<< 学习Perl >>

<< 学习Perl >>

13 ISBN 9787564119287

10 ISBN 7564119284

出版时间：2010-1

页数：639

字数：

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

Do you need tools for making your system administration work easier and more efficient? You've come to the right place. Perl is a powerful programming language that grew out of the traditional system administration toolbox. Over the years it has adapted and expanded to meet the challenges of new operating systems and new tasks. If you know a little Perl, and you need to perform system administration tasks, this is the right book for you. Readers with varying levels of both Perl programming experience and system administration experience will all find something of use within these pages. What's New in This Edition? A tremendous amount of work went into updating this book so it could be even better than the first edition. Here's some of what has been improved in the second edition: New title My editors and I realized that the material in this book was more about how to automate your system administration work in ways that would make your working life more efficient and pleasant than it was about Perl. While Perl is still the toolshed that makes all this possible, it isn't the main focus of the book. New material It's hard to know where to begin on this one. The new edition is four chapters and two appendixes bigger (with a total page count that is 50% greater) than the last one. Included in this edition are a cornucopia of new tools and techniques that you are going to love. I tried to add material on the things I wished I had sysadmintargeted material on, including: XML and YAML best practices (using XML:: LibXML, XML::TwJ. g. and XPath); dealing with config files; more advanced LDAP topics (including updated Net:: LDAP information); email-related topics (including POP3/IMAP, MIME, and spam); new ways of dealing with filesystems; more advanced log file creation and parsing tools; DHCP; mappin/monitoring a network using Nmap and other tools; packet creation and sniffing; information reporting using tools like GraphViz, RRDtool, and Timeline; using SHA-2 instead of MD5; SNMPv3; Mac OS X; converting VBScript code to Perl; geocoding; MP3 file manipulation; using Google Maps; and so on. New advice Part of the value of this book is the advice you can pick up from an experienced system administrator like me who has been doing this stuff for a long time and has compared notes with many other seasoned veterans. This new edition is packed with more sidebars to explain not only the what, but also the why behind the material. Operating system and software information updates All of the text and code has been updated and augmented to work with the latest versions of Unix- (including Linux and Mac OS X) and Windows-based operating systems. Module and code updates/improvements The descriptions and code in this book match the latest versions of the modules mentioned in the first edition. In cases where a module is no longer available or a better alternative has emerged, the appropriate replacement modules have been substituted. Also, all example code is now "use strict" friendly. Errata corrected I have attempted to address all of the errata I received from all of the printings of the first edition. I appreciate the time readers took to report errors to O'Reilly and me so I could fix them at each printing and in this edition. Special thanks go to Andreas Karrer, the German translator for the first edition. Andi pored over every single byte of the original text and submitted almost 200 (mostly layout-related) corrections, all with good cheer. How This Book Is Structured Each chapter in this book addresses a different system administration domain and ends with a list of the Perl modules used in that chapter and references to facilitate deeper exploration of the information presented. The chapters are as follows: Chapter 1, Introduction This introductory chapter describes the material covered in the book in more detail, explaining how it will serve you and what you need to get the most from it. The material in this book is powerful and is meant to be used by powerful people (e.g., Unix superusers and Windows-based operating system administrators). The introduction provides some important guidelines to help you write more secure Perl programs.

Perl   (2)(   )

Perl

        "   "   Perl   *
        *   *   XML   YAML   *   DBI
MySQL   MS SQL   Oracle*   LDAP   Active Directory   *
        *   *   NIS   DNS   DHCP*
                        *
        XML/XPath   LDAP   SNMP   SQL
        Perl   (2)(   )   Perl

