book reviews

ELIZABETH ZWICKY, WITH BRAD KNOWLES, SAM F. STOVER, AND RIK FARROW

BETTER: A SURGEON'S NOTES ON PERFORMANCE

Atul Gawande

Picador, 2007. 257 pages.
ISBN 978-0-312-42765-8

This is obviously not even vaguely a book about system administration. I didn't pick it up intending to review it. I'm not quite sure why I did pick it up. However, I ended up mulling over the similarities between the intractable medical problems it describes and system administration.

This book is about how medicine improves, and it points out that the main problems are not technical, but human: How do you get people to do things that they should do to prevent long-term problems but that give them nothing but annoyance in the short term? This is clearly just as applicable to getting your users to comply with security requirements as it is to getting your doctors to wash their hands to prevent spreading infection.

I also found a useful moral in the tale of Dr. Semmelweiss, who famously figured out that doctors were spreading infection between women giving birth, and then didn't manage to get them to stop. This turns out not to be a simple story of an unheard genius, but the story of somebody who was technically right, but so annoying and unable to explain himself that nobody listened. We've all worked with that guy, right?

If you're looking for some fascinating and useful concepts to wrap your mind around, and are willing to stretch a little to apply them, I think you'll find a good deal of usefulness here.

GEEKONOMICS: THE REAL COST OF INSECURE SOFTWARE

David Rice

Addison-Wesley, 2008. 339 pages.
ISBN 978-0-321-47789-7

Software runs the world and it doesn't work reliably. This is clearly a bad thing. So why doesn't somebody fix it?

This book lays out the forces that keep software unreliable, explains why open source software isn't the cure, and suggests some solutions. There's an interesting discussion of licensing for software engineers, which system administrators should find eerily familiar.

I am reasonably convinced that liability for software manufacturers would improve the world. Every time I think software licensing can't get any more absurd, I discover that I am not yet adequately cynical. (Imagine my surprise when I bought a cookbook with an included CD and discovered that the software license for the CD was printed on the back side of the book's paper slipcover, where there is normally nothing at all.) I also appreciated the discussion of the forces that drive open source toward bloat.

Although I agree with the author, I think he overstates the case in several places. Software isn't the only source of unexpected interactions, and a serious case can be made that better error tolerance is important entirely separate from software problems.

YOUR BRAIN: THE MISSING MANUAL

Matthew MacDonald

O'Reilly, 2008. 247 pages.
ISBN 978-0-596-51778-6

Popular books on the brain are often minefields of attractive but inaccurate information. This one manages to avoid most of the hype and easy faulty generalizations while providing easy to read and digest information about the brain. It has useful tricks without the breathless hype of many popular books.

In particular, it has a nice clear explanation of what is known about gender differences and the brain (none of which is really all that exciting). This is a nice antidote to a lot of the nuttier stuff going around. Unfortunately, there aren't any references, so you're pretty much stuck taking the author's word for it. Comparing what this says to sources I trust, that works out OK. But if you don't happen to know any neuroscientists to ask, you would have a hard time figuring out whether this was in fact more accurate than some very popular books with numerous irrelevant footnotes.

EATING THE IT ELEPHANT: MOVING FROM GREENFIELD DEVELOPMENT TO BROWNFIELD

Richard Hopkins and Kevin Jenkins

IBM Press, 2008. 213 pages.
ISBN: 978-0-13-713012-2

On the face of it, this is the single most relevant book I'm reviewing this issue. It's about building big projects inside an existing IT organization, where you are having to interface with all the existing, crufty systems. The analogy with building on contaminated ground will seem quite apt to anybody who's tried to add anything to a mature IT infrastructure.

Clearly, the authors have worked on many projects like this and learned painfully. They have a grand theory of how to deal with the situation, which they lay out with gripping metaphors. They compare it to other development methodologies and provide some implementation advice.

However, their theory involves a grand unifying program, and implementing that program is going to be a major stumbling block for anybody who wishes to use the process. Furthermore, the grand unifying program needs to contain a knowledge representation of all the parts of the IT infrastructure. They're quite correct that having a working knowledge representation is extremely powerful and enables all sorts of fun stuff, but they imply that it's pretty straightforward for a business analyst to go from an existing program to an appropriate representation of the objects, relationships, and constraints (e.g., every account has one and only one username, every file is owned by one account, and so on and so forth). It's not at all a straightforward matter of discovery and analysis.

