

系统&服务管理进阶

NSD CLOUD

DAY01

内容

上午	09:00 ~ 09:30	KVM构建及管理
	09:30 ~ 10:20	
	10:30 ~ 11:20	virsh控制工具
	11:30 ~ 12:00	
下午	14:00 ~ 14:50	镜像管理
	15:00 ~ 15:50	
	16:10 ~ 17:00	虚拟机技术
	17:10 ~ 18:00	总结和答疑



KVM构建及管理

KVM构建及管理

搭建KVM服务器

虚拟化概述

yum软件组管理

安装虚拟化服务器平台

启动libvirt服务端

管理KVM平台

连接本地/远程KVM

管理虚拟网络

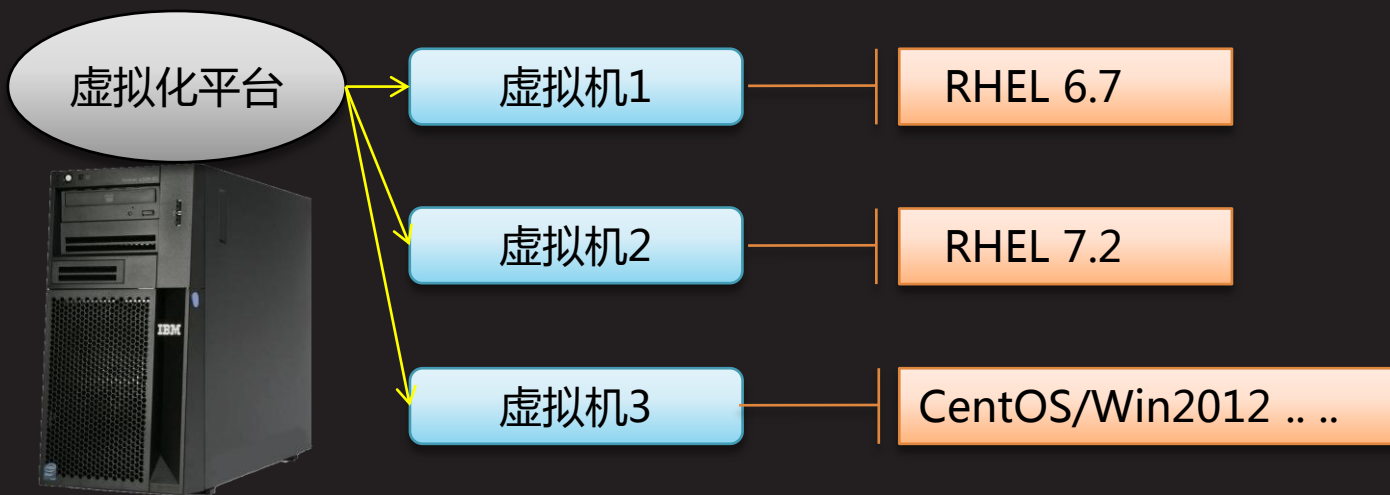
新建一台RHEL7虚拟机

虚拟机的基本控制

搭建KVM服务器

虚拟化概述

- 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）



虚拟化概述（续1）

- 虚拟化主要厂商及产品

系 列	PC/服务器版代表
VMware	VMware Workstation、vSphere
Microsoft	VirtualPC、Hyper-V
RedHat	KVM、RHEV
Citrix	Xen
Oracle	Oracle VM VirtualBox



yum软件组管理

- 快速配置网络 yum 源
 - 配置ftp 或 http 软件
`yum install vsftp`
 - 把 CentOS7 的光盘挂载到可以访问的目录
`mount -t iso9660 -o loop,ro /xx/xx.iso /urdir/`
 - 在客户机里面配置 `/etc/yum.repos.d/xxx.repo`
`[Centos_repo]`
`name= CentOS packet`
`baseurl=ftp://xx.xx.xx.xx/urdir/`
`enabled=1`
`gpgcheck=0`
 - `yum repolist`



安装虚拟化服务器平台

- 必备软件
 - qemu-kvm # 为kvm提供底层的支持
 - libvirt-client # virsh 等软件
 - libvirt-daemon # libvirtd 服务进程
 - libvirt-daemon-driver-qemu # virsh 之 qemu 驱动
- 可选功能
 - virt-install # 系统安装工具
 - virt-manager # 图形管理工具
 - virt-v2v # 虚拟机迁移工具
 - virt-p2v # 物理机迁移工具



