

### 图书基本信息

书名：<<Linux基础及应用教程实验指导与实训>>

13位ISBN编号：9787508461687

10位ISBN编号：7508461681

出版时间：2009-2

出版时间：水利水电出版社

作者：郭迎 等主编

页数：197

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## 前言

随着计算机科学与技术的飞速发展，计算机的应用已经渗透到国民经济与人们生活的各个角落，正在日益改变着传统的人类工作方式和生活方式。

在我国高等教育逐步实现大众化后，越来越多的高等院校会面向国民经济发展的第一线，为行业、企业培养各级各类高级应用型专门人才。

为了大力推广计算机应用技术，更好地适应当前我国高等教育的跨跃式发展，满足我国高等院校从精英教育向大众化教育的转变，符合社会对高等院校应用型人才培养的各类要求，我们成立了“21世纪高等院校规划教材编委会”，在明确了高等院校应用型人才培养模式、培养目标、教学内容和课程体系框架下，组织编写了本套“21世纪高等院校规划教材”。

众所周知，教材建设作为保证和提高教学质量的重要支柱及基础，作为体现教学内容和教学方法的知识载体，在当前培养应用型人才中的作用是显而易见的。

探索和建设适应新世纪我国高等院校应用型人才培养体系需要的配套教材已经成为当前我国高等院校教学改革和教材建设工作面临的紧迫任务。

因此，编委会经过大量的前期调研和策划，在广泛了解各高等院校的教学现状、市场需求，探讨课程设置、研究课程体系的基础上，组织一批具备较高的学术水平、丰富的教学经验、较强的工程实践能力的学术带头人、科研人员和主要从事该课程教学的骨干教师编写出一批有特色、适用性强的计算机类公共基础课、技术基础课、专业及应用技术课的教材以及相应的教学辅导书，以满足目前高等院校应用型人才培养的需要。

本套教材消化和吸收了多年来已有的应用型人才培养的探索与实践成果，紧密结合经济全球化时代高等院校应用型人才培养工作的实际需要，努力实践，大胆创新。

教材编写采用整体规划、分步实施、滚动立项的方式，分期分批地启动编写计划，编写大纲的确定以及教材风格的定位均经过编委会多次认真讨论，以确保该套教材的高质量和实用性。

## 内容概要

本书是以Fedora Core Linux为基础的实际操作应用指南，它是Linux基础及应用教程的配套教材，对Linux系统的应用技术做了全面的实训，主要包括Linux桌面应用、文件系统管理、用户管理、进程管理、网络管理、安全管理等实用操作知识，为Linux的初学者能够快速入门提供了保证。每章都是一个大实训项目，分成若干个子任务，安排了详细的实验内容，使读者能够边学边用，更快地提高使用Linux的实际操作水平。

本书以实际操作对象为实例，内容丰富、讲解清晰，几乎覆盖教程所有的内容，因为是分步进行的，有助于初学者理解、把握问题的精髓，提高对应用操作框架的整体认识，为读者掌握操作技能提供经典典范。

本书可作为高等院校（含高职）电子类、计算机类、信息类等专业的Linux课程实训教材，也可作为广大Linux用户、系统管理员和Linux系统自学者的参考书或培训教材，还可供希望转入嵌入式领域的科研和工程技术人员参考使用。

