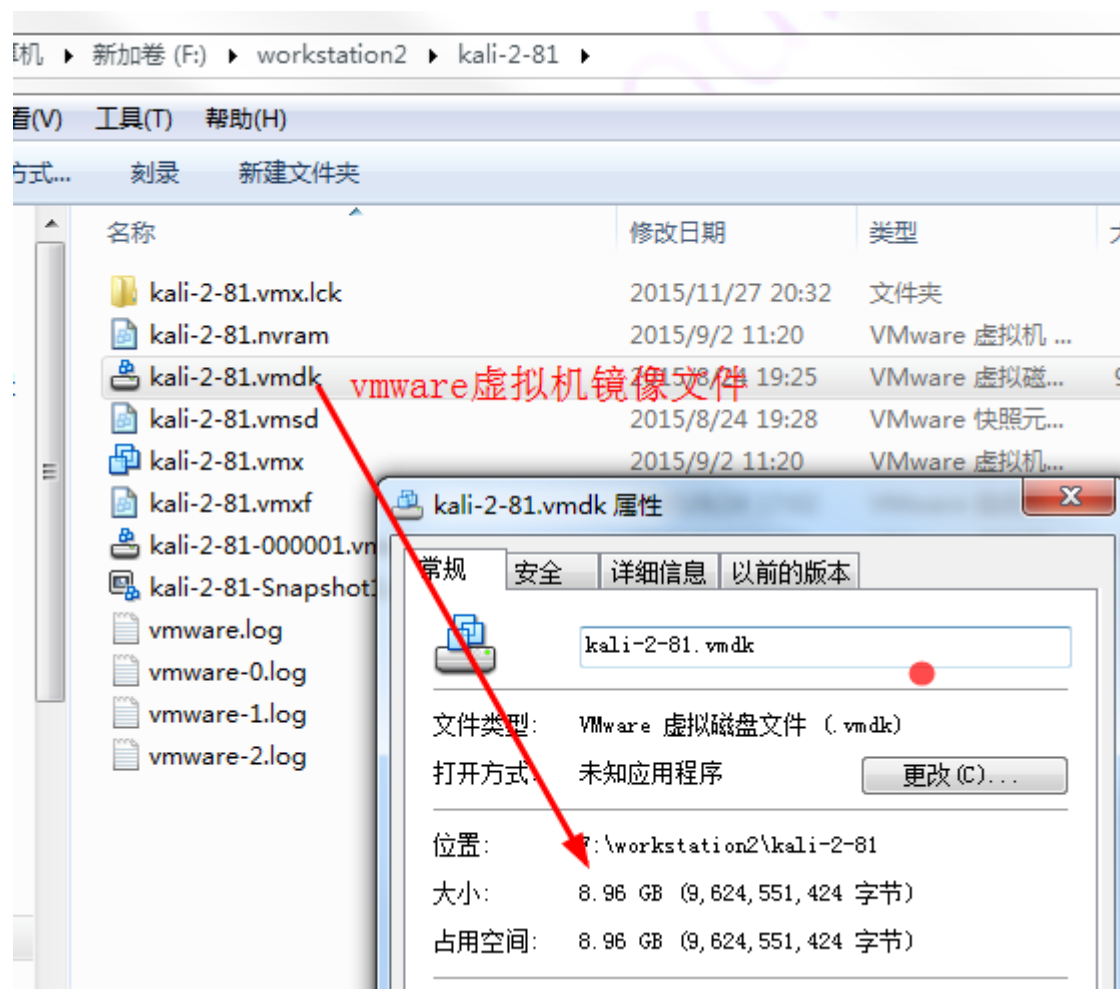
```
[root@xuegod63 images]# ll -h
```

总用量 1.2G #实际镜像占用 1.2G

```
-rw----- 1 root root 9.1G 9月 21 23:33 centos7.0.qcow2
```

#这就是虚拟机镜像文件，不是操作系统镜像格式是 rhel6.5.iso

vmware 的镜像文件：



注：虚拟机克隆完毕后，MAC 地址也是需要修改，网卡配置文件也修改一下。

### 2.1.2 基于 centos7.0 克隆一台虚拟机：xuegod63-kvm2

```
[root@xuegod63 ~]# cd /var/lib/libvirt/images/
```

```
[root@xuegod63 images]# virt-clone -o centos7.0 -n xuegod63-kvm2 -f
```

```
/var/lib/libvirt/images/xuegod63-kvm2.img
```

```
Allocating 'xuegod63-kvm2' | 8.0 GB 03:34
```

