

目 录

1	kvm 环境的搭建	2
1.1	搭建硬件环境	2
1.1.1	检查处理器是否支持 VT 技术	2
1.1.2	设置 BIOS	2
1.2	安装 KVM	2
1.2.1	下载 KVM 源代码	2
1.2.2	配置 KVM	3
1.2.3	编译 KVM	4
1.2.4	安装 KVM	5
1.2.5	加载 kvm 和 kvm_intel 模块	5
1.3	安装 qemu-kvm	5
1.3.1	下载 qemu-kvm 源代码	5
1.3.2	配置 qemu-kvm	6
1.3.3	编译 qemu-kvm	6
1.3.4	安装 qemu-kvm	6
1.4	安装客户机	7
1.4.1	创建镜像文件	7
1.4.2	安装客户机	7
1.4.3	查看客户机	7
1.5	启动 KVM 客户机	8

1 kvm 环境的搭建

首先声明，这个仅在 ubuntu16.04 下配置过，配置日期为 2016.12.16。

1.1 搭建硬件环境

在 x86_64 架构的 INTEL 处理器中，KVM 必需的硬件虚拟化扩展为 INTEL 的虚拟化技术 (INTEL VT)。首先处理器要在硬件上支持 VT 技术。只有在 BIOS 中将 VT 打开，才可以使用 KVM。

1.1.1 检查处理器是否支持 VT 技术

在 linux 系统中，可以通过/proc/cpuinfo 文件中的 CPU 特性标志来查看 CPU 是否支持 VT 技术。如果 CPU 支持 VT 技术，那么文件中的 flags 就包含"vmx"。

这里使用 grep 命令来查看/proc/cpuinfo 中是否包含"vmx"。首先介绍一下 grep 命令。

grep 全称是 Global Regular Expression Print，使用正则表达式搜索文本，并把匹配的行打印出来
grep [pattern] [file] 用于在 file 中查找符合 pattern 的文本，并打印出来
更多的 grep 信息，可以通过 man grep 查看。

使用如下命令就可以查看/proc/cpuinfo 中是否包含了"vmx"：

```
1 grep "vmx" /proc/cpuinfo
```

如果在/proc/cpuinfo 中包含了"vmx"，说明 CPU 支持 VT 技术，否则另外设置。

1.1.2 设置 BIOS

如果 CPU 目前不支持 VT 技术，就需要设置 BIOS 中相应的选项。

VT 的选项一般在 BIOS "Advanced" 栏目的"CPU Configuration" 中，它由"Intel Virtualization Technology" 或"Intel VT" 标识。找到该标识后，将其设为 [Enabled] 就可以了。

1.2 安装 KVM

1.2.1 下载 KVM 源代码

使用如下命令即可下载 KVM 源代码：

```
1 git clone https://git.kernel.org/pub/scm/virt/kvm/kvm.git
```

1.2.2 配置 KVM

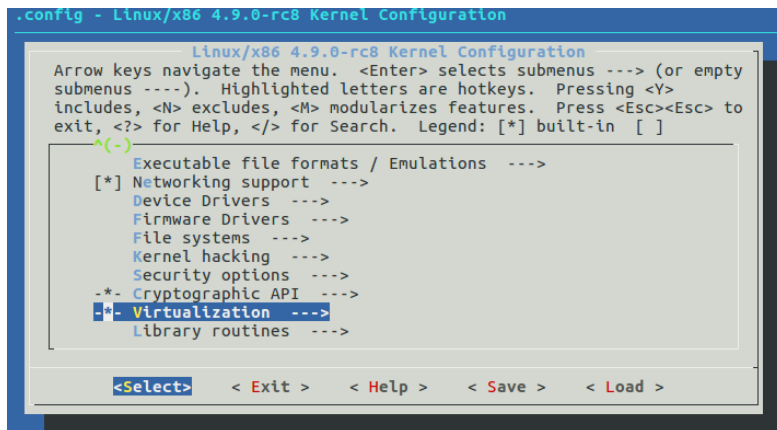
在此我们使用 make menuconfig 对 KVM 进行配置。首先安装 ncurses 库：

```
1 sudo apt-get install libncurses5-dev
```

