

剖析Linux系统启动过程Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/645/2021\\_2022\\_\\_E5\\_89\\_96\\_E6\\_9E\\_90Linu\\_c103\\_645036.htm](https://www.100test.com/kao_ti2020/645/2021_2022__E5_89_96_E6_9E_90Linu_c103_645036.htm) 本文以RedHat9.0和i386平台为例，剖析了从用户打开电源直到屏幕出现命令行提示符的整个Linux启动过程。并且介绍了启动中涉及到的各种文件。阅读Linux源代码，无疑是深入学习Linux的最好方法。在本文对Linux启动过程的介绍中，我们也尝试从源代码的视角来更深入的剖析Linux的启动过程，所以其中也简单涉及到部分相关的Linux源代码，Linux启动这部分的源码主要使用的是C语言，也涉及到了少量的汇编。而启动过程中也执行了大量的shell(主要是bash shell)所写脚本。为了方便读者阅读，笔者将整个Linux启动过程分成以下几个部分逐一介绍，大家可以参考下图：当用户打开PC的电源，BIOS开机自检，按BIOS中设置的启动设备(通常是硬盘)启动，接着启动设备上安装的引导程序lilo或grub开始引导Linux，Linux首先进行内核的引导，接下来执行init程序，init程序调用了rc.sysinit和rc等程序，rc.sysinit和rc当完成系统初始化和运行服务的任务后，返回init；init启动了mingetty后，打开了终端供用户登录系统，用户登录成功后进入了Shell，这样就完成了从开机到登录的整个启动过程。下面就将逐一介绍其中几个关键的部分：第一部分：内核的引导(核内引导) Red Hat9.0可以使用lilo或grub等引导程序开始引导Linux系统，当引导程序成功完成引导任务后，Linux从它们手中接管了CPU的控制权，然后CPU就开始执行Linux的核心映象代码，开始了Linux启动过程。这里使用了几个汇编程序来引导Linux，这一步泛及到Linux源代码树

中的“arch/i386/boot”下的这几个文件：bootsect.S、setup.S、video.S等。其中bootsect.S是生成引导扇区的汇编源码，它完成加载动作后直接跳转到setup.S的程序入口。setup.S的主要功能就是将系统参数（包括内存、磁盘等，由BIOS返回）拷贝到特别内存中，以便以后这些参数被保护模式下的代码来读取。此外，setup.S还将video.S中的代码包含进来，检测和设置显示器和显示模式。最后，setup.S将系统转换到保护模式，并跳转到0x100000。那么0x100000这个内存地址中存放的是什么代码？而这些代码又是从何而来的呢？0x100000这个内存地址存放的是解压后的内核，因为Red Hat提供的内核包含了众多驱动和功能而显得比较大，所以在内核编译中使用了“makebzImage”方式，从而生成压缩过的内核，在Red Hat中内核常常被命名为vmlinuz，在Linux的最初引导过程中，是通过“arch/i386/boot/compressed/”中的head.S利用misc.c中定义的decompress\_kernel()函数，将内核vmlinuz解压到0x100000的。当CPU跳到0x100000时，将执行“arch/i386/kernel/head.S”中的startup\_32，它也是vmlinux的入口，然后就跳转到start\_kernel()中去了。start\_kernel()是“init/main.c”中的定义的函数，start\_kernel()中调用了一系列初始化函数，以完成kernel本身的设置。start\_kernel()函数中，做了大量的工作来建立基本的Linux核心环境。如果顺利执行完start\_kernel()，则基本的Linux核心环境已经建立起来了。在start\_kernel()的最后，通过调用init()函数，系统创建第一个核心线程，启动了init过程。而核心线程init()主要是来进行一些外设初始化的工作的，包括调用do\_basic\_setup()完成外设及其驱动程序的加载和初始化。并完成文件系统初始化和root文件系统的安装。当

do\_basic\_setup()函数返回init()，init()又打开了/dev/console设备，重定向三个标准的输入输出文件stdin、stdout和stderr到控制台，最后，搜索文件系统中的init程序（或者由init=命令行参数指定的程序），并使用execve()系统调用加载执行init程序。到此init()函数结束，内核的引导部分也到此结束了，第二部分：运行init init的进程号是1，从这一点就能看出，init进程是系统所有进程的起点，Linux在完成核内引导以后，就开始运行init程序，。init程序需要读取配置文件/etc/inittab。inittab是一个不可执行的文本文件，它有若干行指令所组成。在Redhat系统中，inittab的内容如下所示(以“###”开始的中注释为笔者增加的)：## inittab This file describes how the INIT process should set up # the system in a certain run-level. ## Author: Miquel van Smoorenburg, # Modified for RHS Linux by Marc Ewing and Donnie Barnes ## Default runlevel. The runlevels used by RHS are: # 0 - halt (Do NOT set initdefault to this) # 1 - Single user mode # 2 - Multiuser, without NFS (The same as 3, if you do not havenetworking) # 3 - Full multiuser mode # 4 - unused # 5 - X11 # 6 - reboot (Do NOT set initdefault to this) # 100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)