

Title Classic Shell Scripting

Author Arnold Robbins & Nelson H.F. Beebe

Publisher O'Reilly

Pages 534

ISBN 0-596-00595-4

Shell scripting has been an integral part of UNIX and system administration for years (which might be cause for the word “classic” in the title). It has been, and remains to be, a very powerful and widely used tool across servers and home computers. Shell scripting provides a POSIX scripting language that interacts directly with the shell in which you are working. This allows the programmer to quickly accomplish large tasks in a very short period of time.

As scripting languages like Python, Ruby, and Perl become more widely used, usage of basic shell scripting has seen a decline. However, there are several reasons why shell scripting still remains a vital tool on UNIX systems. Firstly, shell scripts are portable across all UNIX systems, making it excellent for use across a large number of computers. Secondly, it interacts seamlessly with the UNIX shell on your system. You can call command line tools and utilities from within your script, thereby decreasing the work that the programmer must complete by using pre-existing, widely supported, tools that he or she is likely already familiar with. And finally, as is true with most scripting languages, it is very easy to quickly code a powerful script in a short amount of time – making it a valuable time saver.

This book serves as an excellent tutorial in the field of shell scripting, and leaves the reader with the ability to sit down and knock out scripts with ease. However, before delving into the details of the book it should be made clear that the book is not a tutorial for people new to programming. While the book does an excellent job of explaining most everything it makes reference too, it is simply not addressed to people new to programming. Readers unfamiliar with programming already will likely find themselves constantly looking to other references for definitions and explanations of basic programming concepts that more experienced readers will know almost instinctively.

That said, I would actually recommend the book to those that are relatively new to UNIX systems. Because shell scripting is so closely tied with the command line tools that come standard on UNIX systems, the book gives an excellent tutorial on many of these programs and how to use them. The authors even give a brief tutorial on regular expressions – which is incredibly useful to anyone not already familiar with them.

Perhaps of even more interest is the forty page tutorial on ‘awk’. Those accustomed to a UNIX system most likely know what awk is, and have likely worked with it a little. However, for those new to the language or even for those that would like to learn a little more, the authors give an excellent overview of awk, what it can do, and how to use it. The chapter on awk is slyly titled “Enough awk to Be Dangerous” – which could be taken as a funny, yet subtle, nod to the usage of awk in the hacker world.

The authors' writing itself is easy to read and is very friendly. The book is actually written in first person plural, giving it less of a textbook tone and more of a casual friend-helping-out-a-friend feeling. The text includes an enormous amount of code and screen output examples to follow along with the chapters – most of which are scripts that system administrators would actually be interested in knowing how to code (like displaying a sorted list of processes and the users that are running them). Whenever a new program or tool is introduced, a quick crash-course in its usage is given in a box set off from the rest of the chapter – many of which are easier to read than the 'man' pages that come with them on the UNIX system itself.

The selection of topics in the book is also excellent. The authors move from basic subjects like text processing and looping to inputting files, working with processes, and automating software builds. The book is laid out in an intelligent fashion, and the reader never feels lost or unprepared for the chapter ahead.

Of note is the glossary. Generally, I simply glance over the glossary and make sure that it is adequate for the book. However, this glossary deserves special mention. Almost every term mentioned in the book, from the simple to the complex, has an entry in the glossary. More importantly, however, are the entries themselves. None of the entries are simple one or two sentence definitions. Each entry is at least a paragraph, contains a background and description, and many even contain examples and example usages! This is truly the champion of glossaries – and I hope that future authors take it as an example of what glossaries should be. Anyone can look up the definition to something online – providing a well-written, yet brief, description of something is a far more helpful practice.

This is an excellent example of how tutorial books should be written, and I recommend it to anyone that will be working consistently in an UNIX environment. Even if you have been around the operating system for years, the authors have included little tricks and ideas that will increase productivity and make some of the more monotonous system administrator tasks quick and easy. The authors have done a fantastic job in putting this book together, and it was a pleasure to read.

Reviewed by:
Benjamin Hilburn