


第 3 章 Linux 基本配置

安装完操作系统后，常常需要做一些基本配置，以满足自己的需求。随着 Linux 桌面的日趋成熟和人性化，这种所谓的“基本配置”已经越来越少了。本章选择了入门用户最常问到的一些问题，以便读者能够尽快上手。

3.1 关于超级用户 root


之所以首先介绍 root 用户，是因为这个用户实在太重要了。所有的系统设置都需要使用 root 用户来完成。root 从字面上解释是“根”的意思，所以超级用户也被称作根用户。从某种意义上它相当于 Windows 下的 Administrator 用户。

 **提示：**本节使用的一些命令只是为了能更好地说明问题，读者如果不能马上理解可以暂且不去理会。在后续章节中将逐步讲解 Linux 下命令行的使用。

3.1.1 root 可以做什么

这个问题的答案是 anything。没错，作为整个系统中拥有最高权限的用户，root 可以对系统做任何事情。root 可以访问、修改、删除系统中的任何文件和目录。另外，对于如下这些受限的操作，一般只有 root 用户能够执行。

- ☐ 添加删除用户；
- ☐ 安装软件；
- ☐ 添加删除设备；
- ☐ 启动和停止网络服务；
- ☐ 某些系统调用（例如对内核的请求）；
- ☐ 关闭系统；
- ⋮

 **提示：**像“关闭系统”这样的操作都需要 root 用户来执行，看起来是一件特别古怪的事情。事实上，作为 Linux 的祖先，UNIX 是一种典型的服务器操作系统。而服务器的关闭和启动都必须得到管理员的授权（试想一个普通用户登录服务器，然后随意执行关机命令会怎样）。出于操作简易性的考虑，桌面版的 Linux 允许普通用户在图形界面下关闭系统。但在命令行下执行关机命令仍然需要 root 口令。

Linux 系统上的每个文件和目录都属于某个特定的用户，没有得到许可，其他用户就

不能访问这些对象。但 `root` 用户却可以访问所有用户的文件，就像使用自己的东西一样。因此，拥有 `root` 口令意味着更多的责任，特别是在一台多人协作的服务器上。

3.1.2 避免灾难

正如 3.1.1 节所提到的，`root` 用户可以在系统上做任何事情。那么保证安全性就显得尤为重要。系统不会因为用户输入的命令足够“愚蠢”而拒绝执行。相反，系统会乐滋滋地执行这样一条命令，然后把自己完完整整地删除了。

```
$ rm -fr /*                                ##删除根目录下所有的文件和目录
```

另外，妥善保管 `root` 口令也至关重要。因为任何得到 `root` 口令的人都能够完全控制系统。`Root` 口令应该至少为 8 个字符，7 个字符的密码其实很容易被破解。从理论上讲，最安全的口令应该是由字母、标点符号和数字组成的足够长的随机序列。但这样的密码往往难以记忆，如果为了使用这样所谓“最安全”的密码而不得不把它写在纸上，那么这是得不偿失的。一个比较好的建议是，使用拼音组成的一句话并穿插标点和数字。像 `jintian,qinglang`（今天，晴朗），`woshiyongUbuntu8.04`（我使用 Ubuntu8.04）等都是不错的口令。

和普通用户一样，`root` 账号可以直接用来登录系统。但这显然是一个非常糟糕的选择。既然任何一项误操作都有可能造成灾难性的后果，那么就应该仅在必要的时候才使用 `root` 账号。幸运的是，Linux 提供了这样的特性。用户可以执行不带参数的 `su` 命令将自己提升为 `root` 权限（当然需要提供 `root` 口令）。另一个命令行工具是 `sudo`，它可以临时使用 `root` 身份运行一个程序，并在程序执行完毕后返回至普通用户状态。这两个工具将在第 9 章详细讨论。

3.1.3 Debian 和 Ubuntu 的 root 用户

对于绝大多数的 Linux 发行版而言，安装的最后一步会设置两个用户的口令：一个是 `root` 用户，另一个是用于登录系统的普通用户。而对于 Debian 和 Ubuntu 而言，事情显得有些古怪——只有一个普通用户，而没有 `root`！实际上，这个在安装过程中设置的普通用户账号，在某种程度上充当了 `root`。平时，这个账号安分守己地做自己份内的事，没有任何特殊权限。在需要 `root` 的时候，则可以使用 `sudo` 命令来运行相关程序。`sudo` 命令运行时会要求输入口令，这个口令就是该普通账号的口令。

