

第5章 引导和关机

这部分主要讲解Linux系统引导和关机过程中做了些什么，以及如何才能做到正确引导和关机。如果采取的作法不正确，可能导致文件被讹用或丢失。

5.1 概论

打开计算机，导致加载其操作系统的过程就叫作“引导”（早期的计算机中，只打开计算机是不够的，还必须手工加载操作系统。现在的计算机好多了，一切引导行为都是自动的）。“引导”一词源于计算机通过引导程序，把自己拔起，但自举这一动作更为形象。

引导过程中，计算机先加载一小段被称为“自举程序”的代码，该程序依次加载并启动操作系统。自举程序通常保存在硬盘或软盘上的固定位置。采用两个步骤的原因是操作系统大而复杂，但计算机载入的第一段代码又必须非常之小（只能有几百个字节），以避免固件过于复杂。

不同的计算机执行引导的方式是不同的。对个人电脑来说，计算机（其 BIOS）在软盘或硬盘上的第一个扇道（被称为引导区）进行读取。自举程序就包含在这个扇道内。它从磁盘上的某个地方（或别的地方）载入操作系统。

Linux系统已经载入之后，它便初始化硬件和设备驱动程序，然后运行 init。init启动别的进程，以便允许用户登录和开始工作。关于这部分的详情，参见随后的讨论。

为了关闭Linux系统，首先要求所有的进程终止（其目的是为了让它们关闭所有的文件，执行必要的操作保持干净退出），然后，取消装入文件系统和交换区，最后向控制台打印出一条消息，声明可以关机。如果不遵循正确的关机进程，可能会导致非常严重的后果；更为重要的是，文件系统缓冲区可能没有得到刷新，这意味着该缓冲区内的所有数据会被丢失，磁盘上的文件系统不一致，从而导致文件系统不能使用。

5.2 深入研究引导过程

Linux系统的引导，既可从软盘，又可从硬盘开始。Linux文档项目（<http://MetaLab.unc.edu/LDP/LDP/gs/gs.html>）中，由Matt Welsh等编写的《安装和启动指南》内的安装部分将告诉你如何安装Linux系统，以便按自己的方式启动它。

启动PC机时，BIOS将进行各种各样的测试，检查一切是否正常（这称为电源自检，或简称POST），然后开始真正的引导进程。它将选择磁盘驱动（如果已插入软盘的话，一般是第一个软区；不然的话，如果计算机内已安装硬盘的话，就选择第一个硬盘；但其顺序是可以配置的），然后开始读取最前面的第一个扇区。这就是引导区；对硬盘而言，还可把它称为“主引导记录”，因为一个硬盘中可包含若干个分区，每个分区都有自己的引导区。

引导区内包含一个很小的程序（小得以至于一个扇区就能装下它），其责任是从磁盘上读取并启动真正的操作系统。通过软盘引导Linux系统时，引导区内包含有一小段代码（即引导程序，它从内存中的预定好的位置读取第一个只有几百个字节的数据块，数据块的大小和实际上的内核有关）。Linux启动盘上，没有文件系统，内核只保存在数个连续的扇区内，因为

这样可以简化启动进程。但是，要通过带有文件系统的软盘来启动操作系统，也是可行的，只要你用LILO（即Linux装载程序）。

从硬盘启动时，主引导记录中的代码将检查分区表（同样位于主引导记录内），鉴定活动分区（该分区已被标注为可引导的），再从这个活动分区内读取引导区，然后启动引导区内的代码。分区引导区内的代码和软盘引导区的作用是一样的：从分区内读取并启动内核。但是其细节不同，因为用一个单独的分区来保存内核映像通常是没有用的，所以该分区引导区内的代码不能连续读取硬盘。对于这个问题，有几种解决办法，但最常用的是利用 LILO（如何使用这个程序不在我们的讨论之列；有关详情，参见 LILO文档，该文档讲得非常全面）。

利用LILO启动系统时，它一般会直接读取并启动默认内核。此外，对 LILO进行配置，使其能够启动多个内核之一，甚至除了 Linux之外的其他操作系统也是可能的，对用户来说，也可在启动时，选择准备启动的内核和操作系统。LILO是可以配置的，因此，如果在启动时（装载LILO时），同时按下Alt、Shift和Ctrl这三个键时，LILO会问你准备启动什么，而不是立即启动默认设置。另外，你可以配置 LILO，令其无论如何都询问，同时指定一个可选的超时值，一旦超过这个时间，就启动默认内核。

利用LILO时，还可以在内存或操作系统名之后，指定一个内核命令行参数。

软盘启动和硬盘启动都各有千秋，但一般情况下从硬盘启动更好一些，因为这样可避免在一大堆软盘中找启动盘。而且，从硬盘启动要快得多。但是，从硬盘安装准备启动的系统是非常麻烦的，所以许多人一般先从软盘启动，然后在采取别的方式安装并使用系统时，安装LILO，从硬盘上开始启动进程。