Clone 'xuegod63-kvm2' created successfully.

```
[root@xuegod63 images]# ll -h
```

总用量 3.2G

```
-rw----- 1 root root 9.1G 9月 21 23:33 centos7.0.qcow2 (最大空间是 8G,没立即分配磁)
```

```
-rw----- 1 root root 1.2G 9月 23 18:03 xuegod63-kvm2.img
```

查看克隆完后，使用磁盘情况。

```
[root@xuegod63 images]# df -h
```

文件系统	容量	已用	可用	已用%	挂载点
/dev/sda3	10G	8.2G	1.9G	82%	/
devtmpfs	2.9G	0	2.9G	0%	/dev
tmpfs	2.9G	0	2.9G	0%	/dev/shm

---

tmpfs	2.9G	9.1M	2.9G	1%	/run
tmpfs	2.9G	0	2.9G	0%	/sys/fs/cgroup
/dev/sr0	4.3G	4.3G	0	100%	/mnt
/dev/sda1	197M	174M	24M	88%	/boot
tmpfs	582M	40K	582M	1%	/run/user/0
/dev/sdb1	20G	3.2G	17G	16%	/var/lib/libvirt/images

### 2.1.3 KVM 虚拟机组成

一台 KVM 虚拟机由两部分组成：虚拟机配置文件和镜像 img

查看虚拟机的配置文件：

```
[root@xuegod63 images]# cd /etc/libvirt/qemu
[root@xuegod63 ~]# ls /etc/libvirt/qemu
networks  centos7.0.xml  xuegod63-kvm2.xml
```

查看虚拟机配置文件内容：

```
[root@xuegod63 qemu]# cat  centos7.0.xml
...
<domain type='kvm'>
  <name>centos7.0</name>
  <uuid>893bfff4-a032-408a-8928-9d9e914bea65</uuid>
  <memory unit='KiB'>1048576</memory>
  <currentMemory unit='KiB'>1048576</currentMemory>
  <vcpu placement='static'>1</vcpu>
  ...
  <emulator>/usr/libexec/qemu-kvm</emulator>
  <disk type='file' device='disk'>
    <driver name='qemu' type='qcow2'>
    <source file='/var/lib/libvirt/images/centos7.0.qcow2'>
    <target dev='vda' bus='virtio'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'>
    ...
  <interface type='bridge'>
    <mac address='52:54:00:f5:16:67'>
    <source bridge='br0'>
    <model type='virtio'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'>
```

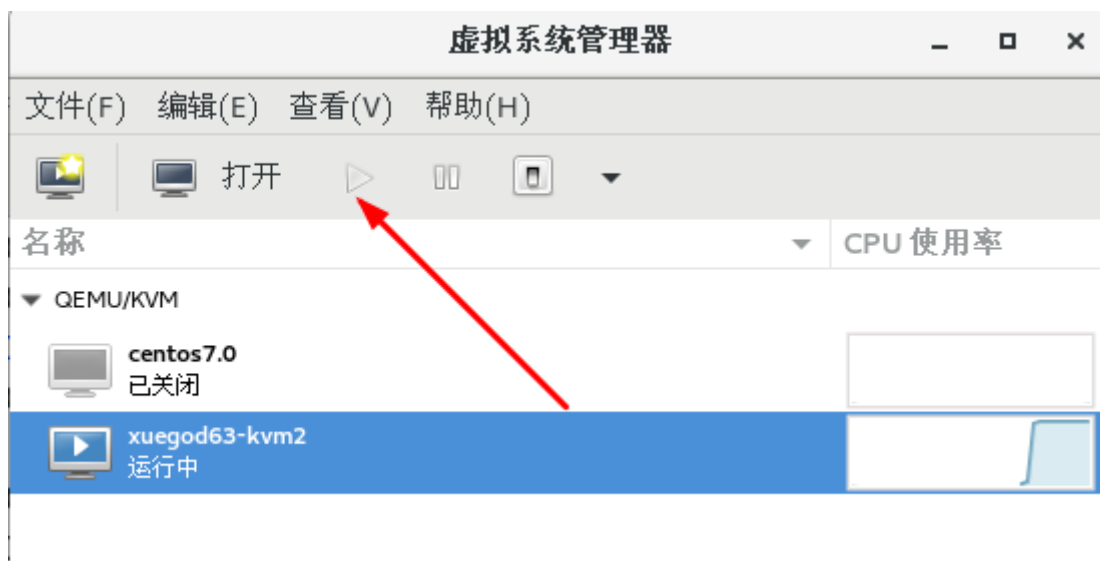
查看原虚拟机和克隆完的虚拟机配置文件的区别：

```
[root@xuegod63 qemu]# vimdiff  centos7.0.xml xuegod63-kvm2.xml
```

