**KVM构建及管理**

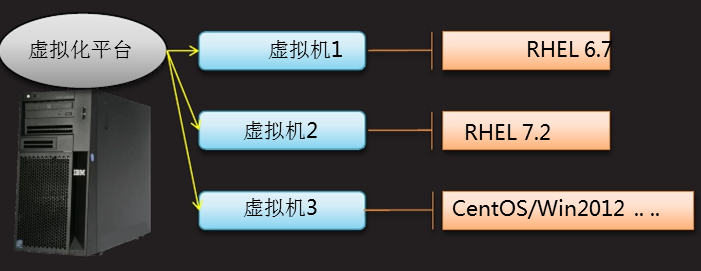
搭建KVM服务器

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、硬件辅劣（CPU）



虚拟化主要厂商及产品

|  |  |
| --- | --- |
| 系列 | PC/服务器版代表 |
| vmware | VMwareWorkstation、vSphere |
| microsoft | VirtualPC、Hyper-V |
| redhat | KVM、RHEV |
| citrix | Xen |
| oracle | OracleVMVirtualBox |

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/

enable=1

gpgcheck=0

yum repolist

安装虚拟化服务器平台

KVM/QEMU/LIBVIRTD

KVM是linux内核的模块，它需要CPU的支持，采用硬件辅助虚拟化技术intel-vt,amd-v,内存的相关如intel的EPT和AMD的RVI技术

QEMU是一个虚拟化仿真工具，通过ioctl与内核KVM交互完成对硬件的虚拟化支持

Libvirt是一个对虚拟化管理的接口和工具，提供用户端程序virsh,virt-install,virt-manager,virt-view与用户交互

必备软件

qemu-kvm #为kvm提供底层支持

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

虚拟化的组成

虚拟机的组成

内核虚拟化模块（KVM）

系统设备仿真（QEMU，虚拟硬件）

虚拟机惯例程序（LIBVIRT）

一个XML文件（虚拟机配置声明文件）

位置/etc/libvirt/qemu

一个磁盘镜像文件（虚拟机的硬盘）

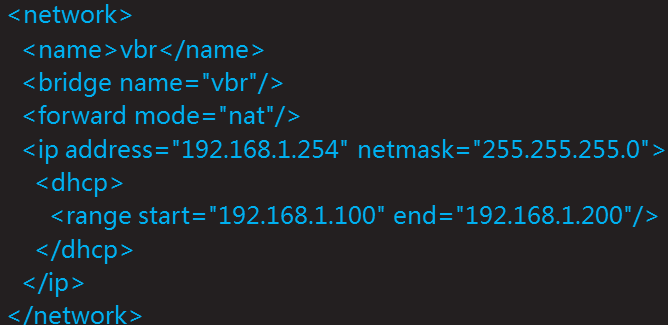
位置/var/lib/libvirt/images

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



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

去除net-undefine virt网络配置

启动virt网络net-start

停止virt网络net-destroy

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

初始化系统并制作模版

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，去除UUID，MAC地址等

重定向console=tty0 console=ttyS0,115200n8

关机并备份xml文件

去除quest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#connect qemu:///system（默认选项）

virsh -c qemu:///system

连接远程

virsh#connect qemu+ssh://user@ip地址:port/system

virsh -c qemu+ssh://user@ip地址:port/system



使用virt-manager客户端管理工具

virsh-manager也可以通过add connection管理其他机器上的虚拟机，一般通过“remote tunnel over ssh”就可以了

需要注意的事virt-manager需要使用ssh免密码登录，如果没有免密码登录需要单独安装python的ssh相关模块，这里使用部署Key的方法解决

**virsh控制工具**

操作虚拟机

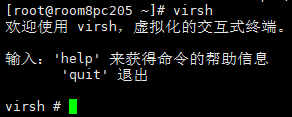
virsh命令工具介绍

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

进入virsh命令终端或者命令行都可以操作

格式：virsh 控制指令 [虚拟机名称] [参数]