Linux内核已经被读入内存之后，真正的启动进程开始了，大致会出现下面这些情况：

Linux内核是压缩过后安装的，所以第一步是自行解压。内核映像的开始处有一个专门为此设计的小程序。

如果你有Linux能够识别的超级VGA卡，而且有某些特殊的文本模式（比如 100列×40行），Linux将询问你，准备采用哪种模式。内核编辑期间，如果预先设置一个映像模式，它就不会提出此类问题。这同样也是通过 LILO或rdev来完成的。

之后，内核检查还存在哪些其他的硬件（硬盘、软盘、网络适配器等），并配置相应的设备驱动程序；在它进行这个步骤期间，它将输出自己找到的信息。例如，我在启动它时，它的输出信息就是这样的：

```
LILO boot:
Loading linux.
Console: colour EGA+ 80x25, 8 virtual consoles
Serial driver version 3.94 with no serial options enabled
tty00 at 0x03f8 (irq = 4) is a 16450
tty01 at 0x02f8 (irq = 3) is a 16450
lp_init: lp1 exists (0), using polling driver
Memory: 7332k/8192k available (300k kernel code, 384k reserved, 176k data)
Floppy drive(s): fd0 is 1.44M, fd1 is 1.2M
Loopback device init
Warning WD8013 board not found at i/o = 280.
Math coprocessor using irq13 error reporting.
Partition check:
  hda: hda1 hda2 hda3
VFS: Mounted root (ext filesystem).
Linux version 0.99.p19-1 (root@haven) 05/01/93 14:12:20
```

不同系统上，文本的具体模式是不一样的，这和硬件、所用的 Linux 版本以及模式的配置方法有关。

然后，内核将试着装入根文件系统。装入点是在编辑时间或任何时候，利用 LILO 或 rdev 配置的。文件系统类型可以自动侦测。如果文件系统装入失败了，比如说因为你忘了把相应的文件系统驱动程序包括在内核内，内核就会手足无措，停止不前了（只能这样了）。

根文件系统通常采用只读装入（可以采用设置装入点的方式来设置）。这样一来，就可以令其在装入期间检查文件系统；检查采用只写装入的文件系统，不见得是个好主意。

在此以后，内核便在后台启动 init 程序（位于 /sbin/init）。init 从事各种各样的启动杂活。其具体活动和它的配置有关；关于这方面的详情，可参考第 6 章。有一点是确定的，那就是它起码会启动某些基本的后台程序。

然后，init 切换到多用户模式，启动虚拟控制台和串行线路的 getty。getty 是一个程序，让人们通过虚拟控制台和串行终端进行登录。init 还可能启动其他一些程序，这和它的具体配置有关。

之后，启动完成，系统激活并开始正常运行。

5.3 关机详情

关闭 Linux 系统时，遵循正确的规则是非常重要的。如若不然，你的文件系统就可能混乱不堪。这是因为 Linux 有一个磁盘缓冲区，这个缓冲区没有将所有数据立即写入磁盘，而是隔段时间后，再将数据写入磁盘。这样可极大地提高性能，但同时也意味着：如果你一时兴起，随手关掉电源，缓冲区内可能还包含有大量的数据，所以磁盘上的文件系统可能不完整（因为只有其中的部分数据已被写入磁盘）。

反对贸然关掉电源的另一个理由是：在多任务系统中，可能有许多程序正置于后台运行，而关掉电源的后果将是不堪设想的。通过正确的关机顺序，你可以保证所有后台进程都能保存自己的数据。

用于正确关闭 Linux 系统的命令是 shutdown。这个命令通常通过两种方式使用。

如果你运行于单用户系统上，使用 shutdown 的通常方式是退出所有正在运行的程序，注销所有虚拟控制台，以 root 的身份登录到其中一个控制台（或仍然以 root 的身份保持登录状态，但应该转而采用 root 的根目录或 root 目录，以避免取消装入的问题），然后，给出命令 shutdown -h now（可是，你通常不会只运行于单用户系统之上，如果你想延迟，就用一个“+”号和一个以 min 为单位的数来代替 now）。

另一种方法是，如果你的系统有多名用户，就采用命令 shutdown -h +time message，time 指的是系统停止之前，等待的时间，以 min 为单位。而 message 是对为什么关闭系统的简要说明。

```
# shutdown -h +10 'We will install a new disk. System should
> be back on-line in three hours.'
#
```

