

TTS 10.0 COOKBOOK

(NSD CLOUD DAY01)

版本编号 10.0

2018-08

达内 IT 培训集团

NSD CLOUD DAY01

1. 案例 1 : virsh 基本管理操作

• 问题

本案例要求熟悉 virsh 的基本操作，可以熟练运用：

- 列出当前正在运行的虚拟机
- 查看虚拟机的信息
- 管理虚拟机
- 设置虚拟机开机自动运行

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：virsh 基本操作

1) 列出当前正在运行的虚拟机

```
[root@room9pc01 ~]# virsh list
```

Id	Name	State
1	node1	running

2) 查看虚拟机的信息

```
[root@room9pc01 ~]# virsh dominfo node1 //查看 node1 的信息
```

```
Id: 1
Name: node1
UUID: 20e15d2f-ea30-4aa3-96dc-91aab6283b10
OS Type: hvm
State: running
CPU(s): 2
CPU time: 92.8s
Max memory: 2048000 KiB
Used memory: 2048000 KiB
Persistent: yes
Autostart: disable
Managed save: no
Security model: none
Security DOI: 0
```

步骤二：管理虚拟机

1) 启动虚拟机

```
[root@room9pc01 ~]# virsh start node1
```

2) 重启虚拟机

```
[root@room9pc01 ~]# virsh reboot node1
```

3) 强制关闭虚拟机

```
[root@room9pc01 ~]# virsh destroy node1
```

4) 设置虚拟机开机自动运行

```
[root@room9pc01 ~]# virsh autostart node1
```

2. 案例 2 : qemu-img 基本操作管理

• 问题

本案例要求：

- 创建一个新的镜像盘文件
- 使用后端模板文件创建一个新的镜像盘文件
- 查看镜像盘文件的信息

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：创建一个新的镜像盘文件

1) 创建一个新的镜像盘文件

qemu-img 命令格式：qemu-img 命令 参数 块文件名称 大小

```
[root@room9pc01 ~]# qemu-img create -f qcow2 disk.img 50G //qcow2 为创建的格式
Formatting 'disk.img', fmt=qcow2 size=53687091200 encryption=off cluster_size=65536
lazy_refcounts=off
```

2) 使用后端模板文件创建一个新的镜像盘文件

备注：-b 使用后端模板文件

```
[root@room9pc01 ~]# qemu-img create -b disk.img -f qcow2 disk1.img
Formatting 'disk1.img', fmt=qcow2 size=53687091200 backing_file='disk.img'
encryption=off cluster_size=65536 lazy_refcounts=off
```

3) 使用后端模板文件创建一个 16G 的镜像盘文件

```
[root@room9pc01 ~]# qemu-img create -b disk.img -f qcow2 disk2.img 16G
Formatting 'disk1.img', fmt=qcow2 size=53687091200 backing_file='disk.img'
encryption=off cluster_size=65536 lazy_refcounts=off
```

步骤二：查看镜像文件的信息

```
[root@room9pc01 ~]# qemu-img info disk1.img
image: disk.img
file format: qcow2
virtual size: 50G (53687091200 bytes)
disk size: 196K
cluster_size: 65536
Format specific information:
  compat: 1.1
```

lazy refcounts: false	help_topic
innodb_index_stats	
innodb_table_stats	
ndb_binlog_index	
plugin	
proc	
procs_priv	
proxies_priv	
server_cost	
servers	
slave_master_info	

3. 案例 3：创建一个虚拟网络

• 问题

创建一个虚拟网络，为之后的自定义安装虚拟机做准备：

- 创建一个名为 vbr 的虚拟网络
- 设置 vbr 的 ip 为 192.168.1.254
- 配置 vbr 虚拟网络的 dhcp 分配地址范围 100-200
- 启动 vbr 虚拟网络并用 ifconfig 验证
- 设置 vbr 虚拟网络开机自启动

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：创建一个名为 vbr 的虚拟网络

```
[root@room9pc01 ~]# vim /etc/libvirt/qemu/networks/vbr.xml
<network>
  <name>vbr</name>                                //vbr 为虚拟网络的名字
  <bridge name="vbr"/>
  <forward mode="nat"/>
  <ip address="192.168.1.254" netmask="255.255.255.0"> //ip 为 192.168.1.254
    <dhcp>
      <range start="192.168.1.100" end="192.168.1.200"/> //ip 范围是 100-200
    </dhcp>
  </ip>
</network>
```

