



Linux公社 (LinuxIDC.com) 于2006年9月25日注册并开通网站, Linux现在已经成为一种广受关注和支持的一种操作系统, IDC是互联网数据中心, LinuxIDC就是关于Linux的数据中心。

LinuxIDC.com提供包括Ubuntu, Fedora, SUSE技术, 以及最新IT资讯等Linux专业类网站。

并被收录到Google 网页目录-计算机 > 软件 > 操作系统 > Linux 目录下。

Linux公社 (LinuxIDC.com) 设置了有一定影响力的Linux专题栏目。

包括:

[Ubuntu专题](#)

[Fedora专题](#)

[RedHat专题](#)

[SUSE专题](#)

[红旗Linux专题](#)

[Android专题](#)

---

[Linux公社简介](#) - [广告服务](#) - [网站地图](#) - [帮助信息](#) - [联系我们](#)

本站 (LinuxIDC) 所刊载文章不代表同意其说法或描述, 仅为提供更多信息, 也不构成任何建议。

本站带宽由[\[6688.CC\]](#)友情提供

Copyright © 2006-2011 [Linux公社](#) All rights reserved

## 系统引导管理器GRUB，为初学者指南

作者：北南南北

来自：LinuxSir.Org

摘要：GRUB是多系统引导管理器，简单的说既能引导Linux，同时也能引导Windows；从LinuxSir.Org 讨论区近四年的观察来看，大多初学者并不能在短时间内掌握GRUB的用法，为了解决初学者在最短时间内掌握GRUB，重写GRUB入门文档还是有必要的；

本文重点介绍了GRUB的menu.lst的写法，另外通过GRUB命令行引导系统也做了详述；这些无论是对Windows版本的WINGRUB还是Linux版本的GRUB都是适用的；

### 目录索引

一、什么是多重操作系统引导管理器，什么是GRUB；

- 1、什么是多重操作系统引导管理器及工作原理；
- 2、什么是GRUB；为什么我要选择GRUB；
  - 1) 什么是GRUB；
  - 2) “GRUB太不好用”——对GRUB的认识的误区；
  - 3) 为什么要选择GRUB；

二、GRUB软件包版本选择和安装；

- 1、Linux版本的GRUB及Windows版本的GRUB的说明；
- 2、GRUB的Windows版本WINGRUB；
- 3、GRUB的Linux版本软件包的安装；

三、在Linux中，GRUB的配置中的安装和写入硬盘的MBR；

- 1、在Linux中，GRUB配置过程中的安装grub-install；
- 2、设定GRUB的/boot分区并写入MBR；

四、GRUB的配置文件的menu.lst的写法；

- 1、menu.lst的写法之一；
  - 1) 在menu.lst中，通过 root (hd[0-n],y)来指定/boot 所在的分区；
  - 2) 在menu.lst中，kernel 命令行的写法；
  - 3) initrd 命令行的写法；
  - 4) menu.lst第一种写法的总结和实践；
    - 1] 用fdisk -l ; df -lh ; more /etc/fstab来确认分区情况；
    - 2] 查看内核vmlinuz的和initrd文件名的全称；
    - 3] 开始写menu.lst ；
- 2、menu.lst的写法之二，精简型；
  - 1) 第一种情况：/boot和Linux的/根分区在同一个分区；
  - 2) 第二种情况：/boot独立一个分区，和Linux的根分区不位于同一分区；

五、通过GRUB命令行来启动Linux操作系统；

- 1、为什么需要学习GRUB的命令行；
- 2、用命令行来引导Linux操作系统的步骤；
  - 1) 进入GRUB的命令行模式 grub>
  - 2) 获取帮助GRUB的 help
  - 3) cat的用法；
  - 4) root (hd[0-n],y) 指令来指定/boot所在的分区；
  - 5) kernel 指令，用来指定Linux的内核，及/所在的分区；
  - 6) initrd 命令行来指定initrd文件；

- 7) boot 引导系统;
- 8) 引导Linux系统实例全程回放;

六、通过GRUB引导Windows操作系统;

- 1、通过编辑 menu.lst 来引导Windows 系统;
- 2、通过GRUB指令来引导Windows ;

七、GRUB丢失或损坏的应对策略;

- 1、由于重新安装Windows或其它未知原因而导致GRUB的丢失;
- 2、如果出现GRUB提示符, 而不出现GRUB的菜单, 如何引导系统;

八、关于GRUB的未尽事宜;

- 九、关于本文;
- 十、参考文档;
- 十一、相关文档;

++++  
正文  
++++

## 一、什么是多重操作系统引导管理器, 什么是GRUB;

### 1、什么是多重操作系统引导管理器及工作原理;

系统启动引导管理器, 是在计算机启动后运行的第一个程序, 他是用来负责加载、传输控制到操作系统的内核, 一旦把内核挂载, 系统引导管理器的任务就算完成退出, 系统引导的其它部份, 比如系统的初始化及启动过程则完全由内核来控制完成;

*Briefly, boot loader is the first software program that runs when a computer starts. It is responsible for loading and transferring control to the operating system kernel software (such as the Hurd or the Linux). The kernel, in turn, initializes the rest of the operating system (e.g. GNU).*

在X86 架构的机器中, Linux、BSD 或其它Unix类的操作系统中GRUB、LILO 是大家最为常用, 应该说是主流;

Windows也有类似的工具NTLOADER; 比如我们在机器中安装了Windows 98后, 我们再安装一个Windows XP, 在机器启动的会有一个菜单让我们选择进入是进入Windows 98 还是进入Windows XP。NTLOADER就是一个多系统启动引导管理器, NTLOADER 同样也能引导Linux, 只是极为麻烦罢了;

在Powerpc 架构的机器中, 如果安装了Linux的Powerpc 版本, 大多是用yaboot 多重引导管理器, 比如Apple机目前用的是IBM Powerpc处理器, 所以在如果在Apple机上, 安装Macos 和Linux Powerpc 版本, 大多是用yaboot来引导多个操作系统;

因为目前X86架构的机器仍是主流, 所以目前GRUB和LILO 仍然是我们最常用的多重操作系统引导管理器;

