

阮一峰的网络日志 » [首页](#) » [档案](#)



分类：[理解计算机](#)

上一篇：[如何让搜索引擎抓取A](#)

下一篇：[人类的心理行为模式--](#)

Linux 的启动流程

作者：阮一峰

日期：2013年8月17日

半年前，我写了[《计算机是如何启动的？》](#)，探讨BIOS和主引导记录的作用。

那篇文章不涉及操作系统，只与主板的板载程序有关。今天，我想接着往下写，探讨操作系统接管硬件以后发生的事情，也就是操作系统的启动流程。



这个部分比较有意思。因为在BIOS阶段，计算机的行为基本上被写死了，程序员可以做的事情并不多；但是，一旦进入操作系统，程序员几乎可以定制所有方面。所以，这个部分与程序员的关系更密切。

我主要关心的是Linux操作系统，它是目前服务器端的主流操作系统。下面的内容针对的是[Debian](#)发行版，因为我对其他发行版不够熟悉。

第一步、加载内核

操作系统接管硬件以后，首先读入 `/boot` 目录下的内核文件。



以我的电脑为例，/boot 目录下面大概是这样一些文件：

```
$ ls /boot  
  
config-3.2.0-3-amd64  
config-3.2.0-4-amd64  
grub  
initrd.img-3.2.0-3-amd64  
initrd.img-3.2.0-4-amd64  
System.map-3.2.0-3-amd64  
System.map-3.2.0-4-amd64  
vmlinuz-3.2.0-3-amd64  
vmlinuz-3.2.0-4-amd64
```

第二步、启动初始化进程

内核文件加载以后，就开始运行第一个程序 /sbin/init，它的作用是初始化系统环境。

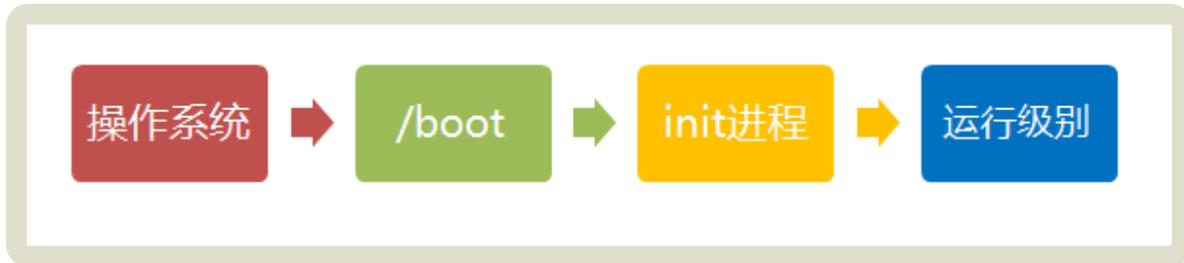


由于init是第一个运行的程序，它的进程编号（pid）就是1。其他所有进程都从它衍生，都是它的子进程。

第三步、确定运行级别

许多程序需要开机启动。它们在Windows叫做"服务" (service) , 在Linux就叫做"守护进程" (daemon) 。

init进程的一大任务, 就是去运行这些开机启动的程序。但是, 不同的场合需要启动不同的程序, 比如用作服务器时, 需要启动Apache, 用作桌面就不需要。Linux允许为不同的场合, 分配不同的开机启动程序, 这就叫做"运行级别" (runlevel) 。也就是说, 启动时根据"运行级别", 确定要运行哪些程序。



Linux预置七种运行级别 (0-6) 。一般来说, 0是关机, 1是单用户模式 (也就是维护模式) , 6是重启。运行级别2-5, 各个发行版不太一样, 对于Debian来说, 都是同样的多用户模式 (也就是正常模式) 。

init进程首先读取文件 /etc/inittab , 它是运行级别的设置文件。如果你打开它, 可以看到第一行是这样的:

```
id:2:initdefault:
```

initdefault的值是2, 表明系统启动时的运行级别为2。如果需要指定其他级别, 可以手动修改这个值。

那么, 运行级别2有些什么程序呢, 系统怎么知道每个级别应该加载哪些程序呢?回答是每个运行级别在/etc目录下面, 都有一个对应的子目录, 指定要加载的程序。

```
/etc/rc0.d  
/etc/rc1.d  
/etc/rc2.d
```

```
/etc/rc3.d  
/etc/rc4.d  
/etc/rc5.d  
/etc/rc6.d
```

上面目录名中的"rc"，表示run command（运行程序），最后的d表示directory（目录）。下面让我们看看 /etc/rc2.d 目录中到底指定了哪些程序。

```
$ ls /etc/rc2.d  
  
README  
S01motd  
S13rpcbind  
S14nfs-common  
S16binfmt-support  
S16rsyslog  
S16sudo  
S17apache2  
S18acpid  
...
```