步骤二：启动 vbr 虚拟网络并用 ifconfig 验证

```
[root@room9pc01 ~]# virsh net-define vbr          //定义 vbr 虚拟网络
[root@room9pc01 ~]# virsh net-start vbr           //启动 vbr 虚拟网络
[root@room9pc01 ~]# ifconfig                      //ifconfig 验证
vbr: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.254 netmask 255.255.255.0 broadcast 192.168.1.255
    ether 52:54:00:b7:1c:10 txqueuelen 1000 (Ethernet)
    RX packets 2460 bytes 176958 (172.8 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1948 bytes 532542 (520.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

步骤三：设置 vbr 虚拟网络开机自启动

```
[root@room9pc01 ~]# virsh net-autostart vbr
```

4. 案例 4：xml 管理

• 问题

熟悉 xml 文件，并对虚拟机的配置进行调整：

- 导出一个虚拟机的 xml 配置文件
- 编辑 xml 文件
- 重新定义虚拟机
- 删除此虚拟机

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：导出虚拟机 xml 的配置文件

1) 查看 xml 配置文件

```
[root@room9pc01 ~]# cd /etc/libvirt/qemu/
[root@room9pc01 qemu]# virsh dumpxml node1
```

2) 导出 xml 配置文件

```
[root@room9pc01 qemu]# virsh dumpxml node1 > node.xml
//导出虚拟机 node1 的配置文件为 node.xml
[root@room9pc01 qemu]# ls
node.xml
```

3) 对虚拟机的配置进行调整

virsh 命令：virsh edit 虚拟机名

备注：可以修改 name，memory，disk、network 等字段

```
[root@room9pc01 qemu]# virsh edit node1
<domain type='kvm'>
  <name>node1</name>                                //node1 为虚拟机的名称，可以随意修改
  <uuid>76d5dc2c-5eef-4e30-8b6c-e58851814f84</uuid> //uuid 可以去掉
  <memory unit='KiB'>2048000</memory>              //内存大小可以调整
  <currentMemory unit='KiB'>2048000</currentMemory>
  <disk type='file' device='disk'>
    <source file='/var/lib/libvirt/images/rhel7.2.qcow2'> //后端盘路径
  .. ..
  <interface type='network'>
    <mac address='52:54:00:91:52:e4'>
//mac 地址和带有 address 字样的全部去掉
  .. ..
```

步骤二：重新定义虚拟机

1) 重新定义虚拟机

```
[root@room9pc01 qemu]# virsh define node1.xml
```

2) 取消定义的虚拟机

```
[root@room9pc01 qemu]# virsh undefine node1
```

5. 案例 5：安装虚拟机

• 问题

本案例要求可以成功安装一个自定义虚拟机：

- 配置一个网络 yum，并安装一个虚拟机
- 制作一个虚拟机模板，包括配置 yum，网卡等

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：配置网络 yum 源

```
[root@room9pc01 ~]# yum -y install vsftpd
[root@room9pc01 ~]# vim /etc/vsftpd/vsftpd.conf
listen=YES
listen_ipv6=NO
[root@room9pc01 ~]# systemctl restart vsftpd
[root@room9pc01 ~]# mkdir /var/ftp/centos
[root@room9pc01 ~]# mount /iso/CentOS-7-x86_64-DVD-1708.iso /var/ftp/centos/
mount: /dev/loop1 写保护，将以只读方式挂载
[root@room9pc01 ~]# vim /etc/yum.repos.d/dvd.repo
[dvd]
name=dvd
baseurl=ftp://192.168.1.254/centos
enabled=1
gpgcheck=0
[root@room9pc01 ~]# yum clean all
[root@room9pc01 ~]# yum repolist
[root@room9pc01 ~]# cd /var/lib/libvirt/images/
[root@room9pc01 images]# qemu-img create -f qcow2 node.qcow2 16G
Formatting 'node.qcow2', fmt=qcow2 size=17179869184 encryption=off
cluster_size=65536 lazy_refcounts=off
```

步骤二：安装一个虚拟机

