[Top](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/SERVICES/DAY02/CASE/01/index.html" \l "page_top_case)

# NSD SERVICES DAY02

1. [案例1：安装一个KVM服务器](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/SERVICES/DAY02/CASE/01/index.html" \l "case1)
2. [案例2：KVM平台构建及简单管理](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/SERVICES/DAY02/CASE/01/index.html" \l "case2)
3. [案例3：virsh基本管理操作](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/SERVICES/DAY02/CASE/01/index.html" \l "case3)
4. [案例4：xml配置文件的应用](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/SERVICES/DAY02/CASE/01/index.html" \l "case4)
5. [案例5：快建新虚拟机](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/SERVICES/DAY02/CASE/01/index.html" \l "case5)

## 1 案例1：安装一个KVM服务器

### 1.1 问题

本例要求准备一台 CentOS7 服务器，将其搭建为KVM平台，主要完成下列操作：

1. 关闭本机的SELinux保护、防火墙服务
2. 挂载光盘到 /mnt/dvd，将其配置为本机YUM源（baseurl = file:///mnt/dvd）
3. 安装KVM相关包组，确保已启用 libvirtd 服务

### 1.2 方案

虚拟化服务软件：

* qemu-kvm为 kvm 提供底层仿真支持
* libvirt-daemon为libvirtd 守护进程，管理虚拟机
* libvirt-client为用户端软件，提供客户端管理命令
* libvirt-daemon-driver-qemu为libvirtd 连接 qemu 的驱动
* virt-manager为图形管理工具

### 1.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：配置本地yum仓库

在光驱中插入CentOS7的系统光盘（若是虚拟机则连接相应ISO文件），然后在系统中将光盘挂载到/mnt目录。

1. [root@kvmsvr ~]# mkdir /mnt/dvd                     //创建挂载点
2. [root@kvmsvr ~]# vim /etc/fstab                     //添加开机挂载配置
3. .. ..
4. /dev/cdrom    /mnt/dvd     iso9660        ro        0 0
5. [root@kvmsvr ~]# mount -a                            //根据fstab配置挂载光盘
6. mount: /dev/sr0 is write-protected, mounting read-only

2）确认/mnt/dvd访问点

1. [root@kvmsvr ~]# ls /mnt/dvd/Packages/\*.rpm         //确认软件包位置
2. .. ..
3. /mnt/Packages/zsh-5.0.2-14.el7.x86\_64.rpm
4. /mnt/Packages/zziplib-0.13.62-5.el7.i686.rpm
5. /mnt/Packages/zziplib-0.13.62-5.el7.x86\_64.rpm

3）将本地目录/mnt/dvd配置为本机的yum源

1. [root@kvmsvr ~]# vim /etc/yum.repos.d/mnt\_dvd.repo
2. [mnt\_dvd]
3. name=added from: file:///mnt/dvd
4. baseurl=file:///mnt/dvd
5. enabled=1
6. gpgcheck=0                                                     //禁止GPG检查
7. [root@kvmsvr ~]# yum repolist                                 //列出可用仓库
8. ……

步骤二：安装KVM虚拟化

1）安装主要KVM软件

1. [root@kvmsvr ~]# yum -y install qemu-kvm libvirt-daemon libvirt-client libvirt-daemon-driver-qemu virt-manager
2. .. ..

2）确保libvirtd服务可用

1. [root@kvmsvr ~]# systemctl restart libvirtd
2. [root@kvmsvr ~]# systemctl enable libvirtd

