book reviews

ELIZABETH ZWICKY, WITH
SAM F. STOVER, BRANDON CHING,
AND RIK FARROW

SLIDE:OLOGY: THE ART AND SCIENCE OF CREATING GREAT PRESENTATIONS

Nancy Duarte

O'Reilly, 2008. 263 pages.
ISBN 978-0-596-52234-6

Duarte's *slide:ology* is the sort of book that you are either going to love or going to hate. It has sensible and often beautiful advice on how to make compelling slide presentations. Done well, these will leave a technical crowd rolling on the floor and crying out for more. Nonetheless, they tend to induce a certain suspicion. And the title *slide:ology* (what is that colon doing there?) doesn't help any. My scientifically selected sampling of typical technical people (that is, the people who wandered through my living room while it was on the table) divided into two camps, which can be characterized as "intrigued" and "deeply hostile."

In fact, one of them said something roughly like, "Good heavens! Why would you need an entire book? Just use fewer slides with bigger type and be done with it." And the author is not in fact unsympathetic to this point of view. She does think there are a few more things about graphic design you might want to know, but I'm sure she'd be happy if she could just get people to stop giving slide presentations that consist mostly of text they're going to say anyway, and they're going to have to say, because nobody can read it on the slide.

If you give presentations and you want them to be more gripping, this book will probably help. Yes, it has spiffy pictures of CEOs in it. Try to get past them, because it also has a bunch of practical advice, and some of it you will certainly sympathize with. As with many other things, the advice is mostly simple, which unfortunately is the exact opposite of easy. You will understand it, but implementing it is much harder.

APPLIED SECURITY VISUALIZATION

Raffael Marty

Addison Wesley, 2008. 506 pages.
ISBN 978-0-321-51010-5

Wait, didn't I just review a book on security visualization? That was *Security Data Visualization*, which I was enthusiastic about a little over six months ago. I wouldn't have thought it was that rich an area, but apparently it is, since the two books don't overlap all that much. *Applied Security Visualization* is a less whiz-bang book, being not so much about the exciting hackers and the novel visualization but more about the SOX and HIPPA compliance and useful visualization with existing tools, all of which I have a lot of sympathy for, since life is in my experience mostly about the less whiz-bang stuff.

Applied Security Visualization is full of useful stuff. It's aimed at technical security people who understand basic security stuff and are comfortable with technical tools and information. If you can't program something (Excel counts) or you can't read graphs comfortably, it's not going to do you much good. And although it talks about visualizations aimed at other people, it's mostly about visualizations for the use of technical people: the pictures you use to help you audit, debug, and figure things out.

I like this book and think it will be of practical use to many security people. I do have some reservations, however. There are a bunch of tactical errors made in graphs. For instance, if you are working with people who are not immersed in your visualizations, do not ever make a graph where the lower left is the good bit and the upper right quadrant is the bad place to be. (Actually, *slide:ology* has some nice coverage of this, which may have made me extra-sensitive to this problem.) And if you are going to talk to the CEO, or even the CIO, I recommend strongly against calling something ROI if it does not involve actual money. The graph that shows that vulnerabilities went down when the risk mitigation program was put in place does not tell you anything about return on investment. It suggests that you are actually changing something with your investment, but having fewer vulnerabilities does not equal more money.

I'm also skeptical about much of the information about the insider threat. Yes, insiders commit a lot of computer crime. Yes, you could probably find some of them earlier if you spent a lot of time looking at data. And I'm sure there are some companies that find this worthwhile. But the vast majority of sites are just not in a position where worrying about any but the most trivial and practical checks is worthwhile, so 150+ pages seems like a lot of space to spend on something that's such a minority interest.

All in all, I think it's a good book of practical interest to people who do security and need help looking through data, but it does try to cover a bit more than it really can.

PRIVACY ON THE LINE

Whitfield Diffie and Susan Landau

MIT Press, 2007. 335 pages.
ISBN 978-0-262-04240-6

Privacy on the Line is a great read—and not as depressing as you might expect a book about wiretapping to be. It talks about privacy, encryption, communication, and government from an educated perspective without assuming that the reader knows anything about either cryptography or history. I recommend it to anybody who's interested in security (personal and national) and how it interacts with encryption and legislation. These are thorny topics indeed, and they are handled here with grace and perspective.

