;login:

Special Focus Issue: Security
Guest Editor: Rik Farrow

inside:

the bookworm

BOOKS REVIEWED IN THIS COLUMN

CYBERREGS
Bill Zoellick

Boston: Addison-Wesley, 2001.
Pp. 307. ISBN 0-201-72230-5.

by Peter H. Salus

Peter H. Salus is a member of the ACM, the Early English Text Society, and the Trollope Society, and is a life member of the American Oriental Society. He is Editorial Director at Matrix.net. He owns neither a dog nor a

peter@matrix.net

I'm writing this on September 17. That means that the issue of security is very different from the one I've written about for more than 15 years.

In the '50s and '60s (and early '70s), computer security meant, in general, the physical security of the multimillion-dollar mainframe. The advent of the DEC 10, the Multics project, and UNIX yielded password security questions. The ARPANET introduced other questions, carefully and wittily discussed by Bob Metcalfe in RFC 602 (December 1973), which I still recommend.

Cryptography, firewalls, etc., help us somewhat where data security is concerned, but I fear that an analogy with our homes or cars is apt. We lock our doors; in some neighborhoods, folks have bars on their windows, burglar alarm systems, "kryptonite" on their bicycles, and "The Club" on their steering wheels.

None of these will avail where a determined attack is concerned, just as moats and armor offered deterrents and shields, not absolute protection.

The news is full of statements that "the world has changed" as of Tuesday morning (Eastern Standard Time). Our sense of security and safety has changed. That's for sure.

On September 14, the FBI sent an advisory to InfraGard members, who are mandated with keeping the nation's digital infrastructure intact, warning them to upgrade their security precautions in light of recent terrorist activity. But

InfraGard comprises federal agencies, which is hardly a significant segment of the Net.

The effect of the devastating terrorist attacks of September 11 have caused businesses and the government to act in anticipation of cyber-assaults.

In a hearing on Wednesday, September 12, the U.S. Senate Governmental Affairs Committee detailed how prone critical-systems computer networks are to cyberterrorism. They concluded that security measures taken where government systems were concerned was poor.

Etc.

Crackers, viruses, DDoS attacks, idiots with backhoes, morons running trawlers in the China Sea, earthquakes, hurricanes – that's what we generally concern ourselves with. Not with terrorist activity.

I was in Copenhagen for uptime(1), the celebration of UNIX's 10^9 second, and got home after midnight on September 11. I got into my office after about six hours' sleep to find everyone clustered around the TV set.

Just terrible.

And a book, too.

Bill Zoellick has written a fine book about the old version of the universe. Perhaps, after a while it will be relevant to the new Internet world, too.

Zoellick has written a couple of other books, and I've always found him reliable. The volume at hand is a thoughtful essay on the ways that government regulation may impinge on business where the Internet and the Web are vital.

My problem is that I fear that government regulation and intrusion will increase, making Zoellick's notions inapplicable.

It's a thought-provoking read, but events may have overtaken it.

92

Vol. 26, No. 7 ;login:

book reviews

PROGRAMMING RUBY: THE PRAGMATIC PROGRAMMER'S GUIDE

DAVID THOMAS, ANDREW HUNT ET AL.

Boston: Addison-Wesley, 2001. Pp. 564.
ISBN 0-201-71089-7.

Reviewed by Raymond M. Schneider
ray@hackfoo.net

For those who do not already know, Ruby is a programming language from Japan. It has been suggested that Ruby is more popular than Python. The Pragmatic Programmers have tackled the task of providing the world with the first book documenting this language.

As any good programmer knows, you need the right tool for the right job. Well, Ruby is another excellent language to add to the toolbox. The book is broken down into five sections: Facets of Ruby, Ruby in Its Setting, Ruby Crystallized, Ruby Library Reference, and Appendices.

The Facets of Ruby section begins by quickly introducing the prospective Ruby programmer to the basics. It covers all of the usual suspects: arrays, hashes, and control structures. For those Perl regular expression folks, you'll be happy to know that Ruby's syntax for regexs is much like Perl. The remaining chapters in this section of *Programming Ruby* provide more depth on the various things Ruby has in common with object-oriented programming languages and completes the brief overview of Ruby's syntax and language characteristics.