可以看到，除了第一个文件README以外，其他文件名都是"字母S+两位数字+程序名"的形式。字母S表示Start，也就是启动的意思（启动脚本的运行参数为start），如果这个位置是字母K，就代表Kill（关闭），即如果从其他运行级别切换过来，需要关闭的程序（启动脚本的运行参数为stop）。后面的两位数字表示处理顺序，数字越小越早处理，所以第一个启动的程序是motd，然后是rpcbing、nfs.....数字相同时，则按照程序名的字母顺序启动，所以rsyslog会先于sudo启动。

这个目录里的所有文件（除了README），就是启动时要加载的程序。如果想增加或删除某些程序，不建议手动修改 /etc/rcN.d 目录，最好是用一些专门命令进行管理（参考[这里](#)和[这里](#)）。

第四步、加载开机启动程序

前面提到，七种预设的"运行级别"各自有一个目录，存放需要开机启动的程序。不难想

到，如果多个“运行级别”需要启动同一个程序，那么这个程序的启动脚本，就会在每一个目录里都有一个拷贝。这样会造成管理上的困扰：如果要修改启动脚本，岂不是每个目录都要改一遍？

Linux的解决办法，就是七个 /etc/rcN.d 目录里列出的程序，都设为链接文件，指向另外一个目录 /etc/init.d，真正的启动脚本都统一放在这个目录中。init进程逐一加载开机启动程序，其实就是运行这个目录里的启动脚本。



下面就是链接文件真正的指向。

```
$ ls -l /etc/rc2.d

README
S01motd -> ../init.d/motd
S13rpcbind -> ../init.d/rpcbind
S14nfs-common -> ../init.d/nfs-common
S16binfmt-support -> ../init.d/binfmt-support
S16rsyslog -> ../init.d/rsyslog
S16sudo -> ../init.d/sudo
S17apache2 -> ../init.d/apache2
S18acpid -> ../init.d/acpid
...
```

这样做的另一个好处，就是如果你要手动关闭或重启某个进程，直接到目录 /etc/init.d 中寻找启动脚本即可。比如，我要重启Apache服务器，就运行下面的命令：

```
$ sudo /etc/init.d/apache2 restart
```

/etc/init.d 这个目录名最后一个字母d，是directory的意思，表示这是一个目录，用来与程序 /etc/init 区分。

第五步、用户登录

开机启动程序加载完毕以后，就要让用户登录了。



一般来说，用户的登录方式有三种：

- (1) 命令行登录
- (2) ssh登录
- (3) 图形界面登录

这三种情况，都有自己的方式对用户进行认证。

(1) 命令行登录：init进程调用getty程序（意为get teletype），让用户输入用户名和密码。输入完成后，再调用login程序，核对密码（Debian还会再多运行一个身份核对程

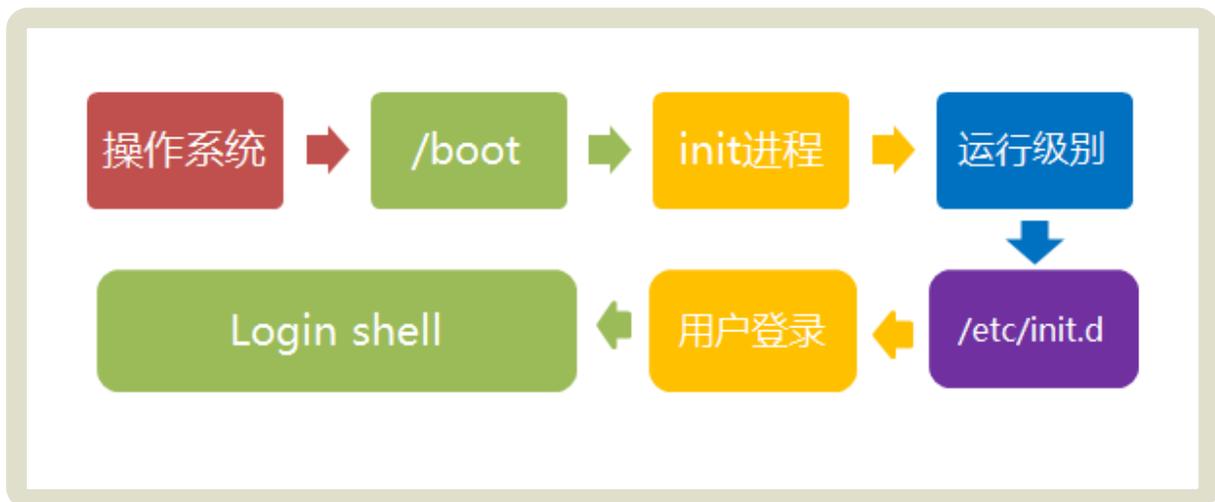
序/etc/pam.d/login)。如果密码正确，就从文件 /etc/passwd 读取该用户指定的 shell，然后启动这个shell。