3)确保虚拟系统管理器（virt-manager）可用

在KVM服务器的桌面环境中，可以通过“应用程序”菜单组找到“系统工具”-->“虚拟系统管理器”快捷方式（如图-1所示）。

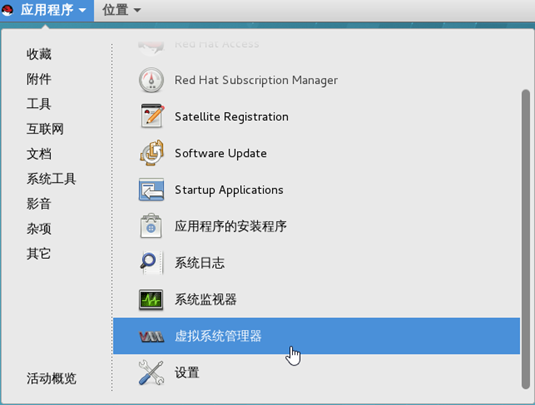


图-1

单击可以成功开启KVM管理工具（如图-2所示），可以看到还没有任何虚拟机。

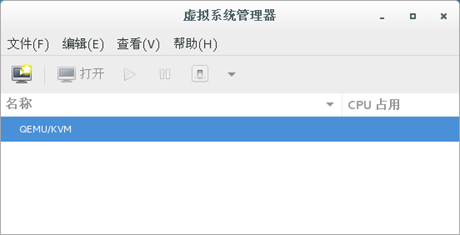


图-2

## 2 案例2：KVM平台构建及简单管理

### 2.1 问题

本例要求在真实KVM服务器上完成以下任务：

1. 新建一个名为 CentOS7 的虚拟机，并为其安装好操作系统（注意禁用SELinux机制、禁用防火墙）
2. 将虚拟机 CentOS7 克隆为 CentOS7-c1
3. 开启虚拟机 CentOS7-c1 ，以 root 用户登入到系统
4. 彻底删除虚拟机 CentOS7-c1