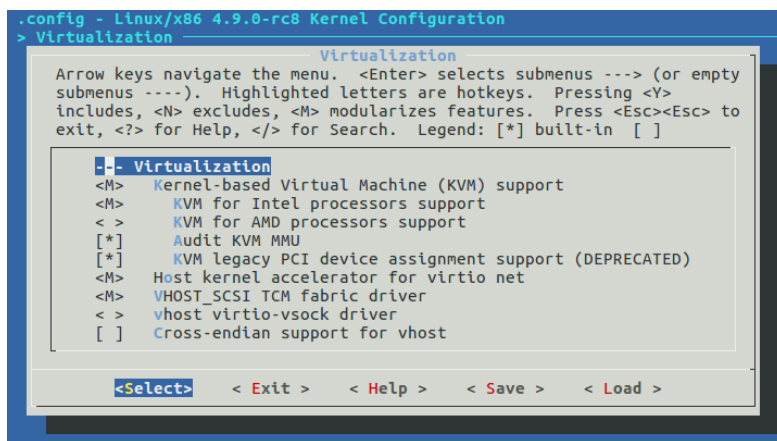
然后在 KVM 文件中打开 terminal，然后输入如下命令：

```
1 make menuconfig
```

然后选择”Virtualization”，如下所示：



进入以后，进行如下图的配置：



在配置完成后，将在 KVM 文件夹下生成 .config 文件。如果想确保 KVM 相关的配置正确，可以检查 .config 文件的相关配置项。查看命令如下：

```
1 vi .config
2 /CONFIG_HAVE_KVM
```

光标跳到对应的配置项后，查看是否如下面几个配置：

```
1 CONFIG_HAVE_KVM=y
2 CONFIG_HAVE_KVM_IROCHIP=y
3 CONFIG_HAVE_KVM_IRQFD=y
4 CONFIG_HAVE_KVM_IRQ_ROUTING=y
5 CONFIG_HAVE_KVM_ENENTFD=y
6 CONFIG_KVM_MMIO=y
7 CONFIG_KVM_ASYNC_PF=y
8 CONFIG_HAVE_KVM_MSI=y
9 CONFIG_HAVE_KVM_CPU_RELAX_INTERCEPT=y
10 CONFIG_KVM_VFIO=y
11 CONFIG_KVM_GENERIC_DIRTYLOG_READ_PROTECT=y
12 CONFIG_KVM_COMPAT=y
13 CONFIG_HAVE_KVM_IRQ_BYPASS=y
14 CONFIG_VIRTUALIZATION=y
15 CONFIG_KVM=m
16 CONFIG_KVM_INTEL=m
```

1.2.3 编译 KVM

KVM 的编译包括三个步骤：

- Kernel 的编译。
- bzImage 的编译。
- 内核模块的编译。

第一步是编译 kernel。因为 kernel 包含了 openssl 的库，所以在编译之前，需要先安装 openssl 和相关的库，命令如下：

```
1 sudo apt-get install openssl
2 sudo apt-get install libssl-dev
```

接下来，就可以直接开始编译 kernel，命令如下：

```
1 make vmlinux
```

第二步是编译 bzImage。命令如下：

```
1 make bzImage
```

第三步是编译内核的模块。命令如下：

```
1 make modules
```

1.2.4 安装 KVM

KVM 的安装包括两个步骤：

- module 的安装。
- kernel 与 initramfs 的安装。

首先安装 module，命令如下：

```
1 sudo make modules_install
```

然后安装 kernel 和 initramfs，命令如下：

```
1 sudo make install
```

最后重启系统，命令如下：

```
1 reboot
```

1.2.5 加载 kvm 和 kvm_intel 模块

使用如下命令即可加载 kvm 和 kvm_intel 模块：

```
1 modprobe kvm
2 modprobe kvm_intel
```

执行 lsmod 指令，会列出所有已载入系统的模块。然后使用 grep，可以查看是否存在 kvm 和 kvm_intel 模块，命令如下所示：

```
1 lsmod | grep kvm
```

如果 terminal 输出有 kvm 和 kvm_intel 有关的信息，就说明 kvm 和 kvm_intel 模块加载成功了。

1.3 安装 qemu-kvm

1.3.1 下载 qemu-kvm 源代码

使用如下命令即可下载 qemu-kvm 源代码：

```
1 git clone https://git.kernel.org/pub/scm/virt/kvm/qemu-kvm.git
```