那么读者就会有这样的疑问：如果再建立一个用户，那么这个用户是不是也能够使用 `sudo` “为所欲为”呢？答案是否定的。`sudo` 通过读取 `/etc/sudoers` 来确定用户是否可以执行相关命令。这个文件默认需要有 `root` 权限才能够修改。关于 `/etc/sudoers` 的修改和配置，将在 9.8 节讨论。

也可以使用 `sudo` 的 `-s` 选项将自己提升为 `root` 用户，使用了 `-s` 选项的 `sudo` 命令相当于 `su`。例如在终端下输入：

```
lewis@lewis-laptop:/station/document$ sudo -s
[sudo] password for lewis:
```

```
root@lewis-laptop:/station/document#
```

🔔注意：出于安全性考虑，在输入密码时屏幕上并不会有任何显示（包括星号）。

最后，可以使用 `exit` 命令回到先前的用户状态。

```
root@lewis-laptop:/station/document# exit
exit
lewis@lewis-laptop:/station/document$
```

3.2 依赖于发行版本的系统管理工具

很多 Linux 发行版本都提供了可视化的系统管理工具。例如 Red Hat 的 Network Administration Tool，以及 SUSE 的 YAST2，如图 3.1 所示。这些可视化工具给 Linux 用户带来了莫大的方便，从而让系统管理工作变得只是单击鼠标这样简单。

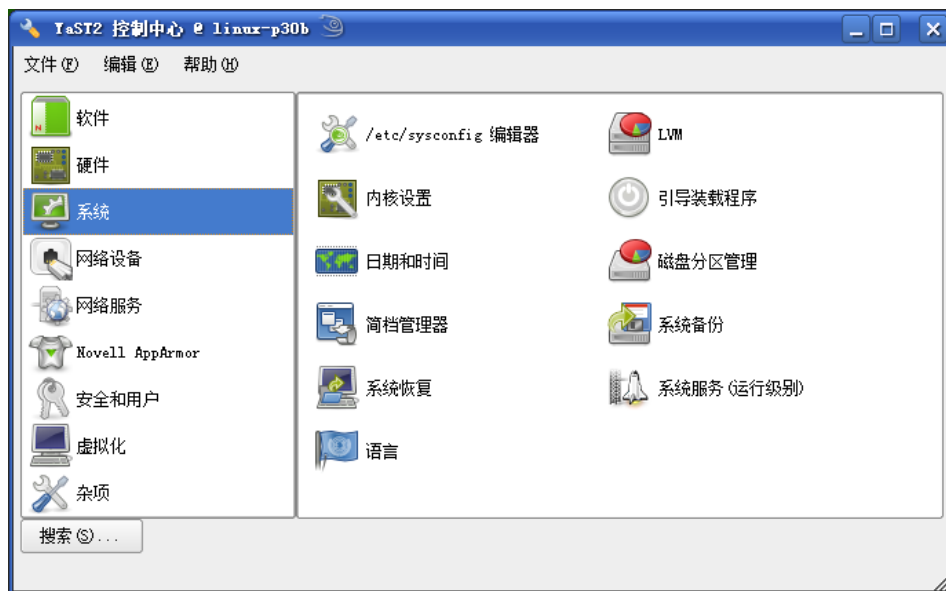


图 3.1 SUSE 的可视化系统管理工具 YAST

然而在另一方面，这些工具向用户隐藏了实施配置的底层机制。尽管这对于初学者而言并不是什么坏事，但图形界面简易性仅仅在系统正常时能够发挥优势。当故障发生时，可视化工具对于解决问题常常无能为力。管理员不得不在命令行下手动解决这些问题。相对于可视化工具而言，命令行往往更灵活、更可靠。

出于以上这些原因，本书将着重讨论如何在命令行下管理系统，这对于所有 Linux 发行版本而言基本都是相同的。同时考虑到读者基础，本书将尽可能多地穿插讲解可视化配置工具。鉴于 Ubuntu 和 openSUSE 在桌面端的普及程度，大部分的可视化工具都将以 Ubuntu Linux 和 openSUSE 为例进行讨论。其他发行版本的配置工具可以基于相似的思想和方法来使用。

