book reviews

ELIZABETH ZWICKY, SAM STOVER, AND RIK FARROW

SECURITY AND USABILITY: DESIGNING SECURE SYSTEMS THAT PEOPLE CAN USE

Lorrie Faith Cranor and Simson Garfinkel, eds.

O'Reilly, 2005. 692 pages.
ISBN 0-596-00827-9

This is a collection of individual papers about security and usability. The collective message is "Security is hard. Usability is also hard. They are not actually in complete opposition, but combining them is really hard." Security people often get away with waving their hands and saying dismissively that making things more secure inherently makes them less usable, so there's no point expecting systems to be usable. This volume is devoted to the proposition that this is a cop-out. Some security interferes with some usability, but most secure systems have usability problems for the same reason that most usable systems have security problems; someone decided that the relevant property could be painted on at the end, but it's actually a design property. You have to actually know what you're doing, think deeply about it, and build it in.

This volume covers a wide range of topics from a wide range of perspectives. I found it pretty uneven and yawned my way through several articles, but on the whole it was enlightening. It includes some great tragicomic moments, such as Simson Garfinkel's paper on used disks, and the infamous "Why Johnny Can't Encrypt," in which 12 people tried to use PGP and only a third of them managed to sign and encrypt a message correctly within 90 minutes and three accidentally exposed the secret. There are also some great case studies where people actually ended up able to use the stuff, which tend to drive home the "usability is also a design property" point, and some good studies on passwords, which drive home the points that actual data can be useful and that text passwords remain popular for really good reasons.

This is a great book to consider before you design your next project that includes authentication and handy to have around to pull out figures to support arguments like "No, I don't think biometrics will solve that for you" and "Yes, I think we ought to run a nice, low-level format on that disk before we let it out of our hands." This is not a book to read from start to finish, at least for me; it starts slowly and doesn't really pick up steam until section three.

PROTECT YOUR WINDOWS NETWORK: FROM PERIMETER TO DATA

Jasper M. Johansson and Steve Riley

Addison-Wesley, 2005. 549 pages.
ISBN 0-321-33643-7

This is a sensible, well-written guide to security from a Windows perspective. Of course, mostly what I mean by "sensible" is that the authors agree with me on almost all the subjects where I already knew my opinion, but I also mean that they have a balanced, rational tone and talk about broad issues instead of exclusively about details. They spend a lot of time encouraging the reader to think about security overall, instead of securing one thing and focusing on the demon of the day.

Although the book comes from a Windows perspective and occasionally gets into Windows specifics, it covers a lot more than just Windows issues. It talks about policies, about users, about physical security. It's a nice guide to the universe of security, even if you're not interested in Windows. If you are interested in Windows, it gives you a lot of important information not available elsewhere and helps sort out the genuinely important issues from the frantic hand-waving and the strange registry-setting obsessions.

This is by far the best, most readable Windows security book I've come across. Admittedly, my experience with the genre is not exhaustive, but it's large enough to have been exhausting; I've certainly hefted a bunch of them, opened them up, groaned in misery, and put them down again. This is not one of those; it's a fine addition to any security library that happens to be about Windows.

WINDOWS SERVER 2003 SECURITY COOKBOOK

Mike Danseglio and Robbie Allen

O'Reilly, 2005. 479 pages.
ISBN 0-596-00753-1

Once you've read *Protect Your Windows Network* so that you understand what you want to do, this will help you figure out how to do it. Don't, under any circumstances, do it in the other order. This is a cookbook; *Protect Your Windows Network* is a menu planner and nutrition handbook. If you start with the cookbook and eat nothing but cookies and steak, it's not the cookbook's

fault, but you're not going to feel good.

The *Cookbook* does attempt to give you some background as to why you might want to do things and what might go wrong if you do them, but it's only really enough to help you if you have a firm background in the underlying issues. Once you have that background, it looks useful, providing command-line and scriptable solutions wherever possible. I'm definitely keeping it around for those moments when I know I need to beat something into submission but don't know how. I will be using caution, however; I note that it doesn't mention the oddities of the "cacls" utility, which doesn't propagate permissions. In Windows, you have to tell files to inherit permissions—cacls doesn't, so changing directory permissions won't change permissions on files that are supposed to inherit the directory permissions, until some future time when something else propagates the inheritance. This is a nasty trap, and anything that suggests you use cacls really ought to mention the issue.

HOME NETWORK SECURITY SIMPLIFIED

Jim Doherty and Neil Anderson

Cisco Press, 2006. 199 pages.
ISBN 1-58720-163-1

This is a book designed for the security-naive but not technology-phobic Windows user—the kind of thing you'd hand to your more self-sufficient relatives to get them to deal with their own security. To fit network security into 199 pages, with lots of color pictures and white space, it simplifies pretty ruthlessly. For instance, it doesn't even mention the existence of non-Windows operating systems. If you were hoping it might apply to your relatives who are bothering you about the security of their Macintoshes, your hopes will be dashed.

That said, I think it does a pretty good job of covering the issues for its audience. They're more tolerant of WEP than I would be. (They do encourage the use of WPA if it's available, but they're willing to accept WEP.) I also have a nontechnical issue with their snooping advice; snooping is just as icky as a parenting technique as it is within a marriage. In either situation, you might consider logs and traces as an agreed-on way to maintain accountability instead of as a secret, an idea they don't mention.

On balance, however, the authors cover the important stuff in a friendly, accessible way, and they manage to be realistic about the dangers of the Internet. I might hand mine off to one of my more distant relatives (the close ones all run operating systems it doesn't believe in).