1.3.2 配置 qemu-kvm

首先安装 glib，命令如下：

```
1 sudo apt-get install libglib2.0-dev
```

使用如下命令即可配置 qemu-kvm：

```
1 ./configure
```

这里介绍一下./configure 命令：

./configure 会根据当前系统环境和指定参数生成 makefile 文件，为下一步的编译做准备。
可以通过在 configure 后加上参数来对安装进行控制。
比如：./configure - prefix=/usr
意思是将该软件安装在 /usr 下面，执行文件就会安装在 /usr/bin

随后，为了不让编译器把警告当作错误处理，应该在 Makefile 中任意一行添加如下代码：

```
1 QEMU_CFLAGS += -w
```

1.3.3 编译 qemu-kvm

使用如下命令即可编译 qemu-kvm：

```
1 make
```

编译的时候可能遇到如下错误：

```
1 ./qemu-options.texi:1526: unknown command 'list'
2 ./qemu-options.texi:1526: table requires an argument: the formatter for @item
3 ./qemu-options.texi:1526: warning: @table has text but no @item
```

这个需要先对源代码打补丁，首先下载补丁，地址如下：

```
1 http://patchwork.ozlabs.org/patch/222212/raw/
```

将补丁文件移动到 qemu-kvm 源代码根目录，然后使用如下命令：

```
1 # fix.patch 是补丁的文件名
2 patch -p1 < fix.patch
```

然后再次编译，命令如下：

```
1 make
```

1.3.4 安装 qemu-kvm

使用如下命令即可安装 qemu-kvm:

```
1 sudo make install | tee make-install.log
```

这里介绍一下 tee 命令:

功能说明: 读取标准输入的数据, 并将其内容输出成文件。

语法: tee [-ai][--help][--version][文件]

参数:

- a 附加到既有文件的后面, 而非覆盖它。
- i 忽略中断信号。
- help 在线帮助。
- version 显示版本信息。

1.4 安装客户机

1.4.1 创建镜像文件

安装客户机之前, 我们需要创建一个镜像文件来存储客户机中的系统和文件。镜像文件将作为客户机的硬盘, 将客户机的操作系统安装在其中。

首先, 使用如下命令行创建一个 8GB 大小的镜像文件 ubuntu1604.img:

```
1 dd if=/dev/zero of=ubuntu1604.img bs=1M count=8192
```

这里介绍一下 dd 命令:

功能说明: 把指定的输入文件拷贝到指定的输出文件中。

语法: dd [选项]

参数:

- if=输入文件
- of=输出文件
- bs=bytes 同时设置读/写缓冲区的字节数
- count=blocks 只拷贝输入的 blocks 块

1.4.2 安装客户机

在联网的情况下, 使用如下命令安装客户机:

```
1 qemu-system-x86_64 -m 2048 -smp 4 -boot order=cd -hda ubuntu1604.img -cdrom ubuntu
  -16.04.iso -vnc 127.0.0.1:2
```

这里介绍一下 qemu-system-x86_64 的参数, 如下所示:

```
1 -m 2048 是给客户机分配2048MB内存
2 -smp 4 是给客户机分配4个CPU
```

```
3  -boot order=cd是指定系统的启动顺序为光驱、硬盘
4  -hda ubuntu1604.img是分配给客户机的IDE硬盘
5  -cdrom ubuntu-16.04.iso是分配给客户机的光驱
6  -vnc 127.0.0.1:2使用vnc方式显示客户机，端口为127.0.0.1:2
```

1.4.3 查看客户机

因为这里使用 vnc 方式显示客户机，所以我们需要先安装 vncserver 和 vncviewer，命令如下：

```
1  sudo apt-get install vncserver
2  sudo apt-get install vncviewer
```

然后使用 vncviewer 查看客户机，命令如下：

```
1  vncviewer 127.0.0.1:2
```

1.5 启动 KVM 客户机

安装好系统之后，就可以使用镜像文件来启动并登陆到自己安装的系统之中。在联网的情况下，使用如下命令即可启动一个 KVM 的客户机：

```
1  qemu-system-x86_64 -m 2048 -smp 4 -hda ubuntu1604.img -vnc 127.0.0.1:2
```

使用如下命令可以查看 KVM 客户机：

```
1  vncviewer 127.0.0.1:2
```