

# KVM 虚拟化

作者：老男孩 Linux 教育-张亚

归档：上课文档

2020/1/6

---

## 快捷键：

Ctrl + 1	标题 1
Ctrl + 2	标题 2
Ctrl + 3	标题 3
Ctrl + 4	实例
Ctrl + 5	程序代码
Ctrl + 6	正文

---

## 格式说明：

蓝色字体：注释

黄色背景：重要

绿色背景：注意

---

## 目 录

第 1 章 KVM 介绍.....	1
第 2 章 安装 KVM.....	2
2.1 安装软件.....	2
第 3 章 创建并安装一台 KVM 虚拟机.....	2
3.1 安装一台 kvm 虚拟机.....	2
第 4 章 KVM 虚拟机常用管理命令.....	3
4.1 查看虚拟机.....	3
4.2 开启虚拟机.....	3
4.3 关闭虚拟机.....	3
4.4 重启虚拟机.....	3
4.5 查看配置文件.....	3
4.6 导出配置文件.....	3
4.7 删除虚拟机.....	3
4.8 导入虚拟机.....	3
4.9 生产故障.....	4
4.10 修改配置文件.....	5
4.11 主机重命名.....	5
4.12 主机挂起.....	6
4.13 主机恢复.....	6
第 5 章 连接 KVM 虚拟机.....	7
5.1 查看 VNC 端口号.....	7
5.2 kvm 虚拟机开机启动.....	6
5.3 查看哪些虚拟机是开机自启动.....	7
5.4 console 管理口.....	8
第 6 章 KVM 磁盘管理.....	8
6.1 虚拟机磁盘格式.....	8
6.2 磁盘工具的常用命令.....	错误！未定义书签。
第 7 章 KVM 快照管理.....	11
7.1 快照管理.....	错误！未定义书签。
第 8 章 KVM 虚拟机克隆.....	11
8.1 完整克隆.....	11
8.2 链接克隆.....	12

---

第 9 章 KVM 虚拟机网络.....	14
9.1 桥接网络.....	14
9.2 NAT 网卡模式.....	错误！未定义书签。
9.3 桥接网卡模式.....	错误！未定义书签。
第 10 章 KVM 虚拟机热添加技术.....	15
10.1 热添加硬盘.....	15
10.2 热添加网卡.....	18
10.3 热添加内存.....	18
第 11 章 KVM 虚拟机迁移.....	19
11.1 冷迁移.....	19
11.2 热迁移.....	19
11.3 KVM 虚拟机迁移 ESXi.....	19
11.4 ESXi 转换为 KVM.....	19

## 第 1 章 虚拟化介绍

### 1.1 什么是虚拟化

通过模拟物理服务器的硬件，实现在一台物理机上可以同时运行多个操作系统，也称为虚拟机

虚拟机共享物理服务器的 CPU，内存，IO 等硬件资源

虚拟机之间逻辑上是互相隔离的

物理机我们一般称为宿主机

宿主机上的虚拟机成为客户机

### 1.2 虚拟化分类

虚拟化主要分为 2 类：

全虚拟化代表：ESXi Xen 都是商业收费

半虚拟化代表：KVM

参考博客：

<https://blog.csdn.net/xiaoyuerp/article/details/84870305>

## 第 2 章 KVM 介绍

kvm 全称是 Kernel-Based Virtual Machine。

也就是说 KVM 是基于 Linux 内核实现的。

KVM 有一个内核模块叫 `kvm.ko`，只用于管理虚拟 CPU 和内存。

IO 的虚拟化比如存储和网络设备则是由 Linux 内核与 Qemu 来实现。

KVM 本身只关注虚拟机调度和内存管理这两个方面

IO 外设的任务交给 Linux 内核和 Qemu

Libvirt 就是 KVM 的管理工具。

其实，Libvirt 除了能管理 KVM 这种 Hypervisor，还能管理 Xen，VirtualBox 等。

Libvirt 包含的内容：

`libvirtd` 是服务程序，接收和处理 API 请求；

`virsh`：是我们经常要用的 KVM 命令行工具

## 第 3 章 安装 KVM

### 3.1 安装软件

```
yum install libvirt virt-install qemu-kvm -y
```

## 第 4 章 创建并安装一台 KVM 虚拟机

### 4.1 安装一台 kvm 虚拟机

```
systemctl start libvirtd.service  
systemctl status libvirtd.service  
virt-install --virt-type kvm --os-type=linux --os-variant rhel7 --name centos7 --memory 1024 --vcpus  
1 --disk /opt/centos7.raw,format=raw,size=10 --cdrom /opt/CentOS-7-x86_64-DVD-1810.iso --network  
network=default --graphics vnc,listen=0.0.0.0 --noautoconsole
```