学到一招：1

注：这里可以看出两者的 MAC 地址是不一样的。所以对克隆或复制出来的虚拟机启动后，要修改网络，因为克隆出来的虚拟机的真实 MAC 地址和操作系统中的 ifcfg-eth0 中的 MAC 地址不一样。

启动 xuegod63-kvm2 测试一下：



#### 2.1.4 测试新克隆的机器：

进入新克隆的机器：

ip addr show 查看 IP 和 mac 地址。系统中没有 ifconfig

```
[root@localhost ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 52:54:00:2b:fb:79 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.71/24 brd 192.168.1.255 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 fe80::6e5b:c45c:aec0:a059/64 scope link
        valid_lft forever preferred_lft forever
```

ping g.cn #发现可以直接上网

```
[root@localhost ~]# ping g.cn
PING g.cn (203.208.51.55) 56(84) bytes of data.
64 bytes from 203.208.51.55 (203.208.51.55): icmp_seq=1 ttl=47 time=53.5 ms
64 bytes from 203.208.51.55 (203.208.51.55): icmp_seq=2 ttl=47 time=47.2 ms
64 bytes from 203.208.51.55 (203.208.51.55): icmp_seq=3 ttl=47 time=52.9 ms
64 bytes from 203.208.51.55 (203.208.51.55): icmp_seq=4 ttl=47 time=47.0 ms
```

查看克隆后 xuegod63-kvm1 的配置文件：

```
[root@xuegod63 qemu]# vim /etc/libvirt/qemu/xuegod63-kvm2.xml
```

```
<interface type='bridge'>
  <mac address='52:54:00:2b:fb:79'/>
```

注：发现虚拟机磁盘中的 mac 地址已经和克隆后的 MAC 地址一样。可以直接上网，不需要再另外删除相关内容。在 centos6 下，kvm 还没有这样的功能。

#### 2.1.5 在 rhel6 下 kvm 克隆后的操作

登录新克隆的虚拟机删除原来的 mac 和 IP 地址，让新克隆的机器可以上网：

```
[root@xuegod63 ~]# rm -rf /etc/udev/rules.d/70-persistent-*
```

```
[root@xuegod63 ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth0 #写入以下内容
```

```
DEVICE=eth0
IPADDR=192.168.1.72
NETMASK=255.255.255.0
ONBOOT=yes
NAME=eth0
~
~
~
~
```

注：记得把 ONBOOT="no" 改为：ONBOOT="yes"

注：把原配置文件中的 MAC 和 UUID 地址删除，然后修改一个和原虚拟机不一样的 IP

```
[root@xuegod63 ~]# reboot    #重启生效
```

或

方法 2：

```
[root@xuegod63 ~]# start_udev    # 重新启动 udev 服务，自动生成刚删除的
```

/etc/udev/rules.d/70-persistent-\*文件，新生成的 udev 文件，会使用新系统的 MAC 地址。

```
[root@xuegod63 ~]# service network restart
```

## 2.2 虚拟机常用镜像格式对比

目前主要虚拟机的镜像格式：raw，cow，qcow，qcow2，vmdk。

### 2.2.1 raw 格式镜像

raw：老牌的镜像格式，用一个字来说就是裸，也就是赤裸裸，你随便 dd 一个 file 就模拟了一个 raw 格式的镜像。由于裸的彻底，性能上说的话还是不错的。centos6 上 KVM 和 XEN 默认的格式还是这个格式。centos7 以上默认是 qcow2。

