

实验 8 Linux内核配置编译安装实验

实验目的

1.掌握配置和编译 Linux 内核的基本方法

实验内容

1.查看当前ubuntu server 1604内核及硬件信息

2.下载、解压linux 内核

3.配置 linux 内核

4.编译和安装 linux 内核

实验前提

请先克隆 ubuntu server 1604 虚拟机，并使用克隆虚拟机安装新kernel，以防原ubuntu linux 失效。

实验步骤

一.查看当前ubuntu server 1604内核及硬件信息

1.使用下列命令查看当前linux的内核版本，并记录结果。

```
uname -a
```

例如,下列命令和结果显示了当前linux使用的内核版本为 4.4.0-116-generic:

```
eo@ubuntu:~/Downloads/linux-5.2.11$ uname -a
```

```
Linux ubuntu 4.4.0-116-generic #140-Ubuntu SMP Mon Feb 12 21:23:04 UTC 2018 x86_64 x86_64 x86_64 GNU/Linux
```

2.学习使用下列命令查看当前linux运行的硬件环境信息，任选三个，记住它。

显示你的硬件以及所需的模块

<p class="mume-header " id="显示你的硬件以及所需的模块"></p>

lshw

查看硬件信息，包括bios、cpu、内存等 信息

<p class="mume-header " id="查看硬件信息包括bios-cpu-内存等-信息"></p>

dmidecode

查看硬件信息

<p class="mume-header " id="查看硬件信息"></p>

dmesg

lspci (比cat /proc/pci更直观)

<p class="mume-header " id="lspci-比cat-procpci更直观"></p>

lspci -v

查看 /proc文件夹下的若干文件:

<p class="mume-header " id="查看-proc文件夹下的若干文件"></p>

查看CPU信息

<p class="mume-header " id="查看cpu信息"></p>

cat /proc/cpuinfo

查看内存信息

<p class="mume-header " id="查看内存信息"></p>

cat /proc/meminfo

查看键盘和鼠标

<p class="mume-header " id="查看键盘和鼠标"></p>

cat /proc/bus/input/devices

查看板卡信息

<p class="mume-header " id="查看板卡信息"></p>

cat /proc/pci

查看USB设备

<p class="mume-header " id="查看usb设备"></p>

cat /proc/bus/usb/devices

查看各设备的中断请求(IRQ)

<p class="mume-header " id="查看各设备的中断请求irq"></p>

cat /proc/interrupts

3.查看当前linux内核的配置文件.

注意, config 后的版本可能有变化。

<p class="mume-header " id="注意config-后的版本可能有变化"></p>

```
sudo less /boot/config-4.4.0-116-generic
```

二.下载、解压linux 内核

1.查找并且下载一份内核源代码

Linux受GNU通用公共许可证（GPL）保护，其内核源代码是完全开放的。现在很多Linux的网站都提供内核代码的下载。

推荐使用Linux的官方网站：<http://www.kernel.org>。在这里你可以找到所有的内核版本。

考虑到下载需要时间，我们已将 Linux 5.2.11 kernel 下载到 ubuntu server 1604 虚拟机的 ~/Downloads/ 中,文件名为linux-5.2.11.tar.xz。

如果要自行下载最新内核版本，可以在虚拟机中~/Downloads/ 目录下运行下列命令：

```
leo@ubuntu:~# cd Downloads/
```

```
leo@ubuntu:~/Downloads$ wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.3.4.tar.xz
```

2.运行下列命令解压kernel包。

```
leo@ubuntu:~/Downloads$ tar -xvf linux-5.2.11.tar.xz
```

3.建立待编译内核的工作目录

```
leo@ubuntu:~# cd Downloads/
```

```
leo@ubuntu:~/Downloads$ mkdir -p mylinux/kernel/
```

三.配置 linux 内核

1.如果不是第一次编译，有必要将内核源代码树置于一种完整和一致的状态。因此推荐执行命令：

```
leo@ubuntu:~/Downloads$ cd linux-5.2.11
leo@ubuntu:~/Downloads/linux-5.2.11$ make mrproper
```

make mrproper命令会删除所有的编译生成文件、内核配置文件(.config文件)和各种备份文件，所以几乎只在第一次执行内核编译前才用这条命令。

make clean命令则是用于删除大多数的编译生成文件，但是会保留内核的配置文件.config，还有足够的编译支持来建立扩展模块。所以你若只想删除前一次编译过程的残留数据，只需执行**make clean**命令。

2.将当前ubuntu server 16.04 的内核配置文件拷贝到内核编译目录。

```
leo@ubuntu:~/Downloads$ cd linux-5.2.11
leo@ubuntu:~/Downloads/linux-5.2.11$ cp -v /boot/config-$(uname -r) .config
```