This book has interesting ideas for people doing large development projects that interoperate with existing systems. For most sites it's going to be much the same as better—not an immediate resource you can implement, but an intriguing starting place for adaptation.

THE BOOK OF IMAP: BUILDING A MAIL SERVER WITH COURIER AND CYRUS

Peer Heinlein and Peer Hartleben

No Starch Press, 2008. 368 pages.
ISBN 978-1593271770

REVIEWED BY BRAD KNOWLES

For us mail server administrators, there aren't many books in print that discuss installing, configuring, and operating IMAP mail systems. If you do a search at your local library or on bookseller Web sites such as Amazon, you will discover that there's only one other book available that is devoted to the subject—*Managing IMAP* by Dianna and Kevin Mullet—published way back in the dark ages of 2000 and now very long in the tooth. There are a few other books that might give us a single chapter, and at least one or two other books on programming as it relates to Internet email, but that's about it. This is the gap that *The Book of IMAP* is intended to fill, specifically for Courier-IMAP and Cyrus.

The book is separated into three standalone parts. Part One, "How to Set Up and Maintain IMAP Servers," discusses topics that are generally applicable to most IMAP servers. Part Two is devoted to Courier-IMAP, and Part Three is about Cyrus. There are also three appendix chapters providing an IMAP command reference, a POP3 command reference, and a guide to installing from source code as opposed to binary packages. The last 20 or so pages are devoted to a fairly extensive index.

The authors clearly work pretty much exclusively with Linux. Everywhere that they talk about differences among specific platforms, they discuss choices such as SuSE, Red Hat, and Debian/Ubuntu, and that's about it. If you can look past their obviously Linux-oriented nature, you should be fine.

One thing that really annoyed me about this book is the occasional misspellings and improper grammar. It is clear that English is not the first language of the authors, although it is likewise clear that the authors care a great deal about proper word usage and sentence structure. This is what makes it doubly annoying when you run across phrases such as "However, these combinations cannot be combined with additional permissions...." There are also typesetting issues, with paths to files broken in the middle of a directory name, when they should be broken at directory separators (e.g., "/"), or where options to "./configure" should not be broken across two lines at all.

Chapter 1 starts with a review of protocols and terms, and Chapter 2 is a step-by-step review of the POP3 and IMAP protocols. Chapter 3 launches into the much weightier topic of load distribution and reliability, starting with load-balancing technologies (including DNS round-robin, round-robin via iptables, and Linux Virtual Server), but where is the discussion of load-balancing switches?

Chapter 3 also covers the subject of IMAP proxies and mentions one reason why you might want to run IMAP proxies even if you have only one IMAP server—certain Webmail IMAP clients are very poorly behaved and open a separate IMAP connection for virtually every single user-visible element on the screen, and caching IMAP proxies help offload much of that work from the back-end IMAP server. However, this ignores the fact that certain other IMAP clients are also equally poorly behaved, a fact we know all too well here at my current employer. Therefore, anyone anywhere who is running an IMAP server of any size may potentially benefit from having a system with a caching IMAP proxy daemon.

Chapter 4 takes us into the subject of choosing a filesystem and filesystem tuning for maximum performance, as well as selecting benchmarking and stress-testing tools to help you during this process. You guessed it; they only tested ext3fs, ReiserFS, and XFS, although they do actually mention that OpenSolaris uses something called ZFS. Fortunately, they at least show the difference that high-performance disk drives can make, and they talk about important things such as RAID and NFS, highlighting various options you may want to look at to help improve your performance.

Chapter 5 is about potentially useful Webmail clients, including Squirrelmail and Horde/IMP, and it goes into more detail about some of the problems that such clients can cause and why you might want to use a caching IMAP proxy to help solve those. The subject of Chapter 6 is migrating IMAP servers, using tools such as imapsync, pop2imap, imap_migrate, imapcopy, and imap_tools. This chapter also talks about converting mailbox formats, changing IMAP folder names, and determining cleartext passwords—all things that you might need to worry about if you're migrating from one type of IMAP server to another.