裸的好处还有就是简单，支持转换成其它格式的虚拟机镜像对裸露的它来说还是很简单的（如果其它格式需要转换，有时候还是需要它做为中间格式），空间使用来看，这个很像磁盘，使用多少就是多少（du -h 看到的大小就是使用大小）。

例：qcow2 转为 vmdk 方法是：qcow2 转为 raw，然后把 raw 转为 vmdk。也可以直接 qcow2 转为 vmdk

扩展：

佳能相机上的高保真用的就是这种 raw 格式。RAW 的原意就是“未经加工”。可以理解为：RAW 图像就是 CMOS 或者 CCD 图像感应器将捕捉到的光源信号转化为数字信号的原始数据。RAW 理解为“数字底片”

缺点：不支持 snapshot 快照。

### 2.2.2 cow、qcow、qcow2 格式

1、cow 格式：还没有成熟，就被放弃了。后来被 qcow 格式所取代。

2、qcow 格式：刚刚出现的时候有比较好的特性，但其性能和 raw 格式对比还是有很大的差距，目前已经被新版本的 qcow2 取代。

3、qcow2 格式：

现在比较主流的一种虚拟化镜像格式，经过一代的优化，目前 qcow2 的性能上接近 raw 裸格式的性能

qcow2 格式支持 snapshot，可以在镜像上做 N 多个快照，具有以下优点：

---

- 更小的存储空间
- 支持创建 image 镜像
- 支持多个 snapshot , 对历史 snapshot 进行管理
- 支持 zlib 的磁盘压缩
- 支持 AES 的加密

### 2.2.3 vmdk 格式 :

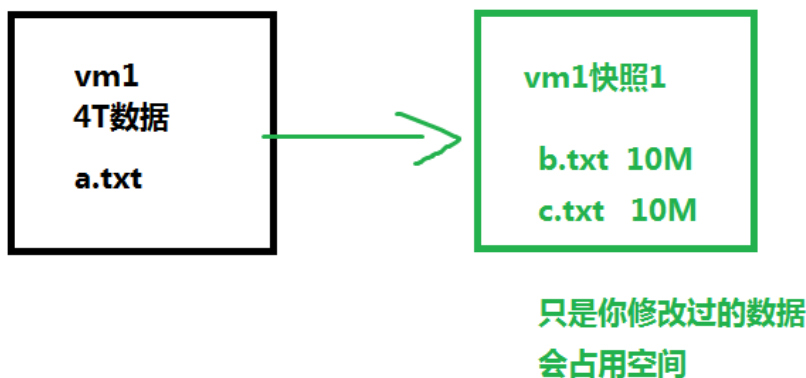
VMware 的格式 整体性能最好 , 因为原本 VMware 就是做虚拟化起家。从性能和功能上来说 , vmdk 应该算最出色的 , 由于 vmdk 结合了 VMware 的很多能力 , 目前来看 , KVM 和 XEN 使用这种格式的情况不是太多。但就 VMware 的企业级虚拟化 Esxi 来看 , 它的稳定性和各方面的能力都很好

## 2.3 KVM 虚拟机快照功能使用方法

### 2.3.1 KVM 快照 :

快照的作用 : 1、热备 2、灾难恢复 3、回滚到历中的某个状态

快照使用空间 :



kvm 快照 , 分两种 :

方法 1 : 使用 lvm 快照 , 如果分区是 lvm , 可以利用 lvm 进行 kvm 的快照备份

方法 2 : 使用 qcow2 格式的镜像创建快照。

### 2.3.2 创建 KVM 快照

在 centos6 下 , kvm 虚拟机默认使用 raw 格式的镜像格式 , 性能最好 , 速度最快 , 它的缺点就是不支持一些新的功能 , 如支持镜像, zlib 磁盘压缩, AES 加密等。

要使用快照功能 , 磁盘格式必须为 qcow2。

查看磁盘格式

```
[root@xuegod63 ~]# qemu-img info /var/lib/libvirt/images/xuegod63-kvm2.img
```

```
image: /var/lib/libvirt/images/xuegod63-kvm2.img
```

```
file format: qcow2
```

```
virtual size: 9.0G (9663676416 bytes)
```

```
disk size: 1.1G # 创建虚拟机时 , 选择不要立即分配磁盘空间
```

```
cluster_size: 65536
```

```
Format specific information:
```

```
compat: 1.1
```