(2) ssh登录：这时系统调用sshd程序（Debian还会再运行/etc/pam.d/ssh），取代getty和login，然后启动shell。

(3) 图形界面登录：init进程调用显示管理器，Gnome图形界面对应的显示管理器为gdm（GNOME Display Manager），然后用户输入用户名和密码。如果密码正确，就读取/etc/gdm3/Xsession，启动用户的会话。

第六步、进入 login shell

所谓shell，简单说就是命令行界面，让用户可以直接与操作系统对话。用户登录时打开的shell，就叫做login shell。



Debian默认的shell是[Bash](#)，它会读入一系列的配置文件。上一步的三种情况，在这一步的处理，也存在差异。

(1) 命令行登录：首先读入 /etc/profile，这是对所有用户都有效的配置；然后依次寻找下面三个文件，这是针对当前用户的配置。

```
~/.bash_profile  
~/.bash_login  
~/.profile
```

需要注意的是，这三个文件只要有一个存在，就不再读入后面的文件了。比如，要是 `~/.bash_profile` 存在，就不会再读入后面两个文件了。

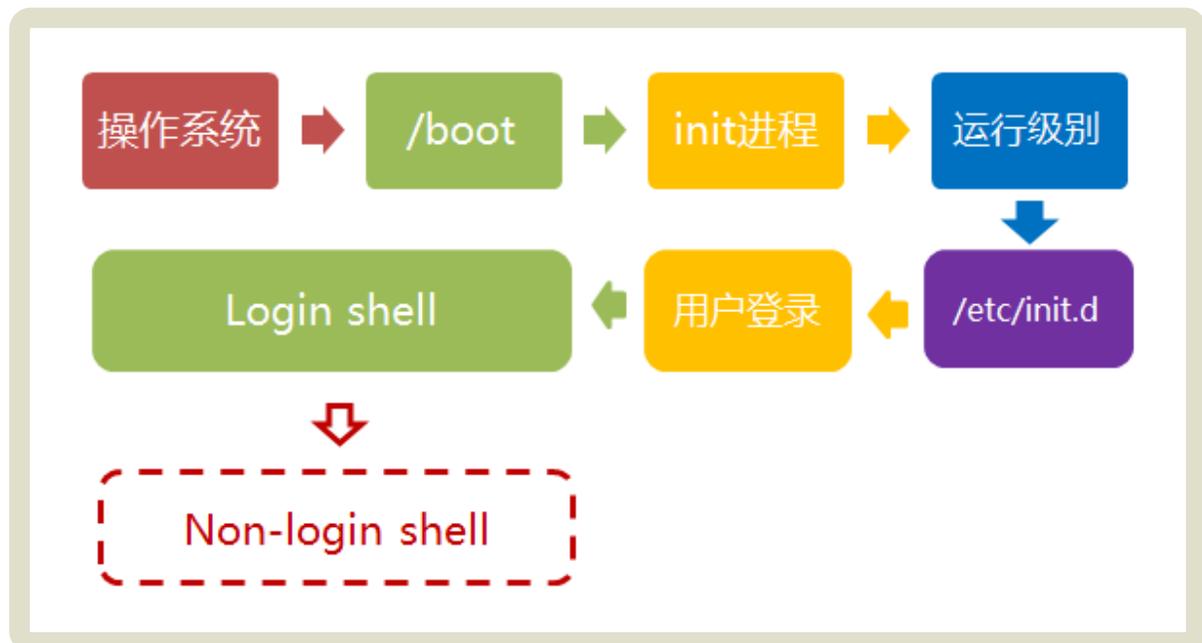
(2) ssh登录：与第一种情况完全相同。

(3) 图形界面登录：只加载 `/etc/profile` 和 `~/.profile`。也就是说，`~/.bash_profile` 不管有没有，都不会运行。

第七步，打开 **non-login shell**

老实说，上一步完成以后，Linux的启动过程就算结束了，用户已经可以看到命令行提示符或者图形界面了。但是，为了内容的完整，必须再介绍一下这一步。

用户进入操作系统以后，常常会再手动开启一个shell。这个shell就叫做 `non-login shell`，意思是它不同于登录时出现的那个shell，不读取`/etc/profile`和`.profile`等配置文件。



`non-login shell`的重要性，不仅在于它是用户最常接触的那个shell，还在于它会读入用户自己的bash配置文件 `~/.bashrc`。大多数时候，我们对于bash的定制，都是写在这个

