

Works®

Technical topics

Evaluation software

Community

Events

My home

Forums

Blogs

Communities

Profiles

Podcasts

Wikis

Activities

IBM Champion program

Search developerWorks



IBM® Bluemix™

A CLOUD PLATFORM
FOR THE WORLD'S IDEAS

Start building for free

My Blogs

Public Blogs

My Updates

This Blog

Search

IBM中国Linux与虚拟化实验室

Log in
to participate☐ IBM中国Linux与虚拟化实验室

Overview

Recent Updates

QEMU1: 使用QEMU创建虚拟机

KVMLTCChina | May 21 | Visits (1191)

g+1 0

赞 0

Share

Tweet 0

About this blog

IBM中国Linux与虚拟化实验室成立于2011年，主要致力于以Linux为核心的系统软件和以KVM为核心的虚拟化及云计算软件的开发、测试、及性能优化等工作。目前参与的社区包括：Linux内核，包

Links

[Linux and Open Virtualization](#)

QEMU1: 使用QEMU创建虚拟机

[Status Updates](#)[Members](#)[Blog](#)[Wiki](#)[Bookmarks](#)[Files](#)[Forums](#)[Ideation Blog](#)

一、QEMU简介

QEMU是一款开源的模拟器及虚拟机监管器(Virtual Machine Monitor, VMM)。QEMU主要提供两种功能给用户使用。一是作为用户态模拟器，利用动态代码翻译机制来执行不同于主机架构的代码。二是作为虚拟机监管器，模拟全系统，利用其他VMM(Xen, KVM, etc)来使用硬件提供的虚拟化支持，创建接近于主机性能的虚拟机。

用户可以通过不同Linux发行版所带有的软件包管理器来安装QEMU。如在Debian系列的发行版上可以使用下面的命令来安装：

```
sudo apt-get install qemu
```

或者在红帽系列的发行版上使用如下命令安装：

```
sudo yum install qemu -y
```

除此之外，也可以选择从源码安装。

1. 获取QEMU源码

可以从[QEMU官网](#)上下载QEMU源码的tar包，以命令行下载2.0版本的QEMU为例：

```
$wget http://wiki.qemu-project.org/download/qemu-2.0.0.tar.bz2
```

```
$tar xjvf qemu-2.0.0.tar.bz2
```

如果需要参与到QEMU的开发中，最好使用Git获取源码：

```
$git clone git://git.qemu-project.org/qemu.git
```

Tags

[Find a Tag](#)

[automation](#) [cim](#) [could](#) [crash](#) [daytrader](#)
[ext2](#) [kerne](#) [kimchi](#) [kvm](#) [libvirt](#) [linux](#)
[networking](#) [openstack](#) [ovirt](#) [ovirt-node](#)
[paging](#) [random](#) [rpm](#) [spec](#) [storage](#) [test](#)
[testing](#) [ubuntu](#) [uto](#) [virtualization](#) [vmware](#)
[虚拟化](#), [虚拟机监控层](#) [软件包管理](#)

[Cloud](#) | [List](#)

2. 编译及安装

获取源码后，可以根据需求来配置和编译QEMU。

```
$cd qemu-2.0.0 //如果使用的是git下载的源码，执行cd qemu
$./configure --enable-kvm --enable-debug --enable-vnc --
enable-werror --target-list="x86_64-sofmmu"
$make -j8
$sudo make install
```