启动libvirtd服务端

- libvirtd 为管理虚拟机提供服务接口
 - 调用 dnsmasq 提供DNS、DHCP等功能
 - 建立 vbr 虚拟网络 (192.168.1.0/24) 配置文件

```
<network>  
  <name>vbr</name>  
  <bridge name="vbr"/>  
  <forward mode="nat"/>  
  <ip address="192.168.1.254" netmask="255.255.255.0">  
    <dhcp>  
      <range start="192.168.1.100" end="192.168.1.200"/>  
    </dhcp>  
  </ip>  
</network>
```



启动libvirtd服务端

- 添加测试新的网桥
 - 显示 virsh 网络配置文件，默认显示已经启动的网络配置，--all 显示全部网络配置文件
 - virsh net-list [--all]
 - 根据 xml 文件配置 virt 网络 net-define
 - 删除 net-undefine virt 网络配置
 - 启动 virt 网络 net-start
 - 停止 virt 网络 net-destroy



案例1：安装一个KVM服务器

1. 在我们的宿主机上禁用 selinux
2. 卸载 firewall 和 NetworkManager
3. 安装虚拟化支持软件包
4. 配置网络 virt.xml 文件
5. 载入 xml ，启动 virt 网络



自定义系统安装配置



系统安装

- 创建一个 qcow2 文件
 - `cd /var/lib/libvirt/images`
 - `qemu-img create -f qcow2 node.cow2 16G`
- virt-manager 系统安装
- 使用刚刚创建的网络 yum 源 安装系统
 - 软件包要选择最小化



系统安装

- 初始化系统并制作模板
 - 配置 yum 源
 - 禁用 Selinux
 - 卸载 NetworkManager , firewalld
 - 安装常用工具软件 net-tools vim-enhanced ...
 - 修改网卡配置文件 , 删除UUID , MAC地址等
 - 重定向 console=tty0 console=ttyS0,115200n8
 - 关机并备份 xml 文件
 - 删除guest域



管理KVM平台



连接本地/远程KVM

- 使用 virsh 客户端工具
 - 连接本地
 - virsh
 - virsh# connect qemu:///system （默认选项）
 - 连接远程
 - virsh# connect
qemu+ssh://user@ip.xx.xx.xx:port/system
 - virt-manager 也可以通过 add connection 管理其它机器上的虚拟机，一般通过"remote tunnel over ssh"就可以了



