

## @ GRUB 学习笔记 @

### 1 首先要了解的几个概念

#### 1.1 启动管理器

启动管理器是存储在磁盘开始扇区中的一段程序，例如，硬盘的 MBR(Master Boot Record)，在系统完成启动测试后，如果系统是从 MBR 启动，则 BIOS(Basic Input/Output System)将控制传送给 MBR。然后存储在 MBR 中的这段程序将运行。这段程序被称为启动管理器。它的任务就是将控制传送给操作系统，完成启动过程》有许多可用的启动管理器，包括 GNU GRUB (Grand Unified Boot Loader), Bootmanager, LILO (Linux LOader), NTLDR (boot loader for Windows NT systems)，等等等。

#### 1.2 什么是 GRUB?

grub 是一个多重启动管理器。grub 是 GRand Unified Bootloader 的缩写，它可以在多个操作系统共存时选择引导哪个系统。它可以引导的操作系统包括：Linux, FreeBSD, Solaris, NetBSD, BeOSi, OS/2, Windows95/98, Windows NT, Windows2000，WinXP。它可以载入操作系统的内核和初始化操作系统（如 Linux, FreeBSD），或者把引导权交给操作系统（如 Windows 98）来完成引导。

#### 1.3 GRUB 的特点

特别适用于 linux 与其它操作系统共存情况。

支持大硬盘 现在大多数 Linux 发行版本的 lilo 都有同样的一个问题：根分区

(/boot 分区)不能分在超过 1024 柱面的地方，一般是在 8.4G 左右的地方，否则 lilo 不能安装，或者安装后不能正确引导系统。而 grub 就不会出现这种情况，只要安装时你的大硬盘是在 LBA 模式下，grub 就可以引导根分区在 8G 以外的操作系统。

**支持开机画面** grub 支持在引导开机的同时显示一个开机画面。对于玩家来说，这样可以制作自己的个性化开机画面；对于 PC 厂商，这样可以在开机时显示电脑的一些信息和厂商的标志等。grub 支持 640x480,800x600,1024x768 各种模式的开机画面，而且可以自动侦测选择最佳模式，与 Windows 那 320x400 的开机画面不可同日而语。

**两种执行模式** grub 不但可以通过配置文件进行例行的引导，还可以在引导前动态改变引导时的参数，还可以动态加载各种设备。例如你在 Linux 下编译了一个新的核心，但不能确定它能不能工作，你就可以在引导时动态改变 grub 的参数，尝试装载这个新的核心进行使用。Grub 的命令行有非常强大的功能，而且支持如 bash 或 doskey 一样的历史功能，你可以用上下键来寻找以前的命令。

**菜单式选择** grub 使用一个菜单来选择不同的系统进行引导。你还可以自己配置各种参数，如延迟时间，默认操作系统等。

**分区大小改变后不必重新配置** grub 是通过文件系统直接把核心读取到内存，因此只要操作系统核心的路径没有改变，grub 就可以引导系统。

除此之外，Grub 还有许多非常强大的功能。例如支持多种外部设备，动态装载操作系统内核，甚至可以通过网络装载操作系统核心。Grub 支持多种文件系统，支持多种可执行文件格式，支持自动解压，可以引导不支持多重引导的操作系

统,支持网络启动等。

#### 1.4 MBR 和第一扇区

你可以简单的理解为 MBR 是整个硬盘的物理第一位置,而第一扇区是硬盘的物理第二位置.

#### 1.5 一个 GRUB 配置文件

基于本例的分区如下：

hda 15G

hda1 8G / RED HAT LINUX8.0

hda5 7G /home

hdc 20G

hdc1 6.4G WinXP

hdc5 6.4G

hdc6 6.4G

hdc7 6.4G

#fdisk -l

# Disk /dev/hdc: 255 heads, 63 sectors, 2434 cylinders

Units = cylinders of 16065 \* 512 bytes

Device Boot Start End Blocks Id System

/dev/hdc1 \* 1 894 7181023+ b Win95 FAT32

/dev/hdc2 895 2434 12370050 f Win95 Ext'd (LBA)

/dev/hdc5 895 1787 7172991 b Win95 FAT32

/dev/hdc6 1788 2434 5196996 b Win95 FAT32

Disk /dev/hda: 255 heads, 63 sectors, 1867 cylinders

Units = cylinders of 16065 \* 512 bytes

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/hda1	*	1	1020	8193118+	83	Linux
-----------	---	---	------	----------	----	-------