```
[root@room9pc01 images]# virt-manager //真机操作，启动 virt-manager
```

使用 virt-manager 软件选择新建虚拟机如图-1 所示：

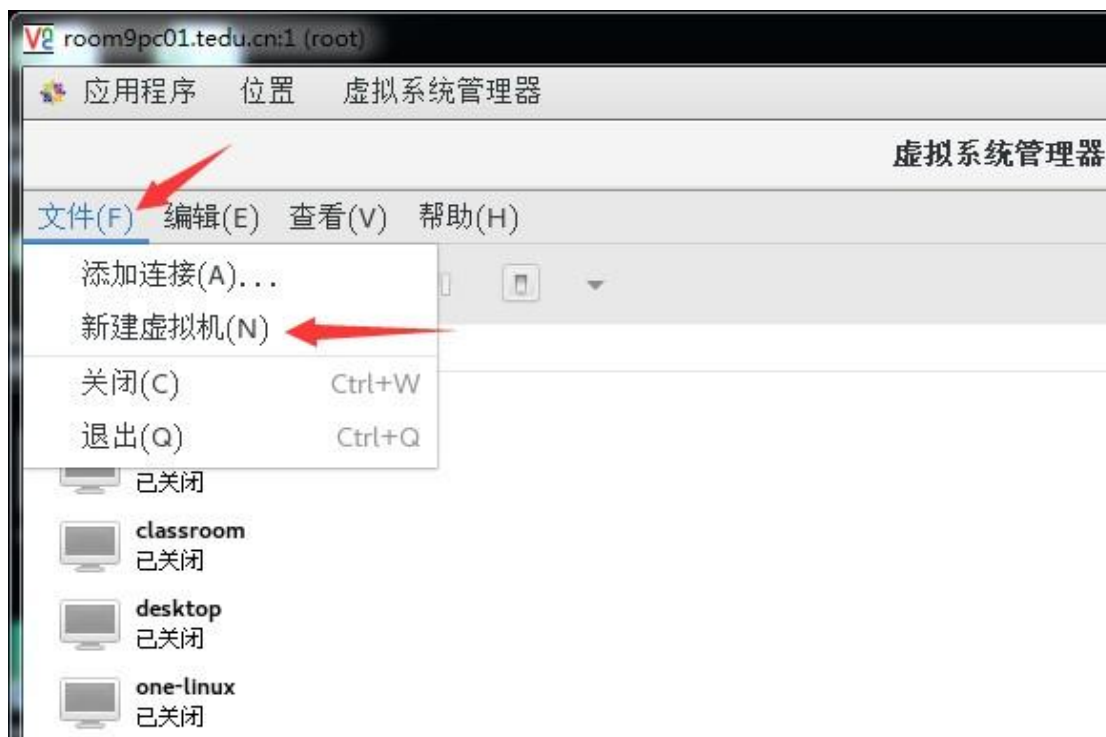


图-1

选择安装方式如图-2 所示：

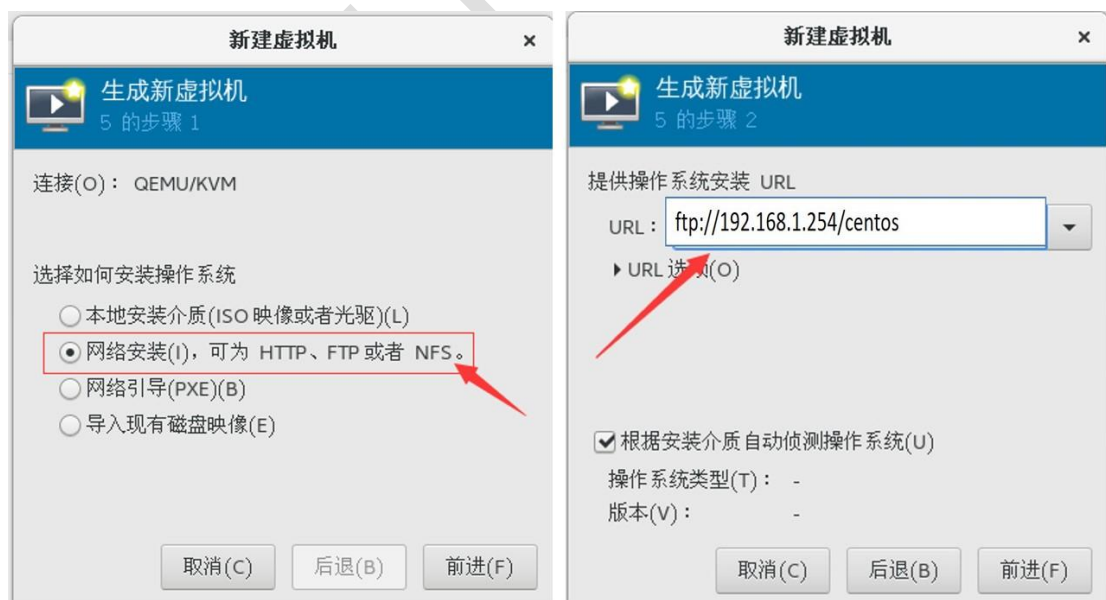