### 2、什么是GRUB; 为什么我要选择GRUB;

#### 1) 什么是GRUB;

GNU GRUB 是一个多重操作系统启动管理器。GNU GRUB 是由GRUB (GRand Unified Bootloader) 派生而来。GRUB 最初由Erich Stefan Boleyn 设计和应用;

*GNU GRUB is a Multiboot boot loader. It was derived from GRUB, GRand Unified Bootloader, which was originally designed and implemented by Erich Stefan Boleyn.*

#### 2) “GRUB太不好用”——对GRUB的认识的误区;

GRUB真的不好用吗？不是的，通过LinuxSir.Org 社区近四年来的运行，我发现了许多新手弟兄还是不太了解GRUB；当然这也有中文Linux社区的责任，虽然有GRUB的中文译本，初学Linux的弟兄可能有点看不懂；

我们欣喜的看到LinuxSir.Org 社区的好多弟兄都曾经或正在写GRUB实践文档，也有的弟兄也总结了GRUB的一些基础知识，比如 probing兄弟的《GRUB 学习笔记》；由于每个人的写文档时风格不同，可能同一份文档不同的人来写就有不同的风格；所以今天也抖胆也一篇入门级的教程，由于北南不会写高级教程，所以还得请高手弟兄指教，先谢过；

### 3) 为什么要选择GRUB；

基于在X86架构的CPU而开发操作系统，系统引导管理器不仅仅有GRUB，而且也有LILO，但对于多重系统引导管理器，你只能选择其一而用；不能两个同时使用；

目前这两个多重系统引导管理器是大家最常用的，也是主流Linux发行版而采用的；有的弟兄喜欢GRUB，比如我个人，有的弟兄喜欢LILO，比如etony兄（谁是etony，请参见 <http://debian.linuxsir.org>）；

主流发行版 Fedora、Redhat、Centos等基于RPM包的系统，在最新版本中都默认GRUB引导；Slackware 目前仍采用LILO；而Debian发行版目前最新的版本也是采用GRUB；

从目前看来，GRUB有逐渐取代LILO之势，GRUB 2.0正在开发之中；所以我们有理由用GRUB，我也有理由写GRUB使用教程；

## 二、GRUB软件包版本选择和安装；

### 1、GRUB的版本选择，Linux版本的GRUB及Windows版本的GRUB的说明；

GRUB不但有Linux版本，也有Windows版本；现我们一一介绍；

如前面所说，目前在在Unix类的操作系统中，大多是都有GRUB；GRUB几乎能引导所有X86架构的操作系统；功能之强，使用简单是GRUB最大的卖点；由于Windows操作系统的先入为主的劣势，使得大家对Windows的NTLOADER了解的比较多，而对开源社区的GRUB显得有点陌生，由此而带来使用上的“心理恐惧”；究其初学者对GRUB“恐惧”的主要原因还是对GRUB没有太多的了解和深入；无论是WINGRUB还是Linux版本的GRUB，最方便的还是对GRUB命令行的操作；一谈到命令行（Command）的操作，可能初学者对此恐惧；其实没有什么难的，象北南这样低级的写手，还能操作得起来，您也应该能行；

### 2、GRUB的Windows版本WINGRUB；

请参考：《以WINGRUB 引导安装Fedora 4.0 为例，详述用WINGRUB来引导Linux的安装》

### 3、GRUB的Linux版本软件包的安装；

其实对于Linux的GRUB，几乎所有的Linux主流发行版都有打包，如果您安装了Linux，并且在开机后出现GRUB字样的，证明您已经安装了GRUB；而无需再次安装；Linux的GRUB软件包安装部份并不是本文的重点；

如果您的Linux系统没有安装GRUB，或者采用的是LILO，而您想用GRUB，可以用系统安装盘自带GRUB软件包来安装，或者到相关发行版本的软件仓库下载后安装；

GRUB 的Linux版本目前在各大发行版中都有打包；比如Fedora/Redhat/Centos/Mandriva/Mandriva/SuSE等以RPM包管理机制的系统，可以通过如下的命令来安装；

请参考《Fedora / Redhat 软件包管理指南》

```
[root@localhost ~]# rpm -ivh grub*.rpm
```

如果是Slackware 您可以用如下的办法来安装；

```
[root@localhost ~]# installpkg grub*.tgz
```

其它的发行版本请用其自己特色的软件包管理工具来安装；

当然您也可以通过源码包，在任何Linux的发行版上安装；至于源码包的安装方法；

请参考：《如何编译安装源码包软件》

```
[root@localhost ~]#tar zxvf grub*.tar.gz
[root@localhost ~]#cd grub-xxx
[root@localhost ~]#./configure;make;make install
```

确认您是否成功安装了GRUB，您可以测试是否有如下两个命令；

```
[root@localhost ~]# grub
[root@localhost ~]# grub-install
```

如果您不能找到这两个命令，可能您的可执行程序的路径没有设置；

请参考：《设置可执行程序路径》，当然您可以用绝对路径；比如下面的；

```
[root@localhost ~]# /usr/sbin/grub
[root@localhost ~]# /usr/sbin/grub-install
```

如果您还是找不到GRUB软件包安装在哪了；您可以用下面的命令来解决和查找；

```
[root@localhost ~]# updatedb 注：这个要花很长时间；是索引slocate 的库，然后再通过locate来查找；
[root@localhost ~]# locate grub
```

比如找到的是有类似如下的；

```
[root@localhost ~]# locate grub
/sbin/grub-md5-crypt
/sbin/grub
/sbin/grub-install
/sbin/grub-terminfo
```

在一般情况下，在路径中带有bin或sbin中字样的，这些路径下都是可执行程序；sbin 是超级权限用户才能使用的管理命令；要使用这些命令一般的情况下得切换到root用户下才能使用；比如

```
[beinan@localhost ~]$ su - 注：切换到root用户，并且切换到其家目录；
Password:
[root@localhost ~]#/sbin/grub 注：用绝对路径来运行grub命令；
```