/dev/hda2		1021	1802	6281415	83	Linux
-----------	--	------	------	---------	----	-------

/dev/hda3		1803	1867	522112+	82	Linux swap
-----------	--	------	------	---------	----	------------

grub.conf,这个文件位于;/boot/grub/grub.conf

# grub.conf generated by anaconda

#

# Note that you do not have to rerun grub after making changes to this file

# NOTICE: You do not have a /boot partition. This means that

# all kernel and initrd paths are relative to /, eg.

# root (hd0,0)

# kernel /boot/vmlinuz-version ro root=/dev/hda1

# initrd /boot/initrd-version.img

#boot=/dev/hda

default=0

timeout=3

```
splashimage=(hd0,0)/boot/grub/splash.xpm.gz

title Red Hat Linux (2.4.18-14)

root (hd0,0)

kernel /boot/vmlinuz-2.4.18-14 ro root=LABEL=/

initrd /boot/initrd-2.4.18-14.img

title Microsoft Windows XP

map (hd0) (hd1)

map (hd1) (hd0)

root (hd1,0)

chainloader (hd1,0)+1

makeactive

boot
```

## 2 解读 grub.conf 文件

我们将来看看 grub.conf 文件内语句,(注:...)内的东西是我们的解读内容.

```
# grub.conf generated by anaconda

#

# Note that you do not have to rerun grub after making changes to this file

# NOTICE: You do not have a /boot partition. This means that

# all kernel and initrd paths are relative to /, eg.

# root (hd0,0)
```

```
# kernel /boot/vmlinuz-version ro root=/dev/hda1
```

```
# initrd /boot/initrd-version.img
```

#boot=/dev/hda (注:以上以符号井 " # " 开头的行表示被注释掉,没有任何意义)

default=0 (注:默认的操作系统就是由 default 控制的。default 后加一个数字 n,表明是第 n + 1 个。需要注意的是,GRUB 中,计数是从 0 开始的,第一个硬盘是 hd0,第一个软驱是 fd0,等等。所以,default 0 表示默认的操作系统在这儿是 Red Hat Linux (2.4.18-14)如果你修改成 1 就是 WinXP 了)

timeout=3 (注:timeout 表示默认等待的时间,这儿是 3 秒钟。超过 3 秒,用户还没有作出选择的话,系统将自动选择默认的操作系统;当然你可以改成任何你乐意的时间)

splashimage=(hd0,0)/boot/grub/splash.xpm.gz (注:指定开机画面文件 splash.xpm.gz 的位置)

title Red Hat Linux (2.4.18-14) (注:表示 Red Hat Linux 的菜单项)

root (hd0,0) (注:表示第一个硬盘第一个分区,这里的 root 和系统内的 root 不是一码事!详细如下说明)

kernel /boot/vmlinuz-2.4.18-14 ro root=LABEL=/ (注:指定内核的位置,详细说明如下文)

initrd /boot/initrd-2.4.18-14.img (注:初始化)

title Microsoft Windows XP (注:表示 Microsoft Windows XP 的菜单项)

map (hd0) (hd1) (注:map 是命令,详细如下)

map (hd1) (hd0)

root (hd1,0) (注:这是指第二个硬盘(从硬盘)上第一个分区)

chainloader (hd1,0)+1 (注:链式装入器,装入一个扇区的数据然后把引导权交给它。详细说明如下)

makeactive

boot

(注:在 Linux 中,当谈到 "root" 文件系统时,通常是指主 Linux 分区。但是, GRUB 有它自己的 root 分区定义。GRUB 的 root 分区是保存 Linux 内核的分区。这可能是您的正式 root 文件系统,也可能不是。我们讨论的是 GRUB,需要指定 GRUB 的 root 分区。进入 root 分区时,GRUB 将把这个分区安装成只读型,这样就可以从该分区中装入 Linux 内核。GRUB 的一个很“酷”的功能是它可以读取本机的 FAT、FFS、minix、ext2 和 ReiserFS 分区。到目前为止,您可能会感到一点疑惑,因为 GRUB 所使用的硬盘/分区命名约定与 Linux 使用的命名约定不同。在 Linux 中,第一个硬盘的第五个分区称作 "hda5"。而 GRUB 把这个分区称作 "(hd0,4)"。GRUB 对硬盘和分区的编号都是从 0 开始计算。另外,硬盘和分区都用逗号分隔,整个表达式用括号括起。现在,可以发现如果要引导 Linux 硬盘 hda5,应输入 "root (hd0,4)"。