图-2

选择内存，cpu 和自定义存储如图-3 所示：



图-3

选择虚拟机名称和网络如图-4 所示：



图-4

选择分区和 KDUMP 如图-5 所示：

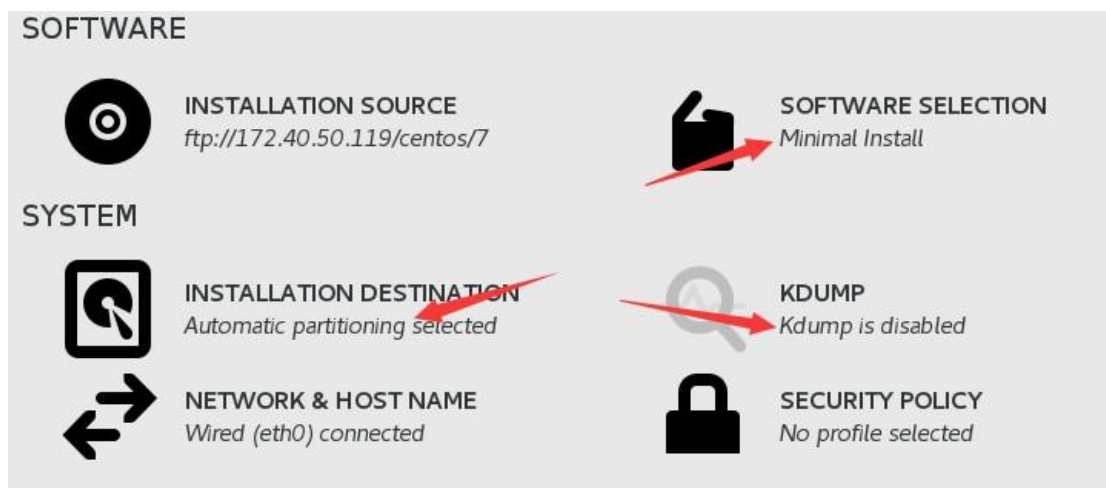


图-5

选择创建分区如图-6 所示：

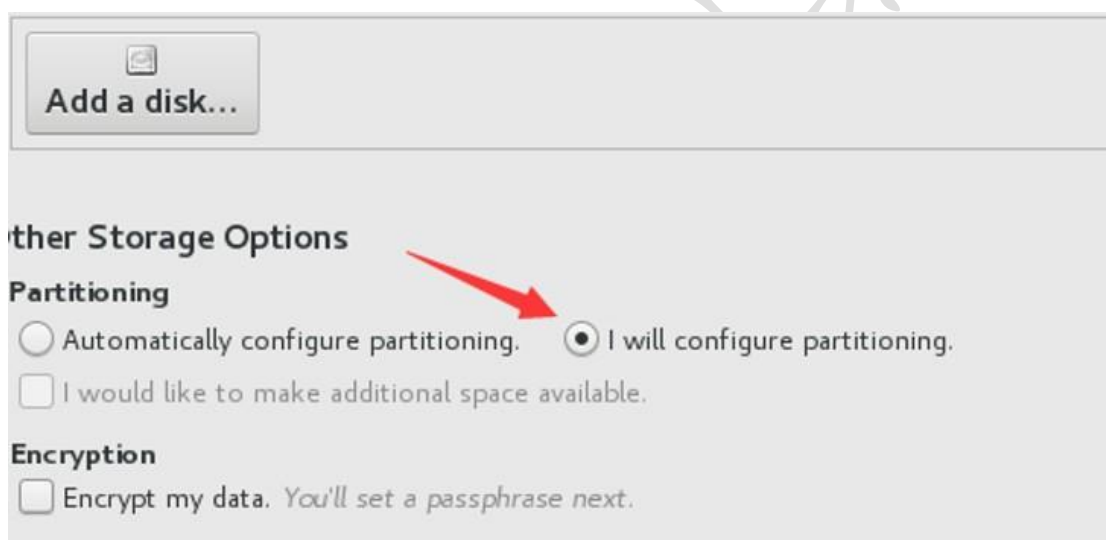