## 书籍目录

序前言第1章 Linux的安装与配置 1.1 任务1 在本地计算机上安装Fedora Core Linux单操作系统 1.1.1 做好安装前的准备工作 1.1.2 安装Fedora Core Linux 1.2 任务2 在本地计算机Windows上安装VMware 1.2.1 准备好软件资源 1.2.2 安装VMware Workstation 1.2.3 在虚拟机上安装Linux 1.2.4 在虚拟机上安装VMware Tools 1.3 本章小结第2章 文件与目录的操作 2.1 任务1 熟悉Linux操作系统的文件和目录结构 2.1.1 识别文件类型 2.1.2 Linux操作系统的目录结构 2.2 任务2 创建文件 2.2.1 使用touch命令 2.2.2 使用vi编辑器 2.3 任务3 文件操作 2.3.1 复制文件 2.3.2 移动文件 2.3.3 删除文件 2.3.4 其他常用的文件操作命令 2.4 任务4 目录的操作 2.4.1 建立子目录命令mkdir 2.4.2 拷贝目录命令cp 2.4.3 删除子目录命令rmdir 2.4.4 其他常用的目录操作命令 2.5 本章小结第3章 桌面应用 3.1 任务1 使用GNOME面板 3.1.1 由主菜单启动GIMP程序 3.1.2 查找MP3文件 3.1.3 使用面板上的启动器 3.1.4 切换工作区 3.1.5 在面板上添加GIMP启动器 3.1.6 自动隐藏配置面板 3.2 任务2 文件管理器Nautilus 3.2.1 启动Nautilus 3.2.2 新建目录 3.2.3 复制、移动、删除文件和目录 3.3 任务3 使用OpenOffice.org办公软件 3.3.1 使用Writer编辑光盘中的doc文档 3.3.2 使用Calc编辑xls文档 3.3.3 使用Impress编辑U盘中的ppt文档 3.3.4 将doc文档生成PDF文档 3.3.5 设置和使用本地打印机 3.4 本章小结第4章 文件系统管理 4.1 任务1 了解Linux文件系统 4.1.1 了解文件系统类型 4.1.2 观察Linux目录结构 4.2 任务2 Linux系统中对硬盘的操作 4.2.1 物理安装硬盘 4.2.2 对硬盘进行分区 4.2.3 创建文件系统 4.2.4 建立挂载点 4.2.5 修改/etc/fstab文件 4.3 任务3 安装和卸载Linux文件系统 4.3.1 安装Linux文件系统 4.3.2 确认安装的Linux文件系统类型 4.3.3 卸载文件系统 4.4 任务4 控制文件和目录的存取权限 4.4.1 准备知识 4.4.2 修改并查看文件的属性及权限 4.5 本章小结第5章 用户管理 5.1 任务1 用户和用户组的配置管理 5.1.1 与用户管理相关的文件 5.1.2 使用adduser增加用户 5.1.3 用户的删除与禁用 5.1.4 设置用户属性 5.1.5 管理用户组 5.2 任务2 配置磁盘配额 5.2.1 启用磁盘配额 5.2.2 对用户指定磁盘配额 5.2.3 对用户组指定磁盘配额 5.2.4 修改过渡期 5.2.5 其他磁盘配额命令 5.3 本章小结第6章 进程管理 6.1 任务1 监视进程 6.1.1 了解进程的组成和生命周期 6.1.2 ps——监视进程 6.1.3 top——更好的监视进程 6.2 任务2 启动进程 6.2.1 手工启动 6.2.2 调度启动 6.3 任务3 进程间通信 6.3.1 kill命令 6.3.2 trap命令 6.3.3 通信实例 6.4 任务4 改变进程优先级nice和renice 6.5 本章小结第7章 网络管理 7.1 网络管理基础知识 7.1.1 TCP/IP网络 7.1.2 端口和IP地址 7.1.3 路由 7.2 任务1 配置网络 7.2.1 配置网络 7.2.2 配置静态路由 7.2.3 配置DNS 7.3 任务2 网络故障的检测与调试 7.3.1 定位故障 7.3.2 操作步骤 7.4 任务3 安装Web服务器 7.4.1 Web服务器简介 7.4.2 Apache的特性 7.4.3 Apache的配置 7.5 虚拟机上Linux与Windows的通信 7.6 本章小结第8章 Shell编程 8.1 任务1 Shell原理初探 8.1.1 Shell的基本工作原理 8.1.2 了解Shell的使用情况 8.2 任务2 Shell变量的操作 8.2.1 Shell变量的定义与分类 8.2.2 本地变量 8.2.3 环境变量 8.2.4 特殊变量 8.2.5 位置变量参数 8.3 任务3 创建和执行Shell脚本 8.3.1 创建Shell脚本 8.3.2 执行Shell脚本 8.3.3 条件测试 8.3.4 控制流结构 8.3.5 向脚本传递参数 8.4 任务4 Shell脚本练习 8.5 本章小结第9章 安全管理 9.1 任务1 控制对服务的访问 9.1.1 使用服务配置工具启动iptables防火墙服务 9.1.2 使用ntsysv应用程序配置httpd万维网服务——停止自动启动 9.1.3 使用chkconfig服务配置工具改变named域名服务器的运行级别 9.1.4 各种服务配置工具 9.2 任务2 控制系统运行级别 9.2.1 修改默认的运行级别 9.2.2 立即改变运行级别 9.2.3 Red Hat Linux的运行级别 9.3 任务3 收集系统进程信息 9.3.1 收集系统进程信息 9.3.2 更多查看进程的方法 9.3.3 收集内存信息 9.3.4 收集文件系统信息 9.3.5 收集硬件信息 9.4 任务4 构建包过滤防火墙 9.4.1 创建防火墙脚本文件 9.4.2 设置防火墙脚本文件为可执行 9.4.3 启动防火墙并设置自动运行 9.4.4 iptables常用命令 9.5 本章小结第10章 构建嵌入式Linux开发环境 10.1 任务1 Linux用户的登录环境 10.1.1 etc/profile文件 10.1.2 运行级别脚本 10.2 任务2 Linux内核的操作 10.2.1 了解Linux内核源码 10.2.2 内核机制 10.2.3 内核模块的装入与卸载 10.3 任务3 Linux I/O端口编程 10.3.1 如何在C语言下使用I/O端口 10.3.2 硬件中断与DMA存取 10.3.3 延迟时间 10.4 任务4 嵌入式Linux开发 10.4.1 构造嵌入式Linux前先要了解的几个关键问题 10.4.2 嵌入式Linux开发环境 10.4.3 开发嵌入式Linux的步骤 10.5 本章小结参考文献