3.3 中文支持

如果读者正在使用 openSUSE 的话，那么只要记得在安装的时候选上中文支持就可以了。受制于安装光盘的容量，Ubuntu 则显得不是那么“聪明”。为此，用户需要在安装结束后手动安装中文包。下面简单介绍在 Ubuntu 下安装中文支持的全过程，其中一些步骤可能已经涉及了本书几百页之后的内容，这里暂且“不求甚解”就可以了。

首先应该确保计算机已经连接到了 Internet（读者可能会在这里遭遇一些麻烦，或许为此不得不参考第 11 章的相关内容）。依次选择“应用程序（Applications）”|“附件”|“终端”命令打开终端模拟器，输入下面这条命令：

```
$ sudo apt-get update
```

这条命令用于从 Internet 更新当前系统软件包的信息，为此需要提供 root 口令。系统会给出一系列下载信息作为回应。这些信息看上去应该像这个样子（用户可以设置速度更快的安装源，参见 7.5 节）：

```
获取: 1 http://mirror.lupaworld.com hardy Release.gpg [189B]
获取: 2 http://mirror.lupaworld.com hardy/main Translation-zh_CN [25.6kB]
忽略 http://mirror.lupaworld.com hardy/restricted Translation-zh_CN
获取: 3 http://mirror.lupaworld.com hardy/universe Translation-zh_CN [25.9kB]
获取: 4 http://security.ubuntu.com hardy-security Release.gpg [189B]
忽略 http://security.ubuntu.com hardy-security/main Translation-zh_CN
:
```

完成更新工作后，依次执行下面的操作步骤。

（1）选择“系统（System）”|“系统管理”|“语言支持（Language Support）”命令，打开 Language Support（语言支持）对话框，如图 3.2 所示。

（2）从 Supported Languages 列表框中选择“汉语”选项，并单击 OK 按钮。系统会要求用户确认安装信息，如图 3.3 所示。



图 3.2 语言支持

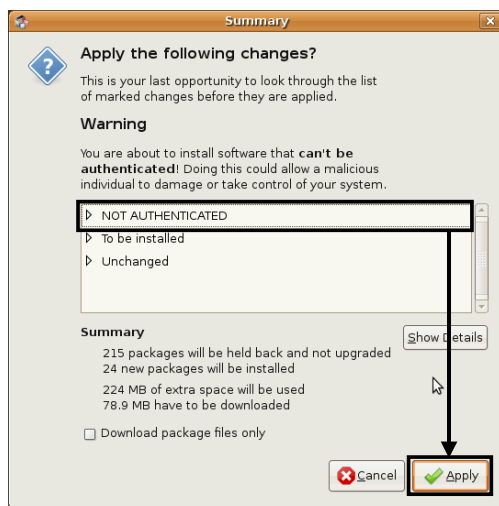


图 3.3 确认语言包安装

(3) 单击 **Apply** 按钮，执行安装，如图 3.4 所示。系统会给出下载软件包的即时进度，这取决于实际的网络环境，会花费几分钟到几个小时不等的时间。

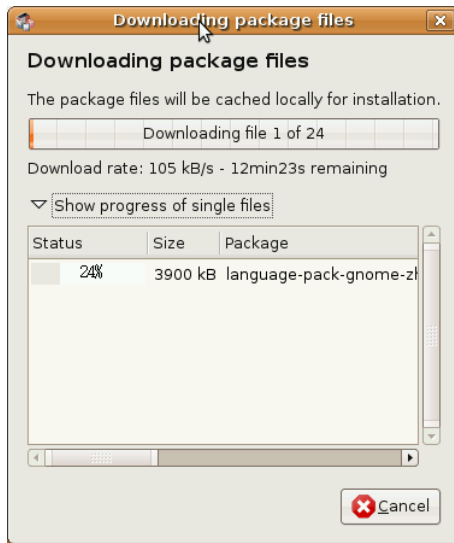


图 3.4 下载语言包

(4) 下载完成后，系统会自动安装这些软件包。如果一切顺利，那么重新启动后就可以让 Ubuntu 全方位地支持中文了。

3.4 关于硬件驱动程序

对于早期的 Linux 而言，寻找特定的硬件驱动程序往往是安装配置中最花费时间的一步。系统管理员甚至不得不自己编写。现在 Linux 已经得到了绝大部分主流硬件厂商的支持。在 Linux 安装完成后，往往已经不需要再安装什么驱动程序了。Linux 安装程序会自动监测系统硬件，并安装相应的驱动程序。在这一点上，Linux 做的甚至比 Windows 更好（读者应该会有安装完 Windows 后疯狂安装硬件驱动的特殊经历）。