With Chapter 7, we get into the second section of the book, where Peer Heinlein discusses the Courier-IMAP server itself. He starts by covering the basics of binary package installation, what gets put where in the filesystem, initial startup, integrating Courier with MTAs (postfix, qmail, and Exim), optimizing the configuration, and what configuration parameters do and where they go. Chapter 8 is about Maildir as an email storage format, how the IMAP namespace impacts that, filenames of email messages, and what flags can be attached to the files.

Chapter 9 focuses on user data and authentication and the myriad different ways that can be achieved, whether through internal-only methods or tying into external systems such as MySQL, PostgreSQL, or LDAP, and whether that's done directly or via PAM, etc. In this long chapter the authors try to make sense of all the hash and organize the information in a reasonable fashion. Courier-IMAP does not support the full SASL standard, but it does support enough of it that you can implement a complete IMAP mail server system. The advantage here is that the authors have left out much of the complexity that Cyrus includes with the SASL Reference Implementation, which has been a large part of why so many mail admins go screaming into the night when they hear the term.

Chapter 10 is for Courier administrators and discusses setting up various different types of shared folders, setting up quotas, using Courier to build an IMAP proxy, configuring systems for "push" IMAP email, and sending emails via IMAP.

Chapter 11 starts the third section of the book, where Peer Hartleben talks about Cyrus. He begins with structure and basic configuration, including installation (binary packages again), optional additional tools that are intended to make it much easier to administrate Cyrus via a WebUI, the Cyrus hierarchy in the filesystem and permissions, features and functions, and authentication (SASL), and then provides a "Quickstart Guide," which includes integration with postix.

Chapter 12 takes us into the Cyrus configuration file, and although it is short, it covers many different configuration options that could have a huge impact on the performance of your server. Chapter 13, on authentication and safeguards, starts with SSL and TLS encryption, SSL certificates, and SASL. For SASL, this chapter gets into detail about how various different modules interact with user authentication schemes, including /etc/passwd logins, Berkeley DB files, and integration with external database systems such as MySQL or LDAP and through intermediary systems such as PAM, and with Kerberos.

Chapter 14 covers advanced Cyrus configuration, including quotas, shared folders and ACLs, virtual domains, sorting email messages into subdirectories and the use of hashed directory schemes for enhanced performance, setting up multiple different partitions for users, the Sieve server-side message filtering language for IMAP servers, integrating Cyrus with other MTAs, backing up and restoring user mailbox data, and performance tuning.

Chapter 15 delves into internal structure and modules, and although this probably isn't strictly necessary, you learn a lot that may turn out to be extremely useful regarding which internal modules do what, what tools are available to do analysis, maintenance, and repairs, the function of a multitude of other lesser-known Cyrus tools, and using the cyradm administration tool.

Chapter 16 details Cyrus at the filesystem level, focusing on the email directory and the administration directory, and we see the function for each of the main subdirectories in these trees. Finally Chapter 17 is all about using Cyrus in a cluster, starting with the cyrus aggregator (a.k.a. "murder," as in "a murder of crows"), the front-end servers, the back-end servers, and the mupdate server, and ending with a brief discussion of replication.

This book is not perfect, but it's much better than what we've had to date. On the whole, I agreed with most of the things the authors wrote, and where I disagreed with them it was more a matter of feeling that they didn't go far enough on a given topic, as opposed to being flat-out wrong. At least the authors introduce the reader to a variety of topics in the field that you just don't find in any other book having to do with mail system administration, and they get the reader started down a good path.

However, this is not a book that does hand-holding for novices. It will be useful to experienced mail system administrators, but if you're not already in this business and you don't already know something of the subject, then you're going to have a steeper learning curve.

At the end of the day, my benchmark is whether or not I would buy the book for myself, if I wasn't given a free copy to review. My answer to that question is most definitely "Yes!"

I've had the opportunity over the years to be a principal person doing the architecture and design of two fairly large-scale email systems, one using commercial products based on Cyrus for a large national ISP and one using purely open-source software and built around Courier-IMAP for a medium-to-large multinational corporation. However, I feel that this book has definitely helped me achieve a better and deeper understanding of these packages.

I will be taking my heavily marked up copy and sharing it with my colleagues with whom I help administer the primary campus mail system for ~50,000 students and ~20,000 faculty and staff here at the University of Texas at Austin. I'm sure this will have an impact on our day-to-day administration of our existing Courier-IMAP based mail system, as well as influencing our future choices for whatever our next-generation campus-wide IMAP server will look like.