### 2.2 方案

使用KVM提供的virt-manager图形化管理程序来操作。

### 2.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：新建名为CentOS7的虚拟机

1）在“虚拟系统管理器”中单击左上方“创建新虚拟机”按钮（如图-3所示）。

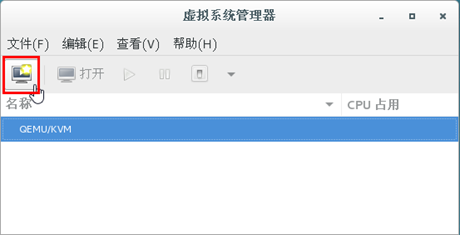


图-3

2）弹出“新建虚拟机”向导，选择“本地安装介质”（如图-4所示），单击“前进”。

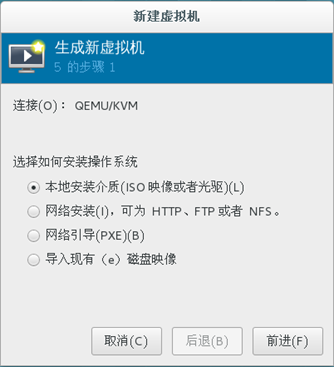


图-4

3）接下来“定位安装介质”，请正确指定CentOS7系统的ISO光盘镜像文件位置（如图-5所示），确认自动识别到操作系统类型，单击“前进”。



图-5

4）选择内存和CPU设置，建议为虚拟机分配内存不小于1024MB（如图-6所示），单击“前进”。



图-6

5）为虚拟机启用存储，例如分配一个40GiB的磁盘（如图-7所示），单击“前进”。



图-7

6）为虚拟机命名，设为CentOS7（如图-8所示），单击“完成”。



图-8

步骤二：为虚拟机CentOS7安装操作系统

1）确认新虚拟机从光盘引导

上述设置全部完成后，将会自动开启新虚拟机，并进入安装过程（如图-9所示）。

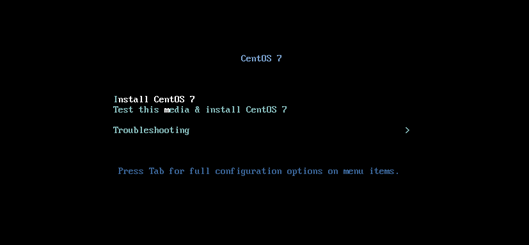


图-9

2）完成后续手动安装过程

具体过程与普通安装相同，注意禁用防火墙、禁用SELinux机制。

3）确认安装结果

新装的虚拟机CentOS7可以正常启动、登录。

在“虚拟系统管理器”中也能够看到此虚拟机。

步骤三：克隆虚拟机并验证

1）将被克隆的虚拟机CentOS7关机

右击选中的虚拟机CentOS7，选择“关机”-->“强制关机”（如图-11所示），根据提示确认即可。



图-11

2）执行克隆操作

右击已关闭的虚拟机CentOS7，选择“克隆”，在弹出对话框中指定名称，确认默认设置（如图-12所示），然后单击右下角的“克隆”按钮完成操作。



图-12

3）确认克隆结果

新克隆的虚拟机CentOS7-c1可以正常启动、登录，可以独立运行。

在“虚拟系统管理器”中也能够看到此虚拟机（图-13所示）。



图-13

步骤四：删除指定的虚拟机

1）删除虚拟机CentOS7-c1

将虚拟机CentOS7-c1关机，右击选择“删除”，在弹出窗口中根据需要确认是否删除虚拟机的磁盘，单击右下角的“删除”按钮即可（如图-14所示）。

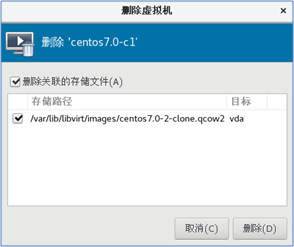


图-14

2）确认删除结果

在“虚拟系统管理器”界面中，虚拟机CentOS7-c1已经没有了（如图-15所示）。

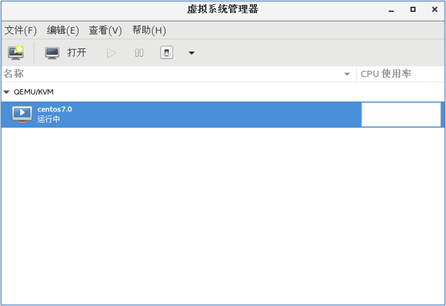


图-15

## 3 案例3：virsh基本管理操作

### 3.1 问题

本例要求学会使用virsh命令管理KVM虚拟机，主要完成下列任务：

1. 查看当前KVM服务器的内存/CPU
2. 列出有哪些虚拟机、查看各虚拟机的状态
3. 启动/重启/关机/强制关机操作
4. 设置虚拟机开机自动运行

### 3.2 方案

virsh命令提供了用来管理各虚拟机的命令接口，支持交互模式，可以实现对虚拟机的查看/创建/停止/关闭等各种操作。

用法参考：

1. virsh 控制指令 [虚拟机名称] [参数]

### 3.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：查看KVM服务器及虚拟机信息

1）查看KVM服务器节点

1. [root@kvmsvr ~]# virsh nodeinfo
2. CPU 型号： x86\_64
3. CPU： 4
4. CPU 频率： 2600 MHz
5. CPU socket： 1
6. 每个 socket 的内核数： 4
7. 每个内核的线程数： 1
8. NUMA 单元： 1
9. 内存大小： 16230564 KiB

2）列出有哪些虚拟机（包括未开启的）及各自的状态

1. [root@kvmsvr ~]# virsh list --all
2. Id Name State
3. ----------------------------------------------------
4. - centos7.0 shut off

3）查看指定虚拟机centos7.0的配置摘要信息

1. [root@kvmsvr ~]# virsh dominfo centos7.0
2. Id: 6
3. 名称： centos7.0
4. UUID: 93fa5915-c060-4631-b4c2-bd4aaaea8fee
5. OS 类型： hvm
6. 状态： running
7. CPU： 1
8. CPU 时间： 28.7s
9. 最大内存： 1048576 KiB
10. 使用的内存： 1048576 KiB
11. 持久： 是
12. 自动启动： 禁用
13. 管理的保存： 否
14. 安全性模式： none
15. 安全性 DOI： 0