HACKING EXPOSED, LINUX THIRD EDITION

ISECOM

McGraw-Hill Osborne Media, 2008. 813 pages.
ISBN 978-0-07-226257-5

REVIEWED BY SAM F. STOVER

I heard some rumors about this book before reading it, and I found that it's fairly controversial. This is not because of the content itself, but because the content is different from that of the previous *Hacking Exposed* books. In fact, that's what drew me to the book: Linux + controversial = yummy. Luckily, I was right: it was yummy. This is one great book. Unfortunately, it's somewhat constrained by the *Hacking Exposed* method of delivering information, which could make it a little tough to swallow for some folks. But rather than seeing the glass as half empty, I see a bigger glass with more stuff in it.

As with any other *Hacking Exposed* book, the primary complaint people have is "It doesn't teach me how to hack!" as if "hacking" is some kind of autonomous activity that you put under your belt as soon as you know how. What this book does do, extremely well I might add, is introduce the Open Source Security Testing Methodology Manual (OSSTMM). The OSSTMM, among other things, teaches you how to follow a penetration testing process from start to finish. So, in all fairness, this rendition of the *Hacking Exposed Linux* book probably does teach you more about "hacking" than other books, as long as you agree with ISECOM on what "hacking" really is.

Enough on what it is; let's talk about what is in it. One of the aspects I love about this book is the miniature case studies that preface every chapter. I think that is a great way to get the readers' attention, as well as give them a fun way to see what the chapter is going to be about before diving in. The first three chapters are all about describing security and controls. Anyone not familiar with OSSTMM definitely shouldn't skip over these chapters, because ISECOM takes a unique approach to risk and threat that you need to understand to get the most out of this book. The next nine chapters belong to the section titled "Hacking the System." This is where the OSSTMM methodology is presented for nine different technologies, ranging from PSTN (Public Switched Telephone Network), to VoIP, to 802.11, to RFID and beyond. I have to say that a lot of different technologies as well as lot of pentesting tools are covered in 320-some-odd pages; this is definitely the bulk of the book. Personally, I found the PSTN section especially intriguing, as I don't have much experience there. Surely in these nine chapters there will be something of interest for just about anyone.

Chapters 13–15 deal with "Hacking the User," which takes a slightly different angle. Each of these chapters deals with different ways to manipulate Web applications, mail services, and name services, in that order. There is plenty of good info for the budding "hacker" in these chapters, with details of different ways bad guys exploit weaknesses, as well as ways to counteract such malicious behavior. Some of the information presented here is pretty basic and some more advanced. Again, there is something for everyone.

The book ends with two chapters on "Care and Maintenance," which deal with source code analysis and Linux kernel tweaks. The first of three appendices lists "best practices" tips, the second presents some basic Linux forensics, and the final appendix talks about the BSD projects.

Overall, the book was well written, with only a few grammatical and spelling errors. The content is consistent with the high-quality output of the ISECOM crew. My only reservation was that I felt the subject matter transcends the *Hacking Exposed* format. However, instead of complaining, I feel that I got more than I bargained for. The OSSTMM isn't just about Linux; it's about security. You'll definitely learn about Linux hacking if you get this book, but you'll also get much more, and that's a good thing.

RAILS FOR PHP DEVELOPERS

Derek DeVires and Mike Naberezny

The Pragmatic Bookshelf, 2008. 406 pages.
ISBN 978-1-934356-04-3

REVIEWED BY BRANDON CHING

As a long-time PHP developer, I never quite found my interest piqued by the advent of Ruby on Rails as a mainstream Web development platform, and the majority of developers I have worked with over the years seemed to agree with my lack of interest. The running joke is that if we simply used Ruby instead of PHP, there would most certainly be a `buildEntireProject()` method that would do all of our work for us. However, times change, and as developers it is our responsibility to explore new and different methods of getting work done, no matter how fruitless our initial expectations are.

Rails for PHP Developers by Derek DeVries and Mike Naberezny was my first serious attempt at practicing another language, aside from PHP, for Web application development. As much as I hate to admit it, I think I like it! The book is broken up into three core sections designed to lead you through a comparative analysis of PHP and the Rails framework, followed by the construction of an entire Rails application.