3.使用下列命令，根据默认模板生成最初的配置文件.config

```
leo@ubuntu:~/Downloads$ cd linux-5.2.11
leo@ubuntu:~/Downloads/linux-5.2.11$ make defconfig
```

4.运行下列命令查看生成的.config文件:

```
leo@ubuntu:~/Downloads/linux-5.2.11$ less .config
```

5.运行下列命令，根据模板精简配置内核。

```
leo@ubuntu:~/Downloads/linux-5.2.11$ make O=../mylinux/kernel/ localmodconfig
```

此后，会出现很多待确认的信息，全部按“Enter”（回车）。直到配置结束。

之后，运行下列命令查看目标目录下出现的配置文件：

```
leo@ubuntu:~/Downloads/linux-5.2.11$ ls ../mylinux/kernel/
```

结果为：

```
.config      .config.old  .gitignore  include/    Makefile    scripts/    source/
```

6.运行下列命令启动配置列表，查看已配置项。

```
leo@ubuntu:~/Downloads/linux-5.2.11$ make O=../mylinux/kernel/ menuconfig
```

可以酌情修改配置，也可不修改。

7.安装编译工具，准备编译。

安装编译工具：

```
<p class="mume-header " id="安装编译工具"></p>
```

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo apt-get install libssl-dev bc libelf-dev
```

以及：

```
<p class="mume-header " id="以及"></p>
```

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo apt-get install build-essential libncurses-dev flex bison
```

8.编译已配置内核。

运行下列命令：

```
leo@ubuntu:~/Downloads/linux-5.2.11$ make mrproper
```

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo make O=../mylinux/kernel/
```

编译内核需要较长的时间，具体与机器的硬件条件及内核的配置等因素有关（几十分钟 ~ 数小时）。

完成后产生的内核文件**bzImage**的位置在 `.../arch/i386/boot` 目录下。这里假设用户的CPU是Intel x86。

如果你在配置时（第4步中）选择了可加载模块，编译完内核后，要对选择的模块进行编译。用下面的命令编译模块并安装到标准的模块目录中：

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo make modules
```

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo make modules_install
```

9.安装新的Linux模块和内核

通常，Linux在系统引导后从/boot目录下读取内核映像到内存中。因此如果想要使用自己编译的内核，就必须先将启动文件安装到/boot目录下。使用命令：

1.安装modules。

<p class="mume-header " id="1安装modules"></p>

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo make O=../mylinux/kernel/ modules_install
```

2.安装内核

<p class="mume-header " id="2安装内核"></p>

```
leo@ubuntu:~/Downloads/linux-5.2.11$ sudo make O=../mylinux/kernel/ install
```

这个操作将编译的内核安装到/boot下。在Ubuntu 16.04下，会在产生如下文件：

- linux image: /boot/vmlinuz-5.2.11
- initrd image: /boot/initrd.img-5.2.11
- 启动配置文件: /boot/config-5.2.11
- 系统文件: /boot/System.map-5.2.11
- 如果选择安装了modules，那么就会在/lib/modules目录下产生一个新的目录“5.2.11”以及其下的若干子文件夹和文件。

说明：Initrd是一个被压缩过的小型根目录，这个目录中包含了启动阶段中必须的驱动模块，可执行文件和启动脚本。当系统启动的时候，bootloader会把initrd文件读到内存中，然后把initrd文件在内存中的起始地址和大小传递给内核。内核在启动初始化过程中会解压缩initrd文件，然后将解压后的initrd挂载为根目录，然后执行根目录中的linuxrc脚本（cpio格式的initrd为/init,而image格式的initrd<也称老式块设备的initrd或传统的文件镜像格式的initrd>为/initrc），您就可以在这个脚本中加载realfs（真实文件系统）存放设备的驱动程序以及在/dev目录下建立必要的设备节点。这样，就可以mount真正的根目录，并切换到这个根目录中来。

10.运行下列命令，修改grub config。

注意：后面的版本号应当为你之前编译使用的版本。

<p class="mume-header " id="注意后面的版本号应当为你之前编译使用的版本"></p>

```
sudo update-initramfs -c -k 5.2.11
```

```
sudo update-grub
```

```
leo@ubuntu:~/Downloads/linux-5.2.11$ cp -R ./arch/i386/boot/. /boot
```

11.在命令行下运行下列命名，重启ubuntu会自动加载新内核。

```
sudo reboot
```

如果失败，会进入initramfs的shell。通常出现进入initramfs，是因为关机不当导致磁盘文件受损还是什么引起的。这里是因为新内核中的配置不对。

12.若成功启动linux server，可以使用下列命令验证内核版本。

```
uname -mrs
```

参考文献

- 1.<https://www.cyberciti.biz/tips/compiling-linux-kernel-26.html>
- 2.<https://www.kernel.org/doc/html/latest/admin-guide/README.html#installing-the-kernel-source>
- 3.<https://blog.csdn.net/hamlee67/article/details/71727705>