图-6

选择 standard Partition 如图-7 所示：

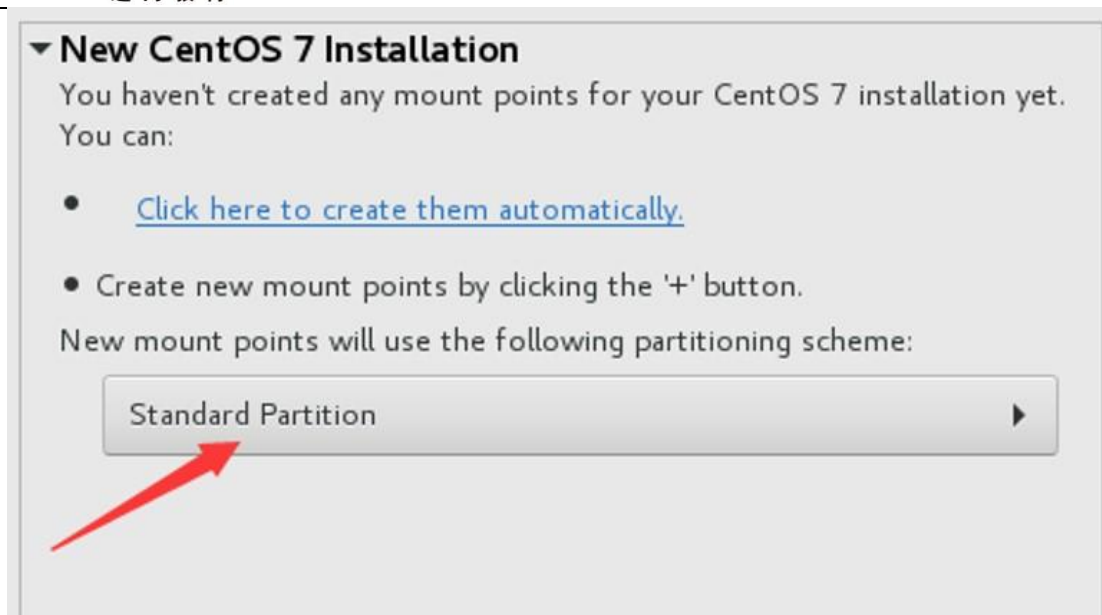


图-7

创建一个根分区如图-8 所示：

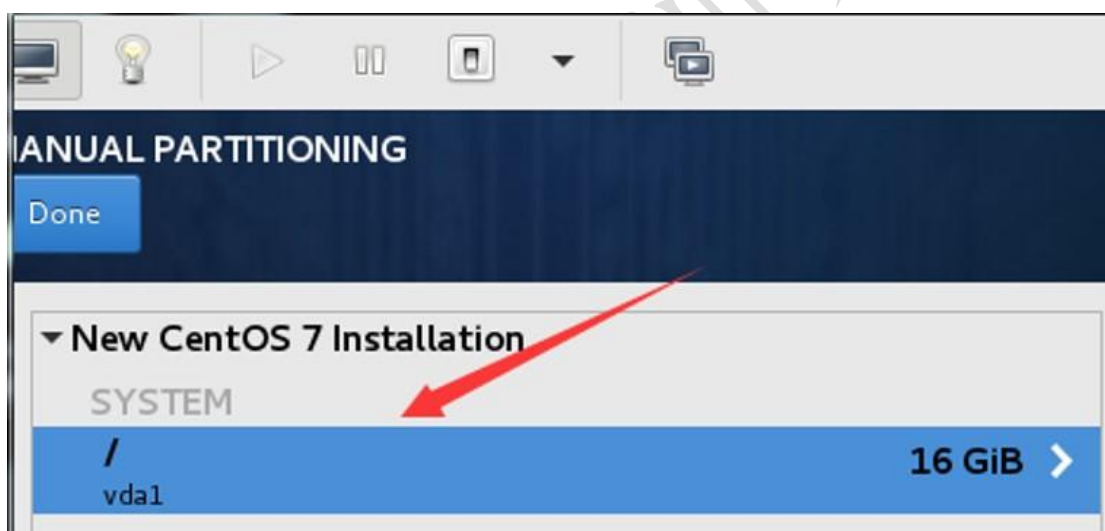


图-8

步骤三：制作一个虚拟机模板

1) 禁用 selinux