And yes, I'm also going to contact the authors and see whether they're interested in getting some help for the second edition.

GOOGLE HACKING FOR PENETRATION TESTERS, VOLUME 2

Johnny Long

Syngress, 2007. 448 pp.
ISBN 978-1-59749-176-1

REVIEWED BY SAM F. STOVER

I don't know what it is about Johnny Long's books, but I just love 'em, and this one is no exception. Good content, good style, and good humor: what more could I ask for? Since I hadn't read Volume 1 (released in 2006), I wasn't sure what to expect, but I was definitely pleased with the end product. Also, I don't want to detract from the other authors; it's apparent that this was a group effort and it was well done.

The first chapter starts off with Google basics, followed by Advanced Operators in Chapter 2. Much of these two chapters could be familiar to the tech-savvy, since we all use Google on a daily basis anyway, right? Chapter 3 is where things start to get interesting, and it just goes on from there. Not that each chapter is that much better than the previous, but each is cool in its own way. Chapter 3 starts diving into basic Google-hacking methods, with some solid guesses on what is actually happening on the back end. Chapter 4 describes how to conduct searches to find data inside of different types of documents, such as databases, log files, and config files.

Chapter 5 was probably my favorite chapter. As a whole it addresses data mining using Google search terms, but it starts with email, phone number, and domain searches—exactly the kind of thing you'd want to be able to do if you find yourself pen-testing a company and need to track down valuable information about a particular individual during said engagement. The obvious next step to this is automation, and although Google does frown slightly on some types of automation, Chapter 5 walks you around the edges and lets you put your computer to work without angering the gods. You'll see a couple of different scrapers and also be introduced to Evolution (now called Maltego), an open source "intelligence"-gathering application.

Chapters 6 and 7 deal with googling for exploits and "simple security" searches, Chapter 8 shows some techniques for finding different kinds of Web servers, portals, and also networking devices, and Chapter 9 focuses on searching for usernames and passwords. Chapter 10 goes back to the automation drawing board and discusses the AJAX Search API provided by Google. The book rounds out with about 50 pages of "Google Hacking Showcase" items in Chapter 11 and ways to protect your assets from Google hackers in Chapter 12.

All in all, this is a very solid book. There were a few more grammatical, spelling, and editorial errors than I'd like to see, but the content was good enough to distract me from the errors and omissions. I've been spending a fair amount of time lately doing engagements in which pen-testing skills and tools are needed, and I think this book, even as big as it is, will be a permanent part of the repertoire.

CRIMEWARE

Markus Jakobsson and Zulfikar Ramzan

Addison-Wesley Professional, 608 pp.
ISBN 978-0-321-50195-0

REVIEWED BY SAM F. STOVER

My biggest gripe with this book, and it's a big one, is the word "crimeware." I just can't buy into that term, and that makes this a hard book to read at times. I'm not saying that it isn't a valid or descriptive term, because it's both, but it just doesn't read or sound right. That said, I think a lot of the content of the book is really spot on, which is why I resolved to get over my anti-crimeware attitude, and I encourage you to do the same if you find that you're turned off by the title.

Chapter 1 provides an introduction to the term "crimeware" and goes into a fair bit of detail on how different types of malicious software can be grouped into what the authors have deemed crimeware, namely, malware designed for the express intent of committing criminal acts. This chapter touches on a lot of different topics and serves to show how bots, trojans, rootkits, transaction generators, proxy attacks, and dns cache poisoning can all be labeled as crimeware. There's a fair bit of technical detail present in this chapter, with indications of more in the following chapters.

The remaining chapters don't really follow any recognizable pattern or hierarchy. They were all written by different authors and could stand on their own. (In fact, I believe some of the chapters were either papers or featured articles in other publications.) I'm going to focus on the chapters that really appealed to me, but on the off-chance that what appealed to me might not be what you are looking for, I definitely recommend checking this book out and seeing whether the topics that interest you are addressed.