### 三、在Linux中，GRUB的配置中的安装和写入硬盘的MBR；

#### 1、在Linux中，GRUB配置过程中的安装grub-install；

grub-install 命令有何用呢？其实就是把我们前面已经安装的软件包中的一些文件复制到/boot/grub中；对于新安装GRUB软件包后，也是一个必经的过程；我们前面所说的GRUB软件包的安装；而现在我们说的是GRUB配置的过程中的安装；虽然在洋文中都是install，但表达的意思是不一样的；

我们首先要运行 fdisk -l 来确认到底是硬盘的标识；

这个过程主要是确认硬盘的标识是哪个调备，到底是/dev/hda还是/dev/hdb 还是其它的；

```
[root@localhost ~]# fdisk -l

Disk /dev/hda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id
System
/dev/hda1    *           1           970       7791493+   7
HPFS/NTFS
/dev/hda2           971          9729       70356667+   5
```

Extended					
/dev/hda5	971	2915	15623181	b	W95
FAT32					
/dev/hda6	2916	4131	9767488+	83	
Linux					
/dev/hda7	4132	5590	11719386	83	
Linux					
/dev/hda8	5591	6806	9767488+	83	
Linux					
/dev/hda9	6807	9657	22900626	83	
Linux					
/dev/hda10	9658	9729	578308+	82	
Linux swap / Solaris					

如果通过fdisk -l 出现有/dev/hda字样的，我们就要用下面的命令来安装；

```
[root@localhost ~]# grub-install /dev/hda
Installation finished. No error reported.
This is the contents of the device map
/boot/grub/device.map.
Check if this is correct or not. If any of the lines is
incorrect,
fix it and re-run the script `grub-install'.

(fd0) /dev/fd0
(hd0) /dev/hda
```

如果是您fdisk -l 出现的有/dev/hdb呢，那就如下运行；

```
[root@localhost ~]# grub-install /dev/hdb
```

如果既有/dev/hda和/dev/hdb 就安装到/dev/hda中；

```
[root@localhost ~]# grub-install /dev/hda
```

值得注意的是如果您有一个/boot分区，应该用如下的办法来安装；

```
[root@localhost ~]#grub-install --root-directory=/boot
/dev/hda

[root@localhost ~]#grub-install --root-directory=/boot
/dev/hdb
```

注解：具体是/dev/hda还是/dev/hdb，请以fdisk -l 为准；如果两个都有，就看您把/boot分区是放在第一块硬盘还是第二块硬盘上了，以实际情况为准；

## 2、设定GRUB的/boot分区并写入MBR；；

在Linux中，GRUB软件包的安装，及在配置过程中安装grub到/boot中还是不够的，还要把GRUB，写入MBR才行；有时我们重新安装了Windows，Windows会把MBR重写，这样GRUB就消失了；如果您出现这样的情况，就要进行这个过程；

```
[root@localhost ~]# grub
```

会出现grub>提示符，这是grub命令行模式，如果能在开机中出现提示符，没有引导不起来的系统，除非您的系统破坏的极为严重。如果仅仅是GRUB被破坏了，GRUB命令行是能让操作系统引导起来的；

接着看例子，我们要找到 /boot/grub/stage1的，在grub>后面输入；

```
grub> find /boot/grub/stage1
(hd0,6)
(fd0) 注：这个是软驱；现在很少用软驱了，如果您有这方面的需要，自己看GRUB的DOC吧；
```

注解：

(hd0,6) 这是/boot所在的分区; 不要误解为是Linux 的/所在的分区, 这是值得注意的;

(fd0) 注: 这个是软驱; 现在很少用软驱了, 如果您有这方面的需要, 自己看GRUB的DOC吧;

```
grub>root (hd0,6)    注: 这是/boot所在的分区;
grub>setup (hd0)    注: 把GRUB写到MBR上;
```

注解:

上面这步骤是根据 find /boot/stage1 而来的, 仔细看一下就明白了; 现在我们一般安装很少会把/boot分区列为一个单独的分区; 不过有的弟兄可能也喜欢这么做; 所以还是有必要说一下为好;

## 四、GRUB的配置文件的menu.lst的写法;

对于GRUB来说, 如果没有配置menu.lst, 无论是Linux版本的GRUB, 还是WINGRUB, 都会有命令行可用, 通过命令行是一样能把操作系统引导起来的; 有些弟兄总以为menu.lst 配置错了, 或者在机器启动后出现grub> 命令行模式就要重新安装系统, 其实根本没有这个必要; 只要学会GRUB的命令行的用法, 根本没有必要重装系统;

menu.lst 位于/boot/grub目录中, 也就是/boot/grub/menu.lst 文件; 您可以用vi或您喜欢的编辑器来编辑他; 如果您不会用vi, 还是去学习一下吧; 简单的用法怎么也得会, 对不对? 毕竟这个文档不是讲vi的用法的;

有的弟兄会说, 我没有menu.lst怎么办? 那就创建一个; 用下面的命令;

```
[root@localhost ~]# touch /boot/grub/menu.lst
```

然后我们再做一个/boot/grub/menu.lst 的链接 /boot/grub/grub.conf

```
[root@localhost ~]# cd /boot/grub
[root@localhost ~]# ln -s menu.lst grub.conf
```

现在我们来写GRUB的menu.lst了, 因为/boot/grub/grub.conf是 /boot/grub/menu.lst的链接文件, 改哪个都行。链接文件相当于Windows的快捷方式, 这样可能能更好的理解;

### 1、menu.lst的写法之一;

首先我们看一下我的Fedora 4.0 中的/boot/grub/menu.lst 的内容;

```
default=0
timeout=5
#splashimage=(hd0,6)/boot/grub/splash.xpm.gz
hiddenmenu
title Fedora Core (2.6.11-1.1369_FC4)
    root (hd0,6)
    kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
    initrd /boot/initrd-2.6.11-1.1369_FC4.img
title WinXp
    rootnoverify (hd0,0)
    chainloader +1
```

注解:

```
default=0
```

default=0 是默认启动哪个系统, 从0开始; 每个操作系统的启动的定义都从title开始的, 第一个title 在GRUB的启动菜单上显示为0, 第二个启动为1, 以此类推;

```
timeout=5
```

