Book Reviews

ELIZABETH ZWICKY, WITH SAM STOVER, EVAN TERAN, AND RIK FARROW

The No Asshole Rule: Building a Civilized Workplace and Surviving One That Isn't

Robert I. Sutton, PhD

Hachette Book Group, 2010. 225 pp.
978-0-446-69820-7

This is a new edition with an extra chapter about the benefits and drawbacks of becoming famous for writing a book with "asshole" in the title. Probably not worth buying a second copy for, but a worthy extra dollop of amusement.

Eyebrow-raising title aside, this is a sweet and useful book. "Sweet" because not only does it argue that letting people be abusive and stupid is bad business, as well as morally wrong, but it does so self-deprecatingly and with great appreciation for the difficult, socially awkward, and technically demanding who may be swept up by people who are aiming for niceness. "Useful" because it gives advice not only on how to set and enforce a "No asshole" rule, but also on how to avoid being one yourself, and how to survive when a plague of assholery sweeps through your work place. (The author argues convincingly that one reason to avoid nasty people is that it's catching.)

Although I fully support the author in distinguishing pure geekery from nastiness, I think he probably gives high technology more of a pass than he should. The most dangerous assholes are the socially competent ones, but hiding in the flock of perfectly nice geeks who don't do social interaction well are some who actually don't care about people.

Surprisingly, this is an uplifting book; if the turkeys are getting you down, it's a good choice to seek out.

Guesstimation: Solving the World's Problems on the Back of a Cocktail Napkin

Lawrence Weinstein and John A. Adam

Princeton University Press, 2010. 293 pp.
978-0-691-12949-5

I expected to love this and didn't, although not through any particular fault on its part. It's an interesting look at how one answers questions like "How many ping-pong balls does it take to go around the world?" and I like a number of the math bits. But I cannot bring myself to care about how many ping-pong balls it might take, even though I know that this is not the point; thinking about things like the size of the world and how big numbers fit together is the point, and I really like those things. But couch them in terms of guesses about ping-pong balls (or piano tuners, or Spider-Man) and my eyes glaze right over. (The appendix of useful numbers I can read quite happily though. Go figure.)

If you find these questions interesting, it is a great introduction to thinking about planetary scales and big numbers, and estimating with honesty. Not to mention it's the most condensed source of information on a certain class of interview questions and how to answer them.

iOS Forensic Analysis: For iPhone, iPad, and iPod Touch

Sean Morrissey

Apress, 12/27/2010
978-1-4302-3342-8

One of the biggest drawbacks for any forensics book is how fast it becomes obsolete. I tried to review an iOS forensics book a while back, but by the time I got my hands on it the content was so outdated, it wasn't worth the time. Luckily, I got to this book right out of the gate, so it's still relevant as well as technical and complete. There are a number of spelling/grammatical errors, and the flow is disjointed at times, but if you want a book that's going to show you how to acquire data from an iDevice, this is it.

Compared to the rest of the book, the first chapter is a little fluffy, as it covers the "History of Apple Mobile Devices." Not terribly relevant, but interesting if you're into that kind of thing (I'm not). After that, though, we jump into the iOS operating and file systems. This is a decent intro to how the iOS system works, partition specifics, SQLite databases that hold all the information, plus some recommendations on tools to use for SQLite analysis. Chapter 3 drops back from the technical for a bit to discuss the legal issues associated with cell phone seizure, which is quite complicated due to legislation being quite behind the curve when applied to rapidly advancing technology. After a brief discussion of whether or not an iDevice can be seized, the actual seizure process is outlined. This sets the stage for the rest of the book, which shows what data lives where, and not just on the iDevice, but also on syncing computers and MobileMe accounts. A number of tools ranging from free to several thousand dollars are evaluated, and evaluated quite fairly, a pleasant surprise. One big takeaway from the book for me was learning about the Lantern forensic tool. Of the tools discussed, this seemed to have the best bang for the buck, and the authors were also very quick to respond to my demo request, so I actually got to play with it within about 30 minutes of discovering it exists.

As with most forensic books, this one walks you through setting up your own forensic workstation, along with a fairly complete list of tools you'll need to do iDevice analysis. A significant amount of time is spent explaining how to use the different tools, leveraging their strengths to find evidence. A by-product of reading this book is seeing firsthand just how much information really lives on a smartphone. I used my own iPhone while going through the book, and I was pretty amazed at what I found. Certain applications such as Pandora and iDisk, in particular, cache files, which I didn't realize. Pandora had thumbnails for the last 20 or so songs I had listened to, and iDisk had about 15 files locally that I had completely forgotten I had read. Overall, a solid technical book that any geek with an iPhone should get their hands on, if for no other reason than to see what's on it.