DICTIONARY OF INFORMATION SECURITY

Robert Slade

Syngress, 2006. 222 pages.
ISBN 1-59749-115-2

This is billed as an essential reference tool; I'm not sure about that. It's not that it's a bad dictionary of information security. It's a perfectly good one, with definitions that are precise enough to be useful without being incomprehensibly technical. The jokes are small enough and rare enough to work as leavening, and there's a nice appendix with a list of other useful dictionaries. I am also somewhat awed that Slade has managed to respond to the one complaint about security books that I didn't expect to see ever handled: Appendix B has a plot and some character development.

Nonetheless, I've never thought "Gee, what I really need here is a dictionary of information security," and I don't expect I will anytime soon. I might possibly look something up in it to answer a question such as "Is that new, or did I just miss it somehow?" but most of my questions about information security terms are more likely to be answered by a search engine (for brand-new terms) or the Jargon File (for the history of terms). This book would be most useful for somebody just entering computer security, but if you're that person, and you're reading things you can't understand without the help of the dictionary, you're in over your head and need some deeper background.

SOCKETS, SHELLCODE, PORTING, & CODING

James C. Foster

Syngress, 2005. 667 pages.
ISBN 1597490059

REVIEWED BY SAM STOVER

This is the final entry in a list of books that I've reviewed by this author. This book, as with the previous ones I've reviewed, comes with an upside and a downside. The upside is that there are a couple of chapters that have really excellent material. The downside is that there are only a couple, and the remaining chapters have been cut-and-pasted from other books.

Let's focus on the good first. Chapters 3, 4, and 5 explain BSD, Windows, and Java Sockets, respectively. There's lots of good material here, and although I'm not a fan of Java, it does provide at least a great foundation for both UNIX (BSD) and Windows sockets.

Chapter 6 is a good introduction to writing portable code, which is the name of the chapter, incidentally. From that, Chapter 7 launches into network-specific

portable coding. I found these two chapters to be the real gems in this book. My day-to-day life does not include any C programming, so working through these chapters was very fun and informative.

Chapter 13, "Writing Security Components," focuses on introducing the reader to the Component Object Model (COM) and implementing it with the Active Template Library (ATL). Once that foundation is laid, the final chapter, "Creating a Web Security Tool," gives a very fun glimpse into the intricacies the authors encountered when writing their own Web scanner. Very good stuff.

Now here's the bad: All of the other chapters exist in other books, namely *Writing Security Tools and Exploits, Buffer Overflow Attacks,* and *The Pen-Testers Open Source Toolkit.* There are 14 chapters in this book, and only half of them contain original material. If you own any of the other three books, be warned. The chapters are all in various stages of cut-and-pasting. It's hard to tell which chapters were lifted from which books, but the fact remains that there is a highly incestuous relationship among the four books.

As I've said before, each book on its own contains a wealth of information. The problem arises when you buy one of the other books in the hopes of moving deeper into the subject, only to find that it's the same material, sometimes verbatim. Regardless of the right or wrong of it, I feel a duty to let people know so that their expectations are managed. It's one thing to buy a cut-and-paste book knowing that the difference among the books is what you want.

To make a long story short, this book has some incredibly valuable information on C coding, but readers and buyers should be aware that 50% of the book could very well already exist on their bookshelf.

DESIGNING EMBEDDED HARDWARE

John Catsoulis

O'Reilly, 2005. 377 pages.
ISBN: 0596007558

REVIEWED BY RIK FARROW

Embedded systems has been one of my interests for years, so when this book appeared, I wanted to read it. And although it lingered on my "to be read" stack for a while, once I got into it I found that Catsoulis writes about a difficult topic clearly, and he held my interest.

Embedded systems are everywhere, more common than desktop computers by far. I had worked with embedded systems, as well as early PC components that used co-processors, back in the early 1980s, and had assumed that things had simply gotten too complex to understand. Instead, Catsoulis explains how embedded system designs have gotten less complex, as chip designers worked to create hardware that is easier to integrate. If you think about it for a moment, it makes perfect sense that ease of use ranks right up there with capabilities, making this a natural evolution.

Catsoulis starts off gently, with chapters as basic as architecture, some assembler and Forth, and Electronics 101; there are even soldering tips. He then moves into the use of specific interconnects, the buses of embedded systems. These chapters initially caught my interest, as they explained concepts I had heard about: the Canbus used in my Prius, or I2C used with real-time clocks in PCs, and even a good explanation of USB. To me, these chapters alone were worth the price of the book. Then Catsoulis describes how these buses are used to tie together sensors, relays, and various microprocessors. Catsoulis's experience teaching college-level classes in embedded system design shows as he points out commonly made design errors, such as forgetting to use draw-down resistors.

If you have ever considered building that network-connected toaster or Web-based wine-cellar temperature sensor, this is the book for you. Even if you won't be designing your own circuit boards, you will certainly understand what is involved in any kit or prebuilt design you may decide to use.

STEALING THE NETWORK: HOW TO OWN A CONTINENT

131ah, Russ Rogers, Jay Beale, Joe Grand, Fyodor, FC, Paul Craig, Timothy Mullen, and Tom Parker

Syngress, 2004. 432 pages.
ISBN 1-931836-05-1

This is another book that sat around gathering dust; I started reading it in the middle, then went on to read the whole thing. Books in this series (the titles of which start with "Stealing") are fictional accounts of hacks and hacking. The chapters vary in quality, but I found I enjoyed reading most of this book, perhaps in part because I know many of the authors and could recognize their hacking styles in their chapters. *Stealing the Network* makes for good idle-time reading.