知道了内核在哪儿,还要具体指出哪个文件是内核文件,这就是 kernel 的工作。

kernel /boot/vmlinuz-2.4.18-14 ro root=LABEL=/说明/boot/vmlinuz-2.4.18-14 就是要载入的内核。后面的都是传递给内核的参数。root=LABEL=/就是 linux 的硬盘分区表示法,ro 是 readonly 的意思。initrd 用来初始的 linux

image，并设置相应的参数。

命令 map:当你有两块硬盘，一个无法从第二块硬盘启动的操作系统，例如 Windowsxp，就可以使用 map 命令.你能将 hd0 映射为 hd1，将 hd1 映射为 hd0。换句话说，你可以虚拟的交换两个硬盘而启动所需要的操作系统。命令形式如下：

```
grub> map (hd0) (hd1)
```

```
grub> map (hd1) (hd0)
```

GRUB 使用了“链式装入器”(chainloader)。链式装入器从分区 (hd1,0) 的引导记录中装入 winxp 自己的引导装入器，然后引导它。这就是这种技术叫做链式装入的原因 -- 它创建了一个从引导装入器到另一个的链。这种链式装入技术可以用于引导任何版本的 DOS 或 Windows。

GRUB 的配置文件要简单就这么简单，如果你要更个性化一点，试一试把“color light-gray/blue”加在 default 语句的下面，下一次启动 GRUB 时，看看有什么变化，再试一试“color light-blue/red”，惊喜吗？有趣吧！)

### 3 如何使用 grub?

#### 3.1 安装

你可以从 <ftp://alpha.gnu.org/pub/gnu/grub> 下载 GRUB 的源代码。

得到文件 grub-0.5.96.1.tar.gz，所以这样做：

```
# tar -xvzf grub-0.5.96.1.tar.gz
```

这个命令将解开的文件和目录放在一个名为 grub-0.5.96.1 的目录中。现在运行

如

下命令：

```
#cd grub-0.5.96.1
```

```
# ./configure
```

如果你想定制 GRUB，使之包含流行的文件系统和对网卡的支持，或者删除你不需要的网卡，可以运行如下命令：

```
# ./configure --help
```

这个命令将输出所有的命令选项。现在可以使用—enable 和—disable 选项增加或

删除对特定网卡的支持。（注意：GRUB 支持网络启动）

为了开始编译过程，键入如下命令：

```
# make
```

```
# make install
```

至此你已经为安装 GRUB 真正做好了准备。

将 GRUB 的相关文件保存在诸如/boot/grub 的目录中是一个好主意。为此，按如下步骤：

1．默认状态下，GRUB 所有文件将安装在/usr/share/grub/i386-pc(或者在 /usr/local/share/grub/i386-pc 目录下，这只取决于你的 shell 变量的设置。)

2．建立一个叫做/boot/grub 的目录，然后拷贝下列文件到这个目录：

```
stage1 stage2 * stage1 5
```

稍候将解释这些文件。同时也要把 GRUB(可能在/usr/sbin 或者/usr/local/sbin 目录下)拷贝到/boot/grub 目录下。

安装 GRUB 可以分为三个单独的步骤：

1. 将“stage1”安装到 MBR 中。
2. 设置“stage2”的地址或者位置。
3. 设置一个菜单或选项，用来决定启动哪一个操作系统。

使用下面的命令开始安装 GRUB。

```
#cd /boot/grub
```

```
# ./grub
```

这个命令根据 BIOS 的设置检测硬件设备，同时产生一些输出信息。这会花较长的时间。

```
end_request: I/O error, dev 02:00 (floppy), sector 0
```

```
GRUB version 0.5.96.1 (640K lower / 3072K upper memory)
```

TAB 键可列出可能的命令列表。其它情况下 TAB 会列出设备或文件名。类似下面的提示会出现：

```
grub>
```

现在，假设你将 Linux 安装在了第一块硬盘的第一个分区中或者/dev/hda1 中。

记住 GRUB 的命名规则，将上面的名字改为(hd0,0)。键入下面的命令：

```
grub> install (hd0,0)/boot/grub/stage1 (hd0) (hd0,0)/boot/grub/stage2
```

```
p (hd0,0)/boot/grub/menu.conf
```