—Sam Stover

Arduino Cookbook

Michael Margolis and Nicholas Robert Weldin

O'Reilly Media, Inc. 2011, 500 pp. (rough cut e-version reviewed)

978-0-596-80247-9

Building Wireless Sensor Networks

Robert Faludi

O'Reilly Media, Inc. 2011, 301 pp. (digital version)

978-0-596-80773-3

Although they focus on different technologies, these two books are interrelated and even have a bit of actual overlap. *Arduino Cookbook* (AC) in typical O'Reilly Cookbook fashion provides almost 200 solutions to various Arduino topics. *Building Wireless Sensor Networks* (BWSN) deals with Xbee radios which use the ZigBee communications protocol. Both Arduino and Xbee are very popular and easy-to-use modules which, although independent, are often used in tandem.

The AC version I read was an O'Reilly Rough Cut, so the finished version might differ slightly from what I discuss here, but probably not enough to make a big difference. There are 18 chapters, each dealing with different types of recipes for using Arduino. Chapter 1 walks you through the Arduino hardware, and Chapters 2–4 introduce the software, some basic programming, and connecting your Arduino to your PC. After that it's off to the races as you learn how to receive sensor data into the Arduino and integrating various devices that provide sensory input (touch, sound, light, etc.). There are so many different things you can do with an Arduino, just listing all the chapters would take up more space than I'm allowed, but some of the really cool recipes include controlling motors, remote controllers, and even functioning on IP networks. In particular, Chapter 14 has four solutions for integrating with Xbee, and Chapter 15 starts out with assigning an IP address, requesting data from a Web server, running a Web server, and interacting with Pachube.

BWSN, not being a Cookbook, goes into a lot more detail, describing the history and technologies in the Xbee radios and the Zigbee protocol. That said, by the end of Chapter 2, assuming you have all the requisite hardware, you'll have two chatting Xbee radios. Unlike Arduino, which is really good at receiving data and acting on it, the Xbee radios are designed to build mesh networks quickly and easily. The Xbees do have some limited sensor capabilities, but when coupled with Arduino the possibilities open up tremendously.

In Chapter 3, Xbees and Arduinos are combined to make a simple doorbell. After getting that under your belt, you are ready to learn more about the different communication protocols available to the Xbee, followed by a solid examination of the flexible and powerful Xbee API. As with all O'Reilly books, the explanations are professional grade and easy to follow, even for someone without an EE background (guilty). Another huge plus for me was the painstakingly detailed pictures and descriptions of the circuits that are used in the various projects. For anyone who's never used a breadboard but wants to learn, this is a great guide, and you'll learn about some pretty cool mesh network protocols in the process.

I really can't recommend these two books enough. I found learning about mesh networks to be fun and easy in BWSN, but it's when you add the sensor capabilities of the Arduino that you begin to see all of the possibilities. Even if you don't have an idea in mind, once you start walking through the example I'd bet that the ideas will start flowing. Well written and with some self-deprecating geek humor, BWSN lays a great foundation for the AC to sit on. Once you know how to connect a bunch of Arduinos together, you can, take over the world, or at least your house.

—Sam Stover

A Guide to Kernel Exploitation

Enrico Perla and Massimiliano Oldani

Syngress, 2010. 442 pp.
978-1597494861

This book is not for the faint of heart. Usually topics like the kernel are reserved for the most hardcore of computer users, and for good reason: the kernel is a vast and complicated piece of software which can take a lot of time and effort to truly understand. *A Guide to Kernel Exploitation* is no exception. This book is an excellent read but is probably not going to be of much interest to the average computer geek.

The book begins by doing a great job of explaining the basics. First, it covers what a kernel is and then, more importantly, the differences between "kernel-land" and "user-land" (the common terms for areas of memory reserved for kernel and user code use). Most notable is the fact that user-space programs rely on the kernel to implement protection schemes, utilizing both software and hardware, to prevent misbehaving software from doing harm, while the kernel can only rely on itself. This means that certain techniques work better in kernel space, and others no longer work at all. I think that this intro is necessary to lay the groundwork for the later chapters, but those who will benefit the most from this book will likely consider it nothing more than a review.