```
[root@localhost ~]# vim /etc/selinux/config
SELINUX=disabled
```

2) 卸载防火墙与 NetworkManager

```
[root@localhost ~]# yum -y remove NetworkManager-* firewalld-* python-firewall
```

3) 配置 yum 源

```
[root@localhost ~]# vim /etc/yum.repos.d/dvd.repo
[dvd]
name=dvd
```

```
baseurl=ftp://192.168.1.254/centos
enabled=1
gpgcheck=0
[root@localhost ~]# yum clean all
[root@localhost ~]# yum repolist
```

4) 导入公钥

注意：把/etc/yum.repos.d/dvd.repo的gpgcheck=0改成gpgcheck=1

```
[root@localhost ~]# lftp 192.168.1.254
lftp 192.168.4.254:~> cd centos
lftp 192.168.4.254:/centos> get RPM-GPG-KEY-CentOs-7
lftp 192.168.4.254:/centos> exit
[root@localhost ~]# rpm --import RPM-GPG-KEY-CentOs-7
[root@localhost ~]# yum -y install net-tools vim-enhanced bridge-utils psmisc
```

5) 配置网卡

```
[root@localhost ~]# vim /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE="eth0"
ONBOOT="yes"
IPV6INIT="no"
TYPE="Ethernet"
BOOTPROTO="dhcp"
[root@localhost ~]# systemctl restart network
```

6) 禁用空路由

```
[root@localhost ~]# vim /etc/sysconfig/network
ZEROCONF="yes"
```

7) 添加 console 配置

```
[root@localhost ~]# vim /etc/default/grub
GRUB_CMDLINE_LINUX="biosdevname=0 net.ifnames=0 console=ttyS0,115200n8"
GRUB_DISABLE_LINUX_UUID="true"
GRUB_ENABLE_LINUX_LABEL="true"
```

8) 重新生成 grub.cfg

```
[root@localhost ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
```

9) 安装扩展分区软件

```
[root@localhost ~]# yum install -y cloud-utils-growpart
```

10) 第一次开机自动扩容

```
[root@localhost ~]# /usr/bin/growpart /dev/vda 1
[root@localhost ~]# /usr/sbin/xfs_growfs /
```

11) 关闭虚拟机后执行信息清理工作

```
[root@room9pc01 ~]# virt-sysprep -d centos7.0
//真机上面操作，centos7.0为虚拟机名称
```

6. 案例 6：离线访问虚拟机问题

本案例要求可以离线访问虚拟机：

- 利用 xml 文件生成一个新的虚拟机
- 利用 guestmount 实现离线访问虚拟机

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：用 xml 生成一个新的虚拟机

```
[root@room9pc01 ~]# cd /etc/libvirt/qemu/
[root@room9pc01 qemu]# cp centos7.0.xml local.xml
[root@room9pc01 qemu]# vim local.xml
<domain type='kvm'>
  <name>local</name>           //修改名字
  <memory unit='KiB'>2048000</memory>           //修改内存
  <currentMemory unit='KiB'>2048000</currentMemory>
  ...
  <source file='/var/lib/libvirt/images/local.img'/>           //img 文件位置
```

注意：除这些外还要把 mac 地址删掉，带 address 字样的全部删除

```
[root@room9pc01 ~]# cd /var/lib/libvirt/images/
[root@room9pc01 images]# qemu-img create -b node.qcow2 -f qcow2 local.img
[root@room9pc01 images]# virsh define /etc/libvirt/qemu/local.xml
[root@room9pc01 images]# virsh start local
[root@room9pc01 images]# virsh console local
```

步骤二：guestmount 实现离线访问

基本用法：guestmount -a 虚拟机磁盘路径 -i /挂载点

-a：指定虚拟磁盘

-i：挂载点

```
[root@room9pc01 ~]# mkdir /mnt/kdisk
[root@room9pc01 ~]# guestmount -a node1.qcow2 -i /mnt/kdisk
[root@room9pc01 ~]# cd /mnt/kdisk
[root@room9pc01 kdisk]# ls
bin      home     media    opt      sbin     tmp
boot     lib      misc     proc     selinux  usr
```