Even though I'm not a software engineer by any stretch, I really enjoyed Chapter 2 by Gary McGraw, which deals with "A Taxonomy of Coding Errors." He has built a whole nomenclature around coding errors and how they contribute to malware infection and propagation. Good stuff. Another chapter I really liked was Chapter 9 on "Virtual Worlds and Fraud." I see this as a ripe market for malicious entrepreneurs, and despite the brevity of this chapter, it was pretty engaging. Another good read was Chapter 11, "Online Advertising Fraud," which dove into several different mechanisms developed by bad guys to make money from folks like Google (and, incidentally, the Google Ad Traffic Quality Team co-wrote this chapter). The last chapter that really grabbed my attention was "Crimeware Business Models," which discusses how the bad guys are making money from all this criminally focused malware.

I think this book takes a reasonably good look at a very diverse and complex topic. There were definitely times when I wished there was more detail, but I guess that's saying that the topics were interesting enough that I wanted more. As with any book that collects from a large author pool, there was a little bit of overlap between certain chapters, but nothing that I couldn't overlook. The topics are interesting, the spelling and grammar are top-notch, and you can easily bounce around the book as your fancy takes you. It gets high marks from me, and I'd definitely be interested if a second edition were to come out.

RUNNING XEN: A HANDS-ON GUIDE TO THE ART OF VIRTUALIZATION

Jeanna N. Matthews, Eli M. Dow, Todd Deshane, Wenjin Hu, Jeremy Bongio, Patrick F. Wilbur, and Brendan Johnson

Prentice Hall, 2008. 586 pages.
ISBN 978-0132349666

REVIEWED BY RIK FARROW

If you read the article in this issue about virtualization in Solaris, you will have a good feeling for the depth of information found in this book, written by some of the same authors. When I first encountered Xen, I installed the right kernel, ran preconfigured guest images, and things just worked. But as soon as I needed to go beyond the basics, I discovered that running Xen is a complex topic. And that is where this book comes in.

The authors start with chapters on the basics of virtualization and the use of a Xen LiveCD, and they quickly move on to configuration options for xend, the interface to the hypervisor that runs in Domain 0. The book continues to dive deeply into setting up and running Xen, from prebuilt guest images to setting up devices that can be accessed only by a guest domain. I particularly appreciated the chapter on Xen networking, as the authors do a good job at explaining the differences among bridging, routing, and NAT-based networking, as well as about having completely virtualized networking among guests.

There is a lot of information in this book that is hard to find online, and it is also clear that the Clarkson University team that wrote this book is intimately familiar with Xen. I found much to like about this book. As an editor, I also have some problems with this book, as there are numerous little editing mistakes—e.g., sentences repeated twice, things that are poorly explained, missing explanations such as how to use a preconfigured guest saved with an .xva file suffix) and other rough edges. It is as if the authors are so familiar with Xen that they sometimes fail to communicate with others who haven't been breathing and living Xen for the past several years. Still, I can recommend this book as a useful resource to anyone tasked with managing Xen systems.

THE HEAD TRIP: ADVENTURES ON THE WHEEL OF CONSCIOUSNESS

Jeff Warren

Random House, 2007. 390 pages.
ISBN 978-1-4000-6484-7

REVIEWED BY RIK FARROW

When I learned that Elizabeth was reviewing *Your Brain: The Missing Manual*, I immediately decided that we needed to contrast that book with another, less pretentiously titled book. Whereas MacDonald's book has no references, Warren's book has 30 pages of them, all neatly listed right before the index. And although Warren's book is also about the brain, its focus is completely different.

Jeff Warren is a freelance producer for the Canadian Broadcasting Company, as well as a freelance science writer. This combination of avocations leads to a delightfully yet rigorously written romp through what he terms "The Wheel of Consciousness." Warren starts with sleep, offering himself as an experimental subject to sleep researchers. He isn't just doing this for the purpose of writing this book, but because he is genuinely curious about his own self. Besides learning the difference between hypnogogic and hypnopompic dreaming (one is quite hallucinogenic in quality, whereas the other is familiar to anyone who uses a snooze alarm), I also learned about The Watch, a reverie-like state that only occurs when you routinely get a chance to be in bed over eight hours.

Warren also examines waking states, wanting to enhance his own ability to be in "the zone" as well as to achieve better focus through the use of biofeedback training. And then there is the lucid dreaming workshop, along with the use of a special device (a topic I don't want to spoil, as it makes for good reading).

Much less of a manual, and with much greater depth, *The Head Trip* teaches you a lot about yourself while never failing to entertain.