### 4.2 指令说明

--name 指定虚拟机的名称

--memory 指定分配给虚拟机的内存资源大小

maxmemory 指定可调节的最大内存资源大小，因为 KVM 支持热调整虚拟机的资源

--vcpus 指定分配给虚拟机的 CPU 核心数量

maxvcpus 指定可调节的最大 CPU 核心数量

--os-type 指定虚拟机安装的操作系统类型

--os-variant 指定系统的发行版本

--location 指定 ISO 镜像文件所在的路径，支持使用网络资源路径，也就是说可以使用 URL

--disk path 指定虚拟硬盘所存放的路径及名称，size 则是指定该硬盘的可用大小，单位是 G

--bridge 指定使用哪一个桥接网卡，也就是说使用桥接的网络模式

--graphics 指定是否开启图形

--console 定义终端的属性，target\_type 则是定义终端的类型

--extra-args 定义终端额外的参数

## 第 5 章 KVM 虚拟机常用管理命令

### 5.1 查看虚拟机

查看正在运行的虚拟机

```
virsh list
```

查看所有的虚拟机

```
virsh list --all
```

### 5.2 开启虚拟机

```
virsh start centos7
```

### 5.3 重启虚拟机

```
virsh reboot centos7
```

### 5.4 关闭虚拟机

```
virsh shutdown centos7
```

```
virsh destroy centos7
```

### 5.5 查看配置文件

```
virsh dumpxml centos7
```

### 5.6 导出配置文件

只要有磁盘文件和配置文件就可以迁移到其他机器

```
virsh dumpxml centos7 >centos7.xml
```

### 5.7 删除虚拟机

```
virsh destroy centos7      #相当于直接断电
```

```
virsh undefine centos7     #只删除配置文件的
```

### 5.8 导入虚拟机

需要把导出的配置文件和磁盘文件一起放在相同目录下

```
virsh define centos7.xml
```

并且会在/var/lib 目录下生成配置文件

```
[root@kvm /opt]# ls /etc/libvirt/qemu
```

```
centos7.xml networks
```

## 5.9 生产故障

问题背景:

关一台少一台

问题原因:

开机情况下 undefine 掉虚拟机，虽然虚拟机还会运行，但是目录下的配置文件没了

此时再关机就会导致虚拟机直接没了

undefine 其实删的是配置文件

只要进程还在，虚拟机就会运行

操作步骤:

```
[root@kvm /opt]# virsh list --all
```

Id	名称	状态
----	----	----

3	centos7	running
---	---------	---------

```
[root@kvm /opt]# ls /etc/libvirt/qemu
```

```
centos7.xml networks
```

```
[root@kvm /opt]# virsh undefine centos7
```

域 centos7 已经被取消定义

```
[root@kvm /opt]# virsh list --all
```

Id	名称	状态
----	----	----

3	centos7	running
---	---------	---------

```
[root@kvm /opt]# ls /etc/libvirt/qemu
```

```
networks
```

```
[root@kvm /opt]# virsh shutdown centos7
```

域 centos7 被关闭

```
[root@kvm /opt]# virsh list --all
```

Id	名称	状态
----	----	----

```
[root@kvm /opt]#
```

## 5.10 修改配置文件

项目背景:

磁盘空间不足, 需要迁移磁盘文件, 但是配置文件里的路径需要修改

建议使用 `edit` 命令编辑, 可以检查语法

```
virsh edit centos7
```

操作步骤:

1. 虚拟机关闭状态下编辑

```
[root@kvm /opt]# virsh list --all
```

Id	名称	状态
—	centos7	关闭

2. 编辑配置文件

```
[root@kvm /opt]# virsh edit centos7
```

```
<source file='/data/centos7.raw' />
```

3. 移动磁盘文件到/data 目录

```
[root@kvm /opt]# mkdir /data
```

```
[root@kvm /opt]# mv /opt/centos7.raw /data/
```