## 章节摘录

插图：1.如何引导当一个微处理器第一次启动时，它开始在预先设置的地址上执行指令。通常在那里有一些只读内存，包括初始化或引导代码。

在PC上，这就是BIOS，它执行了一些低水平的CPU初始化和其他硬件的配置。

BIOS继续辨认哪个磁盘里有操作系统，把操作系统复制到RAM并且转向它。

实际上，这非常复杂，但对我们的目标来说也非常重要。

在PC上运行的Linux依靠PC的BIOS来提供这些配置和OS加载功能。

在一个嵌入式系统里经常没有这种BIOS。

这样就要提供同等的启动代码。

幸运的是，嵌入式系统并不需要PC BIOS引导程序那样的灵活性，因为它通常只需要处理一个硬件的配置。

这个代码更简单也更枯燥。

它只是一个指令清单，将固定的数字塞到硬件寄存器中去。

然而，这是关键的代码，因为这些数值要与你的硬件相符而且要按照特定的顺序进行。

所以在大多数情况下，一个最小的通电自检模块可以检查内存的正常运行、让LED闪烁，并且驱动其他必须的硬件以使主Linux OS启动和运行。

这些启动代码完全根据硬件决定，不可随意移动。

不过，许多系统都有为核心微处理器和内存所定制的菜单式硬件设计。

典型的是，芯片制造商有一个样本主板，可以用来作为设计的参考——或多或少与新设计相同。

通常这些菜单式设计的启动代码是可以获得的，它可以根据用户的需要轻易地修改。

在少数情况下，启动代码需要重新编写。

为了测试这些代码，可以使用一个包含“模拟内存”的电路内置模拟器，它可以代替目标内存。

然后把代码装到模拟器上并通过模拟器调试。

如果这样不行，可以跳过这一步，但这样就要一个更长的调试周期。

这个代码最终要在较为稳定的内存上运行，通常是Flash或EPROM芯片。

接下来需要使用一些方法将代码放在芯片上。

具体怎么做要根据“目标”硬件和工具来定。

一种流行的方法是把Flash或EPROM片插入EPROM或Flash烧制器。

这将把用户的程序“烧”（存）入芯片。

然后，把芯片插入用户的目标板的插座，打开电源。

这个方法需要板上配有插座，但有些设备是不能配插座的。

另一个方法是通过一个JTAG（Joint Test Action Group，联合测试行动小组）界面，它是一种国际标准测试协议（IEEE1149.1兼容），主要用于芯片内部测试。

一些芯片有JTAG界面可以用来对芯片进行编程。

这是最方便的方法。

芯片可以永远被焊在主板上，一个小电缆从板上的JTAG连接器（通常是一个PC片），连到JTAG界面。

下面是PC运行JTAG界面所需的一些惯用程序。

这个设备还可以用来小量生产。

编辑推荐

《Linux基础及应用教程实验指导与实训》由中国水利水电出版社出版。

#### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>