In Chapter 2 we finally get into the good stuff. This chapter is an overview of the different types of bugs that can be used to achieve successful exploitation. The usual suspects, such as memory corruption and integer overflow/underflow, are here, plus there are a few that while present in user-land, are typically not exploitable, such as NULL pointer usages.

Chapter 3 focuses on the typical architecture of x86 and x86-64 kernels and how the little details affect the success of typical attacks. This information is absolutely critical to an attacker for creating a working exploit. The book discusses some basic but interesting techniques such as placing the shellcode in a buffer allocated in a user-space program instead of directly in the kernel memory. For a local privilege escalation exploit, this may simplify things a bit, since the program will be able to explicitly shut off many of the memory protections kernel pages may have: for example, the NX (no-execute) bit. It also discusses some concepts which are unique to kernel-space exploits, such as the possibilities of overwriting interrupt table pointers and how to properly leverage a successful overwrite.

Having established the basics, the book spends the next few chapters going into how to apply these core ideas to several of the more popular operating systems such as "the UNIX family," OS X, and Windows. These chapters do a good job of demonstrating the design principles that the various OSes utilize and the different structures and defense measures that an attacker needs to be familiar with when targeting a system.

All in all, this was a good read. The beginning covered a bit of ground that, to be honest, the reader should already know. On the other hand, some of the later chapters delved really deeply into the fine details, almost to the point of being a little overwhelming. But the book is well balanced and often uses good code samples to help demonstrate the concepts being discussed. For anyone who is typically a user-land-only exploit writer or perhaps even an existing kernel hacker who wants to expand their knowledge to new operating systems, this is likely a worthwhile read.

—Evan Teran

Sleights of Mind: What the Neuroscience of Magic Reveals about Our Everyday Deceptions

Stephen Macknik and Susana Martinez-Conde,
with Sandra Blakeslee

Henry Holt and Co.: 2010. 297 pp.
978-0805092813

I bought this book to read on my own over the holidays and can say that I thoroughly enjoyed it. That said, you might be wondering why I consider including a review of it in a CS publication. In truth, computers are never mentioned (as far as I can recall, and that's important). But a lot of what the authors have to say has a real bearing on everyday life, and more importantly, on the designers of computer interfaces.

Macknik and Martinez-Conde are neuroscientists who became interested in magicians as a way of exploring how the human brain works. As they got into their project, they learned that the magicians themselves have a working knowledge of various weaknesses in how humans process sensory information. As an example, Apollo, a magician who specializes in picking the pockets of his volunteers, pointed out that if he wants people to focus on his hand as he moves it, he must move it in an arc. If he moves it in a straight line, the watchers simply move their focus to where his hand *will be*, as it is easy for the brain to predict this.

The authors cover different forms of misdirection as well as the quirks of human vision. These topics are important to anyone who is designing an interactive Web page or the front end to an application. An interface can just as easily confuse the mark, er, user, as help the user work with it without becoming frustrated.

If you want a fun read that is also related to your work, I recommend this book. It does include both an index and references, as well as URLs for videos of many of the magic tricks that are performed. Oh, I almost forget to mention that they also explain how tricks are done, as it is this that makes it very clear how imperfect we are at perceiving reality.

-Rik Farrow

Some Follow-Up on Previous Reviews

I reviewed *Building the Perfect PC*, by Robert and Barbara Thompson, for the February 2011 issue, and said I would try building their "mainstream system." I bought all the components necessary (well, almost all the components) from Newegg for $650, and spent about four times as long as the authors did in building a similar system. My experience pretty much mirrored theirs, and I can still recommend their book, but with a minor caveat: they assumed that you have extra SATA cables lying around and didn't mention that you will need more than the pair that comes with the Intel motherboard. Fortunately, I did have a couple (as I build a system every year), so instead of waiting a couple of days for cables to appear, I just had to search my office to complete the project.

I also received the second edition of *The Myths of Innovation*. I reviewed this book when it first came out and can still recommend it highly. Scott Berkun offers words of encouragement to anyone who has banged up against a wall of complacency with existing systems, and to those who find themselves wondering why innovation is so hard. I lent this book to an out-of-work tech friend, and she reported back to me that she was inspired by reading it, as well as surprised by what she learned.

—Rik Farrow