文件里面的。

你也许会问，要是不进入 non-login shell，岂不是.bashrc就不会运行了，因此bash也就不能完成定制了？事实上，Debian已经考虑到这个问题了，请打开文件 ~/.profile，可以看到下面的代码：

```
if [ -n "$BASH_VERSION" ]; then
  if [ -f "$HOME/.bashrc" ]; then
    . "$HOME/.bashrc"
  fi
fi
```

上面代码先判断变量 \$BASH_VERSION 是否有值，然后判断主目录下是否存在 .bashrc 文件，如果存在就运行该文件。第三行开头的那个点，是source命令的简写形式，表示运行某个文件，写成"source ~/.bashrc"也是可以的。

因此，只要运行 ~/.profile文件， ~/.bashrc文件就会连带运行。但是上一节的第一种情况提到过，如果存在 ~/.bash_profile文件，那么有可能不会运行 ~/.profile文件。解决这个问题很简单，把下面代码写入.bash_profile就行了。

```
if [ -f ~/.profile ]; then
  . ~/.profile
fi
```

这样一来，不管是哪种情况，.bashrc都会执行，用户的设置可以放心地都写入这个文件了。

Bash的设置之所以如此繁琐，是由于历史原因造成的。早期的时候，计算机运行速度很慢，载入配置文件需要很长时间，Bash的作者只好把配置文件分成了几个部分，阶段性载入。系统的通用设置放在 /etc/profile，用户个人的、需要被所有子进程继承的设置放在.profile，不需要被继承的设置放在.bashrc。

顺便提一下，除了Linux以外，Mac OS X 使用的shell也是Bash。但是，它只加载.bash_profile，然后在.bash_profile里面调用.bashrc。而且，不管是ssh登录，还是在图形界面里启动shell窗口，都是如此。

参考链接

- [1] Debian Wiki, [Environment Variables](#)
- [2] Debian Wiki, [Dot Files](#)
- [3] Debian Administration, [An introduction to run-levels](#)
- [4] Debian Admin, [Debian and Ubuntu Linux Run Levels](#)
- [5] Linux Information Project (LINFO), [Runlevel Definition](#)
- [6] LinuxQuestions.org, [What are run levels?](#)
- [7] Dalton Hubble, [Bash Configurations Demystified](#)

(完)

文档信息

- 版权声明：自由转载-非商用-非衍生-保持署名（[创意共享3.0许可证](#)）
- 发表日期：2013年8月17日
- 更多内容：[✈ 档案](#) » [✈ 理解计算机](#)
- 付费支持：[🛒 购买文集](#)
- 社交媒体：[🐦 twitter](#) , [👁 weibo](#)
- Feed订阅：[📡](#)

相关文章

- **2014.11.11:** [编译器的工作过程](#)
| 源码要运行，必须先转成二进制的机器码。这是编译器的任务。
- **2014.09.07:** [数据压缩与信息熵](#)
| 1992年，美国佐治亚州的WEB Technology公司，宣布做出了重大的技术突破。
- **2014.07.04:** [数据库的最简单实现](#)
| 所有应用软件之中，数据库可能是最复杂的。
- **2013.11.29:** [Stack的三种含义](#)
| 学习编程的时候，经常会看到stack这个词，它的中文名字叫做"栈"。

广告（购买广告位）

留言（67条）

mizpah. 说：

linux感觉突然从今年突然名声噪起。

[2013年8月17日 16:48](#) | [档案](#) | [引用](#)

zhiyelee 说：

perfect , 豁然开朗

[2013年8月17日 16:58](#) | [档案](#) | [引用](#)

simon 说：

引用mizpah.的发言：

linux感觉突然从今年突然名声噪起。

linux一直很火，服务器上面的王者。。

[2013年8月17日 17:00](#) | [档案](#) | [引用](#)

imorz 说：

还有很多 .d 结尾的文件夹，比如 if-up.d, cron.d, pam.d, apparmor.d，但似乎不是都与 deamon 有关？

[2013年8月17日 17:12](#) | [档案](#) | [引用](#)

kkk 说：

现在好多都改systemd了..和init不一样啊
能再讲讲systemd的流程么？

[2013年8月17日 17:20](#) | [档案](#) | [引用](#)

呵呵呵 说：

引用mizpah.的发言：

linux感觉突然从今年突然名声噪起。

windown感觉突然从今年突然名声噪起。

[2013年8月17日 17:42](#) | [档案](#) | [引用](#)

orzFly 说：

引用imorz的发言：

还有很多 .d 结尾的文件夹，比如 if-up.d, cron.d, pam.d, apparmor.d，但似乎不是都与 daemon 有关？

参考：

The .d suffix here means directory. Of course, this would be unnecessary as Unix doesn't require a suffix to denote a file type but in that specific case, something was necessary to disambiguate the commands (/etc/init, /etc/rc0, /etc/rc1 and so on) an the directories they use (/etc/init.d, /etc/rc0.d, ...)