案例2：KVM平台远程管理

远程管理实验

1、使用 virsh 连接本地 libvirtd

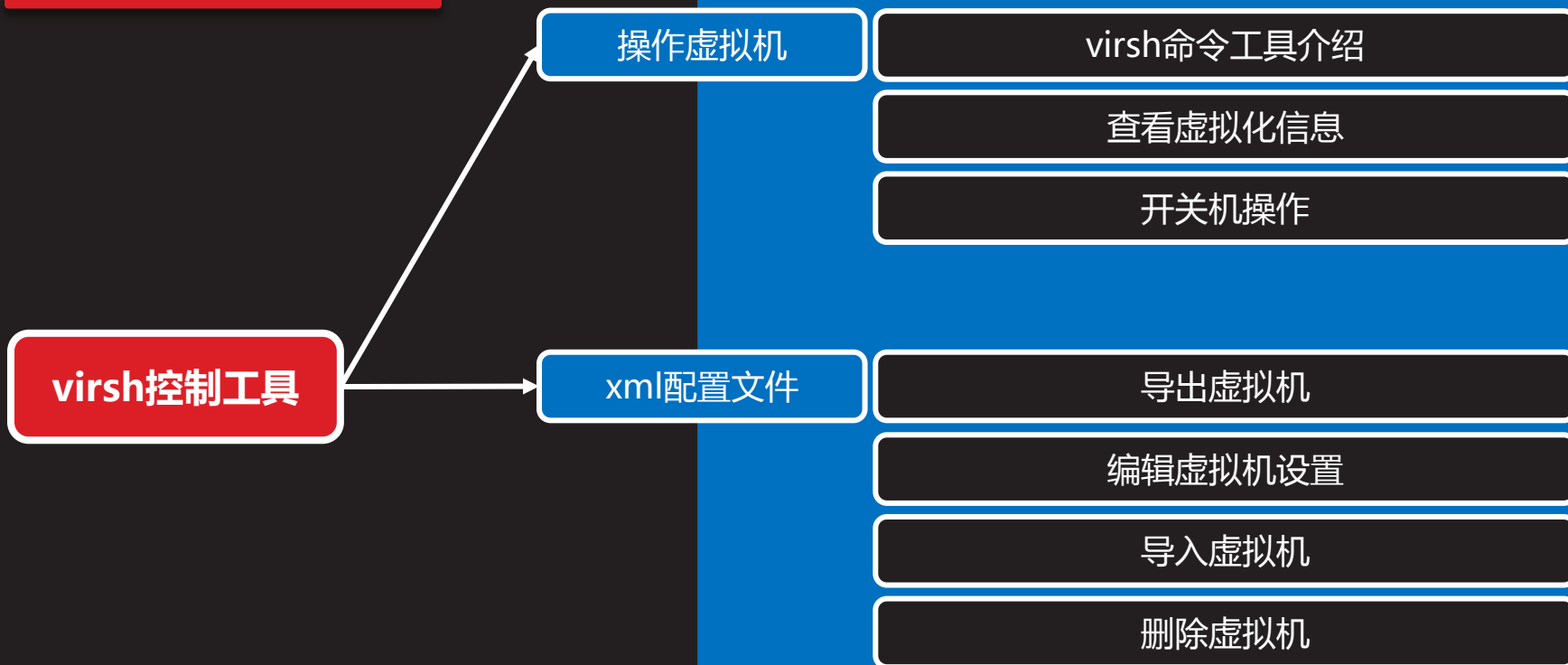
查看本地虚拟机配置

2、使用 virsh 连接远程 libvirtd

查看远程虚拟机配置



virsh控制工具



操作虚拟机

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]`

```
[root@kvmsvr ~]# virsh
欢迎使用 virsh，虚拟化的交互式终端。
```

```
输入：'help' 来获得命令的帮助信息
      'quit' 退出
```

```
virsh #
```



查看虚拟化信息

- 查看KVM节点（服务器）信息
 - `virsh nodeinfo`
- 列出虚拟机
 - `virsh list [--all]`
- 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`



开关机操作

- 运行|重启|关闭指定的虚拟机
 - `virsh start|reboot|shutdown` 虚拟机名称
- 强制关闭指定的虚拟机
 - `virsh destroy` 虚拟机名称
- 将指定的虚拟机设为开机自动运行
 - `virsh autostart [--disable]` 虚拟机名称



案例3：virsh基本管理操作

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



xml配置文件

导出虚拟机

- xml配置文件
 - 定义了一个虚拟机的名称、CPU、内存、虚拟磁盘、网卡等各种参数设置
 - 默认位于 `/etc/libvirt/qemu/虚拟机名.xml`
- 导出xml配置文件
 - 查看：`virsh dumpxml 虚拟机名`
 - 备份：`virsh dumpxml 虚拟机名 > 虚拟机名.xml`



编辑虚拟机设置

- 对虚拟机的配置进行调整
 - 编辑：`virsh edit 虚拟机名`
 - 若修改 name、memory、disk、network，可自动保存为新虚拟机配置

```
[root@kvmsvr ~]# virsh edit rhel-207
<domain type='kvm'>
  <name>rhel-207</name>
  <uuid>76d5dc2c-5eef-4e30-8b6c-e58851814f84</uuid>
  <disk type='file' device='disk'>
    <source file='/var/lib/libvirt/images/rhel7.2.qcow2'/>
  .. ..
  <interface type='network'>
    <mac address='52:54:00:91:52:e4'/>
  .. ..
```



导入虚拟机

- 根据修改后的独立xml文件定义新虚拟机
 - `virsh define XML描述文件`

```
[root@kvmsvr ~]# virsh define /root/rhel-207.xml
```

定义域 rhel-207 (从 /root/rhel-207.xml)

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

Id	名称	状态

-	rhel-207	关闭
-	rhel7.2	关闭



删除虚拟机

- 必要时可去除多余的xml配置
 - 比如虚拟机改名的情况
 - 避免出现多个虚拟机的磁盘或MAC地址冲突
 - `virsh undefine 虚拟机名`

```
[root@kvmsvr ~]# virsh undefine rhel7.2
```