注: 表示在开机后, GRUB画面出现几秒后开始以默认启动; 如果在启动时, 移动上下键, 则解除这一规则;

#splashimage=(hd0,6)/boot/grub/splash.xpm.gz 注: GRUB的背景画面, 这个是可

选项；我不喜欢GRUB的背景画面，所以加#号注掉，也可以删除；

```
hiddenmenu
```

注解：隐藏GRUB的启动菜单，这项也是可选的，也可以用#号注掉；

一般的情况下对Linux操作系统的启动，一般要包括四行；title 行；root行；kernel行；initrd 行；

1) 在menu.lst中，通过 root (hd[0-n],y)来指定/boot 所在的分区；

title XXXXX 注：title 后面加一个空格，title 是小写的，后面可以自己定义；比如FC4，自己定义一个名字就行；

root (hd[0-n],y)，在本例中，我们看到的是root (hd0,6) ,root (hd[0-n],y)表示的是/boot所在的分区；有时我们安装Linux的时候，大多是不设置/boot的，这时/boot和/所在的同一个分区；这个root (hd[0-n],y)很重要，因为/boot目录中虽然有grub目录，最为重要的是还有kernel 和initrd文件，这是Linux能启动起来最为重要东西；

有的弟兄会问，root (hd[0-n],y)是怎么来的？

请参考：《在Linux系统中存储设备的两种表示方法》

2) 在menu.lst中，kernel 命令行的写法；

kernel 一行，是通指定内核及Linux的/分区所在位置；

比如例子中是；

```
kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro root=LABEL=/
```

在这里以kernel 起始，指定Linux的内核的文件所处的绝对路径；因为内核是处在/boot目录中的，如果/boot是独立的一个分区，则需要把boot省略；如果/boot是独立的分区，这行要写成：

```
kernel /vmlinuz-2.6.11-1.1369_FC4 ro root=LABEL=/
```

因为/boot所处的分区已经在title 下一行root (hd[0-n],y)中指定了，所以就无需再指明内核处在哪个分区了；另外Linux系统的硬盘分区的挂载配置文件在/etc/fstab，原理是通过 mount /dev/hd[a-z]X /boot 来进行的；您可以对照着来理解；

ro 表示只读；root=LABEL=/ 来表示Linux的根所处的分区。LABEL=/ 这是硬盘分区格式化为相应文件系统后所加的标签；如果您不了解什么是标签，也可以直接以/dev/hd[a-z]X 或者/dev/sd[a-z]X来表示；就看您的Linux是根分区是在哪个分区了。比如我是在/dev/hda7，那这里就可以写成root=/dev/hda7；

如果查看系统运行所挂载的分区，请用 df -lh 来查看，就能明白是不是/boot是独立的分区，或者查看/etc/fstab也能知道；

```
[root@localhost ~]# df -lh
Filesystem          容量  已用  可用  已用%  挂载点
/dev/hda7           11G   9.2G  1.2G   90%   /
/dev/shm            236M     0   236M    0%   /dev/shm
```

在这个例子中，我们可以发现 /boot并没有出现只有/dev/hda7，这表示/boot并不是独立的一个分区；所有的东西都包含在/中；于是我们在/boot中查看内核版本；

```
[root@localhost ~]# ls /boot/vmlinuz*
/boot/vmlinuz-2.6.11-1.1369_FC4  注：看到内核vmlinuz所处的目录；
```

于是我们就可以这样kernel 这行了；

```
kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro root=/dev/hda7
```

3) initrd 命令行的写法；

如果是/boot独立一个分区，initrd 一行要把/boot中省略；如果/boot不是处于一个分区，而是和Linux的/分区处于同一分区，不应该省略；

比如我们在2) 中用的例子；现在拿到这里，我们应该首先查看 /boot中的initrd 的文件名到底是什么；



```
[root@localhost ~]# ls /boot/initrd*
/boot/initrd-2.6.11-1.1369_FC4.img
```

如果是通过df -lh 得知或查看/etc/fstab 也行，得知/boot是独立的分区；这时initrd 应该写成；

```
initrd /initrd-2.6.11-1.1369_FC4.img
```

如果是 /boot不是独处一个分区，而是在/同一处一个分区，则要写成；

```
initrd /boot/initrd-2.6.11-1.1369_FC4.img
```

4) menu.lst第一种写法的总结和实践；

在这里，我们只说重要的，不重要的就一带而过了；

1] 用fdisk -l ; df -lh ; more /etc/fstab来确认分区情况；

我们过fdisk -l ; df -lh ; more /etc/fstab 来确认/boot所在的分区，及Linux的根分区所在位置；

比如我们确认/boot和Linux的分区同处一个分区；

```
[root@localhost ~]# df -lh
Filesystem          容量  已用  可用  已用%  挂载点
/dev/hda7            11G  9.2G  1.2G   90%  /
/dev/shm             236M    0  236M    0%  /dev/shm
```

然后我们/etc/fstab 中,查看/分所在的分区或分区标签是什么；

```
[root@localhost ~]# more /etc/fstab
# This file is edited by fstab-sync - see 'man fstab-sync'
for details
LABEL=/              /                    ext3
defaults            1 1                    /dev/pts            devpts
gid=5,mode=620      0 0                    /dev/shm            tmpfs
defaults            0 0                    /dev/proc            proc
defaults            0 0                    /dev/sys              sysfs
defaults            0 0                    LABEL=SWAP-hda1      swap
defaults            0 0                    /dev/hdc              /media/cdrecorder   auto
pamconsole,exec,noauto,managed 0 0
```

经过上面的df -lh 和more /etc/fstab 的对照中得知，/boot并非独处一个分区，而是和/在同一个分区；这个Linux系统安装在/dev/hda7上，文件系统（此分区）的标签为LABEL=/，/boot也是处于/dev/hda7，/dev/hda7也可以说是 root (hd0,6)；