希望博主能够重新考虑一下这个问题。

[2013年8月17日 19:02](#) | [档案](#) | [引用](#)

江南胡杨 说：

引用mizpah.的发言：

linux感觉突然从今年突然名声噪起。

是你才开始关注Linux：)

[2013年8月17日 19:02](#) | [档案](#) | [引用](#)

哇咔咔 说：

改版了，不错

[2013年8月17日 19:14](#) | [档案](#) | [引用](#)

Anonymous 说：

很多东西说的不足。

1. 普适性缺乏

现在debian自己都在讨论要不要引入systemd，RHEL7 alpha1 ~ 3(别问为啥我能拿到alpha)/Fedora/Archlinux都已经完全迁移到systemd，ubuntu用自家的upstart，大概只有debian和gentoo还在用SystemV的这一套了，因该先将通用的内容，比如init程序，然后再说各种init程序的实现，如systemV/systemd/upstart等等由于init有不同的实现，所以那些进程脚本位置都有差别。

2. 缺乏细节

即便这篇文章不是针对Linux内核启动的代码分析级别的细致分析，也应该更多的提到bootloader。

最误导人的是读取内核那边，明明是内核+initrd一起读取的。

并且几乎所有非嵌入式发行版通用的initrd这个细节完全没有提到。

其中提到的init程序实际上只要用了initrd都是从initrd里面起起来的。还有initrd的双rootfs等重要的排错机制没提到。

同样第六步提到的各种配置文件，这个一样少了很多东西。

比如pam_env的pam模块就会读取/etc/environment的配置文件，对于这种可以灵

活配置的东西因该介绍的是如何自己去寻找读取顺序，而不是只说有哪些。

[2013年8月17日 19:16](#) | [档案](#) | [引用](#)

Terry 说：

这是 SysV init 别忘了提醒大家还有 Ubuntu 的 Upstart 和未来的标准 systemd (Arch Linux, Fedora 和 Suse 已经在用了)。

[2013年8月17日 19:27](#) | [档案](#) | [引用](#)

Terry 说：

提到 initrd 不得不提一下这货从 ramdisk 到 ramfs 的转变。详见：

<https://www.kernel.org/doc/Documentation/filesystems/ramfs-rootfs-initramfs.txt>

另外 IBM DeveloperWorks 上有几篇关于 Linux 启动流程和 initrd / initramfs 的文章很不错，可以参考来补充这篇。

[2013年8月17日 20:51](#) | [档案](#) | [引用](#)

Jimmie 说：

引用 **imorz** 的发言：

还有很多 .d 结尾的文件夹，比如 if-up.d, cron.d, pam.d, apparmor.d，但似乎不是都与 daemon 有关？

.d 可以是 directory 的缩写，比如我熟悉的 emacs 会读取配置文件 .emacs，或者进入文件夹 .emacs.d 读取配置文件。加 .d 可能是提示这是一个文件夹吧

[2013年8月17日 21:32](#) | [档案](#) | [引用](#)

lzprgmr 说：

照你的图的描述好似先运行init.d中所有的程序，然后运行rcN.d里面所有的程序 - 但逻辑上判断貌似只是运行 rcN.d中的所有程序（选择性的symbolic link到init.d） - 应该是后者吧

[2013年8月17日 21:39](#) | [档案](#) | [引用](#)

比尔盖子 说：

其中提到的 non-login Shell 感觉容易和 nologin shell 混淆，建议注明。

[2013年8月17日 21:44](#) | [档案](#) | [引用](#)

阮一峰 说：

引用orzFly的发言：

希望博主能够重新考虑一下这个问题。

谢谢指出，已经改过来了。不好意思，我一直以外这个d是 daemon 的意思。

[2013年8月17日 22:03](#) | [档案](#) | [引用](#)

henry_wu001 说：

是不是文章修改过后rss里面会出现2个，上一个点出来无效了

[2013年8月17日 22:32](#) | [档案](#) | [引用](#)

dcxy0 说：

不错，写的非常棒，长了很多知识啊！

[2013年8月17日 22:47](#) | [档案](#) | [引用](#)

sheep 说：

引用orzFly的发言：

希望博主能够重新考虑一下这个问题。

.d 结尾是代表与 SysV init相关的配置文件，init是有很多种实现的，上一代的是 SysV init,现在是两种，即Ubuntu所使用的Upstart实现以及主流的systemd实现，两种新的实现都兼容老的SysV init配置文件，但实际上几乎没有什么主流distro真的在用SysV init了，都是通过兼容实现的,所以有必要看一下systemd和upstart的手册和文档（runlevel这个概念其实也是老的SysV init里面的）