---

注：qcow2 格式可以直接创建快照

### 2.3.3 对虚拟机进行快照管理

对 xuegod63-kvm2 虚拟机创建快照

语法：virsh snapshot-create 虚拟机的名字

例：创建一个快照

```
[root@xuegod63 images]# virsh snapshot-create xuegod63-kvm2
```

Domain snapshot 1440726046 created

注：创建快照时不需要关闭虚拟机，关机创建快照比较快，开机创建快照需要把内存中的内容写到磁盘上，记录虚拟机这一时刻的状态。

```
[root@xuegod63 images]# virsh shutdown xuegod63-kvm2
```

域 xuegod63-kvm2 被关闭

```
[root@xuegod63 images]# virsh snapshot-create xuegod63-kvm2 #这时创建快照很快  
已生成域快照 1513776621
```

创建快照时启个名字：

语法：virsh snapshot-create-as KVM 虚拟机名 快照名

例：生一个快照，名字是：httpd

```
[root@xuegod63 ~]# virsh list --all ##查看虚拟机名字
```

Id	Name	State
-	centos7.0	shut off
-	xuegod63-kvm2	shut off

```
[root@xuegod63 ~]# virsh snapshot-create-as xuegod63-kvm2 httpd
```

Domain snapshot httpd created

查看虚拟机镜像快照列表

```
[root@xuegod63 images]# virsh snapshot-list xuegod63-kvm2
```

名称	生成时间	状态
1513776483	2017-12-20 21:28:03 +0800	running
1513776621	2017-12-20 21:30:21 +0800	shutoff
httpd	2017-12-20 21:32:09 +0800	shutoff

快照也支持中文：

```
[root@xuegod63 ~]# virsh snapshot-create-as xuegod63-kvm2 安装好 httpd
```

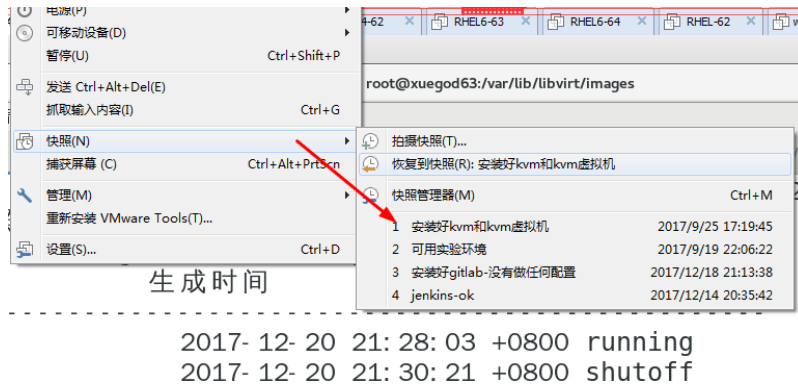
Domain snapshot 安装好 httpd created

```
[root@xuegod63 ~]# virsh snapshot-list xuegod63-kvm2
```

Name	Creation Time	State
1440741062	2015-08-28 13:51:02 +0800	shutoff
httpd	2015-08-28 13:58:26 +0800	shutoff
安装好 httpd	2015-08-28 14:00:15 +0800	shutoff

查看最近一次使用的快照版本。

对于 vmware : 第一个就是当前的快照



可以看到为当前最新的快照版本。

```
[root@xuegod63 images]# virsh snapshot-current xuegod63-kvm2
```

```
<domainsnapshot>
```

```
<name>安装好 httpd</name>
```

```
<state>shutoff</state>
```

```
<creationTime>1440726046</creationTime>
```

快照配置文件在/var/lib/libvirt/qemu/snapshot/虚拟机名称下

```
[root@xuegod63 ~]# ll -h /var/lib/libvirt/qemu/snapshot/xuegod63-kvm2/
```

总用量 24K

```
-rw----- 1 root root 4.8K 9月 23 18:32 1506162674.xml
```

```
-rw----- 1 root root 4.8K 9月 23 18:32 httpd.xml
```

```
-rw----- 1 root root 4.9K 9月 23 18:32 安装好 httpd.xml
```