查看虚拟化信息

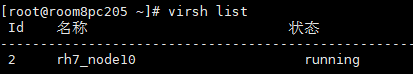
查看KVM节点（服务器）信息

virsh nodeinfo

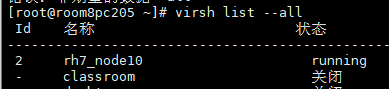


列出虚拟机

virsh list [--all]

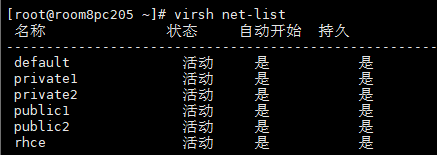


all会把关机状态的也列出来



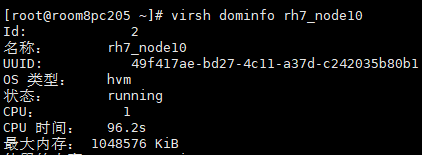
列出虚拟网络

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





**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

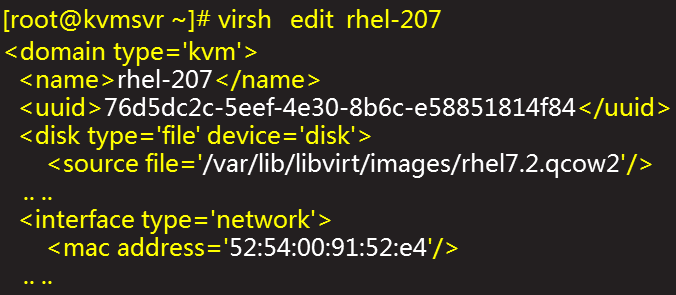
备份：virsh dumpxml 虚拟机名 > 虚拟机名.xml

编辑虚拟机设置

对虚拟机的设置进行调整

编辑：virsh edit 虚拟机名

若修改name、memory、disk、network，可自动保存为新的虚拟机配置



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

删除虚拟机

必要时可以去除多余的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MAC地址冲突

virsh undefine 虚拟机名

**镜像管理（默认的在/var/lib/libvirt/images/下面）**

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

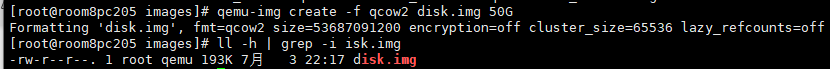
|  |  |  |
| --- | --- | --- |
| 特点/类型 | RAW | QCOW2 |
| KVM默认 | 否 | 是 |
| I/O效率 | 高 | 较高 |
| 占用空间 | 大 | 小 |
| 压缩 | 不支持 | 支持 |
| 后端盘复用 | 不支持 | 支持 |
| 快照 | 不支持 | 支持 |

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

qemu-img create -f qcow2 disk.img 50G

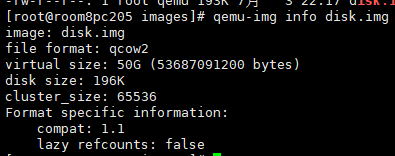


当使用量变大，它会变大，最大不会超过预设值50G的大小

查看镜像盘文件的信息

qemu-img info 磁盘路径

qemu-img info disk.img



-b使用后端模版文件

qemu-img create -f qcow2 -b 模版名 新磁盘名 size

size不指定，使用模版大小

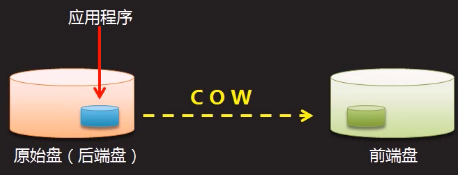
qemu-img create -b disk.img -f qcow2 disk1.img



QCW2技术

cow技术原理

copy on write，写时复制



直接映射原始盘的数据

当原始盘的旧数据由修改时，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导原始盘

原理：用户请求前端盘，如果没有数据，前端盘会转发到后端盘，如果要进行写操作，后端盘会把数据写到前端盘，用户对前端盘修改，不会保存到后端盘

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

必要时可将虚拟机回复到指定的快照

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