[2013年8月18日 00:52](#) | [档案](#) | [引用](#)

阮一峰 说：

引用Lzprgmr的发言：

照你的图的描述好似先运行init.d中所有的程序，然后运行rcN.d里面所有的程序 - 但逻辑上判断貌似只是运行 rcN.d中的所有程序（选择性的 symbolic link到init.d） - 应该是后者吧

谢谢指出，你说得有道理，我已经改过来了，把第三步和第四步换了一下顺序，图也改了。

[2013年8月18日 01:15](#) | [档案](#) | [引用](#)

D 说：

太笼统了....而且 SystemV5的启动流程现在也不流行了...

[2013年8月18日 11:27](#) | [档案](#) | [引用](#)

meow 说：

现在已经是systemd的天下了，还一遍又一遍的介绍这些旧东西？更新一下知识库吧。

[2013年8月18日 14:31](#) | [档案](#) | [引用](#)

zaeneas 说：

指手画脚一下，我记得ssh的bash的profile读取顺序和tty的读取顺序不一样

原因是 no_interactive shell和interactive shell的区别

[2013年8月18日 15:49](#) | [档案](#) | [引用](#)

hikerell 说：

请教一下图片使用什么做的？我一般用debian，写文章时，再开一台windows用PS，在传到debian。。。

[2013年8月18日 18:16](#) | [档案](#) | [引用](#)

Metoo 说：

憋了半年就整理出这点东西，真心看不下去。这是在科普DOS开机吗

如果要讲开机引导，就彻底讲透，除了上面的兄弟写的，还有很多原理没有说透，比如：Android是如何引导的，Xbox 360是如何引导的...等等。它们共同的逻辑和基本步骤是什么，各自有什么差异（什么原因造成的）？

看了半天，觉得我们就停留在刷机商的水平，然后意淫自己开发了神器，能锤掉乔布斯...

[2013年8月19日 09:17](#) | [档案](#) | [引用](#)

uframer 说：

引用mizpah.的发言：

linux感觉突然从今年突然名声噪起。

村通电

[2013年8月19日 09:29](#) | [档案](#) | [引用](#)

苗苗老师 说：

引用**imorz**的发言：

还有很多 .d 结尾的文件夹，比如 if-up.d, cron.d, pam.d, apparmor.d，但似乎不是都与 daemon 有关？

.d的意思是 directory，并不是daemon啊。

[2013年8月19日 09:39](#) | [档案](#) | [引用](#)

feinfo 说：

谢谢分享，对我这种linux小白来说很有用

[2013年8月19日 10:19](#) | [档案](#) | [引用](#)

老雷 说：

第六步这一句“（3）图形界面登录：只加载 /etc/prfile 和 ~/.profile。也就是说，~/.bash_profile 不管有没有，都不会运行。”的“/etc/prfile”少了一个“o”

[2013年8月19日 10:53](#) | [档案](#) | [引用](#)

godghost 说：

引用**Metoo**的发言：

憋了半年就整理出这点东西，真心看不下去。这是在科普DOS开机吗

你行你上啊 ~ 不要光说不练 ~

[2013年8月19日 11:41](#) | [档案](#) | [引用](#)

QFAN 说：

Linux发行版实在够多够乱，而且就是相同发行版也是一个版本一大改，三年彻底不认识..... GRUB为例，以前双系统改个GRUB启动顺序多容易，然后天雷一声震天响冒出个GRUB2。改文件不算数还要运行工具..... init也是，以前强切换直接init几就行，现在干脆就不好使了。/etc下各种配置文件也是，不同发行版里乱起名乱放子目录。头文件的放置位置什么的就更别提了...

[2013年8月19日 17:43](#) | [档案](#) | [引用](#)

祝·广东 说：

真心觉得应该讲透彻点，这些感觉真的很浮。

[2013年8月20日 02:16](#) | [档案](#) | [引用](#)

阮一峰 说：

引用老雷的发言：

第六步这一句“(3) 图形界面登录：只加载 /etc/prfile 和 ~/.profile。也就是说，~/.bash_profile 不管有没有，都不会运行。”的“/etc/prfile”少了一个“o”

谢谢指出，已经改正了。

[2013年8月20日 04:45](#) | [档案](#) | [引用](#)

总结者 说：

它们在Windows叫做"服务" (service)，在Linux就叫做"守护进程" (daemon)

不对哦，我理解daemon是service的实体文件，他可以作为service执行才对，我记得鸟哥是这么说的

[2013年8月20日 09:26](#) | [档案](#) | [引用](#)

eccstartup 说：

Ubuntu系统，~/.profile 文件更改之后，想对在terminal里面让其生效，就要source一下，以后打开的terminal也要source，直到下次重启。

有办法解决这一问题吗？

而mac os x系统source一遍，以后打开的terminal都生效了。

[2013年8月20日 17:11](#) | [档案](#) | [引用](#)

胡小侠 说：

刚看文章的时候觉的不错,通俗易懂,太牛叉了.
看了楼上的评价我就不敢看了,你懂的.