#快照的配置文件

互动：腾讯云，阿里云中快照功能收费吗？

快照要收费！

- 自动快照策略已在2016年3月31日升级为2.0版本，新版本中您需要为每个磁盘重新配置快照策略，其自动快照才能生效；
- 目前2.0版本的快照服务处于免费试用阶段，商用时间及收费标准另行通知，在正式商业化收费之前，您可以选择删除所有快照和自动快照费用；
- 您如果已经设置了2.0版本的快照策略，商用之后，已生成的快照会按照收费标准进行收费。

快照空间占用的是哪的空间？虚拟机自身 img 镜像空间？还是云平台自身的空间？

查看快照占用的大小：

查看快照占用的大小：

```
[root@xuegod63 ~]# qemu-img info /var/lib/libvirt/images/xuegod63-kvm2.img
```

```
image: /var/lib/libvirt/images/xuegod63-kvm2.img
```

```
file format: qcow2
```

```
virtual size: 9.0G (9663676416 bytes)
```

```
disk size: 3.1G
```

```
cluster_size: 65536
```

```
Snapshot list:
```



---

ID	TAG	VM SIZE	DATE	VM CLOCK
1	1506162674	0	2017-09-23 18:31:15	00:00:00.000
2	httpd	0	2017-09-23 18:32:04	00:00:00.000
3	安装好 httpd	0	2017-09-23 18:32:15	00:00:00.000

Format specific information:

compat: 1.1

lazy refcounts: true

#### 2.3.4 恢复虚拟机快照

恢复虚拟机快照必须关闭虚拟机。注：阿里云也需要关闭后再恢复快照

```
[root@xuegod63 snapshot]# virsh domstate xuegod63-kvm2 #查看虚拟机状态
shut off
```

确认需要恢复的快照，这里恢复到 **httpd**

```
[root@xuegod63 snapshot]# virsh snapshot-list xuegod63-kvm2
```

Name	Creation Time	State
-----		
1448633311	2015-11-27 22:08:31 +0800	shutoff
httpd	2015-11-27 22:10:32 +0800	shutoff

#### 执行恢复

```
[root@xuegod63 snapshot]# virsh snapshot-revert xuegod63-kvm2 httpd
```

#### 确认恢复版本

```
[root@xuegod63 snapshot]# virsh snapshot-current xuegod63-kvm2 | more
<domainsnapshot>
  <name>httpd</name>
  <state>shutoff</state>
  <parent>
    <name>1448633311</name>
  </parent>
  <creationTime>1448633432</creationTime>
```

#### 2.3.5 删除快照和开机创建快照

##### 1、删除快照

```
[root@xuegod63 snapshot]# virsh snapshot-delete xuegod63-kvm2 1448633311
Domain snapshot 1448633311 deleted
```

查看：

```
[root@xuegod63 snapshot]# qemu-img info
/var/lib/libvirt/images/xuegod63-kvm2.qcow2
image: /var/lib/libvirt/images/xuegod63-kvm2.qcow2
file format: qcow2
virtual size: 8.0G (8589934592 bytes)
disk size: 1.1G
cluster_size: 65536
```

---

Snapshot list:

ID	TAG	VM SIZE	DATE	VM CLOCK
2	httpd	0	2015-11-27 22:10:32	00:00:00.000

上面操作已经还原到快照：httpd。 现在测试使用这个快照启动系统：

```
[root@xuegod63 snapshot]# virsh start xuegod63-kvm2
Domain xuegod63-kvm2 started
```

## 2、开机创建快照：

```
[root@xuegod63 ~]# virsh snapshot-create-as xuegod63-kvm2 http-start
Domain snapshot http-start created
```

查看开机创建的快照占用的磁盘空间：

```
[root@xuegod63 ~]# qemu-img info /var/lib/libvirt/images/xuegod63-kvm2.img
image: /var/lib/libvirt/images/xuegod63-kvm2.img
file format: qcow2
virtual size: 9.0G (9663676416 bytes)
disk size: 3.1G
cluster_size: 65536
```

Snapshot list:

ID	TAG	VM SIZE	DATE	VM CLOCK
2	httpd	0	2017-09-23 18:32:04	00:00:00.000
3	安装好 httpd	0	2017-09-23 18:32:15	00:00:00.000
4	http-start	127M	2017-09-23 18:35:19	00:00:27.178