现在分析一下这个命令的细节。

```
install
```

一个内置的命令，它告诉 GRUB 将(hd0,0)/boot/grub/stage1 安装到 hd0 的主引导纪录中。

(hd0,0)/boot/grub/stage2

告诉 GRUB stage2 映象的位置。

p with the the following options: (hd0,0)/boot/grub/menu.conf

为菜单的显示设置配置文件。

下面是对这个命令用法的总结：

- 1 . install
- 2 . ource\_of\_stage1
- 3 . where\_to\_install
- 4 . source\_of\_stage2
- 5 . p source\_of\_configuration\_file

现在，你完成了硬盘上的基本安装工作。

在软盘上的安装：

为了在软盘上安装 GRUB，你要会用“dd”命令，并且了解它是如何工作的。为了

做一张 GRUB 启动盘，你要将 stage1 和 stage2 文件放到软盘的开始扇区中。

安装 stage1 到软盘上：

插入一张已格式化的软盘，键入命令：

```
# dd if=stage1 of=/dev/fd0 bs=512 count=1
```

命令的详细说明如下：

if=input file

i.e., stage1

of=output file

i.e., floppy drive (this may be different on your computer)

bs=bytes to read and write

Here it is 512 bytes.

count=how many times to perform this operation

每次拷贝“bs”数目的块到目标地址。

安装 stage2 到软盘

```
# dd if=stage2 of=/dev/fd0 bs=512 seek=1
```

这里的步骤同 stage1 的相同，除了一个 seek 选项外。Seek 选项将调过一个“bs”。例如，在上面的命令中 bs 的值是 512,这样 seek=1 意味着将跳过硬盘上前 512 个字节，从 513 字节开始。这将使得第一步操作不会覆盖 stage1 的前 512 字节。

现在你完成了基本的软盘驱动器安装。

如果已经安装了 grub 要把 grub 重新安装到主引导扇区上，只需要简单打入 makebootable 命令就可以了。

### 3.2 配置 grub

grub 启动时会在 /boot/grub/ 中寻找一个名字为 menu.lst 的配置文件，如果找不到此文件则不进入菜单模式而直接进入命令行模式。

现在，我们来看一下如何在启动后进入各种操作系统，如何建立 menu.conf 文件。我们就从 GRUB 支持的启动过程开始。可以有两种方法来完成启动过程：

- A.通过调用内核本地启动
- B.连续启动或者将控制转给另一个引导器

## A 模式启动过程

1. 配置跟设备或者告诉 GRUB 你的根文件系统。
2. 告诉 GRUB 你的内核影像的位置，然后将参数传送给内核。
3. 重新启动，试一下。

为了启动 Linux，将内核以 bzImage 的文件名放在 /boot/ 目录中，跟文件系统是

/dev/hda1，或者 GRUB 中的(hd0,0)。启动过程如下：

- 1.root (hd0,0) [This sets the root partition]
- 2.kernel /boot/bzImage root=/dev/hda1 [This sets the kernel]

B 模式启动过程（这种模式假设当前的分区中安装了另一个启动管理器，例如

LILO

或者 NTLDR）：

1. 设置根分区但不要安装它
2. 激活这个分区
3. 配置需要启动的分区的第一个扇区
4. 重新启动，看一下效果。

我们在试试启动安装在 /dev/hdc1 或者(hd1, 0)的 windows。启动 windows 的过程如下：

- 1.rootnoverify (hd1,0)
- 2.makeactive
- 3.chainloader +1 [+1 sets the first sector of the current root partition]

4.boot [transfers the control and quits GRUB]

menu.conf 文件：它用于建立启动多操作系统时的菜单。建立 menu.conf 并不难。它使用简单的英语，就象你在这一节看到的那样。

所有的菜单项目都以没有逗号分隔的“title TITLENAM”开头。你可以随意设置 TITLENAM。

设置 Linux 启动菜单步骤如下：

- 1．设置标题
- 2．设置根分区
- 3．设置内核的相应参数
- 4．启动

一个菜单例子：

```
title Red Hat Linux (2.4.18-14)

root (hd0,0)

kernel /boot/vmlinuz-2.4.18-14 ro root=LABEL=/
initrd /boot/initrd-2.4.18-14.img
```

