TER Volume 6, Issue 4, June 1999

Volume 6, Issue 4, June 1999

Telecommunications Electronic Reviews (TER) is a publication of the Library and Information Technology Association.

Contents:

ter issues (/lita/publications/archive/ter)

Windows NT Security Information on the Web. Part II: NTSecurity.net (http://www.ntsecurity.net)

by Marshall Breeding

In an earlier column (TER, volume 5, issue 5. http:/www.lita.org/ter/ter-5-5.html#winnt (http:/www.lita.org/ter/ter-5-5.html#winnt)) I reviewed Microsoft Corporation's Web site as it relates to security resources for Windows NT. I noted that Microsoft's site abounds with information and resources related to this topic, but suffers from the tendency of Microsoft to put a good face on the security gaps in NT. I found it somewhat troublesome that Microsoft does not acknowledge security problems on their site until a Hotfix or Service Pack is available. A thorough network administrator cannot rely only on Microsoft's Web site, but should be familiar with other resources that specialize in NT Security. In an effort to balance the limitations of Microsoft's own Web site for NT security issues, I surfed about for other sites dedicated to this topic.

In this column, I'll review a Web site called NTSecurity.Net (http://www.ntsecurity.net/ (http://www.ntsecurity.net/)), also known as the NT Shop (http://www.ntshop.net/ (http://www.ntshop.net/)). Both URLs point to the same server. This site often comes up at the top of the heap when using one of the major search engines for "NT Security." Although there are not explicit statements of authorship or ownership of this site, much of the content has been created or gathered by Mark Joseph Edwards. The footers of each page note Copyright ownership by "M.E." The site shows considerable corporate sponsorship, with a large variety of banner advertisements. In addition to the adds of products related to NT security products, there are also banner adds for "The Government of Tibet in Exile," linking to a lengthy article on human rights issues in China. This political issue gains even more attention under a section titled "Hack Attacks" through an article "Chinese Society Hacked." The article focuses much more attention on an anti-China rhetoric than on the actual security attack mentioned in the title.

One of the primary resources on NTSecurity is a directory of Security Tools. This directory contains brief descriptions for a number of important applications and utilities, though it is not at all comprehensive. Several of the categories defined have no entries. For example, there is a heading for "Backup and Archival Systems" with no products listed. This is a particularly egregious omission considering the importance of having a sound backup system in place for any network and the availability of several major commercial products in this genre. In general, the products for the site's advertisers are well represented, while coverage for others is spotty.

The site includes a resource listing known security risks in the NT environment. In this section, extensive information is given on particular applications, viruses, Trojans, and the methods and techniques that can used to attack a server or network. A typical page in this section would describe the problem, give some background on the issue, include a program that demonstrates the exploit, and describe what needs to be done to secure a system from this type of attack. Such an approach emphasizes the need to stay ahead in the cat-and-mouse security game. These demonstration programs, in the wrong hands, can be used to assail an unprotected system.

The top-level menu bar includes a category called "Risks", where one can view the documented security issues either by category or in reverse chronological order. At first take, it appears that this section is unforgivably out of date. The latest entry when listed by reverse chronological order when I reviewed the site in May 1999 was from September 1998--showing a lapse of almost nine months. It takes a more thorough investigation of the site to see that a "Recent Discoveries" section is available but not incorporated into these listings. This omission certainly detracts from the usefulness of this section. When one chooses to view the Risks sorted by category, the resulting page first lists some entries sorted in chronological order waiting to be categorized before the category lists themselves display. This section indicated it had been last updated eight months before this review. The non-categorized entries are from the older list of chronological entries, and do not include the items in the "Recent Discoveries." Not only are the last nine months of security problems not included in the security risks by category, neither are the more recent part of the archive. Most of the items that have been placed into the category lists date from 1997. Again, this section suffers from lack of recent attention.

The NTSecurity site includes a Books section, that identifies books that can be purchased from Amazon.com and Fatbrain.com, an online bookstore specializing in technical books. Of the twenty books listed, none carried 1999 imprints, 8 were from 1998, and the remainder were published in 1997 - 1994. A quick search of Amazon.com directly revealed at least three books on this topic published in 1999.