4. 启动虚拟机

```
[root@kvm /data]# virsh start centos7
```

域 centos7 已开始

```
[root@kvm /data]# virsh list
```

Id	名称	状态
7	centos7	running

## 5.11 主机重命名

关机状态下重命名

```
[root@kvm ~]# virsh domrename centos7 web-blog
```

Domain successfully renamed

```
[root@kvm ~]# virsh list --all
```



Id	名称	状态
1	web-blog	关闭

## 5.12 主机挂起

```
[root@kvm ~]# virsh suspend web-blog
```

域 web-blog 被挂起

```
[root@kvm ~]# virsh list --all
```

Id	名称	状态
1	web-blog	暂停

## 5.13 主机恢复

```
[root@kvm ~]# virsh resume web-blog
```

域 web-blog 被重新恢复

```
[root@kvm ~]# virsh list --all
```

Id	名称	状态
1	web-blog	running

## 5.14 kvm 虚拟机开机启动

```
[root@kvm ~]# virsh autostart web-blog
```

域 web-blog 标记为自动开始

测试能否开机自启

```
[root@kvm ~]# virsh list --all
```

Id	名称	状态
1	web-blog	running
2	web-www	running

```
[root@kvm ~]# virsh shutdown web-blog
```

域 web-blog 被关闭

```
[root@kvm ~]# virsh shutdown web-www
```

域 web-www 被关闭

```
[root@kvm ~]# virsh list --all
```

Id	名称	状态
-	web-blog	关闭
-	web-www	关闭

重启 libvirtd 服务

```
[root@kvm ~]# systemctl restart libvirtd.service
```

再次查看虚拟机

```
[root@kvm ~]# virsh list --all
```

Id	名称	状态
1	web-blog	running
-	web-www	关闭

## 5.15 查看哪些虚拟机是开机自启动

本质是软链接，只要存在这个软链接，就会开机自启动

```
[root@kvm ~]# ll /etc/libvirt/qemu/autostart/
```

总用量 0

```
lrwxrwxrwx 1 root root 30 12月 11 11:47 web-blog.xml -> /etc/libvirt/qemu/web-blog.xml
```

## 第 6 章 连接 KVM 虚拟机

### 6.1 查看 VNC 端口号

查看当前运行的主机：

```
[root@kvm ~]# virsh list --all
```

Id	名称	状态
1	web-blog	running
2	web-www	running

```
[root@kvm ~]# virsh vncdisplay web01
```

:0

```
[root@kvm ~]# virsh vncdisplay web-www
```

:1

## 6.2 console 管理口

默认进入 console 是没有回显的, 需要系统开启相关的参数

```
[root@kvm ~]# virsh console web-blog
```

连接到域 web-blog

换码符为 ^]

ssh 连接进入 kvm

```
[root@kvm ~]# ssh root@192.168.122.62
```

root@192.168.122.62's password:

Last login: Wed Dec 11 12:02:30 2019 from gateway

```
[root@localhost ~]#
```

添加参数并重启 115200 为波特率

```
[root@localhost ~]# grubby --update-kernel=ALL --args="console=ttyS0,115200n8"
```

```
[root@localhost ~]# grep "115200" /boot/grub2/grub.cfg
```

```
linux16 /vmlinuz-3.10.0-957.el7.x86_64 root=/dev/mapper/centos-root ro rd.lvm.lv=centos/root
rd.lvm.lv=centos/swap rhgb quiet LANG=zh_CN.UTF-8 console=ttyS0,115200n8
```

```
linux16 /vmlinuz-0-rescue-8fda2ef411f94d568c7b2eff38b677c1 root=/dev/mapper/centos-root ro
rd.lvm.lv=centos/root rd.lvm.lv=centos/swap rhgb quiet console=ttyS0,115200n8
```

```
[root@localhost ~]# reboot
```

再次使用 console 登陆, 然后多按几次回车, 发现可以登录了

```
[root@kvm ~]# virsh console web-blog
```

连接到域 web-blog

换码符为 ^]

CentOS Linux 7 (Core)

Kernel 3.10.0-957.el7.x86\_64 on an x86\_64

localhost login:

## 第 7 章 KVM 磁盘管理

### 7.1 虚拟机磁盘格式介绍

raw: 不支持做快照, 性能好

qcow2: 支持快照, 性能不如 raw 好

## 7.2 查看磁盘信息

```
[root@kvm /data]# qemu-img info centos7.raw
image: centos7.raw
file format: raw
virtual size: 10G (10737418240 bytes)
disk size: 1.4G
```

## 7.3 创建磁盘

```
[root@kvm /data]# qemu-img create -f qcow2 /data/oldboy.qcow2 1G
Formatting '/data/oldboy.qcow2', fmt=qcow2 size=1073741824 encryption=off cluster_size=65536
lazy_refcounts=off
[root@kvm /data]# ls -lh oldboy.qcow2
-rw-r--r-- 1 root root 193K 12月 11 17:03 oldboy.qcow2
[root@kvm /data]# qemu-img info oldboy.qcow2
image: oldboy.qcow2
file format: qcow2
virtual size: 1.0G (1073741824 bytes)
disk size: 196K
cluster_size: 65536
Format specific information:
    compat: 1.1
    lazy refcounts: false
```

