

<<Linux系统及编程基础>>

图书基本信息

书名：<<Linux系统及编程基础>>

13位ISBN编号：9787302286318

10位ISBN编号：7302286310

出版时间：2012-8

出版时间：清华大学出版社

作者：唐晓君 编

页数：271

字数：449000

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>

<<Linux系统及编程基础>>

内容概要

《21世纪高等学校规划教材·计算机科学与技术：Linux系统及编程基础》从易用性和实用性角度出发，以Red Hat Enterprise Linux 6为基础，全面、系统、深入浅出地介绍了Linux系统的概念、使用、管理和开发方面的知识。

《21世纪高等学校规划教材·计算机科学与技术：Linux系统及编程基础》共9章，主要讲述Linux基础及安装、Linux的文件系统、Linux的vim编辑器、Linux系统管理基础、Linux的网络管理及应用、Linux下shell编程、Linux下C编程、GTK+图形界面程序设计、Qt图形界面程序设计。各章内容经过细心设计和精心组织，让读者能够在最短的时间内学到最多的知识。

本书非常适合作为高等学校计算机及相关专业的Linux操作系统课程的教材，也可作为广大Linux用户及Linux系统自学者学习用书。

<<Linux系统及编程基础>>

书籍目录

第1章 Linux基础及安装

1.1 Linux概述

1.1.1 Linux的简介

1.1.2 Linux的特点

1.1.3 Linux的主要组成

1.1.4 Linux的版本介绍

1.1.5 Linux与其他操作系统

1.2 Red Hat Enterprise Linux 6安装

1.2.1 Red Hat Enterprise Linux介绍

1.2.2 选择安装方式

1.2.3 安装步骤

1.2.4 启动Linux

1.3 Linux基础

1.3.1 Linux的用户类型

1.3.2 Linux的登录

1.3.3 Linux的注销

1.3.4 Linux的运行级别

1.3.5 Linux的关机和重启

1.3.6 Linux的基本命令

1.4 Linux的GUI

1.4.1 X Window介绍

1.4.2 GNOME

1.4.3 KDE

本章小结

本章习题

第2章 Linux的文件系统

2.1 Linux文件系统基础

2.1.1 Linux文件系统概述

2.1.2 Linux文件系统的特性

2.1.3 Linux文件系统的组成

2.2 Linux文件系统类型

2.2.1 ext文件系统

2.2.2 其他文件系统

2.3 Linux文件系统操作

2.3.1 文件和目录的基本概念

2.3.2 文件操作命令

2.3.3 目录操作命令

2.3.4 改变文件或目录存取权限的命令

2.3.5 改变用户组和文件主的命令

2.3.6 链接文件的命令

本章小结

本章习题

第3章 Linux的vim编辑器

3.1 vim简介

3.2 vim工作模式

<<Linux系统及编程基础>>

3.3 vim的进入与退出

3.3.1 进入vim

3.3.2 退出vim

3.4 vim的编辑命令

3.4.1 移动光标

3.4.2 文本插入

3.4.3 文本删除

3.4.4 文本复制与粘贴

3.4.5 文本移动

3.4.6 文本查找与替换

3.4.7 重复与取消

3.4.8 vim中执行Shell命令

3.4.9 文件的读写

3.5 使用vim创建Shell脚本

3.6 使用vim创建c程序

本章小结

本章习题

第4章 Linux系统管理基础

4.1 系统启动、运行与系统关闭

4.1.1 系统启动

4.1.2 系统运行级别

4.1.3 系统关闭

4.2 用户管理

4.2.1 用户管理简介

4.2.2 用户管理方法

4.2.3 用户管理相关配置文件

4.2.4 用户管理命令

4.3 设备管理

4.3.1 设备标识

4.3.2 磁盘和分区管理命令

4.3.3 存储设备的挂载与卸载

4.4 进程管理

4.4.1 进程的概念

4.4.2 启动进程

4.4.3 进程管理命令

4.5 日志管理

4.5.1 Linux的日志

4.5.2 常用日志文件

4.5.3 日志分析工具

本章小结

本章习题

第5章 Linux的网络管理及应用

5.1 Linux网络管理命令

5.1.1 ifconfig

5.1.2 ping

5.1.3 netstat

5.1.4 其他常用命令

<<Linux系统及编程基础>>

5.2 文件服务器配置NFS , samba

5.2.1 NFS配置

5.2.2 samba配置

5.3 DNS服务器配置

5.4 Web服务器配置

5.5 Mail服务器配置

5.6 FTP服务器配置

本章小结

本章习题

第6章 Linux下Shell编程

6.1 Shell简介

6.1.1 什么是Shell

6.1.2 Shell种类介绍

6.2 Shell基础

6.2.1 Shell命令处理过程

6.2.2 标准输入 / 输出和重定向

6.2.3 管道

6.2.4 特殊字符

6.2.5 别名

6.2.6 命令历史

6.3 Shell脚本

6.3.1 什么是Shell脚本

6.3.2 Shell脚本执行方式

6.4 Shell变量

