Expert Shell Scripting

Ron Peters

Apress®

Expert Shell Scripting

ISBN-13 (pbk): 978-1-4302-1841-8

ISBN-13 (electronic): 978-1-4302-1842-5

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Lead Editor: Frank Pohlmann
Technical Reviewer: Brian Culp
Editorial Board: Clay Andres, Steve Anglin, Mark Beckner, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Michelle Lowman, Matthew Moodie, Jeffrey Pepper, Frank Pohlmann, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Project Manager: Sofia Marchant
Copy Editor: Candace English
Associate Production Director: Kari Brooks-Copony
Production Editor: Liz Berry
Compositor: Pat Christenson
Proofreader: Lisa Hamilton
Indexer: Julie Grady
Artist: April Milne
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit http://www.springeronline.com.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales–eBook Licensing web page at http://www.apress.com/info/bulksales.

The source code for this book is available to readers at http://www.apress.com. You will need to answer questions pertaining to this book in order to successfully download the code.

Contents at a Glance

PART 3 Useful Scripts

Contents

About the Author

RON PETERS has worked as a system administrator for most of the last 15 years. He was a senior administrator at Intel in a 24/7 production environment and was the primary administrator of a large compute cluster dedicated to design work. He is now a Linux/UNIX administrator for Columbia Sportswear. He enjoys spending time with his family, restoring his Dodge Challenger, and playing racquetball.

About the Technical Reviewer

BRIAN CULP has worked professionally in the information-services industry for 20 years. Throughout those years he has worked with startups and small businesses, and spent a dozen years employed by a leading company in the IT industry. Brian has spent time on service desks, as a UNIX systems admin, a project manager, an e-commerce/business web site administrator, and a solutions developer.

Brian's development and use of UNIX shell scripts has always grown out of specific needs, as he always seems to be in some stage of trying to solve a technical problem. He hopes you will find the scripts and methods described in this book useful in building your own problem-solving toolkit.

Acknowledgments

For most things in this world, we depend on others. This book is no exception; this project is larger than most I have taken on, and I could not have done it alone. I would firstly and most importantly like to thank my God for the free gift of life as well as the skills and abilities that enabled me to write this book. I would also like to thank my wife, Kathleen, and my two boys, Austin and Grant, for enduring the seemingly endless hours and evenings I've been spending with my laptop.

I want to express my gratitude to the two Brians: to Brian Grell for giving me ideas and discussing many topics that have found their way into this book, and to Brian Culp for reviewing the whole book and keeping me focused on what I was trying to say, and asking the right questions so I could maintain clarity.

Finally, I want to thank all the other editors who have had a hand in helping me remove the Englilsh[1] from my writing.

1. http://www.homestarrunner.com/sbemail64.swf

Introduction

I learned the basics of programming when I was in school; I learned how to shell-script by example.

I've met and worked with many system administrators and other *NIX folks, each of whom has their own bag of tricks when it comes to managing a system, interacting with their environment, or coding a script. It's always very useful to have conversations and interact with people like this because you invariably gain some tidbits that you can throw into your own collection of tricks. I decided to collect all the useful shell-scripting and interaction techniques I have learned through the years and combine them into one beneficial reference guide. In fact, I used some of my own notes about those techniques while writing this book. Since I haven't memorized everything present in this book, I would periodically look up items when I was working on various tasks. I want this book to be the beginning of a higher-level reference library that can be added to and can grow continually.

You might be aware of the large number of shell-scripting books and online resources aiding in the mastery of shell scripting. Many are excellent and cover a wide range of topics. The main purpose of this book is to combine some of the most unique tools, code snippets, and scripts that go beyond the level of basic scripts. I wanted to create a cookbook of sorts—lesser-known recipes and fairly advanced algorithms that have proved useful to me.

I have included scripts you can use as is, and sample scripts illustrating a specific algorithm. I also demonstrate a few complex commands that may be useful on the command line. I have tried to tailor the scripts to be useful at multiple levels. Most times, however, there is little or no error-checking since that is not necessarily the point of a specific script. You must be prepared to make modifications to fit your local environment.

How This Book Came About

My friend Brian Culp and I have worked together as UNIX system administrators for many years. Periodically, Brian or I will be working on some script and run into a problem. One of us will stop, walk over to the other, and say something like, "Do you have any code that does *X*?" The answer may be no, in which case we'll launch into a discussion on how we might tackle the problem, or come up with a few different solutions. However, many times it might be something like, "Hmm, yeah, I think I remember doing something like that in a script that does *X* on system *Y*. Let me look for a minute." A few carefully chosen `grep` commands, and the solution is at hand.

Although finding the solution we want is great, it's not the most efficient use of our time. To go from having a place to store and organize all of our (and, of course, other programmers') gems and having them in a heavily documented form, to writing a book on the subject was only a short step. Even though it is possible to search online references quickly for specific code, there were many occasions when I just wanted to pull a book off my shelf. It's not always obvious what to search for when you have a specific itch that needs scratching, so you're not sure exactly to what search for online.

This is to some extent an expression of my own limitations: my family and friends think I'm a computer guru, but rest assured, I know better. There are many programmers out there who are much more adept at shell coding than I am. I mainly intended to collect, order, and explain code that I have found to be highly useful in my professional experience as a system administrator, and share that information with others.

Who Should Read This Book

The book is meant for the intermediate shell coder up to the advanced shell-code hacker, because I don't explain many basic programming structures. If you're looking for that type of book, you should look to the resources mentioned in Appendix C.

This is not to say that the beginner won't find this book useful; it may work well as a supplementary reference to a more traditional shell-scripting training guide. But there is a difference between learning English as a second language and learning how to apply sarcasm. This book is like sarcasm in that example; it assumes some basic shell-code literacy.

I go into great detail about how and why the scripts were written in their present form, and I include some explanation of how to avoid certain problems. Much of my learning came from sources heavy in obfuscation and light on clarity, so I tried to be as explicit as possible, and favored explaining too much rather than too little. You can think of many chapters I included as shell scripts with extremely detailed commentary.

The book is divided into three parts: "Basic Scripting Techniques," "System Interaction and Advanced Techniques," and "Useful Scripts." Most chapters serve as stand-alone discussions, although they may refer to other chapters on some minor points.

Issues and Ideas

I have made every effort to test the code that I included in this book to validate that it works. With a project of this size, however, even with the number of eyes that have reviewed it, there may be mistakes. I would like to know about the mistakes as well as, and more importantly, any other ideas and scripts that could be used for future revisions of this book. Please drop me a note at `rbpeters@peterro.com`.