

### 3\_Services02 虚拟化搭建

#### 一 手动图形安装虚拟机名为 nsd01

利用光盘镜像安装

内存 2048M

虚拟机名为:nsd01

网络类型选择为:private1

软件包:带 GUI 的服务器

分区:自动分区

设置 root 的密码为 1,建立 lisi 用户设置密码 1

#### 二 虚拟化概述

##### 2.1 virtualization 资源管理

x 个物理资源 --> y 个逻辑资源

实现程度:完全、部分、硬件辅助(CPU)

CPU 支持虚拟化所有的指令集(vmx)

```
[student@room9pc01 ~]$ lscpu | grep vmx
```

##### 2.2 虚拟化主要厂商及产品

VMware VMware Workstation、vSphere

Microsoft VirtualPC、Hyper-V

RedHat KVM、RHEV

Citrix Xen

## 二 虚拟机 nsd01 进行配置

### 2.1 配置永久主机名:kvm.tedu.cn

```
~]# echo kvm.tedu.cn > /etc/hostname
```

### 2.2 配置永久 IP 地址:192.168.4.30/24

```
~]# nmcli connection modify eth0 ipv4.method manual  
      ipv4.addresses 192.168.4.30/24 connection.autoconnect yes
```

```
~]# nmcli connection up eth0    //激活网卡
```

```
~]# ifconfig
```

### 2.3 修改防火墙默认区域与 SELinux

```
~]# firewall-cmd --set-default-zone=trusted
```

```
~]# setenforce 0
```

```
~]# getenforce
```

```
Permissive
```

```
~]# vim /etc/selinux/config
```

```
SELINUX=permissive
```

### 2.4 构建 Yum 仓库

## 三 安装虚拟化服务器平台

### 3.1 主要软件包

qemu-kvm :为 kvm 提供底层仿真支持

libvirt-daemon : libvirtd 守护进程, 管理虚拟机

libvirt-client : 用户端软件, 提供客户端管理命令

libvirt-daemon-driver-qemu : libvirtd 连接 qemu 的驱动

virt-manager : 图形管理工具

```
]# yum -y install qemu-kvm
```

```
]# yum -y install libvirt-daemon
```

```
]# yum -y install libvirt-client
```

```
]# yum -y install libvirt-daemon-driver-qemu
```

```
]# yum -y install virt-manager
```

虚拟化服务: libvirtd

```
[root@kvm /]# systemctl status libvirtd #查看服务的状态
```

## 四 通过命令管理虚拟机

### 4.1 virsh 命令工具介绍

提供管理各虚拟机的命令接口

支持交互模式, 查看/创建/停止/关闭 ...

**格式: virsh 控制指令 [虚拟机名称] [参数]**

查看 KVM 节点(服务器)信息

```
]$ virsh nodeinfo
```

列出虚拟机

```
]$ virsh list [--all] #重点
```

列出虚拟网络

```
]$ virsh net-list [--all]
```

查看指定虚拟机的信息

```
]$ virsh dominfo 虚拟机名称    #重点
```

```
]$ virsh autostart nsd01    #设置虚拟机开机自启动
```

```
]$ virsh dominfo nsd01
```

```
]$ virsh autostart --disable nsd01    #禁止虚拟机开机自启动
```

```
]$ virsh dominfo nsd01
```

开关机操作

```
]$ virsh start 或 reboot 或 shutdown 虚拟机名称    #重点
```

强制关闭指定的虚拟机

```
]$ virsh destroy 虚拟机名称    #重点
```

## 五 一台 KVM 虚拟机的组成

xml 配置文件(虚拟机描述文件):定义虚拟机的名称、UUID、CPU、内存、虚拟磁盘、网卡等各种参数设置

```
~]$ ls /etc/libvirt/qemu/    #xml 配置文件路径
```

磁盘镜像文件:保存虚拟机的操作系统及文档数据,镜像路径取决于 xml 配置文件中的定义

```
~]$ ls /var/lib/libvirt/images/    #镜像文件路径
```

### 5.1 查看虚拟机 xml 文件内容

```
[student@room9pc01 ~]$ virsh dumpxml nsd01
```

虚拟机名称

虚拟机的 UUID 唯一标识

虚拟机磁盘文件路径

```
<source file='/var/lib/libvirt/images/nsd01.qcow2' />
```

虚拟机网卡信息

```
<interface type='network'>
```

```
<mac address='52:54:00:d4:fe:59' />
```

```
[student@room9pc01 ~]$ virsh dumpxml nsd01 | less
```

## 5.2 在命令行手动制作一台虚拟机

### 5.2.1 磁盘镜像文件

```
]$ cd /var/lib/libvirt/images/
```

```
]$ du -sh .node_base.qcow2
```

```
]$ cp .node_base.qcow2 nsd02.qcow2
```

```
]$ du -sh nsd02.qcow2
```

### 5.2.2 建立新的 xml 配置文件，uuidgen 可以生成随机的 UUID 值

```
]$ cd /etc/libvirt/qemu/
```

```
]$ virsh dumpxml nsd01 > /etc/libvirt/qemu/nsd02.xml
```

```
]$ vim /etc/libvirt/qemu/nsd02.xml
```