David N   Blank-Edelman   25

Preface 1. Introduction Automation Is a Must How Perl Can Help You This Book Will Show You How What You Need Some Notes About the Perl Versions Used for This Book What About Perl 5.10? What About Strawberry Perl? What About Perl 6? Some Notes About Using Vista with the Code in This Book Locating and Installing Modules Installing Modules on Unix Installing Modules on Win32 It's Not Easy Being Omnipotent Don't Do It Drop Your Privileges As Soon As Possible Be Careful When Reading Data Be Careful When Writing Data Avoid Race Conditions .Enjoy References for More Information 2. Filesystems Perl to the Rescue Filesystem Differences Unix Windows-Based Operating Systems Mac OS X Filesystem Differences Summary Dealing with Filesystem Differences from Perl Walking or Traversing the Filesystem by Hand Walking the Filesystem Using the File::Find Module Walking the Filesystem Using the File::Find::Rule Module Manipulating Disk Quotas Editing Quotas with edquota Trickery Editing Quotas Using the Quota Module Editing NTFS Quotas Under Windows Querying Filesystem Usage Module Information for This Chapter References for More Information 3. User Accounts Unix User Identities The Classic Unix Password File Changes to the Password File in BSD 4.4 Systems Shadow Passwords Windows-Based Operating System User Identities Windows User Identity Storage and Access Windows User ID Numbers Windows Passwords Don't Play Nice with Unix Passwords Windows Groups Windows User Rights Building an Account System to Manage Users The Backend Database The Low-Level Component Library The Process Scripts Account System Wrap-Up Module Information for This Chapter References for More Information Unix Password Files Windows User Administration 4. User Activity Process Management Windows-Based Operating System Process Control Unix Process Control File and Network Operations Tracking File Operations on Windows Tracking Network Operations on Windows Tracking File and Network Operations in Unix Module Information for This Chapter Installing Win32::Setupsup References for More Information 5. TCP/IP Name and Configuration Services Host Files Generating Host Files Error-Checking the Host File Generation Process Improving the Host File Output Incorporating a Source Code Control System NIS, NIS+, and WINS NIS+ Windows Internet Name Server (WINS) Domain Name Service (DNS) Generating DNS (BIND) Configuration Files DNS Checking: An Iterative Approach DHCP Active Probing for Rogue DHCP Servers Monitoring Legitimate DHCP Servers Module Information for This Chapter References for More Information 6. Working with Configuration Files Configuration File Formats Binary Naked Delimited Data Key/Value Pairs Markup Languages All-in-One Modules Advanced Configuration Storage Mechanisms Module Information for This Chapter References for More Information XML and YAML 7. SQL Database Administration Interacting with a SQL Server from Perl Using the DBI Framework Using ODBC from Within DBI Server Documentation MySQL Server via DBI Oracle Server via DBI Microsoft SQL Server via ODBC Database Logins Monitoring Space Usage on a Database Server Module Information for This Chapter References for More Information DBI Microsoft SQL Server ODBC Oracle 8. Email Sending Mail Getting sendmail (or a Similar Mail Transport Agent) Using the OS-Specific IPC Framework to Drive a Mail Client Speaking the Mail Protocols Directly Common Mistakes in Sending Email Overzealous Message Sending Subject Line Waste Insufficient Information in the Message Body Fetching Mail Talking POP3 to Fetch Mail Talking IMAP4rev1 to Fetch Mail Processing Mail Dissecting a Single Message Dissecting a Whole Mailbox Dealing with Spam Support Mail Augmentation Module Information for This Chapter References for More Information 9. Directory Services What's a Directory? Finger: A Simple Directory Service The WHOIS Directory Service LDAP: A Sophisticated Directory Service LDAP Programming with Perl The Initial LDAP Connection Performing LDAP Searches Entry Representation in Perl Adding Entries with LDIF Adding Entries with Standard LDAP Operations Deleting Entries Modifying Entry Names Modifying Entry Attributes Deeper LDAP Topics Putting It All Together Active Directory Service Interfaces ADSI Basics Using ADSI from Perl Dealing with Container/Collection Objects Identifying a Container Object So How Do You Know Anything About an Object? Searching Performing Common Tasks Using the WinNT and LDAP Namespaces Working with Users via ADSI Working with Groups via ADSI Working with File Shares via ADSI Working with Print Queues and Print Jobs via ADSI Working with