2]查看内核vmlinuz的和initrd文件名的全称；

```
[root@localhost ~]# ls -lh /boot/vmlinuz*
-rw-r--r-- 1 root root 1.6M 2005-06-03 /boot/vmlinuz-
2.6.11-1.1369_FC4
[root@localhost ~]# ls -lh /boot/initrd*
-rw-r--r-- 1 root root 1.1M 11月 26 22:30 /boot/initrd-
2.6.11-1.1369_FC4.img
```

3]开始写menu.lst ；

我们根据上面所提到的，可以写成如下的样子；

```
default=0
```

```
timeout=5
title FC4
    root (hd0,6)
    kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
    initrd /boot/initrd-2.6.11-1.1369_FC4.img
```

也可以写成;

```
default=0
timeout=5
title FC4
    root (hd0,6)
    kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
    initrd /boot/initrd-2.6.11-1.1369_FC4.img
```

注解: 上面两个不同之处在于一指定Linux的根/所在的分区时, 一个是用了文件系统的标签, 另一个没有用标签;

## 2、menu.lst的写法之二, 精简型;

本写法主要是把指定/boot所位于的所分区直接写入kernel 指令行; 这样就省略了通过root (hd[0-n],y)来指定/boot所位于的分区;

### 1)第一种情况: /boot和Linux的/根分区在同一个分区;

有前面的那么多的讲解, menu.lst写法之二就好理解多了; 也得分两种情况, 咱们先把/boot并不是独处一个分区, 而是和Linux的根分区处于同一个分区; 我们以

4) menu.lst第一种方法的写法总结 的实例为例子;

```
default=0
timeout=5
title FC4x
    kernel (hd0,6)/boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
    initrd (hd0,6)/boot/initrd-2.6.11-1.1369_FC4.img
```

注解:

**title FC4x** 注: 自己为这个Linux 起个简单的名, 以title开头, 然后一个空格, 后面就自己发挥吧, FC4或FC4x都行;

**kernel** 空格 (hd0,6)/boot/vmlinuz-2.6.11-1.1369\_FC4 空格 ro 空格 root=/dev/hda7

**kernel** 这行这样理解 kernel (boot所在的分区)/boot/内核文件全称 ro  
**root=Linux**根所位于的分区或标签

**initrd** 空格 (hd0,6)/boot/initrd-2.6.11-1.1369\_FC4.img

**initrd** 这行可以这样理解 initrd (/boot所在的分区)/boot/内核文件名全称

### 2) 第二种情况: /boot独立一个分区, 和Linux的根分区不是同一个分区;

比如我们查看到df -lh 得到的是

```
[root@localhost ~]# df -lh
Filesystem          容量  已用  可用  已用%  挂载点
/dev/hda6           200M  120M   80M   60%  /boot
/dev/hda7           11G   9.2G  1.2G   90%  /
```

我们再进行一行查看/etc/fstab 得知;

```
LABEL=/              /                    ext3
defaults            1 1
LABEL=/boot          /boot                ext3
defaults            1 2
```

所以我们应该写成如下的;

```
title FC4x
    kernel (hd0,5)/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
    initrd (hd0,5)/initrd-2.6.11-1.1369_FC4.img
```

因为Linux的根分区是/dev/hda7，通过/etc/fstab和df -h的内容得知标签为LABEL=/的分区就是/dev/hda7，所以有；

```
title FC4x
    kernel (hd0,5)/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
    initrd (hd0,5)/initrd-2.6.11-1.1369_FC4.img
```

## 五、通过GRUB命令行来启动Linux操作系统；

GRUB的命令行才是王道，如果知道怎么用命令行来启动操作系统，那理解menu.lst的写法也不难；也就是说在开机的时候，不用GRUB的菜单，通过GRUB的命令也是一样能把操作系统引导起来。

因为menu.lst的内容就是GRUB的一个一个的指令集合；是不是Linux这玩意很神奇？

### 1、为什么需要学习GRUB的命令；

当我们把GRUB的menu.lst写错的时候，或者丢掉了menu.lst的时候，比如在开机的时候，GRUB会出现grub>类似的命令提示符，这时我们需要用命令行启动系统；当然您可以不用定义GRUB的菜单，直接用命令行来启动系统，比如我现在就是，为了写GRUB的文档，就把menu.lst删除了，直接用命令来启动系统；

### 2、用命令行来引导Linux操作系统的步骤；

通过命令行来引导操作系统的流程，也没有什么难的；无非是把指令手工输入到grub>提示符的后面；在这个过程中，tab键的命令补齐功能就显得很重要了。如果您不知道有哪些命令，可以输入help；

#### 1) 进入GRUB的命令行模式 grub>

如果开机时，GRUB出现的是grub>，说明你没有/etc/grub/menu.lst，您需要自己写一个才会GRUB的菜单，让我们来选择进入哪个系统。如果有GRUB的菜单，您可以按Ctrl+c组合键进入GRUB的命令行模式，会出现grub>提示符；

```
grub>
```

#### 2) 获取帮助GRUB的 help

只要您在grub>提示符的后面输入help就能得到GRUB所有的命令提示；

```
grub> help
blocklist FILE                boot
cat FILE                      chainloader [--force]
FILE
clear                          color NORMAL
[HIGHLIGHT]
configfile FILE              device DRIVE DEVICE
displayapm                   displaymem
find FILENAME                geometry DRIVE
[CYLINDER HEAD SECTOR [
halt [--no-apm]              help [--all]
[PATTERN ...]
hide PARTITION               initrd FILE [ARG ...]
kernel [--no-mem-option] [--type=TYPE] makeactive
map TO_DRIVE FROM_DRIVE     md5crypt
module FILE [ARG ...]       modulenounzip FILE
[ARG ...]
```

```

pager [FLAG]                                partnew PART TYPE
START LEN
parttype PART TYPE                          quit
reboot                                       root [DEVICE
[HDBIAS]]
rootnoverify [DEVICE [HDBIAS]]              serial [--unit=UNIT]
[--port=PORT] [--
setkey [TO_KEY FROM_KEY]                   setup [--prefix=DIR]
[--stage2=STAGE2_
terminal [--dumb] [--no-echo] [--no-ed terminfo [--
name=NAME --cursor-address
testvbe MODE                                unhide PARTITION
uppermem KBYTES                             vbeprobe [MODE]