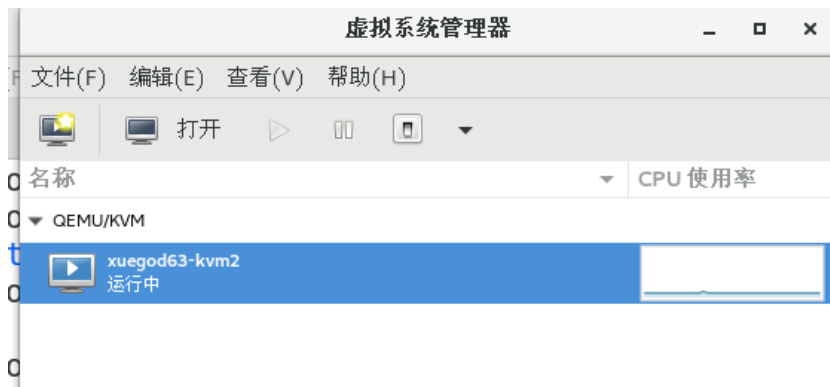
注：127M，存储虚拟机开机时的运行状态，比如内存中的数据。

## 2.4 virsh 常用命令

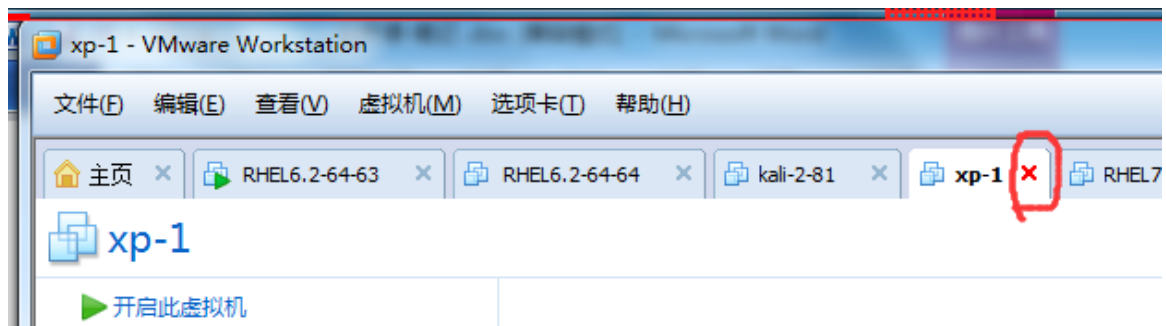
```
# virsh list //查看已打开虚拟机列表
# virsh list --all //查看所有虚拟机列表
# virsh version //查看 virsh 版本号
# virsh start xuegod63-kvm2 //启动 xuegod63-kvm2 虚拟机
# virsh shutdown xuegod63-kvm2 //关机 xuegod63-kvm2 虚拟机
# virsh dumpxml xuegod63-kvm2 > xuegod63-kvm2.xml //导出 xuegod63-kvm2 虚拟机
```

配置文件

```
[root@xuegod63 images]# virsh dumpxml centos7.0 > centos7.0.xml
[root@xuegod63 ~]# virsh undefine centos7.0 //取消 centos7.0 定义
域 centos7.0 已经被取消定义
```