## 7.4 调整磁盘容量: 只能加不能减

```
[root@kvm /data]# qemu-img resize /data/oldboy.qcow2 1T
Image resized.
[root@kvm /data]# ls -lh oldboy.qcow2
-rw-r--r-- 1 root root 272K 12月 11 17:07 oldboy.qcow2
[root@kvm /data]# qemu-img info oldboy.qcow2
image: oldboy.qcow2
file format: qcow2
virtual size: 1.0T (1099511627776 bytes)
disk size: 260K
cluster_size: 65536
```

```
Format specific information:
```

```
compat: 1.1
```

```
lazy refcounts: false
```

## 7.5 磁盘格式转换

操作步骤:

- 1.将虚拟机关机
- 2.转换磁盘格式
- 3.编辑配置文件修改为 qcow2 格式

```
[root@kvm /data]# virsh shutdown web-blog
[root@kvm /data]# qemu-img convert -f raw -O qcow2 centos7.raw centos7.qcow2
[root@kvm /data]# virsh edit web-blog
[root@kvm /data]# virsh dumpxml web-blog|grep qcow2
    <driver name='qemu' type='qcow2' />
    <source file='/data/centos7.qcow2' />
[root@kvm /data]# virsh start web-blog
域 web-blog 已开始
[root@kvm /data]# virsh list
Id      名称                      状态
-----
2       web-blog                  running
[root@kvm /data]# virsh vncdisplay web-blog
:0
```

然后进入虚拟机 dd 生成文件

```
[root@kvm /data]# ll -h centos7.qcow2
-rw-r--r-- 1 qemu qemu 1.4G 12月 11 17:29 centos7.qcow2
[root@kvm /data]# virsh console web-blog
[root@localhost ~]#
[root@localhost ~]# dd if=/dev/zero of=/opt/test.raw bs=500M count=1
记录了 1+0 的读入
记录了 1+0 的写出
524288000 字节(524 MB)已复制, 4.16398 秒, 126 MB/秒
[root@localhost ~]#
```

退出虚拟机之后再次查看磁盘发现增加了 500M

```
[root@kvm /data]# ll -h centos7.qcow2
-rw-r--r-- 1 qemu qemu 1.9G 12月 11 17:32 centos7.qcow2
```

## 第 8 章 KVM 快照管理

### 8.1 创建快照

raw 格式不支持做快照，qcow2 格式快照就保存在本地  
后面的数字是 unix 时间戳

```
[root@kvm /data]# virsh snapshot-create-as --name init web-blog
已生成域快照 init
```

### 8.2 查看快照

```
[root@kvm /data]# virsh snapshot-list web-blog
```

名称	生成时间	状态
-----		
init	2019-12-11 17:45:23 +0800	running

### 8.3 还原快照

```
[root@kvm /data]# virsh snapshot-revert web-blog --snapshotname init
```

### 8.4 删除快照

```
[root@kvm /data]# virsh snapshot-delete web-blog --snapshotname init
```

## 第 9 章 KVM 虚拟机克隆

### 9.1 完整克隆

```
[root@kvm /data]# virt-clone --auto-clone -o web-blog -n web-blog-backup
ERROR  必须暂停或者关闭有要克隆设备的域。
[root@kvm /data]# virsh shutdown web-blog
域 web-blog 被关闭
[root@kvm /data]# virt-clone --auto-clone -o web-blog -n web-blog-backup
WARNING 当卷已经被全部分配后，需要的卷容量将超过可用池空间。(10240 M 需要的容量 > 8526 M 可用容量)
正在分配 'centos7-clone.qcow2'
```

```
| 10 GB 00:00:03
```

成功克隆 'web-blog-backup'。

```
[root@kvm /data]# virsh list --all
```

Id	名称	状态
-----		
-	web-blog	关闭
-	web-blog-backup	关闭
-	web-www	关闭

```
[root@kvm /data]# virsh dumpxml web-blog-backup |grep "qcow2"
```

```
<driver name='qemu' type='qcow2' />
<source file='/data/centos7-clone.qcow2' />
```

查看空间发现比原来的小

```
[root@kvm /data]# ls -lh /data/centos7*
```

```
-rw----- 1 root root 1.4G 12月 11 17:52 /data/centos7-clone.qcow2
-rw-r--r-- 1 root root 2.1G 12月 11 17:52 /data/centos7.qcow2
-rw----- 1 root root 10G 12月 11 17:22 /data/centos7.raw
```