configure脚本用于生成Makefile，其选项可以用`./configure --help`查看。这里使用到的选项含义如下：

--enable-kvm：编译KVM模块，使QEMU可以利用KVM来访问硬件提供的虚拟化服务。

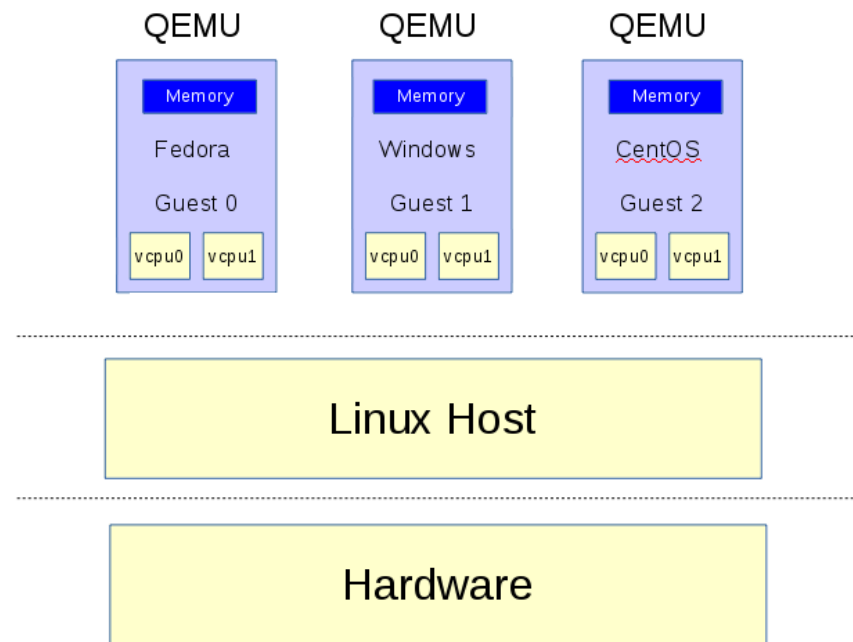
--enable-vnc：启用VNC。

--enable-werror：编译时，将所有的警告当作错误处理。

--target-list：选择目标机器的架构。默认是将所有的架构都编译，但为了更快的完成编译，指定需要的架构即可。

二、基本原理

QEMU作为系统模拟器时，会模拟出一台能够独立运行操作系统的虚拟机。如下图所示，每个虚拟机对应主机(Host)中的一个QEMU进程，而虚拟机的vCPU对应QEMU进程的一个线程。



QEMU结构图

系统虚拟化最主要是虚拟出CPU、内存及I/O设备。虚拟出的CPU称之为vCPU，QEMU为了提升效率，借用KVM、XEN等虚拟化技术，直接利用硬件对虚拟化的支持，在主机上安全地运行虚拟机代码(需要硬件支持)。虚拟机vCPU调用KVM的接口来执行任务的流程如下(代码源自QEMU开发者Stefan的[技术博客](#)):

```
open("/dev/kvm")
ioctl(KVM_CREATE_VM)
ioctl(KVM_CREATE_VCPU)
for (;;) {
    ioctl(KVM_RUN)
    switch (exit_reason) {
        case KVM_EXIT_IO: /* ... */
        case KVM_EXIT_HLT: /* ... */
```

```
}  
}
```

QEMU发起ioctl来调用KVM接口，KVM则利用硬件扩展直接将虚拟机代码运行于主机之上，一旦vCPU需要操作设备寄存器，vCPU将会停止并退回到QEMU，QEMU去模拟出操作结果。

虚拟机内存会被映射到QEMU的进程地址空间，在启动时分配。在虚拟机看来，QEMU所分配的主机上的虚拟地址空间为虚拟机的物理地址空间。

QEMU在主机用户态模拟虚拟机的硬件设备，vCPU对硬件的操作结果会在用户态进行模拟，如虚拟机需要将数据写入硬盘，实际结果是将数据写入到了主机中的一个镜像文件中。

三、创建及使用虚拟机

1. 命令行创建及启动虚拟机

成功安装QEMU之后便可创建自己的虚拟机。具体步骤如下：

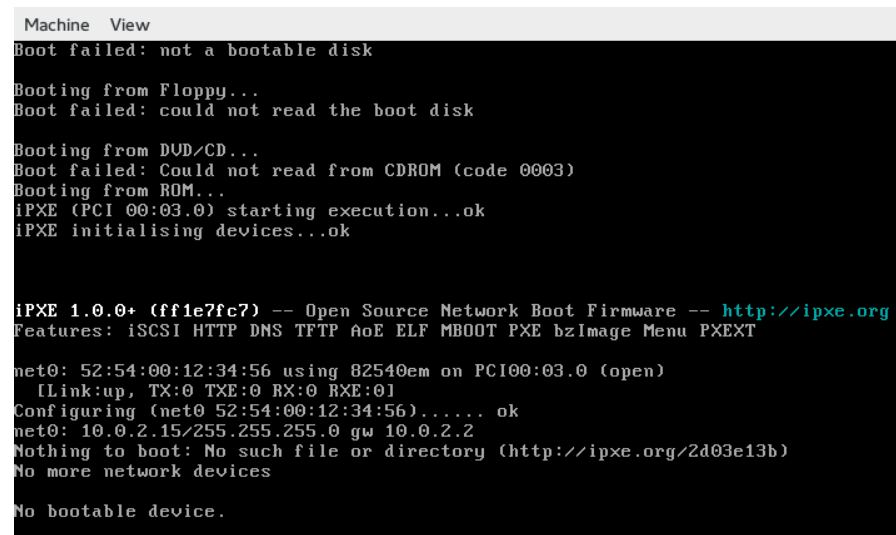
1, 使用qemu-img创建虚拟机镜像。虚拟机镜像用来模拟虚拟机的硬盘，在启动虚拟机之前需要创建镜像文件。

```
[kelvin@kelvin tmp]$ qemu-img create -f qcow2 fedora.img 10G  
Formatting 'fedora.img', fmt=qcow2 size=10737418240  
encryption=off cluster_size=65536 lazy_refcounts=off  
[kelvin@kelvin tmp]$ ls  
fedora.img
```