```

如果需要得到某个指令的帮助，就在 `help` 后面空一格，然后输入指令，比如；

```
grub>help kernel
```

### 3) cat的用法；

`cat`指令是用来查看文件内容的，有时我们不知道Linux的/boot分区，以及/根分区所在的位置，要查看/etc/fstab的内容来得知，这时，我们就要用到`cat (hd[0-n],y)/etc/fstab`来获得这些内容；注意要学会用`tab`键命令补齐的功能；

```

grub> cat (      按tab 键会出来hd0或hd1之类的；
grub> cat (hd0, 注：输入hd0,然后再按tab键；会出来分区之类的；
grub> cat (hd0,6)
Possible partitions are:
  Partition num: 0,  Filesystem type unknown, partition
type 0x7
  Partition num: 4,  Filesystem type is fat, partition
type 0xb
  Partition num: 5,  Filesystem type is reiserfs,
partition type 0x83
  Partition num: 6,  Filesystem type is ext2fs, partition
type 0x83
  Partition num: 7,  Filesystem type unknown, partition
type 0x83
  Partition num: 8,  Filesystem type is reiserfs,
partition type 0x83
  Partition num: 9,  Filesystem type unknown, partition
type 0x82

grub> cat (hd0,6)/etc/fstab 注：比如我想查看一下
(hd0,6)/etc/fstab的内容就这样输入；

LABEL=/                                /                                ext3
defaults                                1 1
/dev/devpts                             /dev/pts                         devpts
gid=5,mode=620                          0 0
/dev/shm                                 /dev/shm                         tmpfs
defaults                                0 0
/dev/proc                                /proc                             proc
defaults                                0 0
/dev/sys                                 /sys                              sysfs
LABEL=SWAP-hda1                          swap                              swap
defaults                                0 0
/dev/hdc                                  /media/cdrecorder                auto
pamconsole,exec,noauto,
managed 0 0

```

有的弟兄可能会说，我不知道Linux安装在了哪个分区。那根据文件系统来判断一个

一个的尝试总可以吧我；只要能cat出/etc/fstab就能为以后引导带来方便；

主要查看/etc/fstab中的内容，主要是Linux的/分区及/boot是否是独立的分区；如果没有/boot类似的行，证明/boot和Linux的/处于同一个硬盘分区；比如上面的例子中LABEL=/ 这行是极为重要的；说明Linux系统就安在标签为LABEL=/的分区中；

如果您的Linux系统/boot和/没有位于同一个分区，可能cat (hd[a-n],y) 查到的是类似下面的；

LABEL=/		/	ext3
defaults	1 1		
LABEL=/boot		/boot	ext3
defaults	1 2		

4) root (hd[0-n],y) 指令来指定/boot所在的分区；

其实这个root (hd[0,n],y)是可以省略的，如果省略了，我们要在kernel 命令中指定；我们前面已经说过 (hd[0-n],y) 硬盘分区的表示方法的用途；主要是用来指定 /boot所在的分区；

比如我们确认/boot和 (hd0,6)，所以就可以这样来输入root (hd0,6)

```
grub> root (hd0,6)
```

如果发现不对，可以重新来过；没有什么大不了的；

5) kernel 指令，用来指定Linux的内核，及/所在的分区；

kernel 这个指令可能初学者有点怕，不知道内核在哪个分区，及内核文件名的全称是什么。不要忘记tab键的命令补齐的应用；

如果我们已经通过root (hd[0-n],y) 指定了/boot所在的分区，语法有两个：

如果/boot和Linux的/位于同一个分区，应该是下面的一种格式；

kernel /boot/vmlinuz在这里按tab键来补齐，就看到内核全称了 ro  
root=/dev/hd[a-z]X

如果/boot有自己独立的分区，应该是；

kernel /vmlinuz在这里按tab键来补齐，就看到内核全称了 ro  
root=/dev/hd[a-z]X

在这里 root=/dev/hd[a-z]X 是Linux 的/根所位于的分区，如果不知道是哪个分区，就用tab出来的来计算，一个一个的尝试；或用cat (hd[0-n],y)/etc/fstab 中得到Linux的/所在的分区或分区的标签；

```
grub> kernel /boot/ 在这里按tab键；这样就列出/boot中的文件了；
Possible files are: grub initrd-2.6.11-1.1369_FC4.img
System.map-2.6.11-1.1369
_FC4 config-2.6.11-1.1369_FC4 vmlinuz-2.6.11-1.1369_FC4
grubBAK memtest86+-1.55
.l xen-syms xen.gz

grub> kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
[Linux-bzImage, setup=0x1e00, size=0x18e473]
```

注解：root=LABEL=/ 是Linux的/所在的分区的文件系统的标签；如果您知道Linux的/在哪个具体的分区，用root=/dev/hd[a-z]X来指定也行。比如下面的一行也是可以的；

```
grub> kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
```

也可以把/boot所在的分区的指定 root (hd[0-n],y)这行省掉，直接在kernel 中指定/boot所在的分区；所以就在下面的语法；

如果是/boot和Linux的根同处一个分区；

```
kernel (hd[0-n],y)/boot/vmlinuz ro root=/dev/hd[a-z]X
```

比如:

```
grub>kernel
```

如果是/boot和Linux所在的根不在一个分区; 则是;

```
kernel (hd[0-n],y)/vmlinuz ro root=/dev/hd[a-z]X
```

```
grub> kernel (hd0,6)/boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
[Linux-bzImage, setup=0x1e00, size=0x18e473]
```

或下面的输入, 以cat 出/etc/fstab内容为准;

```
grub> kernel (hd0,6)/boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
[Linux-bzImage, setup=0x1e00, size=0x18e473]
```

6) initrd 命令行来指定initrd文件;

```
grub> initrd /boot/initrd在这里tab 来补齐;
grub> initrd /boot/initrd-2.6.11-1.1369_FC4.img
[Linux-initrd @ 0x2e1000, 0x10e685 bytes]
```

