

- 云计算
 - KVM安装
 - 安装条件
 - 安装系统和软件
 - 验证安装
 - 透传
 - KVM使用
 - 镜像管理
 - 安装虚拟机
 - 字符
 - 图形
 - 附加配置
 - 管理工具
 - virsh
 - virt-manager

云计算

KVM安装

安装条件

VT-x

BIOS

Intel(R) Virtualization Tech [Enabled]

```
cat /proc/cpuinfo | grep -e vmx -e nx -e svm
grep vmx /proc/cpuinfo
grep svm /proc/cpuinfo
egrep '(vmx|svm)' /proc/cpuinfo
```

模块

```
lsmod | grep kvm_intel
modprobe -r kvm_intel
modprobe kvm_intel
```

安装系统和软件

- 图型
 - 工作站
 - 附加工具
 - 开发工具
- 字符

```
yum install kvm virt-manager libvirt libvirt-python python-virtinst libvirt-client qemu-kvm qemu-img
yum install kvm virt-manager libvirt*
yum install qemu-kvm libvirt virt-install virt-manager
yum install spice-protocol spice-client spice-server
```

```
yum install qemu-kvm* virt-* libvirt*
yum install spice*
systemctl [ start | status | enable ] libvirtd
```

qemu-kvm -- 该软件包主要包含KVM内核模块和基于KVM重构后的QEMU模拟器。KVM模块作为整个虚拟化环境的核心工作在系统空间，负责CPU和内存的调度。QEMU作为模拟器工作在用户空间，负责虚拟机I/O模拟。

qemu-img -- 主要用来QEMU磁盘镜像的管理，如新建一块磁盘镜像给虚拟机。

libvirt -- 提供Hypervisor和虚拟机管理的API。

libvirt-client -- KVM客户端命令行管理工具virsh，负责虚拟机的启动、停止和管理等。

libvirt-daemon -- libvirtd守护进程，作为客户端管理工具跟Hypervisor和虚拟机之间的桥梁。

libvirt-daemon-driver-xxx -- 从名字来看属于libvirtd服务的驱动文件，作为libvirtd服务跟Hypervisor不同对象(如qemu模拟器，网络，存储等)间的接口。

bridge-utils -- 网桥管理工具包，负责桥接网络的创建、配置和管理等工作。

virt-install -- 创建和克隆虚拟机的命令行工具包。

virt-manager -- 图形界面的KVM管理工具。<https://virt-manager.org/>

验证安装

```
virsh -c qemu:///system list
```

透传

KVM on KVM

Vmware	2	物理	软件
KVM	N	物理	物理

```
modprobe -r kvm_intel
modprobe kvm_intel nested=1

echo -e "options kvm_intel nested=1" > /etc/modprobe.d/kvm-nested.conf

cat /sys/module/kvm_intel/parameters/nested
```

KVM使用

镜像管理

```
qemu-img create -f qcow2 /var/lib/libvirt/images/os.qcow2 10G
qemu-img resize os.img +2G
qemu-img convert os.img vmware.vmdk
qemu-img info vmware.vmdk
```

安装虚拟机

字符

```
qemu-img create -f qcow2 /var/lib/libvirt/images/ac260c73.qcow2 260G
```

```
virt-install \
--connect qemu:///system \

--name ac260c73 \
--memory 4096 \

--cpu host \
--cpu host-passthrough \
--vcpus 4,maxvcpus=8,sockets=2,cores=1,threads=2 \

--disk size=8 \
--disk path=/var/lib/libvirt/images/ac260c73.qcow2 \
--disk /var/lib/libvirt/images/ac260t73.qcow2,size=260,format=qcow2,bus=virtio,sparse \

--cdrom CentOS-7-x86_64-Minimal-1708.iso \
--location /var/lib/libvirt/images/CentOS-7-x86_64-DVD-1611.iso \
--location http://example.com/path/to/os \

--os-type linux \
--os-variant rhel7 \
--accelerate \

--network bridge=virbr0 \
--network bridge=br0 \

--graphics none \
--console pty,target_type=serial \
--extra-args 'console=ttyS0,115200n8 serial'

--import

--network=bridge:br0
--pxe

--initrd-inject /path/to/ks.cfg
--extra-args="ks=file:/ks.cfg console=tty0 console=ttyS0,115200n8"

--extra-args="ip=192.168.1.2::192.168.1.1:255.255.255.0:test.example.com:eth0:none"
```

图形

```
--graphics spice \
--graphics vnc \
--graphics vnc,listen=0.0.0.0,port=5910 \
--input tablet,bus=usb \
--accelerate \

--os-type linux          类型
--os-variant rhel7        型号
--accelerate              提高性能
--input tablet,bus=usb     鼠标
--console pty,target_type=serial  控制端口
--extra-args 'console=ttyS0,115200n8 serial'  文本输出
--network                 网络
```

附加配置

```
<domain>
  <devices>
    <input type='tablet' bus='usb' />
    <input type='mouse' bus='ps2' />
    <input type='keyboard' bus='ps2' />
  </devices>
</domain>
```

```
<domain>
  <cpu mode='host-passthrough'>
    <topology sockets='1' cores='1' threads='2' />
  </cpu>
</domain>
```

管理工具

virsh

```
uri

list
list --all

console

autostart ci
start ci
stop ci
destroy ci
suspend ci
reboot ci

define ci.xml
undefind ci

dumpxml ci
edit ci

dominfo ci
domstate ci
domstats ci
domid ci
domname ci

cpu-stats ci
vcpucount ci
vcpuinfo ci

dommemstat ci

domblkstat ci
domblklist ci
domblkinfo ci vda

domiflist ci
domifstat ci vnet3
domif-getlink ci vnet3

nodeinfo
nodememstats
nodecpustats
nodecpumap
node-memory-tune
hostname
cpu-models x86_64

pool-list
pool-info kvm
pool-dumpxml kvm
```

virt-manager