查看快照发现并没有把原来机器的快照一起复制过来

```
[root@kvm /data]# virsh snapshot-list web-blog-backup
```

名称	生成时间	状态
-----		

```
[root@kvm /data]# virsh snapshot-list web-blog
```

名称	生成时间	状态
-----		

init	2019-12-11 17:45:23 +0800	running
------	---------------------------	---------

## 9.2 链接克隆

### 1.生成虚拟机磁盘文件

```
[root@kvm /data]# qemu-img create -f qcow2 -b centos7-clone.qcow2 centos7-clone-link.qcow2
```

```
Formatting 'centos7-clone-link.qcow2', fmt=qcow2 size=10737418240 backing_file='centos7-clone.qcow2'
encryption=off cluster_size=65536 lazy_refcounts=off
```

```
[root@kvm /data]# qemu-img create -f qcow2 -b centos7.raw centos7-clone-link-raw.qcow2
```

```
Formatting 'centos7-clone-link-raw.qcow2', fmt=qcow2 size=10737418240 backing_file='centos7.raw'
encryption=off cluster_size=65536 lazy_refcounts=off
```

```
[root@kvm /data]# ll -h centos7-clone*
```

```
-rw-r--r-- 1 root root 193K 12月 11 18:09 centos7-clone-link.qcow2
-rw-r--r-- 1 root root 193K 12月 11 18:09 centos7-clone-link-raw.qcow2
-rw----- 1 root root 1.4G 12月 11 17:52 centos7-clone.qcow2

[root@kvm /data]# qemu-img info centos7-clone-link.qcow2
image: centos7-clone-link.qcow2
file format: qcow2
virtual size: 10G (10737418240 bytes)
disk size: 196K
cluster_size: 65536
backing file: centos7-clone.qcow2
Format specific information:
    compat: 1.1
    lazy refcounts: false
```

## 2.生成虚拟机配置文件

重命名虚拟机名称

删除 uid

删除 mac 地址

修改磁盘地址

```
[root@kvm /data]# vim web-clone.xml
```

## 3.导入配置文件

```
[root@kvm /data]# virsh define web-clone.xml
```

定义域 web-blog-clone (从 web-clone.xml)

## 4.启动虚拟机

```
[root@kvm /data]# virsh list --all
```

Id	名称	状态
–	web-blog	关闭
–	web-blog-backup	关闭
–	web-blog-clone	关闭
–	web-www	关闭

```
[root@kvm /data]# virsh start web-blog-clone
```

域 web-blog-clone 已开始

## 5.克隆虚拟机脚本



## 第 10 章 KVM 虚拟机网络

### 10.1 桥接网络

默认虚拟机网络是 NAT 模式，网段 192.168.122.0/24

默认 NAT 模式

```
virt-install --virt-type kvm --os-type=linux --os-variant rhel7 --name web04 --memory 1024 --vcpus 1 --disk /opt/web04.qcow2 --boot hd --network network=default --graphics vnc,listen=0.0.0.0 --noautoconsole
```

桥接模式

为了方便测试，可以在 VM 虚拟机上打开 DHCP 功能

#### 1.创建桥接网卡

```
[root@kvm /data]# virsh iface-bridge eth0 br0
```

已启动桥接接口 br0

#### 2.链接克隆一台新磁盘

```
[root@kvm /data]# qemu-img create -f qcow2 -b centos7.qcow2 bridge.qcow2
Formatting 'bridge.qcow2', fmt=qcow2 size=10737418240 backing_file='centos7.qcow2' encryption=off cluster_size=65536 lazy_refcounts=off
```

#### 3.创建新虚拟机

```
[root@kvm /data]# virt-install --virt-type kvm --os-type=linux --os-variant rhel7 --name web04 --memory 1024 --vcpus 1 --disk /data/bridge.qcow2 --boot hd --network bridge=br0 --graphics vnc,listen=0.0.0.0 --noautoconsole
```

开始安装.....