前面有#的行是一个注释。

建立启动 Windows 或者 DOS 的菜单：

```
title Windoze

rootnoverify (hd0,0)

makeactive

chainloader +1

boot
```

#----

又或者:

```
title Microsoft Windows XP
```

```
map (hd0) (hd1)
```

```
map (hd1) (hd0)
```

```
root (hd1,0)
```

```
chainloader (hd1,0)+1
```

```
makeactive
```

```
boot
```

----

注意:root 和 rootnoverify 都是一样的,把 rootnoverify 改成 root 也行。不过经过实践来看。有时引导 win 时,系统安装好后,是 rootnoverify (hdX.Y)这样形式的,这样会出现 windows 起不来,出现什么 windows 什么文件损坏的情况。这时,我们就要把在 grub 中,引导 windows 的那段中的 rootnoverify 改为 root  
root 英文的意思就是根的意思,在这里是让 linux 知道自己所处的位置,也就是我们所安装 linux 的 / 根分区所在的位置。

----

如果你安装了两个版本的 Windows——一个是你自己用的,另一个给你的家人用的——这样第二个就无法安装,因为有提示说 Windows 已经安装了。

有一种简单的方法可以安装两个版本的 Windows,即在启动时隐藏一个分区而使用另一个。你甚至可以为你的分区设置密码保护,这样可以避免别人错误的加载你的分区。下面介绍如何完成两个 Windows 的安装, hda1 和 hda2 或者

(hd0,0) 和(hd0,1) , 用到的命令有 lock, password, hide 和 unhide。

建立 windows 项目 "My Entry":

```
title My Entry
```

```
lock
```

```
unhide (hd0,0)
```

```
hide (hd0,1)
```

```
rootnoverify (hd0,0)
```

```
makeactive
```

```
chainloader +1
```

```
boot
```

```
#----
```

为了更好的使用 lock 命令，你需要在配置文件开头使用 password 命令。

Password 命令的语法如下：password secret(“secret”就是密码)。任何时候你都

可以通过按 p 键儿输入密码。

建立 Windows 项目 "Family Entry"

```
title Family Entry
```

```
unhide (hd0,1)
```

```
hide (hd0,0)
```

```
rootnoverify (hd0,1)
```

```
makeactive
```

```
chainloader +1
```

```
boot
```

任何人都可以启动这个项目，而无须输入密码。

这有一个使用 password 命令的有趣的窍门。为了在缺省菜单列表或配置文件中隐藏某些项目，你可以使用下面的命令加载一个定制列表：

```
password secret
```

在这个命令中，“secret”是密码，而/boot/grub/secret-list.conf 是密码文件。这样做之前，你要先进入到根目录或者给出全路径名。例如：

```
password secret (hd0,4)/boot/grub/secret-list.conf
```

还有一个更重要的命令是“map”。当你有两块硬盘，一个无法从第二块硬盘启动的操作系统，例如 Windows，就可以使用 map 命令。例如，你能够将 hd0 映射为 hd1，将 hd1 映射为 hd0。换句话说，你可以虚拟的交换两个硬盘而启动所需要的操作系统。命令形式如下：

```
grub> map (hd0) (hd1)
```

```
grub> map (hd1) (hd0)
```

启动 FreeBSD：

```
title FreeBSD 4.0
```

```
root (hd0,4,a)
```

```
kernel /boot/loader
```

```
boot
```

```
#----
```

这里我们调用了 FreeBSD 的启动管理器。Root (hd0,4,a)由四个参数，是因为 FreeBSD 对一个单独分区进行了虚拟分割。我们称根分区为“a”。如果 FreeBSD

占

据了整个第二块硬盘，这里就应该是 root (hd0,a)。这样，就不是调用内核而是调用 FreeBSD 的启动管理器，它要比调用内核更易使用。

(注意：推荐在使用 OpenBSD 和 GNU/Hurd 之前，要先试一下链式加载。)