对于 Linux 安装程序没有集成的驱动程序，就需要手动安装。主流硬件厂商一般都会在其官方网站上提供驱动程序的 Linux 版本。安装方法视不同的驱动提供商和用户的 Linux 版本而定。读者应该仔细阅读安装说明。需要注意的是，驱动程序的安装往往存在风险。所以必须选择与自己的硬件完全匹配的驱动，否则会让硬件无法使用，甚至损坏硬件。

如果硬件厂商并没有提供 Linux 版本的驱动程序，那么只能寄希望于第三方开发了。很多 Linux 爱好者会开发一些硬件的驱动程序，如果读者碰巧找到了，那就可以安装使用。但这些驱动程序往往没有得到硬件厂商的支持，使用上存在一定的风险，应该谨慎对待。

Ubuntu Linux 的更新程序会自动从互联网上探测适合当前系统的驱动程序，并在适当的时候提示用户安装。依次选择“系统”|“系统管理”|“硬件驱动”命令可以打开“硬件驱动”对话框，其中列出了当前可用的硬件驱动程序，如图 3.5 所示。选择相应的设备驱动，系统会自动从互联网上下载并完成安装和相关配置。



图 3.5 安装硬件驱动程序

3.5 获得更新

无论是 Ubuntu、openSUSE，还是其他一些主流 Linux 发行版本，都会不定期地提供相关软件包的更新。这些更新通常是出于升级版本或是修补安全漏洞的目的。“不定期”是显然的，安全漏洞不会“定期”出现，世界上所有的软件发行商也不会同时发布升级版本。系统不会盲目地更新“不存在”的东西，因此更新列表的长度总是同当前系统上安装的软件数量成正比。

以 Ubuntu Linux 为例，每当有可用更新，系统会在右上角显示一个红色的箭头。单击这个红色箭头即可打开“更新管理器”对话框，其中列出了所有可用更新，包括安全更新（一定要安装）和推荐软件更新，如图 3.6 所示。Ubuntu 的更新非常迅速，笔者在半个多月里积累了接近 200MB 的更新内容。

在更新列表中选择需要更新的软件包（通常使用推荐组合即可），单击“安装更新”按钮即可从互联网上下载并安装更新，如图 3.7 所示。完成安装后可能需要重启系统。

3.6 进阶：配置 Grub

本节继续讨论引导加载程序 Grub。在第 2 章的“进阶”部分已经介绍了如何修复被损坏的 Grub，这里将更深入地讲解 Grub 的使用。当然所谓的“深入”是相对的，这个引导程序本身可以被拿出来写一本书，本节所涉及的只是一些皮毛而已。

3.6.1 Grub 的配置文件

Grub 启动时通常从 `/boot/grub/menu.lst` 读取引导配置，并且严格地依此行事。下面是引导一个 Linux 系统所做的配置，这段内容取自 Grub 配置文件给出的示例。



图 3.6 更新管理器

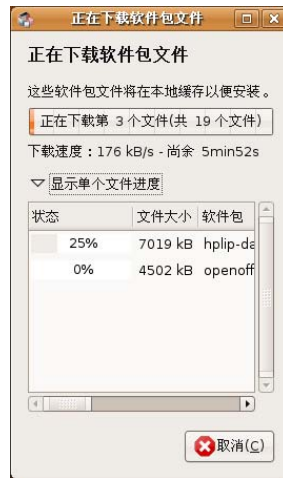


图 3.7 下载更新

```
title      Linux
root       (hd0,1)
kernel    /vmlinuz root=/dev/hda2 ro
```

关键字 `title` 指定了引导时显示的标题，用户可以把它改成任意的字符串，但至少应该有意义。关键字 `root` 指定了从哪个硬盘分区开始引导（即根分区），这里是第 1 块硬盘的第 2 个主分区（回忆 2.4 节对 `Grub` 分区表述方式的介绍）。最后，关键字 `kernel` 指定了 `Linux` 内核文件所在的位置。

引导 `Windows` 的配置段则有些不同，下面这段内容同样取自 `Grub` 配置文件的示例。

```
title      Windows 95/98/NT/2000
root       (hd0,0)
makeactive
chainloader +1
```