步骤二：虚拟机开关机操作

1）将虚拟机centos7.0开启

1. [root@kvmsvr ~]# virsh start centos7.0                             //开机
2. Domain centos7.0 started
3. [root@kvmsvr ~]# virsh list                                     //检查结果
4. Id Name State
5. ----------------------------------------------------
6. 5 centos7.0 running

2）将虚拟机centos7.0关机

1. [root@kvmsvr ~]# virsh shutdown centos7.0                         //关机
2. Domain centos7.0 is being shutdown
3. .. .. //稍等片刻
4. [root@kvmsvr ~]# virsh list --all                                //检查结果
5. Id Name State
6. ----------------------------------------------------
7. - centos7.0 shut off

3）将虚拟机centos7.0强制关机（shutdown无效时适用）

1. [root@kvmsvr ~]# virsh destroy centos7.0                             //强制关机
2. Domain centos7.0 destroyed
3. [root@kvmsvr ~]# virsh list --all                                //检查结果
4. Id Name State
5. ----------------------------------------------------
6. - centos7.0 shut off

步骤三：虚拟机自启设置

1）将虚拟机centos7.0设为自动启动

1. [root@kvmsvr ~]# virsh autostart centos7.0                         //设置自启动
2. Domain centos7.0 marked as autostarted
3. [root@kvmsvr ~]# virsh dominfo centos7.0                         //确认结果
4. Id: -
5. Name: centos7.0
6. .. ..
7. Autostart: enable
8. .. ..

2）将虚拟机centos7.0取消自动启动

1. [root@kvmsvr ~]# virsh autostart --disable centos7.0             //取消自启动
2. Domain centos7.0 unmarked as autostarted
3. [root@kvmsvr ~]# virsh dominfo centos7.0                         //确认结果
4. Id: -
5. Name: centos7.0
6. .. ..
7. Autostart: disable
8. .. ..

## 4 案例4：xml配置文件的应用

### 4.1 问题

本例要求在KVM服务器上通过使用xml文件完成下列任务：

1. 将虚拟机 centos7.0 改名为 centos-207
2. 将虚拟机 centos-207 复制为 centos-7
3. 上述虚拟机的CPU/内存/网络类型保持不变
4. 但这2个虚拟机有可能会同时运行，不应出现冲突

### 4.2 方案

KVM虚拟机的xml配置文件也就是通常所说的虚拟机的描述文件，主要用来定义一个虚拟机的名称、UUID、CPU、内存、虚拟磁盘、网卡等各种参数设置。

KVM虚拟机的xml配置文件默认位于： /etc/libvirt/qemu/虚拟机名.xml 。

修改虚拟机配置的基本思路：

1. 编辑虚拟机配置：virsh edit 虚拟机名
2. 根据需要修改，保存配置结果

### 4.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：将虚拟机centos7.0改名为centos-207

1）生成新的UUID字串，并复制备用

1. [root@kvmsvr ~]# uuidgen
2. 76d5dc2c-5eef-4e30-8b6c-e58851814f84

2）编辑虚拟机centos7.0的配置

调整name、uuid值，保存修改：

1. [root@kvmsvr ~]# virsh edit centos7.0
2. <domain type='kvm'>
3. <name>centos-207</name>                                     //新名称
4. <uuid>76d5dc2c-5eef-4e30-8b6c-e58851814f84</uuid>         //新UUID值
5. .. ..

3）确认已自动导入的新配置

1. [root@kvmsvr ~]# virsh list --all
2. Id Name State
3. ----------------------------------------------------
4. - centos-207 shut off                 //新名称的虚拟机
5. - centos7.0 shut off

4）删除旧名称的虚拟机配置

1. [root@kvmsvr ~]# virsh undefine centos7.0                     //取消定义虚拟机
2. Domain centos7.0 has been undefined
3. [root@kvmsvr ~]# virsh list --all                         //确认结果
4. Id Name State
5. ----------------------------------------------------
6. - centos-207 shut off