[2013年8月21日 00:32](#) | [档案](#) | [引用](#)

Ray 说：

"所以第一个启动的程序是motd，然后是rpcbing、nfs.....数字相同时..."

虽然是个打字错误，为求严谨，还是说一下吧，rpcbing (rpcbind)

[2013年8月22日 13:31](#) | [档案](#) | [引用](#)

Joseph 说：

想问一下这个画图是怎么实现的？

[2013年8月27日 14:48](#) | [档案](#) | [引用](#)

小白龙 说：

引用Metoo的发言：

憋了半年就整理出这点东西，真心看不下去。这是在科普DOS开机吗

我也觉得文章写得很搞笑，他的读者定义是什么？小学生的科普文章吗？

把一些业内早已知道的基本知识定义毕恭毕敬的整整齐齐整理一下。搬运工吗？

既想要做为一个专业者或者爱好者，又要写一些科普小学生的文章。

一点在旧概念的上的新见解或者是新的认知都没有。

然后难道下一篇又是转到评论时政去了吗？

[2013年8月29日 21:39](#) | [档案](#) | [引用](#)

JunkFood 说：

引用Metoo的发言：

憋了半年就整理出这点东西，真心看不下去。这是在科普DOS开机吗

引用小白龙的发言：

我也觉得文章写得很搞笑，他的读者定义是什么？小学生的科普文章吗？

我去，看不下去了。。。

博主本来就不是程序员，写这篇博文也只是为了记录一下自己的学习而已

如果你们觉得水平太低，那么你们也根本不是这篇文章的目标群体，大可以去找更高水平的文章来看，何必在这里浪费时间呢？

[2013年9月 1日 19:36](#) | [档案](#) | [引用](#)

guoqiao 说：

看了评论,发现心理扭曲的人真多.

[2013年9月 4日 13:17](#) | [档案](#) | [引用](#)

无家可归 说：

唉，留言真是没法看。给钱了吗？就在这里冷嘲热讽的。有没有人逼你看

[2013年9月17日 22:30](#) | [档案](#) | [引用](#)

tkiller 说：

大家可以试试安装流行的linux发行版，现在的linux已经是非常不错的了，如果只是看看电影，上网，基本的图片和文字处理的话，完全可以脱离win了，我这就是用的排名非常靠前的linux mint发行版，呵呵非常好用，而且linux下面的中文输入法比win下的还好用，真心不错

想2005的时候折腾红帽子，插个usb移动硬盘挂上去都要整半天，现在直接识别挂载，amd 785g的芯片组安装完全不需要驱动，哈哈，所有的东西基本都给你搞好了，自己设置好软件更新源，推荐清华或者163的，更新那是嗖嗖的，试试吧，变革的时代也要不断的尝试与学习！

而且firefox在linux的字体渲染更好，硬件加速也不会字体发虚，滚动的感觉非常好，不错！

[2013年9月25日 18:41](#) | [档案](#) | [引用](#)

Ming 说：

引用**JunkFood**的发言：

我去，看不下去了。。。

博主本来就不是程序员，写这篇博文也只是为了记录一下自己的学习而已

如果你们觉得水平太低，那么你们也根本不是这篇文章的目标群体，大可以去找更高水平的文章来看，何必在这里浪费时间呢？

即使是程序员写出来我也觉得挺好的，自己学到的不一定是全面的，正确的，学习心得写出来和大家交流一下，互相进步。牛人们有耐心的请指出不足，没耐心的请跳过，何必冷言冷语。

[2013年9月29日 11:17](#) | [档案](#) | [引用](#)

lcy 说：

謝謝樓主，有些評論真實莫名其妙！

[2013年10月 1日 23:17](#) | [档案](#) | [引用](#)

王东杰 说：

期待讲一下android的启动过程和这个有什么区别？

[2013年10月 8日 18:29](#) | [档案](#) | [引用](#)

Atlas2013 说：

楼主，你好 发现你的文章中的图做的非常好，能请教一下用什么工具做的么？谢谢

[2013年10月10日 10:40](#) | [档案](#) | [引用](#)

陈钢 说：

写得不错，通俗易懂。继续加油！

[2013年10月13日 17:52](#) | [档案](#) | [引用](#)