Ruby in Its Setting shows some of what's in store for someone utilizing Ruby in various scenarios. The first chapter of this section covers Ruby's command-line syntax and flags. There are no surprises here. The following two chapters, Ruby and the Web and Ruby TK, are fun with plenty of examples to play with.

Ruby Crystallized, the third section, is an even more in-depth look at dealing with Ruby's semantics. This section has more information on the behavior of classes and objects in Ruby, looking at safe levels and tainting as well.

The Ruby Library Reference section is just that, designed for those of us who do not feel the need to read a book on programming in yet another language but just want a reference. This section of the book (starting on page 279 and going through to the Appendices) is just excellent, not only providing explanations of what things are but also including many examples of the way things should be implemented.

This book is excellent in its approach, whether you like to read through the basics with each language you learn or just want to tackle the references. It should satisfy you no matter what sort of reader you are.

Lastly, this book has actually been made available under the Open Publications License. It is on the Web, and you can download it at:
http://www.rubycentral.com/book/index.html.
Happy Ruby Hacking!

THE OPERA 5.X BOOK
J.S. LYSTER

San Francisco: No Starch, 2001.

Reviewed by Rick Leir
We have seen IE clobber Netscape in the browser market, and many companies now don't much care whether their Web applications or Web sites work with anything but IE. This sets the stage for the "big company in the Northwest" to dominate the Web as it does the desktop. I am interested in any browser which has the merits to buck the trend. Opera might have a chance.

Dominate? There are Web sites which only work in IE, forcing you to have a desktop machine with a Microsoft OS. Most people have such a machine available to them, so Web designers tend to cater to IE users, and make use of the corresponding Microsoft server software. I sense a vicious circle here.

What browser should you use? IE has a user interface (UI) that I just don't like. Netscape sadly lost its war and is now being sent in the wrong direction by AOL. Mozilla 9.1 is good for the knowledgeable Linux or *nix user. Amaya is the W3C example browser. Konqueror is a KDE/Linux project. Opera is an excellent alternative. I like its UI, speed, size, and availability for lots of OSes (MS Windows, Solaris, OS/2, Macintosh, Linux [x86, SPARC, PowerPC], EPOC, and BeOS).

Opera is a free browser with ads. Turn off the ads for a (low) purchase price. If you as a professional expect your browser to be free of cost, go soak your head.

New users: This book will be useful to anyone new to Opera, even if they are new to Web browsers in general. It is said that you can divide users of an application into categories: beginner (10%), average (80%), and advanced (10%).

This book will help a beginner quickly get into the average user category and will encourage the average user to make the step to advanced. Sysadmins who provide Opera on their systems will want to have a few copies of this book to make available to users.

Web developers: If you want a nice, standards-compliant browser, Opera is a good choice, and you will find this book useful for Opera configuration. If you have to develop for both IE and Netscape, you may find it easier to develop for Opera and then test in IE and Netscape since Opera is more compliant to standards.

This book starts with a good description of the UI: all menus, buttons, and options. There are some subtleties here, such as an option to pretend to be the IE

book reviews

browser so certain sites don't go into a "lowest common denominator mode" or reject you completely. Keyboard shortcuts are presented, and they can speed you up considerably. The book continues with a description of mail and newsgroup access that will help even people who are new to Usenet.

Next is a description of Java and plugin configuration, which is important because Opera defaults to minimal plugins so it can be lightweight.

For more advanced users, there is a very brief introduction to CSS so that Opera styles can be configured and a very brief introduction to security protocols so that cookies and certificates will make more sense. An introduction to HTML leads into using Opera in full-screen mode for presentations or kiosks.

Up to this point ,the book focuses on Opera 5 for Windows, but it follows with two chapters on the differences for other OSes.

There is an attached CD containing versions of Opera for various OSes, plugins, and sample HTML/CSS code. This will save you hours if you don't have a broadband link.

To wrap it up, there is a "Brief Contents" before a "Contents in Detail," which together make it easier to get around, and an index.

This book is typeset with a nice choice of fonts in a casual style. Look for a purple cover with a conductor's baton!