相当于在 VMware 上执行了：点一下 xp-1 那个叉



[root@xuegod63 qemu]# ls /etc/libvirt/qemu #这下面，也没有虚拟机配置文件

# virsh define 虚拟机.xml 的绝对路径 //把虚拟机重新加入到 virt-manager 目录列表。

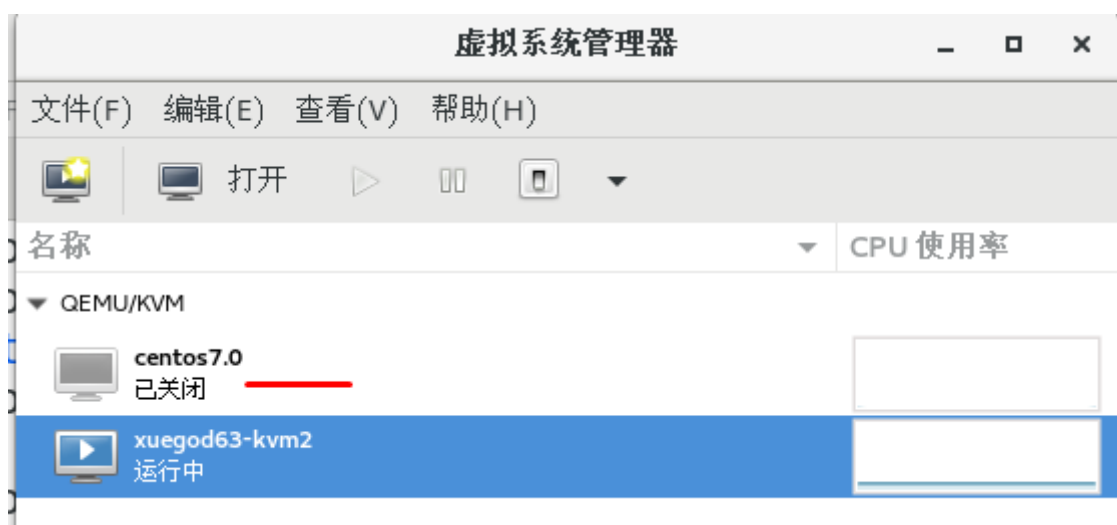
例：

[root@xuegod63 ~]# virsh define /opt/centos7.0

定义域 centos7.0 (从 /opt/centos7.0)

[root@xuegod63 qemu]# ls

centos7.0.xml networks xuegod63-kvm2.xml



# virsh destroy xuegod63-kvm2 //强制关闭 xuegod63-kvm2 虚拟机。正常关不了机时，用这个。

---

```
[root@xuegod63 ~]# virsh autostart centos7.0 //设置开机自启动 node1。
[root@kvm01 ~]# virsh autostart --disable vm1 #取消虚拟机随宿主机开机自启
[root@kvm01 ~]# virsh suspend vm1 #挂起虚拟机
[root@kvm01 ~]# virsh resume vm1 #恢复虚拟机
[root@kvm01 ~]# virsh console vm1 #控制台管理虚拟机
```

## 2.5 实战 1 : qcow2 格式转换成 raw

### 2.5.1 关闭 xuegod63-kvm1 , 转换格式

```
[root@xuegod63 ~]# virsh shutdown xuegod63-kvm2
```

转换磁盘格式

qemu-img 的参数

-f 源镜像的格式

-O 目标镜像的格式

例 : qcow2 格式转换成 raw

```
[root@xuegod63 images]# qemu-img convert -f qcow2 -O raw /var/lib/libvirt/images/
xuegod63-kvm2.img /var/lib/libvirt/images/xuegod63-kvm2.raw
```

查看转换后的格式, 已经转换成了 raw

```
[root@xuegod63 images]# qemu-img info /var/lib/libvirt/images/xuegod63-kvm2.raw
image: /var/lib/libvirt/images/ xuegod63-kvm2.raw
file format: raw
```

### 2.5.2 其他镜像格式转换方法 :

例 1 : 将 vmdk 转换为 qcow2

```
qemu-img convert -f vmdk -O qcow2 source-name.vmdk target-name.qcow2
```

例 2 : 将 qcow2 转换为 vmdk

```
qemu-img convert -f qcow2 -O vmdk source-name.qcow2 target-name.vmdk
```

### 2.5.3 修改虚拟机配置文件, 使用 raw 格式镜像文件, 来启动虚拟机 :

方法 1 :

```
[root@xuegod63 ~]# virsh edit xuegod63-kvm2 #修改配置文件
```

改 :

```
<driver name='qemu' type='qcow2' cache='none'/>
```

```
<source file='/var/lib/libvirt/images/xuegod63-kvm2.img'/>
```

为 :

```
23 <driver name='qemu' type='raw' cache='none'/>
```

```
24 <source file='/var/lib/libvirt/images/xuegod63-kvm2.raw'/>
```

方法 2 :

#注意 vim 直接编辑配置文件/etc/libvirt/qemu/xuegod63-kvm2.xml 不生效

修改后, 需要重启服务

```
[root@xuegod63 ~]# /etc/init.d/libvirtd restart
```

---

## 总结：

- 2.1 KVM 虚拟机克隆方法
- 2.2 虚拟机常用镜像格式对比
- 2.3 KVM 虚拟机快照功能使用方法
- 2.4 virsh 命令常见用法
- 2.5 KVM 常用镜像格式转换