Section I begins with a brief introduction to the Ruby language and outlines some of the basic differences between PHP and Ruby. Although far from an exhaustive introductory reference on the Ruby language, these first few chapters utilize your existing PHP knowledge and comparatively show you how to get things done in Ruby. For instance, in Section 2.6, outlining method creation and parameter passing, the authors show how to create a method in PHP, and they follow it by showing the same code in Ruby. The authors proceed to explain the Ruby-specific how and why, which gives good context and surprisingly helpful insight given the relatively short length of each section.

The first section is also where you will be introduced to the Rails framework and build your first basic Rails application. By the end of Chapter 3, you will have covered a good majority of Ruby's object-oriented features, including attributes, namespaces, typing, and overriding. Again, each topic is placed within a comparative code context, with both PHP and Ruby examples.

Section II is where you really get into the heart of the Rails framework. Under the pretext of building a meeting management application, the authors guide you through the major concepts of the Rails framework, including database modeling, controllers and views, validation, user management, associations, and deployment. This section of the book is quite extensive in both its descriptions and its code samples. As you progress through building the messaging application, you are exposed to everything from form creation and validation to caching and even some production server recommendations and configuration help.

In the book's final section, the authors present three reference chapters devoted to relating PHP to Ruby syntactically. Akin to a foreign language dictionary, these chapters bring back the comparative code examples seen in the first section but now laid out reference-style. Each topic contains both code comparisons and brief details of Ruby specifics. This section seems incredibly handy to have, as it covers everything from strings and array manipulation, to object cloning, to header redirection and so much more.

Overall, I was very impressed with *Rails for PHP Developers*. The wording was down to earth, the flow of the book was coherent, and the content was relative and informative. Each of the main chapters has a good summary plus a number of practical exercises to reinforce your learning of the material. Although not a replacement for a strict Ruby language instructional or reference book, it certainly lives up to its title and capitalizes upon the existing development knowledge of its intended audience (which, by the way, should be an intermediate- to advanced-level PHP developer). If you are a current PHP developer serious about learning Ruby on Rails, then I would certainly recommend this book.

So, am I a Ruby convert? Well, maybe not just yet. However, *Rails for PHP Developers* has certainly provided me with the guidance and piqued my interest in Ruby on Rails, and I can promise that I will at least be dabbling in some Ruby in the near future.

ANATHEM

Neal Stephenson

William Morrow, 2008. 960 pages.
ISBN 978-0061474095

REVIEWED BY RIK FARROW

My tech reading this past couple of months has been either online or in books too old to be reviewed fairly. I did take time out to read Neal Stephenson's new tome, *Anathem*, and I thoroughly enjoyed it.

Stephenson, of *Cryptonomicon* and *The Baroque Cycle* fame, has created a richly thought-out world that parallels our own in many ways, while being more advanced in others. The people of Arbe have forced their scientists, mathematicians, and philosophers to live in cloisters, called concents, partially because of a past disaster known only as the Terrible Events, but just as much because of irrational fears that their research will create new worldwide disasters. This system has prevailed for thousands of years, with people living in concents watching the rise and fall of civilizations on the outside several times over. At the same time, the researchers are limited to pure research, with pen on leaf, by both their internal watchers (the Inquisition) and the external world which has invaded and sacked the concents three times.

Stephenson has invented his own vocabulary for key elements of this world, and these terms take time to get accustomed to. I avoided some of this adjustment by reading the Dictionary [1] first. As I read, I could appreciate just why Stephenson wants to force us out of our familiar track and into seeing the world differently.

The narrator of the story, a 19-year-old man "collected" 10 years ago because of his intelligence, provides a thoughtful view of the tensions between the world of the scientists versus the world outside the concent's walls. These tensions are heightened by the discovery of an unusual object orbiting Arbe. As in the past, the Powers that Be, rulers of the outside world, overcome their fear to enlist the scientists in untangling a possibly world-threatening event.

Stephenson's depictions of his key characters had me laughing out loud, as he has created people recognizable to any geek. His descriptions of interactions with the politicians of Arbe and the researchers and scientists clearly parallel those of our own world.

I was left wishing the book had been longer than its already immense length. I can heartily recommend this book to anyone smart, with both a sense of humor and a willingness to explore different ways of being and thinking and a desire to recognize bulshytt (see The Dictionary) when it is encountered.

REFERENCE

[1] The Dictionary, 4th Edition, A.R. 3000: http://www.nealstephenson.com/anathem/dict.htm.