栗子壳李逵 说：

打印回寝室看~ 露珠加油 上次的Git个人主页就是看你的搞的 虽然最后没时间搞成功 但是涨了不少姿势~

[2013年10月14日 21:38](#) | [档案](#) | [引用](#)

windywater 说：

阮老师的博文向来是通俗易懂，适于初学者，之前看过一篇讲TCP/IP的文章非常赞！评论里有些人花30块买了份肯德基套餐，吃完又指责人家没有做出五星级酒店的味道。

唉！好在阮老师不跟这种人一般见识。

[2013年11月 2日 20:29](#) | [档案](#) | [引用](#)

蓝燕光 说：

哥，你用的什么画图工具

[2013年11月15日 11:02](#) | [档案](#) | [引用](#)

Sn0rt 说：

要说启动流程的话,推荐 linux内核设计的艺术,靠用启动流程为主线讲操作系统的.

[2013年12月 2日 12:32](#) | [档案](#) | [引用](#)

zzj 说：

讲的真清楚，之前的很多疑问迎刃而解，谢谢你！

[2014年1月25日 20:29](#) | [档案](#) | [引用](#)

落没古堡 说：

学究气加蛋疼的说：『确切的说init并不是真正的第一个程序，也并不是所有程序都从它衍生。』它的进程编号（pid）是1。无疑init超NB

[2014年4月 5日 21:57](#) | [档案](#) | [引用](#)

gshmu 说：

哈哈，写的太深入浅出了，非常易懂。

导致有些人认为他都了解Linux多年了，可是至今我也没看到结果哈。

[2014年6月 9日 13:09](#) | [档案](#) | [引用](#)

nudo 说：

引用Ming的发言：

自己学到的不一定是全面的，正确的，学习心得写出来和大家交流一下，互相进步。牛人们有耐心的请指出不足，没耐心的请跳过，何必冷言冷语。

为这个“态度”点个赞。

[2014年6月17日 11:27](#) | [档案](#) | [引用](#)

Hebe 说：

大牛们请嘴下留情吧，坦白说这是我第一个看懂了的linux启动流程，像我这样目前什么都还不清楚的，真的不太需要代码级别的分析，起码这两天还不需要，就这个很好!!!

[2014年6月24日 16:03](#) | [档案](#) | [引用](#)

阿道夫 说：

我觉得这个作者写的非常好，具有大师的风范

[2014年8月20日 20:50](#) | [档案](#) | [引用](#)

umount 说：

请教一下，操作系统刚接管硬件时，应该还没有加载文件系统吧，那Linux的GRUB是如何找到/boot下面的各种内核文件的？

[2014年9月15日 21:14](#) | [档案](#) | [引用](#)

Boatman 说：

引用mizpah.的发言：

linux感觉突然从今年突然名声噪起。

因为Android的兴起使然。个人觉得是一个原因。

[2014年10月14日 10:40](#) | [档案](#) | [引用](#)

Ox1D 说：

为什么历史不好的地方不进行更正？

bash 存放在那么多目录，修改起来那么麻烦。如果将他简化到一个目录不是很好吗？
毕竟这又不像 windows 的 C 盘命名那么无足轻重。

[2014年10月28日 21:47](#) | [档案](#) | [引用](#)

ARM Linux 说：

解释很有调理，看完基本明白了Linux启动过程

[2014年10月30日 11:12](#) | [档案](#) | [引用](#)

dirk 说：

楼主的文章最可贵的地方就是文章的篇幅不大，但内容丰富，句句重点，简明扼要，容易理解，支持.....

[2014年10月30日 12:53](#) | [档案](#) | [引用](#)

Alice 说：

引用mizpah.的发言：

linux感觉突然从今年突然名声噪起。

linux的桌面用户也在大幅增加！

[2014年11月 4日 18:38](#) | [档案](#) | [引用](#)

haotian 说：

终于找到一个通俗易懂的blog了

[2014年11月12日 13:47](#) | [档案](#) | [引用](#)

geeksword 说：

阮老师有时间讲讲Linux启动过程中FS的挂载，initrd的作用...

[2014年12月19日 15:08](#) | [档案](#) | [引用](#)

hah 说：

非常之好，正合我这初学者所需，看完之后从一头雾水一下豁然开朗！

另外，看到有些评论不能说那是刻薄

简直是人品问题~~好像他们就想关上linux大门给他们几个人玩似的，赚钱多？！

[2015年1月28日 11:26](#) | [档案](#) | [引用](#)

我要发表看法

您的留言（HTML标签部分可用）

您的大名：

«-必填

电子邮件：

«-必填，不公开

个人网址：

«-我信任你，不会填写广告链

接

记住个人信息？

«- 点击按钮

联系方式 | ruanyifeng.com 2003 - 2015 