1) 虚拟机名称    <name>nsd02</name>

2)虚拟机的 UUID 唯一标识      删除整行内容    UUID

3)虚拟机磁盘文件路径

```
<source file='/var/lib/libvirt/images/nsd02.qcow2'/>
```

4)虚拟机网卡信息

```
<mac address='52:54:00:d4:fe:59' />      #删除整行内容
```

### 5.2.3 导入虚拟机配置信息

```
]$ virsh define /etc/libvirt/qemu/nsd02.xml
```

```
]$ virsh list --all
```

```
]$ virsh start nsd02
```

### 5.2.4 命令行删除虚拟机

```
]$ virsh list --all
```

```
]$ virsh destroy nsd02      #强制关闭虚拟机
```

域 nsd02 被删除

```
]$ virsh list --all
```

```
]$ virsh undefine nsd02      #删除虚拟机 xml 配置文件
```

域 nsd02 已经被取消定义

```
]$ virsh list --all
```

```
]$ rm -rf /var/lib/libvirt/images/nsd02.qcow2      #手动删除
```

### 5.2.5 在命令行手动创建虚拟机

a 建立新的磁盘文件

```
]$ cp /var/lib/libvirt/images/.node_base.qcow2 /tmp/nsd03.qcow2

]$ du -sh /tmp/nsd03.qcow2
```

## b 建立 xml 配置文件

```
]$ virsh dumpxml nsd01 > /home/student/nsd03.xml

]$ ls /home/student/nsd03.xml

]$ vim /home/student/nsd03.xml

1)虚拟机名称    <name>nsd03</name>

2)虚拟机的 UUID 唯一标识    删除整行内容    UUID

3)虚拟机磁盘文件路径    <source file='/tmp/nsd03.qcow2' />

4)虚拟机网卡信息

<mac address='52:54:00:d4:fe:59' /> 删除整行内容
```

## c 导入虚拟机配置信息

```
]$ ls /etc/libvirt/qemu

$ virsh define /home/student/nsd03.xml #导入虚拟机信息

]$ ls /etc/libvirt/qemu

]$ virsh start nsd03

]$ virsh list
```

## 5.3 三合一:导出 修改 导入

对虚拟机的配置进行调整

编辑:virsh edit 虚拟机名

### 5.3.1 建立磁盘文件

```
l$ cp /var/lib/libvirt/images/.node_base.qcow2 /var/lib/libvirt/images/nsd04.qcow2
```

```
l$ du -sh /var/lib/libvirt/images/nsd04.qcow2
```

### 5.3.2 建立 xml 配置文件

```
[student@room9pc01 /]$ virsh edit nsd01
```

1) 虚拟机名称    <name>nsd04</name>

2) 虚拟机的 UUID 唯一标识    删除整行内容    UUID

3) 虚拟机磁盘文件路径    <source file='/var/libvirt/images/nsd04.qcow2' />

4) 虚拟机网卡信息

<mac address='52:54:00:d4:fe:59' />    #删除整行内容

```
[student@room9pc01 /]$ virsh list --all
```

## 5.4 常用镜像盘类型

虚拟机的磁盘镜像文件格式

特点\类型	RAW	QCOW2
KVM 默认	否	是
I/O 效率	高	较高
占用空间	大	小



压缩	不支持	支持
后端盘复用	不支持	支持
快照	不支持	支持

查看磁盘镜像文件信息

```
]$ qemu-img info /var/lib/libvirt/images/nsd04.qcow2
    virtual size: 2.0G (2147483648 bytes) #虚拟大小
    disk size: 633M #实际占用空间
```

## 六 COW 技术原理

Copy On Write, 写时复制

直接映射原始盘的数据内容

当原始盘的旧数据有修改时,在修改之前自动将旧数据存入前端盘

对前端盘的修改不回写到原始盘

原始盘的数据不能做修改

### 6.1 快速创建 qcow 前端盘

qemu-img 通过 -b 选项复用指定原始盘

```
qemu-img create -f qcow2 -b 原始盘 前端盘 大小
```

#### 6.1.1 命令手动创建虚拟机磁盘文件

```
]$ cd /var/lib/libvirt/images/
```

```
]$ qemu-img create -f qcow2 -b .node_base.qcow2 nsd05.qcow2
```

5G #通过原始盘产生前端盘

```
]$ qemu-img info nsd05.qcow2      #查看虚拟磁盘文件信息

    virtual size: 5.0G (5368709120 bytes) #虚拟大小
    disk size: 196K                  #实际占用空间大小
    backing file: .node_base.qcow2   #原始盘
```

## 6.2 建立 xml 文件

```
]$ virsh dumpxml nsd01 > /tmp/nsd05.xml  #导出虚拟机信息

]$ vim /tmp/nsd05.xml  #修改 虚拟机名 UUID 磁盘路径 网卡 MAC

]$ virsh define /tmp/nsd05.xml
```

## 6.3 离线访问虚拟机

使用 `guestmount` 工具

支持离线挂载 `raw`、`qcow2` 格式虚拟机磁盘

可以在虚拟机关机的情况下,直接修改磁盘中的文档

方便对虚拟机定制、修复、脚本维护

```
]$ virsh destroy nsd01

]$ guestmount -a /var/lib/libvirt/images/nsd01.qcow2 -i /mnt/
```