-f选项用于指定镜像的格式，qcow2格式是QEMU最常用的镜像格式，采用来写时复制技术来优化性能。fedora.img是镜像文件的名称，10G是镜像文件大小。镜像文件创建完成后，可使用qemu-system-x86来启动x86架构的虚拟机：

```
$qemu-system-x86_64 fedora.img`
```

此时会弹出一个窗口来作为虚拟机的显示器，显示内容如下：



```
Machine View
Boot failed: not a bootable disk

Booting from Floppy...
Boot failed: could not read the boot disk

Booting from DVD/CD...
Boot failed: Could not read from CDROM (code 0003)
Booting from ROM...
iPXE (PCI 00:03.0) starting execution...ok
iPXE initialising devices...ok

iPXE 1.0.0+ (ff1e7fc7) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: iSCSI HTTP DNS TFTP AoE ELF MBOOT PXE bzImage Menu PXEXT

net0: 52:54:00:12:34:56 using 82540em on PCI00:03.0 (open)
  (Link:up, TX:0 TXE:0 RX:0 RXE:0)
Configuring (net0 52:54:00:12:34:56)..... ok
net0: 10.0.2.15/255.255.255.0 gw 10.0.2.2
Nothing to boot: No such file or directory (http://ipxe.org/2d03e13b)
No more network devices

No bootable device.
```

QEMU虚拟机显示器输出

因为fedora.img中并未给虚拟机安装操作系统，所以会提示“No bootable device”，无可启动设备。

2. 准备操作系统镜像。

可以从不同Linux发行版的官方网站上获取安装镜像，以fedora20为例：

```
`[kelvin@kelvin tmp]$ wget http://ftp6.sjtu.edu.cn/fedora/linux/releases/20/Live/x86\_64/Fedora-Live-Desktop-x86\_64-20-1.iso`
```

3, 检查KVM是否可用。

QEMU使用KVM来提升虚拟机性能，如果不启用KVM会导致性能损失。要使用KVM，首先要检查硬件是否有虚拟化支持：

```
`[kelvin@kelvin ~]$ grep -E 'vmx|svm' /proc/cpuinfo`
```

如果有输出则表示硬件有虚拟化支持。其次要检查kvm模块是否已经加载：

```
[kelvin@kelvin ~]$ lsmod | grep kvm  
kvm_intel 142999 0  
kvm 444314 1 kvm_intel
```

如果kvm_intel/kvm_amd、kvm模块被显示出来，则kvm模块已经加载。最好要确保qemu在编译的时候使能了KVM，即在执行configure脚本的时候加入了--enable-kvm选项。

4, 启动虚拟机安装操作系统。

执行下面的命令启动带有cdrom的虚拟机：

```
[kelvin@kelvin tmp]$ qemu-system-x86_64 -m 2048 -enable-kvm  
fedora.img -cdrom ./Fedora-Live-Desktop-x86_64-20-1.iso
```

-m 指定虚拟机内存大小，默认单位是MB，-enable-kvm使用KVM进行加速，-cdrom添加fedora的安装镜像。可在弹出的窗口中操作虚拟机，安装操作系统，安装完成后重起虚拟机便会从硬盘(fedora.img)启动。之后再启动虚拟机只需要执行：

```
[kelvin@kelvin tmp]$ qemu-system-x86_64 -m 2048 -enable-kvm  
fedora.img
```

即可。

2. 图形界面创建及启动虚拟机

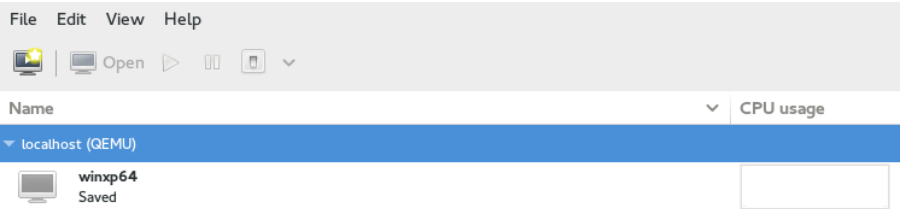
命令行启动虚拟机比较繁琐，适合开发者，但对于用户来说，采用图形界面管理虚拟机则更为方便。采用图形界面管理QEMU虚拟机需要安装virt-manager，红帽系列的发行版只需要执行命令：

```
$sudo yum install virt-manager -y
```

安装完成后用root用户启动virt-manager：

```
$su -  
#virt-manager
```

启动后的界面如下图所示：



virt-manager界面

点击左上角电脑图标即可创建虚拟机。按照步骤操作即可完成对虚拟机的创建。

[Add a Comment](#) | [More Actions](#)

Comments (0)

[Add a Comment](#) | [More Actions](#)

[Previous Entry](#) | [Main](#) | [Next Entry](#)

About
Help
Contact us
Submit content

Feeds
Newsletters
Follow
Like

Report abuse
Terms of use
Third party notice
IBM privacy
IBM accessibility

Faculty
Students
Business Partners