如果/boot是独立的一个分区, 应该是如下样子的语法; 比如下面的;

```
grub> initrd /initrd在这里tab 来补齐;
grub> initrd /initrd-2.6.11-1.1369_FC4.img
[Linux-initrd @ 0x2e1000, 0x10e685 bytes]
```

7) boot 引导系统;

```
grub>boot
```

前面的几个步骤都弄好。就进入引导; 尝试一下就知道了。。

8) 引导Linux系统实例全程回放;

实例: /boot和Linux的/处于同一个硬盘分区;

```
grub> cat (hd0,6)/etc/fstab
# This file is edited by fstab-sync - see 'man fstab-sync'
for details
LABEL=/                                /                                ext3
defaults                                1 1
/dev/devpts                              /dev/pts                          devpts
gid=5,mode=620 0 0
/dev/shm                                  /dev/shm                          tmpfs
defaults                                0 0
/dev/proc                                  /proc                              proc
defaults                                0 0
/dev/sys                                  /sys                              sysfs
defaults                                0 0
LABEL=SWAP-hda1                          swap                              swap
defaults                                0 0
/dev/hdc                                  /media/cdrecorder                 auto
pamconsole,exec,noauto,managed 0 0

grub> root (hd0,6)
Filesystem type is ext2fs, partition type 0x83

grub> kernel /boot/在这里按tab补齐, 全列出/boot所有的文件;
Possible files are: grub initrd-2.6.11-1.1369_FC4.img
```

```

System.map-2.6.11-1.1369_FC4 config-2.6.11-1.1369_FC4
vmlinuz-2.6.11-1.1369_FC4
memtest86+-1.55.1 xen-syms xen.gz

grub> kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7 注: 输入
    [Linux-bzImage, setup=0x1e00, size=0x18e473]

grub> initrd /boot/在这里按tab补齐
Possible files are: grub initrd-2.6.11-1.1369_FC4.img
System.map-2.6.11-1.1369_FC4 config-2.6.11-1.1369_FC4
vmlinuz-2.6.11-1.1369_FC4
memtest86+-1.55.1 xen-syms xen.gz

grub> initrd /boot/initrd-2.6.11-1.1369_FC4.img 注: 输
入initrd文件名的全名;
    [Linux-initrd @ 0x2e1000, 0x10e685 bytes]

grub> boot

```

我们指定Linux的根时，可以用cat出来的fstab的内容中Linux的/分区文件系统标签来替代；也就是kernel那行中root=/dev/hd[a-z]X；

```

grub> cat (hd0,6)/etc/fstab
# This file is edited by fstab-sync - see 'man fstab-sync'
for details
LABEL=/                /                    ext3
defaults              1 1
/dev/devpts           /dev/pts             devpts
gid=5,mode=620       0 0
/dev/shm              /dev/shm             tmpfs
defaults              0 0
/dev/proc             /proc                proc
defaults              0 0
/dev/sys              /sys                 sysfs
LABEL=SWAP-hda1      swap                 swap
defaults              0 0

/dev/hdc              /media/cdrecorder    auto
pamconsole,exec,noauto,managed 0 0

grub> root (hd0,6)
Filesystem type is ext2fs, partition type 0x83

grub> kernel /boot/在这里按tab补齐, 全列出/boot所有的文件;
Possible files are: grub initrd-2.6.11-1.1369_FC4.img
System.map-2.6.11-1.1369_FC4 config-2.6.11-1.1369_FC4
vmlinuz-2.6.11-1.1369_FC4
memtest86+-1.55.1 xen-syms xen.gz

grub> kernel /boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
    [Linux-bzImage, setup=0x1e00, size=0x18e473]

grub> initrd /boot/在这里按tab补齐
Possible files are: grub initrd-2.6.11-1.1369_FC4.img
System.map-2.6.11-1.1369_FC4 config-2.6.11-1.1369_FC4
vmlinuz-2.6.11-1.1369_FC4 grubBAK
memtest86+-1.55.1 xen-syms xen.gz

grub> initrd /boot/initrd-2.6.11-1.1369_FC4.img 注: 输
入initrd文件名的全名;

```

```
[Linux-initrd @ 0x2e1000, 0x10e685 bytes]
```

```
grub> boot
```

如果是/boot和Linux的根分区不在同一个分区,要把kernel和initrd 指令中的/boot去掉,也就是/vmlinuzMMMMMM 或 /initrdNNNN

也可以不用root (hd[0-n])来指定/boot所在分区,要在kernel 和initrd 中指定;比如Linux的/根所位于的分区和/boot所位于的分区都是(hd0,6),并且我们cat出来的/etc/fstab是Linux的/根分区的文件系统的标签为LABEL=/,引导操作系统的例子如下;

```
grub>kernel (hd0,6)/boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
grub>initrd (hd0,6)/boot/initrd-2.6.11-1.1369_FC4.img
grub>boot
```

或

```
grub>kernel (hd0,6)/boot/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
grub>initrd (hd0,6)/boot/initrd-2.6.11-1.1369_FC4.img
grub>boot
```

如果/boot位于 /dev/hda6,也就是(hd0,5),Linux的根/位于分区/dev/hda7,并且我们cat 出来的/etc/fstab 中/分区的标签为 LABEL=/.下面的两种方法都可以引导;

```
grub>kernel (hd0,5)/vmlinuz-2.6.11-1.1369_FC4 ro
root=LABEL=/
grub>initrd (hd0,5)/initrd-2.6.11-1.1369_FC4.img
grub>boot
```

或

```
grub>kernel (hd0,5)/vmlinuz-2.6.11-1.1369_FC4 ro
root=/dev/hda7
grub>initrd (hd0,5)/initrd-2.6.11-1.1369_FC4.img
grub>boot
```

## 六、通过GRUB引导Windows操作系统;

### 1、通过编辑 menu.lst 来引导Windows 系统;

如果您的Windows所处的分区在(hd0,0),可以在menu.lst 加如下的一段就能引导起来了;

```
title WinXp
    rootnoverify (hd0,0)
    chainloader +1
```