现在你已经完成了基本的 GRUB 编译、安装和配置。你对 GRUB 了解的越多，就越会发现 GRUB 是一种控制启动的易用和高效的方法。

各种 GRUB 命令：

default xx

表明 xx 是缺省的启动项目

timeout yy

表明在 yy 秒后缺省启动项目将启动

fallback zz

在第一个启动项目失败后，过了 timeout 时间后，就将启动 zz 项目。

color

这个命令将生成彩色菜单。它的语法是：color normal current\_selection。这两个域都可以有两个值，形如 foreground/background。例如：

color green/black or light-gray/blue

你也可以使用相关的数字。

记住：所有的值都是从 0 开始，所以 0 是第一个

### 3.3 GRUB 启动盘

要制作引导盘，需执行一些简单的步骤。首先，在新的软盘上创建 ext2 文件系

统。然后，将其安装，并将一些 GRUB 文件复制到该文件系统，最后运行 "grub" 程序，它将负责设置软盘的引导扇区。

将一张空盘插入 1.44MB 软驱，输入：

```
# mke2fs /dev/fd0
```

创建了 ext2 文件系统后，需要安装该文件系统：

```
# mount /dev/fd0 /mnt/floppy
```

现在，需要创建一些目录，并将一些关键文件（原先安装 GRUB 时已安装了这些文件）复制到软盘：

```
# mkdir /mnt/floppy/boot
```

```
# mkdir /mnt/floppy/boot/grub
```

```
# cp /boot/grub/stage1 /mnt/floppy/boot/grub
```

```
# cp /boot/grub/stage2 /mnt/floppy/boot/grub
```

再有一个步骤，就能得到可用的引导盘。

在 linux bash 中，从 root 用户运行“grub”，该程序非常有趣并值得注意，因为它实际上是 GRUB 引导装入器的半功能性版本。尽管 Linux 已经启动并正在运行，您仍可以运行 GRUB 并执行某些任务，而且其界面与使用 GRUB 引导盘或将 GRUB 安装到硬盘 MBR 时看到的界面(即 GRUB 控制台)完全相同。

在 grub> 提示符处，输入：

```
grub> root (fd0)
```

```
grub> setup (fd0)
```

```
grub> quit
```

现在，引导盘完成了。

如果要把 GRUB 装到硬盘上，也很容易。这个过程几乎与引导盘安装过程一样。首先，需要决定哪个硬盘分区将成为 root GRUB 分区。在这个分区上，创建 /boot/grub 目录，并将 stage1 和 stage2 文件复制到该目录中，可以通过重新引导系统并使用引导盘，或者使用驻留版本的 GRUB 来执行后一步操作。在这两种情况下，启动 GRUB，并用 root 命令指定 root 分区。例如，如果将 stage1 和 stage2 文件复制到 hda5 的 /boot/grub 目录中，应输入 "root (hd0,4)"。接着，决定在哪里安装 GRUB -- 在硬盘的 MBR，或者如果与 GRUB 一起使用另一个“主”引导装入器，则安装在特定分区的引导记录中。如果安装到 MBR，则可以指定整个磁盘而不必指定分区，如下（对于 hda）：

```
grub> setup (hd0)
```

如果要将 GRUB 安装到 /dev/hda5 的引导记录中，应输入：

```
grub> setup (hd0,4)
```

现在，已安装 GRUB。引导系统时，应该立即以 GRUB 的控制台方式结束（如果安装到 MBR）。现在，应创建引导菜单，这样就不必在每次引导系统时都输入那些命令。

### 从软盘启动 grub

制作启动盘后可以用软盘启动引导硬盘上的操作系统，插入制作好的启动软盘，进入 BIOS 设定软盘启动。软盘启动成功后就会进入 grub 的命令行模式

```
grub>
```

要启动一个操作系统，首先指定引导哪个分区上的系统，例如要引导第一个硬盘上的第一个分区的操作系统，先键入

```
grub>root (hd0,0)
```

接着如果要启动的是 Windows 系统，键入

```
grub>chainloader (hd0,0)+1
```

注意(hd0,0)要随着硬盘和分区不同而改变数字。

如果要引导 Linux 或其他系统，应键入

```
grub>kernel (hd0,0) /boot/vmlinuz root=/dev/hda1
```

注意 hda1 参数也要随着硬盘和分区不同而改变，如从第二个硬盘的第一个分区引导则用 hdb1。

最后敲入 boot 就可以启动系统了。

在任何时候不能确定命令或者命令的参数都可以按 Tab 获得相关的帮助。用上下键可以获得命令的历史记录。其实这些命令就是 menu.lst 的启动描述，您也可以根据那些描述来自己键入启动命令，最后敲入 boot 就可以引导系统了。