步骤二：将虚拟机centos-207复制为centos-7

1）生成新UUID，并复制备用

1. [root@kvmsvr ~]# uuidgen
2. 90908905-bde4-4c4a-90b0-8a8f5bba1e25

2）修改导出后的配置（调整名称、UUID、磁盘路径、网卡MAC）

修改导出的xml配置文件，调整name、uuid、disk路径、mac地址值：

1. [root@kvmsvr ~]# virsh edit centos-207
2. <domain type='kvm'>
3. <name>centos-7</name>                                             //新名称
4. <uuid>90908905-bde4-4c4a-90b0-8a8f5bba1e25</uuid>             //新UUID值
5. .. ..
6. <disk type='file' device='disk'>
7. <driver name='qemu' type='qcow2'/>
8. <source file='/var/lib/libvirt/images/centos-7.qcow2'/>         //新磁盘路径
9. .. ..
10. </disk>
11. .. ..
12. <interface type='network'>
13. <mac address='52:54:00:91:50:07'/>                         //新MAC地址
14. <source network='default'/>
15. <model type='virtio'/>
16. <address type='pci' domain='0x0000' bus='0x00' slot='0x03
17. ' function='0x0'/>
18. </interface>
19. .. ..

3）确认已自动导入的新配置

1. [root@kvmsvr ~]# virsh list --all                         //确认结果
2. Id Name State
3. ----------------------------------------------------
4. - centos-207 shut off
5. - centos-7 shut off                 //新虚拟机

4）复制虚拟机磁盘文件

为新虚拟机提供一份独立的磁盘文件：

1. [root@kvmsvr ~]# cd /var/lib/libvirt/images/
2. [root@kvmsvr images]# cp centos7.0.qcow2 centos-7.qcow2
3. [root@kvmsvr images]# ls                                     //确认结果
4. centos-7.qcow2 centos7.0.qcow2

## 5 案例5：快建新虚拟机

### 5.1 问题

本例要求利用qcow2磁盘特性快建2台新的KVM虚拟机，配置要求如下：

1. svr7：svr7.tedu.cn，192.168.4.7/24
2. pc207： pc207.tedu.cn，192.168.4.207/24
3. 为上述虚拟机配好网络，确认yum源可用
4. 从CentOS真机可ssh远程访问这两台虚拟机

### 5.2 方案

快建新虚拟机的基本思路：

1. 提前准备好一台模板虚拟机（镜像磁盘+xml配置文件）
2. 基于qcow2磁盘复用技术快建新虚拟机的磁盘
3. 通过调整模板机的配置快建新虚拟机的xml配置文件
4. 导入新虚拟机

Copy On Write，写时复制技术原理：

* 直接映射原始盘的数据内容
* 当原始盘有修改时，在修改之前将旧数据存入前端盘
* 对前端盘的修改不会回写到原始盘

### 5.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：准备模板虚拟机

找一台已经装好CentOS7系统、配置好本地yum源、关闭SELinux的虚拟机，提取模板虚拟机磁盘、xml配置文件备用。

1）准备磁盘目录、模板虚拟机磁盘文件

1. [root@kvmsvr ~]# qemu-img info /data/images/centos7\_muban.qcow2
2. image: /data/images/centos7\_muban.qcow2
3. file format: qcow2
4. virtual size: 300G (322122547200 bytes)                 //虚拟机磁盘容量
5. disk size: 3.2G                                         //在KVM服务器占用容量

2）准备模板虚拟机配置文件

1. [root@kvmsvr ~]# cat /data/images/centos7\_muban.xml
2. <domain type='kvm'>
3. <name>centos7.0</name>
4. <uuid>a1992150-5cc7-e19d-20df-cd5cea7d8aa2</uuid>
5. <memory unit='KiB'>2097152</memory>
6. .. ..