域 rhel7.2 已经被取消定义



案例4：xml配置文件的应用

使用 virsh 调整虚拟机配置，完成下列任务

- 1) 将虚拟机 rhel7.2 改名为 rhel-207
- 2) 将虚拟机 rhel-207 复制为 rhel-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



镜像管理

镜像管理

虚拟机磁盘镜像

常用镜像盘类型

创建虚拟机磁盘镜像

创建/还原/删除快照

虚拟机磁盘镜像



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



创建虚拟机磁盘镜像

- 创建新的镜像盘文件
 - `qemu-img create -f 格式 磁盘路径 大小`
- 查询镜像盘文件的信息
 - `qemu-img info 磁盘路径`

```
[root@kvmsvr ~]# qemu-img create -f qcow2 disk1.qcow2
20G
```

.. ..

```
[root@kvmsvr ~]# qemu-img info disk1.qcow2
```

```
image: disk1.qcow2
```

```
file format: qcow2
```

```
virtual size: 20G (21474836480 bytes)
```

```
disk size: 196K
```

.. ..

//虚拟大小

//实际数据大小



创建/还原/删除快照

- 快照的作用
 - 在虚拟机磁盘镜像内记录不同时间点的状态数据备份
 - 必要时可将虚拟机恢复到指定的快照
- 基本用法
 - `qemu-img snapshot -c 快照名 qcow2磁盘`
 - `qemu-img snapshot -l qcow2磁盘`
 - `qemu-img snapshot -a 快照名 qcow2磁盘`
 - `qemu-img snapshot -d 快照名 qcow2磁盘`