域创建完成。

#### 4.查看虚拟机网卡信息

```
[root@kvm /data]# virsh list
```

Id	名称	状态
3	web-blog-clone	running
4	web04	running

```
[root@kvm /data]# virsh console web04
```

```
[root@localhost ~]#
```

```
[root@localhost ~]# ip a
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    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 group default qlen 1000
    link/ether 52:54:00:0c:3f:f1 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.128/24 brd 10.0.0.255 scope global noprefixroute dynamic eth0
        valid_lft 1751sec preferred_lft 1751sec
    inet6 fe80::e36f:3266:69cf:b064/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
[root@localhost ~]#
```

查看网卡信息

```
[root@kvm ~]# brctl show
```

bridge name	bridge id	STP enabled	interfaces
br0	8000.000c294c35da	yes	eth0 vnet1
virbr0	8000.52540025ee29	yes	virbr0-nic vnet0

## 第 11 章 KVM 虚拟机热添加技术

### 11.1 热添加硬盘

热添加硬盘，网卡，内存，CPU

kvm 虚拟机在线热添加硬盘

创建一块硬盘

```
[root@kvm /data]# qemu-img create -f qcow2 web01-add.qcow2 10G

Formatting 'web01-add.qcow2', fmt=qcow2 size=10737418240 encryption=off cluster_size=65536
lazy_refcounts=off
```

临时生效

```
[root@kvm /data]# virsh attach-disk web01 /data/web01-add.qcow2 vdb --subdriver qcow2
```

成功附加磁盘

永久生效

老男孩教育官网 <http://www.oldboyedu.com>

```
[root@kvm /data]# virsh attach-disk web01 /data/web01-add.qcow2 vdb --subdriver qcow2
```

成功附加磁盘

```
[root@kvm /data]# virsh attach-disk web01 /data/web01-add.qcow2 vdb --subdriver qcow2 --config
```

成功附加磁盘

临时剥离硬盘

```
[root@kvm /data]# virsh detach-disk web01 vdb
```

永久剥离硬盘

```
[root@kvm /data]# virsh detach-disk web01 vdb --config
```

调整磁盘大小实验

```
[root@kvm /data]# qemu-img create -f qcow2 web01-add.qcow2 10G
```

```
Formatting 'web01-add.qcow2', fmt=qcow2 size=10737418240 encryption=off cluster_size=65536
lazy_refcounts=off
```

```
[root@kvm /data]# virsh attach-disk web01 /data/web01-add.qcow2 vdb --subdriver qcow2
```

成功附加磁盘

```
[root@kvm /data]# virsh attach-disk web01 /data/web01-add.qcow2 vdb --subdriver qcow2 --config
```

成功附加磁盘

```
[root@kvm /data]# virsh console web01
```

```
[root@localhost ~]# mkfs.xfs /dev/vdb
```

```
meta-data=/dev/vdb          isize=512    agcount=4, agsize=655360 blks
               =             sectsz=512    attr=2, projid32bit=1
               =             crc=1        finobt=0, sparse=0
data        =             bsize=4096    blocks=2621440, imaxpct=25
               =             sunit=0      swidth=0 blks
naming      =version 2      bsize=4096    ascii-ci=0 ftype=1
log         =internal log   bsize=4096    blocks=2560, version=2
               =             sectsz=512    sunit=0 blks, lazy-count=1
realtime    =none          extsz=4096    blocks=0, rtextents=0
```

```
[root@localhost ~]# mount /dev/vdb /mnt/
```

```
[root@localhost ~]# df -h|tail -1
```

```
/dev/vdb          10G   33M   10G    1% /mnt
```

```
[root@localhost ~]# touch /mnt/web01.txt
```

```
[root@localhost ~]# ll /mnt/
```

总用量 0

```
-rw-r--r--. 1 root root 0 12月 12 12:16 web01.txt
```

```
-----  
# ^+] 退出  
-----
```

```
[root@kvm /data]# virsh detach-disk web01 vdb
```

成功分离磁盘

```
[root@kvm /data]# qemu-img resize /data/web01-add.qcow2 +10G
```

Image resized.

```
[root@kvm /data]# qemu-img info /data/web01-add.qcow2
```

image: /data/web01-add.qcow2

file format: qcow2

virtual size: 20G (21474836480 bytes)

disk size: 260K

cluster\_size: 65536

Format specific information:

compat: 1.1

lazy refcounts: false

```
[root@kvm /data]# virsh attach-disk web01 /data/web01-add.qcow2 vdb --subdriver qcow2
```

成功附加磁盘

```
[root@kvm /data]# virsh console web01
```

```
[root@localhost ~]# fdisk -l /dev/vdb
```

磁盘 /dev/vdb: 21.5 GB, 21474836480 字节, 41943040 个扇区

Units = 扇区 of 1 \* 512 = 512 bytes

扇区大小(逻辑/物理): 512 字节 / 512 字节

I/O 大小(最小/最佳): 512 字节 / 512 字节

```
[root@localhost ~]# mount /dev/vdb /mnt/
```

```
[root@localhost ~]# df -h|tail -1
```

```
/dev/vdb          10G   33M   10G    1% /mnt
```

```
[root@localhost ~]# ll /mnt/
```