如果您的机器有两块硬盘,而Windows 位于第二个硬盘的第一个分区,也就是(hd1,0)

您可以用grub的map来指令来操作把两块硬盘的序列对调,这样就不用在BIOS中设置了;在menu.lst中加如下的内容,比如下面的;

```
title WinXp
    map (hd0) (hd1)
    map (hd1) (hd0)
    rootnoverify (hd0,0)
    chainloader +1
    makeactive
```

如果Windows的分区不位于硬盘的第一个分区怎么办呢?比如在(hd0,2);这个也好办吧,把rootnoverify 这行的(hd0,0)改为 (hd0,2)



```
title WinXp
    rootnoverify (hd0,2)
    chainloader +1
makeactive
```

如果Windows的在第二个硬盘的某个分区，比如说是位于(hd1,2)，则要用到map指令；

```
title WinXp
    map (hd0) (hd1)
    map (hd1) (hd0)
    rootnoverify (hd1,2)
    chainloader +1
makeactive
```

如果有多个Windows系统，怎么才能引导出来呢？应该用hide和unhide指令操作；比如我们安装了两个Windows，一个是位于(hd0,0)的windows 98，另一个是安装的是位于(hd0,1)的WindowsXP；这时我们就要用到hide指令了；

```
title Win98
    unhide (hd0,0)
    hide (hd0,1)
    rootnoverify (hd0,0)
    chainloader +1
makeactive

title WinXP
    unhide (hd0,1)
    hide (hd0,0)
    rootnoverify (hd0,1)
    chainloader +1
makeactive
```

## 2、通过GRUB指令来引导Windows；

其实我们会写menu.lst了，在menu.lst中的除了title外，都是一条条指令；如果我们启动Windows，只是输入指令就行了；

比如Windows的分区在(hd0,0)，我们在开机后，按ctrl+c，进入GRUB的命令模式；就可以用下面的

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

其它同理... ..

## 七、GRUB丢失或损坏的应对策略；

如果GRUB是Linux版本才会出现这样的问题；WINGRUB可以不写在MBR上；所以不会出现这样的问题。WINGRUB用起来比较简单。menu.lst和命令行的用法和Linux版本的GRUB是一样的；

### 1、由于重新安装Windows或其它未知原因而导致GRUB的丢失；

您可以通过系统安装盘、livecd进入修复模式；

请参考：《Linux系统的单用户模式、修复模式、跨控制台登录在系统修复中的运用》

首先：您根据前面所说grub-install来安装GRUB到/boot所在的分区；要仔细看文档，/boot是不是处于一个独立的分区是重要的，执行的命令也不同；

其次：要执行grub，然后通过root(hd[0-n],y)来指定/boot所位于的分区，然后接着执行setup(hd0)，这样就写入MBR了，比如下面的例子；

```
grub>root (hd0,6)
grub>setup (hd0)
grub>quit
```

重新引导就会再次出现MBR的菜单了或命令行的提示符了；

## 2、如果出现GRUB提示符，而不出现GRUB的菜单，如何引导系统；

存在的问题可能是/boot/grub/menu.lst丢失，要自己写一个才行；您可以用命令行来启动系统，进入系统后写一写menu.lst就OK了。前面已经谈过了；

写好后还要建一个grub.conf的链接，如下：

```
[root@localhost ~]# cd /boot/grub
[root@localhost grub]# ln -s menu.lst grub.conf
```

## 八、关于GRUB的未尽事宜；

GRUB有很多内容，比如对BSD的引导，还有一些其它指令的用法，我并没有在本文提到；主要我目前还未用到，如果您需要了解更多，请查看《GNU GRUB 手册和FAQ》

## 九、关于本文；

本文前后写了三四天，中间发现并不能把Linux设备的两种表现形式说的清楚，于是被迫写了《在Linux系统中存储设备的两种表示方法》；由于没有BSD系统，所以没有写关于BSD的引导；如果正在用BSD的弟兄如果有时间不妨写一写；写的时候注意文档的结构，这样方便大家的阅读；

GRUB有很多内容，需要大家慢慢的学习和研究；有的弟兄抑制GRUB，说不如NTLOADER，其实这是错误的；如果您想学习和使用Linux就得学习和适应Linux的操作；习惯成自然，如果您抵制学习Linux，那可能您永远会说“Linux不如Windows”；

## 十、参考文档；

《GNU GRUB 手册和FAQ》

## 十一、相关文档；

《以WINGRUB 引导安装Fedora 4.0 为例，详述用WINGRUB来引导Linux的安装》

《系统引导管理器GRUB学习笔记》

《如何为GRUB系统引导管理器加上密码》

《在Linux系统中存储设备的两种表示方法》

《合理规划您的硬盘分区》

《系统引导过程及硬盘分区结构论述》

《Linux 查看磁盘分区、文件系统、使用情况的命令和相关工具介绍》

《实例解说 fdisk 使用方法》



Linux公社 (LinuxIDC.com) 于2006年9月25日注册并开通网站, Linux现在已经成为一种广受关注和支持的一种操作系统, IDC是互联网数据中心, LinuxIDC就是关于Linux的数据中心。

LinuxIDC.com提供包括Ubuntu, Fedora, SUSE技术, 以及最新IT资讯等Linux专业类网站。

并被收录到Google 网页目录-计算机 > 软件 > 操作系统 > Linux 目录下。

Linux公社 (LinuxIDC.com) 设置了有一定影响力的Linux专题栏目。

包括:

[Ubuntu专题](#)

[Fedora专题](#)

[RedHat专题](#)

[SUSE专题](#)

[红旗Linux专题](#)

[Android专题](#)

---

[Linux公社简介](#) - [广告服务](#) - [网站地图](#) - [帮助信息](#) - [联系我们](#)

本站 (LinuxIDC) 所刊载文章不代表同意其说法或描述, 仅为提供更多信息, 也不构成任何建议。

本站带宽由[\[6688.CC\]](#)友情提供

Copyright © 2006-2011 [Linux公社](#) All rights reserved