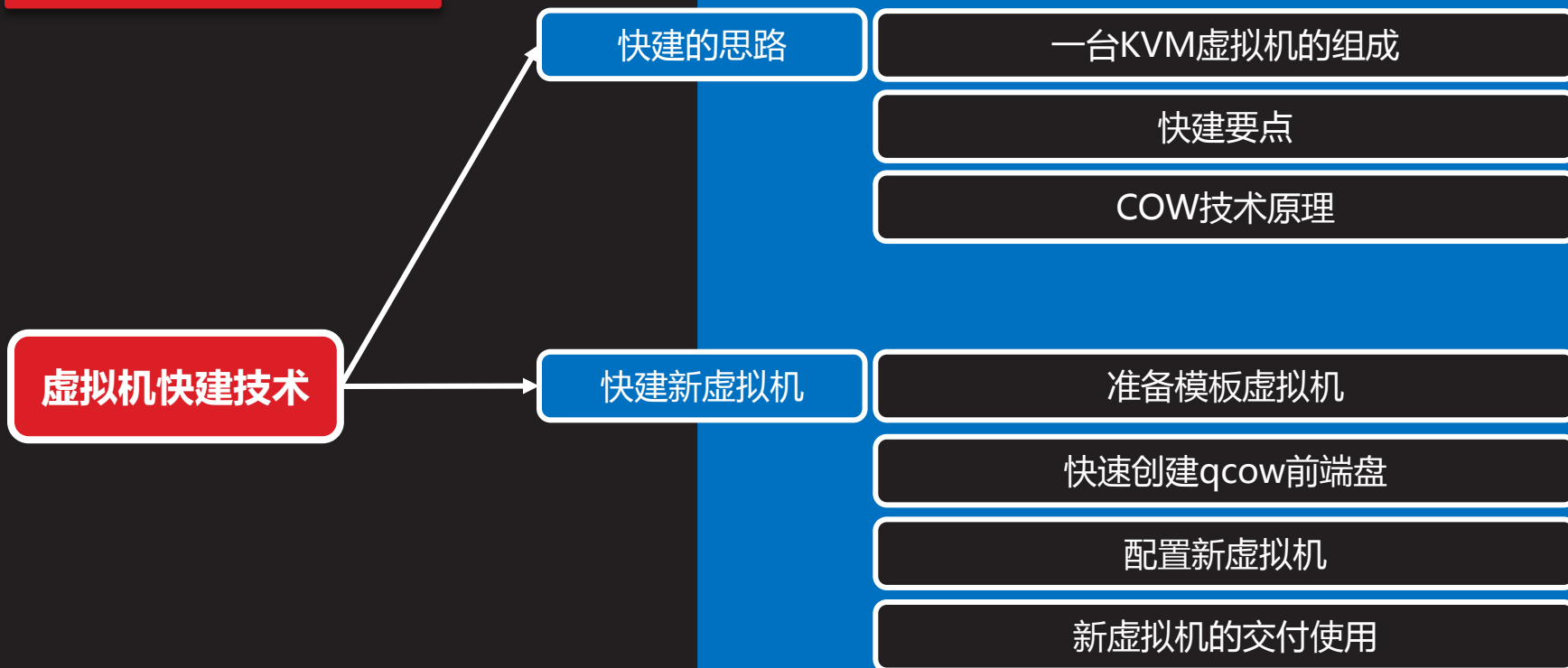
案例5：为虚拟机制作快照备份

使用 qemu-img 及必要的工具完成下列任务

- 1) 将虚拟机 rhel-7 关机
- 2) 为虚拟机 rhel-7 的磁盘制作名为 snap1 的快照
- 3) 开启并登入虚拟机 rhel-7，在桌面上新建文件 1.txt
- 4) 再次关闭虚拟机 rhel-7，还原到快照 snap1
- 5) 重新开启并登入虚拟机 rhel-7，检查 1.txt 文件



虚拟机快建技术



快建的思路

一台KVM虚拟机的组成

- 一台KVM虚拟机的组成
 - **xml配置文件**：定义虚拟机的名称、CPU、内存、虚拟磁盘、网卡等各种参数设置
 - **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



快建要点

- 基本思路

- 1) 准备一台模板虚拟机（镜像磁盘+xml配置文件）
- 2) 基于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



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



快建新虚拟机

准备模板虚拟机

- 模板机的选择
 - 可以用现有的虚拟机，也可再安装一台新虚拟机
- 模板机的初始化
 - 1) 预装软件/系统优化/关闭SELinux等
 - 2) 生产环境可以标记 `/.unconfigured`，方便新虚拟机的配置（相当于出厂设置，提示用户初始化）
 - 3) 备份好模板磁盘、xml配置文件
 - 4) 删除模板机（`undefine`）



快速创建qcow前端盘

- qemu-img 通过 -b 选项复用指定后端盘
 - qemu-img create -f qcow2 -b 后端盘 前端盘

```
[root@kvmsvr ~]# cd /data/images/
[root@kvmsvr images]# qemu-img create -f qcow2 -b
rhel7_muban.qcow2 rh7_node1.qcow2
...
[root@kvmsvr images]# ls -lh rh*.qcow2
-rw-r--r-- 1 root root    198K 12月  9 13:55 rh7_node1.qcow2
-rw-r--r-- 1 qemu qemu  3.2G 12月  7 11:08 rhel7_muban.qcow2
//对比前端盘、后端盘大小
```



配置新虚拟机

- 拷贝模板机的xml配置
 - 按照需要修改，并据此定义新虚拟机

```
[root@kvmsvr ~]# cp /data/images/rhel7_muban.xml /root/  
[root@kvmsvr ~]# vim /root/rhel7_muban.xml
```

```
.. ..
```

```
[root@kvmsvr images]# virsh define /root/rhel7_muban.xml  
定义域 rh7_node1 ( 从 /root/rhel7_muban.xml )
```



新虚拟机的交付使用

- 正常运行快建的新虚拟机
 - 检查已有装好的操作系统，登入后与模板机一样
 - 用户自行修改主机名/IP地址等参数

```
[root@kvmsvr images]# virsh list --all
```

Id	名称	状态
----	----	----

-	rhel-207	关闭
---	----------	----

-	rhel-7	关闭
---	--------	----

-	rh7_node1	关闭
---	-----------	----

.. ..



案例6：快建新虚拟机

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



总结和答疑

总结和答疑

离线访问虚拟机

配置虚拟机系统有奇招

如何挂载虚拟盘

SSH免密码登录

问题现象

故障分析及排除

离线访问虚拟机

配置虚拟机系统有奇招

- 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
- !!! 需要注意 SELinux 机制的影响



如何挂载虚拟盘

- 基本用法

`guestmount -a 虚拟机磁盘路径 -i /挂载点`

```
[root@kvmsvr ~]# mkdir /mnt/kdisk
```

```
[root@kvmsvr ~]# guestmount -a node1.qcow2 -i /mnt/kdisk
```

```
[root@kvmsvr ~]# ls /mnt/kdisk
```

```
bin      home    media   opt      sbin     tmp
```

```
boot     lib     misc    proc     selinux  usr
```

```
...
```