总用量 0

```
-rw-r--r--. 1 root root 0 12月 12 12:16 web01.txt
```

这个时候我们发现问题，就是虽然 fdisk 看到了 20G，但是挂载后还是 10G 的空间  
通过执行命令可以解决：

```
[root@localhost ~]# xfs_growfs /dev/vdb
```

```
meta-data=/dev/vdb          isize=512    agcount=4, agsize=655360 blks
```

```
          =                  sectsz=512   attr=2, projid32bit=1
```

老男孩教育官网 <http://www.oldboyedu.com>

```

=                                crc=1          finobt=0 spinodes=0
data    =                        bsize=4096      blocks=2621440, imaxpct=25
=                                sunit=0         swidth=0 blks
naming  =version 2              bsize=4096      ascii-ci=0 ftype=1
log     =internal              bsize=4096      blocks=2560, version=2
=                                sectsz=512     sunit=0 blks, lazy-count=1
realtime =none                 extsz=4096      blocks=0, rtextents=0
data blocks changed from 2621440 to 5242880

```

此时再次查看，发现容量已经增加了

```

[root@localhost ~]# df -h|tail -1
/dev/vdb          20G   33M   20G    1% /mnt

```

如果是 ext4 的分区命令需要使用 resize2fs

```
resize2fs /dev/vdb
```

## 11.2 热添加网卡

```
virsh attach-interface web04 --type bridge --source br0 --model virtio detachinterface web04 --type bridge --mac 52:54:00:35:d3:71
```

## 11.3 热添加内存

```

virt-install --virt-type kvm --os-type=linux --os-variant rhel7 --name web04 --
memory 512,maxmemory=2048 --vcpus 1 --disk /data/web04.qcow2 --boot hd --network bridge=br0 --
graphics vnc,listen=0.0.0.0 --noautoconsole

```

临时热添加

```
setmem web04 1024M --live
```

永久增大内存

```
setmem web04 1024M --config
```

## 11.4 热添加 CPU

```

virt-install --virt-type kvm --os-type=linux --os-variant rhel7 --name web04 --
memory 512,maxmemory=2048 --vcpus 1,maxvcpus=10 --disk /data/web04.qcow2 --boot hd --network
bridge=br0 --graphics vnc,listen=0.0.0.0 --noautoconsole

```

热添加 cpu 核数

```
setvcpus web04 4 --live
```

永久添加 cpu

老男孩教育官网 <http://www.oldboyedu.com>

```
核数 setvcpus web04 4 --config
```

## 第 12 章 KVM 虚拟机迁移

### 12.1 冷迁移

环境：新克隆一个虚拟机，然后删除/data 和/opt 下所有的文件，然后批量删除所有的虚拟机

```
virsh undefine $(virsh list --all|awk 'NR>1{print $2}')
```

拷贝磁盘文件和配置文件到第二台服务器：

```
[root@kvm-100 /data]# scp web01.qcow2 web01.xml 10.0.0.200:/data/
```

第二台机器编辑配置文件，然后删除 uuid,mac 地址，最后导入配置

```
[root@kvm-200 /opt]# cd /data/
[root@kvm-200 /data]# ls
web01.qcow2  web01.xml
[root@kvm-200 /data]# vim web01.xml
[root@kvm-200 /data]# virsh define web01.xml
定义域 web01 (从 web01.xml)
```

### 12.2 热迁移

三台服务器：

nfs,kvm-100,kvm-200

NFS 服务端安装配置

注意！要做 host 解析

```
virsh migrate --live --verbose web01 qemu+ssh://10.0.0.12/system --unsafe
```

### 12.3 KVM 虚拟机迁移 ESXi

1.安装 ESXi 虚拟机

2.KVM 主机执行转换磁盘格式

```
[root@kvm-100 /data]# qemu-img convert -f qcow2 -O vmdk web01.qcow2 web01.vmdk
```

3.ESXi 主机执行转换命令

```
mkfstools -i web01.vmdk -d thin web01-exsi.vmdk
```

### 12.4 ESXi 转换为 KVM

1.ESXi 导出虚拟机为 OVA 格式

```
yum install virt-v2v -y
virt-v2v -i ova centos7.ova -o local -os /data -of qcow2
```

2.直接拷贝 vmdk 磁盘文件然后转换

```
qemu-img convert -f vmdk -O qcow2 web01-exsi web01.qcow2
```