#### 4 GRUB 的交互性

GRUB 最好的优点之一就是其强健的设计 -- 在不断使用它时请别忘了这点。如果更新内核或更改它在磁盘上的位置，不必重新安装 GRUB。事实上，如有必要，只要更新 menu.lst 文件即可，一切将保持正常。

只有少数情况下，才需要将 GRUB 引导装入器重新安装到引导记录。首先，如果更改 GRUB root 分区的分区类型（例如，从 ext2 改成 ReiserFS），则需要重新安装。或者，如果更新 /boot/grub 中的 stage1 和 stage2 文件，由于它们来自更新版本的 GRUB，很有可能要重新安装引导装入器。其它情况下，可

以不必理睬！

GRUB 的最大的特点就是交互性特别强。在开机时，按一下“c”，将进入 GRUB 控制台。显示如下：

```
GRUB version 0.5.96.1 (640K lower / 3072K upper memory)
```

```
[ Minimal BASH-like line editing is supported. For the first word, TAB
```

```
lists possible command completions. Anywhere else TAB lists the possible  
completions of a device/filename. ]
```

```
grub>
```

欢迎使用 GRUB 控制台。现在，再研究命令：

将通过 GRUB 控制台绕过 lilo 来启动 RedHat linux，

```
grub> root (h
```

现在，按一次 Tab 键。如果系统中有多个硬盘，GRUB 将显示可能完成的列表，从 "hd0" 开始。如果只有一个硬盘，GRUB 将插入 "hd0,"。如果有多个硬盘，继续进行，在 ("hd2") 中输入名称并在名称后紧跟着输入逗号，但不要按 Enter 键。部分完成的 root 命令看起来如下：

```
grub> root (hd0,
```

现在，继续操作，再按一次 Tab 键。GRUB 将显示特定硬盘上所有分区的列表，以及它们的文件系统类型。在我的系统中，按 Tab 键时得到以下列表：

```
grub> root (hd0, (tab, 按 tab 一下键)
```

Possible partitions are:

Partition num: 0, Filesystem type is fat, partition type 0x6

Partition num: 2, Filesystem type is ext2fs, partition type 0x83

Partition num: 4, Filesystem type unknown, partition type 0x7

Partition num: 5, Filesystem type is ext2fs, partition type 0x83

Partition num: 6, Filesystem type is fat, partition type 0xb

Partition num: 7, Filesystem type is fat, partition type 0xb

Partition num: 8, Filesystem type is ext2fs, partition type 0x83

Partition num: 9, Filesystem type unknown, partition type 0x82

如您所见，GRUB 的交互式硬盘和分区名称实现功能非常有条理。这些，只需要好好理解 GRUB 新奇的硬盘和分区命名语法，然后就可以继续操作了

```
grub> root (hd0,8)
```

现在已安装了 root 文件系统，到装入内核的时候了

```
grub> kernel /boot/vmlinuz-2.4.2 root=/dev/hda5 ro
```

```
[Linux-bzImage, setup=0x1200, size=0xe1a30]
```

您已经安装了 root 文件系统并装入了内核。现在，可以引导了。只要输入 "boot"，Linux 引导过程就将开始。是不是很 cool 啊，GRUB 的 menu.lst 更像一个 linux 下的脚本程序。

## 5 常见 grub 除错方法的思路

首先进去 Linux 的 rescue 模式！

用软盘或光盘启动，然后在启动的提示符输入:linux rescue

按照提示进入一个 Shell 状态，你可以到/mnt/下面看到一个 sysimage 这么目录，进去以后，就是你安装 linux 的/分区。

使用命令将根分区变为当前目录的根分区:chroot /mnt/sysimage

然后转到/sbin/这个目录中.

使用 fdisk -l 显示当前分区情况，然后使用#grub-install /dev/hdx(x 为你使用的是那块硬盘安装的，一般情况下是 hda)

使用 exit 推出 chroot，再使用 exit 退出 linux rescue 模式，系统将重新启动！取出光盘，应该可以看到 grub 安装好了.

在具体的环境中，编辑/boot/grub/grub.conf 文件和 menu.lst 文件.