它将警告各位用户，系统将在 10min 之后关闭，他们最好马上注销，不然就会丢失所有未保存的数据。这条警告消息将在有人登录的每台终端上打印出来，其中包括所有的 X 终端：

```
Broadcast message from root (tty0) Wed Aug 2 01:03:25 1995...
```

```
We will install a new disk. System should  
be back on-line in three hours.
```

```
The system is going DOWN for system halt in 10 minutes !!
```

关机之前，这条警告消息会自动出现若干次，随着时间的推进，其出现的时间间隔越来越短。

任何延迟之后，开始真正的关机时，所有文件系统（除开根文件系统）都将卸装，用户进程（如果有人仍在登录状态的话）将被杀死，后台程序将被关闭，所有的文件系统都被卸装，所有的一切活动都将停止下来。事后，init打印出你可以关机的消息。然后，而且只有在这以后，你才能把手伸向电源开关关机。

有些时候，我们不能正确关闭系统，尽管这在好的系统上很少出现。例如，如果内核出错或崩溃，而且通常在出现异常时，我们就完全不可能执行任何新命令。因此，正确的关闭系统，有时会显得很困难。我们只能指望不要出现太严重的数据损毁现象，并正确关闭电源。假如出现的问题并不严重（比如有人拿斧头砍你的键盘），而且内核和更新程序仍然在正常运行，那么一种更好的做法是稍等几分钟，让更新程序有机会刷新你的缓冲区。最后，再关闭电源。

有人喜欢连续执行三次sync命令（sync命令用于刷新缓冲区），等待磁盘I/O终止，再关掉电源。假如当前没有程序运行，利用这个命令，其效果便等同于shutdown命令。但是，由于它不会卸载任何文件系统，所以有可能在ext2fs的“清除文件系统”标记中出现问题。因此，我们一般不推荐使用三次sync命令的方法。

假如你想打破砂锅问到底的话，这里便告诉你：之所以会出现“三次sync”这种看似可笑的做法，原因在于，在早期年代的Unix系统中，磁盘I/O操作的速度非常慢，所以假如你慢慢地连续键入三次sync命令的话，便能为I/O操作留下充裕的时间，让它“刚好”能够完成。

5.4 重新启动

重新启动意味着再一次启动系统。整个过程是：首先将系统完全关闭，切断电源，再重新加电开机。一个较简单的方法是直接要求shutdown命令重新启动系统，而非只是简单地关闭。要想做到这一点，方法是为shutdown命令增加一个-r参数。举个例子来说，你可执行下述命令，要求立即重启：

```
shutdown -r now
```

大多数Linux系统都能在用户用键盘同时按下Ctrl + Alt + Del的时候，自动执行shutdown -r now命令。这样一来，便能立即开始系统的重启。但是，用户按下Ctrl + Alt + Del时系统的反应也是可能修改的。事实上，一种更好的做法是，一旦用户按下该组合键，便预留出一段延迟时间，以便完成一个多用户系统的重启。至于那些任何人都能亲自操作的机器（谁都能用它的键盘），则最好配置成按下Ctrl + Alt + Del后，什么事情也不做！

5.5 单用户模式

shutdown命令亦可用于将系统切换为单用户模式。在这种模式中，任何人都不能登录，只有root才能使用控制台。这对系统管理员来说特别有用，因为对他们的某些任务而言，在系

统正常运行时候，是根本无法进行的。

5.6 应急盘

显然，并非总能从硬盘正常启动一台计算机。举个例子来说，假如你在配置 LILO的时候犯了一个错误，那么系统也许会根本无法启动。在这样的情况下，我们需要通过另一种方式，来完成系统的启动（只要硬件正常，这种方式就一定是可行的）。对典型的PC来说，这便意味着要从软盘启动！

在目前发布的大多数 Linux系统中，都允许操作者在安装时创建一张应急启动软盘。这是一个不错的选项。但请注意，在这样的启动盘中，可能只包含了内核，而且会假定你会用安装盘上的程序来修正以后碰到的任何问题。但某些情况下，那些程序并不够用。例如，可能需要恢复一些以后制作的文件，而那些文件并非是用安装盘上的程序（软件）来制作的。

所以，最好同时再创建一张自定义的根软盘。在由 Graham Chapman编写的《Bootdisk HOWTO》中，给出了一系列如何制作应急盘的指示，极具参考价值。网址是 <http://MetaLag.unc.edu/LDP/HOWTO/Bootdisk-HOWTO.html>。当然，应急盘也不能制作完毕后便搁在一边不管了，还得注意经常更新它的内容。

对于用来装入根软盘的软盘驱动器来说，不得把它用于其他任何目的。因此，假如你只有一部软盘驱动器的时候，便显得非常不便。但是，假如你的内存足够大，便可考虑配置你的启动盘，将根盘载入一个 RAM虚拟盘里（启动软盘的内核需要为此专门配置）。一旦根软盘正确装入RAM盘，软盘驱动器便可自由地装入其他软盘了。