Windows-Based Operating System Services via ADSI Module Information for This Chapter References for More Information LDAP ADSI 10. Log Files Reading Text Logs Reading Binary Log Files Using unpack() Calling an OS (or Someone Else's) Binary Using the OS's Logging API Structure of Log File Data Dealing with Log File Information Space Management of Logging Information Log Parsing and Analysis Writing Your Own Log Files Logging Shortcuts and Formatting Help Basic/Intermediate Logging Frameworks Advanced Logging Framework Module Information for This Chapter References for More Information 11. Security Noticing Unexpected or Unauthorized Changes Local Filesystem Changes Changes in Data Served Over the Network Noticing Suspicious Activities Local Signs of Peril Finding Problematic Patterns Danger on the Wire, or "Perl Saves the Day" Preventing Suspicious Activities Suggest Better Passwords Reject Bad Passwords Module Information for This Chapter References for More Information 12. SNMP Using SNMP from Perl Sending and Receiving SNMP Traps, Notifications, and Informs Alternative SNMP Programming Interfaces Module Information for This Chapter References for More Information 13. Network Mapping and Monitoring Network Mapping Discovering Hosts Discovering Network Services Physical Location Presenting the Information Textual Presentation Tools Graphical Presentation Tools Monitoring Frameworks Extending Existing Monitoring Packages What's Left? Module Information for This Chapter References for More Information 14. Experiential Learning Playing with Timelines Task One: Parsing crontab Files Task Two: Displaying the Timeline Task Three: Writing Out the Correct XML File Putting It All Together Summary: What Can We Learn from This? Playing with Geocoding Geocoding from Postal Addresses Geocoding from IP Addresses Summary: What Can We Learn from This? Playing with an MP3 Collection Summary: What Can We Learn from This? One Final Exploration Part One: Retrieving the Wiki Page with WWW::Mechanize Part Two: Extracting the Data Part Three: Geocoding and Mapping the Data Summary: What Can We Learn from This? Remember to Play Module Information for This Chapter Source Material for This Chapter A. The Eight-Minute XML Tutorial B. The 10-Minute XPath Tutorial C. The 10-Minute LDAP Tutorial D. The 15-Minute SQL Tutorial E. The Five-Minute RCS Tutorial F. The Two-Minute VBScript-to-Perl Tutorial G. The 20-Minute SNMP Tutorial Index

FAT filesystems are case-insensitive. In Unix, an attempt to open a file using thewrong case   i.e., MYFAVORITEFILE versus myfavoritefile   will fail, but with FATor VFAT, this will succeed with no problem.Instead of a forward slash, FAT uses the backward slash   as its path separator.This has a direct ramification for the Perl programmer, because the backslash is aquoting character in Perl. Paths written in single quotes with only single separators   e.g, Spathe' dirdirfilename'   are just fine. However, situations in which youneed to place multiple backslashes next to each other   e.g, serverdirfile   arepotential trouble. In those cases, you have to be vigilant in doubling any multiplebackslashes. Some Perl functions and some Perl modules will accept paths withforward slashes, but you shouldn't count on this convention when programming.It is better to bite the bullet and write winnttemp than to learn that yourcode breaks because the conversion hasn't been done for you.FAT files and directories have special flags associated with them that are calledattributes. Example attributes include "Read-only" and "System."The root of a FAT filesystem is specified starting with the drive letter on which thefilesystem resides. For instance, the absolute path for a file might be specified asc : homecindydocsresumecurrent.doc.FAT32 and NTFS filesystems have the same semantics as VFAT filesystems. They sharethe same support for long filenames and use the same root designator. NTFS is moresophisticated in its name support, however, because it allows these names to be specified using Unicode. Unicode is a multibyte character encoding scheme that can be usedto represent all of the characters of all of the written languages on the planet.NTFS also has some functional differences that distinguish it from the other Windowsand basic Unix filesystems. Later in this chapter, we will write some code to take advantage of some of these differences, such as filesystem quotas. NTFS supports ACLs,which provide a fine-grained permission mechanism for file and directory access.

"　　　　　　　　　　　　　　　　　　　　　　　　(　　　)
Peri　　　　　　　　　David
"　　——Peter Baer Galvin　Corporate Technologies　　　　　　　IT

　　Perl　　　　　　　　　(　2 　)(　　　)

PDF

:http://www.tushu007.com