关键字 `makeactive` 将 `root` 指定的分区设置为活动分区；关键字 `chainloader` 从指定位置加载 `Windows` 引导程序。

要设置 `Grub` 引导多个系统，只要简单地把各个 `title` 组合在一起就可以了。下面这个配置示例可以让用户选择从分区 4 启动 `Ubuntu 8.04`，也可以从分区 1 启动 `Windows Vista`。

```
default    0
timeout    10

title      Ubuntu 8.04.1, kernel 2.6.24-21-generic
root       (hd0,4)
kernel     /boot/vmlinuz-2.6.24-21-generic ro quiet splash locale=zh_CN
initrd     /boot/initrd.img-2.6.24-21-generic
quiet

title      Windows Vista/Longhorn (loader)
root       (hd0,2)
savedefault
```

```
makeactive
chainloader +1
```

`timeout 10` 告诉 Grub 在接收到用户的输入之前需要等多久，这里的单位是秒。如果 `timeout` 被设置为 0，那么用户就没有任何选择余地，Grub 自动依照第 1 个 `title` 的指示引导系统。

3.6.2 使用 Grub 命令行

用户可以在 Grub 引导时手动输入命令来指导 Grub 的行为。在 Grub 启动画面出现时按下 C 键可以进入 Grub 的命令行模式，如图 3.8 和图 3.9 所示。

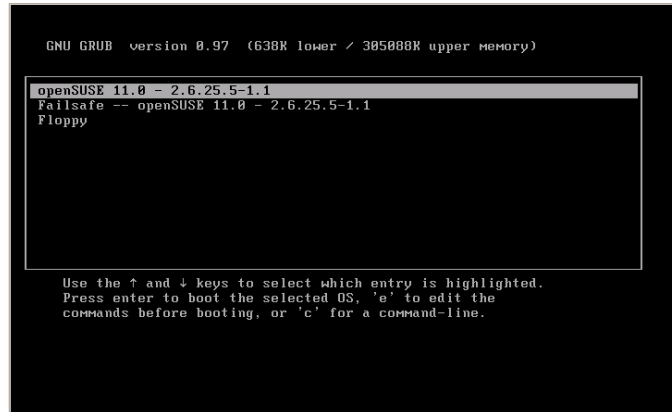


图 3.8 Grub 引导界面

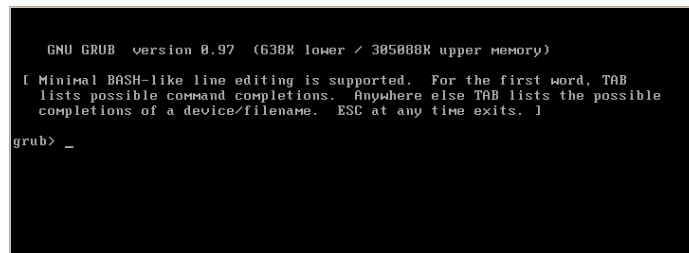


图 3.9 Grub 命令行

使用 Grub 命令行是一门艺术，也是足够复杂的一件事情。如表 3.1 列出了一些最基本和最常用的命令，读者如果对此感兴趣，可以到 www.gnu.org/software/grub/manual/ 上下载其官方手册。

表 3.1 引导程序Grub的常用命令

命 令	说 明
help	显示帮助信息
reboot	重新引导系统
root	指定根分区
kernel	指定内核所在的位置
find	在所有可以安装的分区上寻找一个文件
boot	依照配置引导系统

3.7 小 结

- ❑ 超级用户 `root` 是 Linux 中最重要的用户，拥有执行系统管理任务的完整权限。
- ❑ 注意妥善保管 `root` 口令，并在执行某些“破坏性”的任务时格外小心。
- ❑ Debian 和 Ubuntu 强制用户通过 `sudo` 命令提升权限。
- ❑ Linux 发行版本通常包含有自己的可视化管理工具，但命令行始终是管理员最可靠的伙伴。
- ❑ openSUSE 用户可以直接从安装光盘获取中文支持；Ubuntu 用户则需要从互联网上下载中文安装包。
- ❑ Linux 能够自动检测并安装绝大部分硬件的驱动程序。
- ❑ Ubuntu 和 openSUSE 都能自动获取软件更新信息，并提示用户下载安装。
- ❑ 引导程序 Grub 的配置文件是 `/boot/grub/menu.lst`。这是一个文本文件，可以用任何文本编辑器修改。
- ❑ 在 Grub 启动画面出现时按下 `C` 键可以进入 Grub 的命令行模式。