步骤二：快建虚拟机svr7

1）为虚拟机svr7快速建立前端盘（复用模板机的磁盘数据）

1. [root@kvmsvr ~]# qemu-img create -f qcow2 -b /data/images/centos7\_muban.qcow2 /data/images/svr7.qcow2
2. Formatting '/data/images/svr7.qcow2', fmt=qcow2 size=322122547200 backing\_file='/data/images/centos7\_muban.qcow2' encryption=off cluster\_size=65536

2）为虚拟机svr7准备xml配置

1. [root@kvmsvr ~]# cp /data/images/centos7\_muban.xml /tmp/svr7.xml    //拷贝配置
2. [root@kvmsvr ~]# vim /tmp/svr7.xml                             //修改配置
3. <domain type='kvm'>
4. <name>svr7</name>                                             //新名称
5. <uuid>b20a1a1c-a2de-4b2f-bb03-91a3e36257c7</uuid>             //新UUID值
6. .. ..
7. <channel type='unix'>
8. <source mode='bind' path='/var/lib/libvirt/qemu/channel/t
9. arget/domain-centos-7/org.qemu.guest\_agent.0'/>                     //改套接字路径
10. .. ..
11. </channel>
12. <disk type='file' device='disk'>
13. <driver name='qemu' type='qcow2'/>
14. <source file='/var/lib/libvirt/images/svr7.qcow2'/>         //新磁盘路径
15. .. ..
16. </disk>
17. .. ..
18. <interface type='network'>
19. <mac address='52:54:00:11:00:07'/>                         //新MAC地址
20. <source network='default'/>
21. <model type='virtio'/>
22. <address type='pci' domain='0x0000' bus='0x00' slot='0x03
23. ' function='0x0'/>
24. </interface>
25. .. ..

3）定义新虚拟机svr7

1. [root@room9pc00 ~]# virsh define /tmp/svr7.xml
2. 定义域 svr7（从 /tmp/svr7.xml）

4）为虚拟机svr7配置主机名、IP地址

开启虚拟机svr7，使用root用户登入系统。

1. [root@svr7 ~]# vim /etc/hostname                     //配置主机名
2. svr7.tedu.cn
3. [root@svr7 ~]# nmcli connection modify eth0 ipv4.method manual ipv4.addresses 192.168.4.7/24 connection.autoconnect yes         //配置IP地址
4. [root@svr7 ~]# nmcli connection up eth0             //激活连接
5. 成功激活的连接（D-Bus 激活路径：/org/freedesktop/NetworkManager/ActiveConnection/8）
6. [root@svr7 ~]# ifconfig eth0                             //确认配置结果
7. eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
8. inet 192.168.4.7 netmask 255.255.255.0 broadcast 192.168.4.255
9. inet6 fe80::20c:29ff:fe5e:f686 prefixlen 64 scopeid 0x20<link>
10. ether 52:54:00:11:00:07 txqueuelen 1000 (Ethernet)
11. RX packets 112143 bytes 9388455 (8.9 MiB)
12. RX errors 0 dropped 0 overruns 0 frame 0
13. TX packets 507844 bytes 771354289 (735.6 MiB)
14. TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

步骤三：快建虚拟机pc207

操作与步骤二类似，此处略。

步骤四：快建虚拟机的运行测试

1）启动两台虚拟机svr7、pc207，均可正常运行

2）从svr7可ping通pc207

1. [root@svr7 ~]# ping 192.168.4.207
2. PING 192.168.4.207 (192.168.4.207) 56(84) bytes of data.
3. 64 bytes from 192.168.4.207: icmp\_seq=1 ttl=64 time=0.392 ms
4. 64 bytes from 192.168.4.207: icmp\_seq=2 ttl=64 time=0.369 ms
5. .. ..

3）从CentOS真机可ssh远程访问这两台虚拟机

1. [root@room9pc00 ~]# ssh -X root@192.168.4.7
2. root@192.168.4.7's password:                         //验证svr7的root密码
3. Last login: Sun Mar 26 11:30:35 2017 from 192.168.4.254
4. [root@svr7 ~]# exit
5. [root@room9pc00 ~]# ssh -X root@192.168.4.207
6. root@192.168.4.207's password:                         //验证pc207的root密码
7. Last login: Sun Mar 26 11:32:35 2017 from 192.168.4.254
8. [root@pc207 ~]# exit
9. [root@room9pc00 ~]#