6.4.1 用户自定义变量

6.4.2 环境变量

6.4.3 位置变量

6.4.4 特殊变量

6.5 Shell编程基础

6.5.1 Shell脚本的输入 / 输出

6.5.2 Shell的逻辑运算

6.5.3 Shell的算术运算

6.6 Shell的控制结构

6.6.1 if语句

6.6.2 case语句

6.6.3 for语句

6.6.4 while语句

6.6.5 until语句

6.6.6 break和continue语句

6.7 Shell函数

6.7.1 函数定义

6.7.2 函数调用

本章小结

本章习题

第7章 Linux下C编程

7.1 Linux下C编程基础

7.2 常用开发工具

<<Linux系统及编程基础>>

- 7.2.1 GCC简介
- 7.2.2 GCC的使用
- 7.2.3 简单的C语言程序
- 7.2.4 程序调试工具gdb
- 7.2.5 程序维护工具make
- 7.2.6 Linux下make示例
- 7.3 进程控制系统调用
 - 7_3.1 进程创建
 - 7.3.2 进程执行
 - 7.3.3 获取指定进程标识符
 - 7.3.4 进程终止
 - 7.3.5 进程等待
 - 7.3.6 进程间信号通信
- 7.4 线程控制系统调用
 - 7.4.1 线程控制系统调用
 - 7.4.2 线程控制函数示例
- 7.5 文件系统调用
 - 7.5.1 创建文件
 - 7.5.2 打开文件和关闭文件
 - 7.5.3 读写文件
 - 7.5.4 文件定位
- 本章小结
- 本章习题
- 第8章 GTK+图形界面程序设计
 - 8.1 GTK+程序设计简介
 - 8.2 使用GTK+开发图形界面程序
 - 8.2.1 按钮和标签
 - 8.2.2 文本输入框
 - 8.2.3 复选按钮
 - 8.2.4 组合框
 - 8.2.5 框架
 - 8.2.6 图像控件
 - 8.2.7 菜单的制作
 - 8.2.8 工具栏的制作
 - 8.2.9 状态栏的制作
 - 本章小结
 - 本章习题
- 第9章 Qt图形界面程序设计
 - 9.1 Qt程序设计简介
 - 9.2 开发Qt图形界面程序
 - 9.2.1 标准输入框例子
 - 9.2.2 标准对话框的实例
 - 9.2.3 记事本小程序
 - 本章小结
 - 本章习题
- 参考文献

<<Linux系统及编程基础>>

章节摘录

版权页：插图：Linux有图形化的管理工具以及日常办公的工具，比如电子邮件、网络浏览器和文档处理工具等。

不过在Linux中，图形化的管理工具通常是控制台（命令行）工具的扩展，也就是说，用图形化工具能够完成的所有工作，用控制台命令行同样能够完成。

而使用图形化的工具并不妨碍用户对配置文件进行手工修改，其实际意义可能并不是显而易见，但是如果在图形化管理工具中所做的任何工作都可以以命令行的方式完成，这就表示这些工作同样可以使用一个脚本来实现。

脚本化的命令可以成为自动执行的任务。

Linux中的配置文件是可读的文本文件，这与过去的Windows中的INI文件类似，但与Windows操作系统的注册思路有本质的区别。

每一个应用程序都有自己的配置文件，而通常不与其他配置文件放在一起。

不过大部分配置文件都存放于一个目录树（ect）下的单独位置，所以在逻辑上看起来是一起的。

文本文件的配置方式可以不通过特殊的系统工具就可以完成配置文件的备份、检查和编辑工作。

（3）文件名扩展。

Linux不使用文件名扩展来识别文件的类型，这与Windows操作系统不同。

Linux操作系统是根据文件的头内容来识别其类型。

为了提高用户的可读性，Linux仍可以使用文件名扩展，这对Linux系统来说没有任何影响。

不过有一些应用程序，比如Web服务器，可能使用命名约定来识别文件类型，但这只是特定应用程序的需要而不是Linux系统本身的要求。

Linux通过文件访问权限来判断文件是否为可执行文件，任何一个文件都可以赋予可执行权限，程序和脚本的创建者或管理员可以将它们识别为可执行文件，这样做有利于安全，使得保存到系统上的可执行文件不能自动执行，可以防止许多脚本病毒。

（4）重新引导。

在使用Windows系统时，也许已经习惯出于各种原因而重新引导系统（即重新启动），但在Linux系统中这一习惯需要改变。

一旦开始运行，它将保持运行状态，直到受到外来因素的影响，比如硬件故障。

实际上，Linux系统的设计使得应用程序不会导致内核的崩溃，因此不必经常重新引导，所以除了Linux内核之外，其他软件的安装、启动、停止和重新配置都不用重新引导系统。

如果用户确实重新引导了Linux系统，问题很可能得不到解决，甚至还会使问题更加恶化，因此在学习Linux操作系统时，要克服重新引导系统的习惯。

另外，可以远程地完成Linux中的很多工作，只要有一些基本的网络服务在运行，就可以进入到那个系统。

而且，如果系统中一个特定的服务出现了问题，用户还可以在故障诊断的同时让其他服务继续运行。

当用户在一个系统上同时运行多个服务的时候，这种管理方式更为重要。

（5）命令区分大小写。

所有的Linux命令和选项都区分大小写，如—R和—r不同，会去做不同的事情。

控制台命令几乎都使用小写，在后面的章节中会对Linux操作系统中的命令进行详细讲解。

<<Linux系统及编程基础>>

编辑推荐

<<Linux系统及编程基础>>

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>