## 第 13 章 KVM 图形化管理工具

### 13.1 项目地址

<https://github.com/retspen/webvirtmgr/wiki/Install-WebVirtMgr>

### 13.2 安装依赖命令

```
yum -y install git python-pip libvirt-python libxml2-python python-websocketify supervisor gcc
python-devel
pip install numpy -i https://pypi.tuna.tsinghua.edu.cn/simple
```

### 13.3 安装 python 的 Django 环境

```
git clone git://github.com/retspen/webvirtmgr.git
cd webvirtmgr
pip install -r requirements.txt -i https://pypi.tuna.tsinghua.edu.cn/simple
./manage.py syncdb
./manage.py collectstatic
```

### 13.4 安装配置 Nginx

```
cat > /etc/yum.repos.d/nginx.repo<<EOF
[nginx-stable]
name=nginx stable repo
baseurl=http://nginx.org/packages/centos/\$releasever/\$basearch/
gpgcheck=1
enabled=1
gpgkey=https://nginx.org/keys/nginx_signing.key
module_hotfixes=true

[nginx-mainline]
```

```
name=nginx mainline repo
baseurl=http://nginx.org/packages/mainline/centos/\$releasever/\$basearch/
gpgcheck=1
enabled=0
gpgkey=https://nginx.org/keys/nginx_signing.key
module_hotfixes=true
EOF
yum install nginx -y
mkdir /code
mv /opt/webvirtmgr /code/
chown -R nginx:nginx /code
rm -rf /etc/nginx/conf.d/default.conf
cat >/etc/nginx/conf.d/webvirtmgr.conf<<EOF
server {
    listen 80 default_server;

    server_name localhost;

    access_log /var/log/nginx/webvirtmgr_access_log;

    location /static/ {
        root /code/webvirtmgr;
        expires max;
    }
    location / {
        proxy_pass http://127.0.0.1:8000;
        proxy_set_header X-Real-IP \$remote_addr;
        proxy_set_header X-Forwarded-for \$proxy_add_x_forwarded_for;
        proxy_set_header Host \$host:\$server_port;
        proxy_set_header X-Forwarded-Proto \$scheme;
        proxy_connect_timeout 600;
        proxy_read_timeout 600;
        proxy_send_timeout 600;
        client_max_body_size 1024M;
    }
}
```



```
EOF
nginx -t
systemctl start nginx
netstat -lntup|grep 80
```

### 13.5 配置 Supervisor

```
cat >/etc/supervisord.d/webvirtmgr.ini<<EOF
[program:webvirtmgr]
command=/usr/bin/python /code/webvirtmgr/manage.py run_gunicorn -c
/code/webvirtmgr/conf/gunicorn.conf.py
directory=/code/webvirtmgr
autostart=true
autorestart=true
logfile=/var/log/supervisor/webvirtmgr.log
log_stderr=true
user=nginx

[program:webvirtmgr-console]
command=/usr/bin/python /code/webvirtmgr/console/webvirtmgr-console
directory=/code/webvirtmgr
autostart=true
autorestart=true
stdout_logfile=/var/log/supervisor/webvirtmgr-console.log
redirect_stderr=true
user=nginx
EOF
systemctl start supervisord.service
supervisorctl status
```

### 13.6 创建用户

```
mkdir /var/lib/nginx/.ssh/ -p
chown -R nginx:nginx /var/lib/nginx/
su - nginx -s /bin/bash
ssh-keygen
touch ~/.ssh/config && echo -e "StrictHostKeyChecking=no\nUserKnownHostsFile=/dev/null" >>
~/.ssh/config
```

```
chmod 0600 ~/.ssh/config
exit
adduser webvirtmgr
echo "123456"|passwd --stdin webvirtmgr
su - nginx -s /bin/bash
ssh-copy-id root@10.0.0.100
```

## 13.7 web 页面操作



### 登录

root
.....
登录

### Add Connection



TCP连接

SSH连接

TLS Connection

Local Socket

You must create ssh [authorization key](#). If you have another SSH port on your server, you can add IP:PORT like '192.168.1.1:2222'.

Label

10.0.0.100

FQDN / IP

10.0.0.100

用户名

root|

关闭

添加

10.0.0.100

New Instance

虚拟机实例

	名称	状态	VCPUs	内存	执行
存储池	web01	关闭	1	1024MB	<div><div></div><div></div><div></div><div></div><div></div><div></div></div>
网络池					
Interfaces					
Secrets					
概览					

13.8 报错解决

```
python -m pip install --upgrade --force pip
pip install setuptools==33.1.1 -i https://pypi.tuna.tsinghua.edu.cn/simple
```