The more that I deal with Windows NT, the more I realize that there is very little free software available for this operating system--particularly related to security. Especially annoying is the lack of any free virus protection software. Don't expect to find any freeware on this site. The software available on this site only includes links to evaluation and demonstration programs of commercial products.

One article on NTSecurity describes the threat of the CIH or Chernobyl virus. Upon careful reading, this article is little more than a press release from Sophos Plc, a company that sells anti-virus software. This article has multiple links to the Sophos Web site where on can download a limited evaluation of their product.

The table of contents bar linked to a document called "IE Security FAQ" that yielded a "404 Not Found" error, an embarrassing mistake.

Other examples of dated information included announcements for conferences that have come and gone, for example the 1999 RSA Data Security Conference, January 17-21, 1999 and The Internet Security Conference, April 19-22, 1999. Both of these conferences had taken place by the time of the review, yet the information in NTSecurity described them as upcoming events. It would have been better, of course, to have seen some summary or wrap-up of information that was given at the conferences.

This site does include a number of useful articles related to NT security. Almost all the content of any real value is linked from the top-level home page, not from the deeper levels of the site. Some of the better pieces are links to articles on Microsoft's Web site, including "Kerberos authentication in Win2K domains" and "Single sign-on in Win2K domains." In a section titled Hands-on Help, the article "What Hotfixes are Most Important to Load" by Mark Edwards was particularly helpful. Also worthwhile are links to product reviews that the site's maintainers have contributed to Windows NT Magazine and InfoWorld.

I began this review with a fairly positive opinion about this site. In the past I had come across this site and found a couple of tidbits that helped me with a particular issue. But as I returned to this site to do a more comprehensive evaluative review, I was generally disappointed. My earlier visits must have been during a period when the site was more active. Now the site seems to be quite out of date with little recent material. Security issues, more than any other topic demand up-to-date information. While there are some recent articles in the "Recent Discoveries" section on the site's front page, little recent information has been incorporated into the rest of the site. I am also concerned with the slant toward the site's advertisers and the lack of information about tools and products from other vendors. NTSecurity's organization and structure seemed cumbersome. It was difficult to discern the relation of the table of contents along the left side of the page to the menu of sections along the top. The two were more contradictory than complementary. A search facility would have made the information in this site much more accessible, but this too was unfortunately lacking.

All-in-all, I found this site not to be nearly as helpful as I expected.

Marshall Breeding (breeding@library.vanderbilt.edu (mailto:breeding@library.vanderbilt.edu) or http://www.library.vanderbilt.edu/libtech/breeding/home.html (http://www.library.vanderbilt.edu/libtech/breeding/home.html)) is Library Technology Analyst at Vanderbilt University.

table of contents ter issues (/lita/publications/archive/ter)

REVIEW OF: Michael Leventhal, David Lewis, and Matthew Fuchs. Designing XML Internet Applications. New Jersey: Prentice Hall, 1998.

by Brad Eden

XML (eXtensible Markup Language) has been designated the successor to HTML (HyperText Markup Language) as the computer language of choice for structured information in the networked age. This book examines the history of XML, discusses the options the use of XML presents for computer information systems in the future, and assists readers of the book in implementing XML applications on the Internet. This book, the foreword of which is written by Charles Goldfarb, considered the inventor/developer of SGML (Standard Generalized Markup Language), presents a philosophy on the use of XML in order to convince the reader how XML technology can transform their Internet investment.

The book is not an introduction to XML, although it would be of interest to readers with varying levels of XML experience. Knowledge of XML is not assumed, but readers should not expect simple language. Progression through the chapters is relatively fast-paced, and a CD-ROM is included so that readers can have access to many of the programs discussed in the book. The CD contains document type definitions (DTD), source code, XML stylesheets, JDK (Java Development Kit), Perl, and distributions of the public domain tools such as AElfred, Jade, SP, sgrep, NXP, and SAX.

The book is divided into five parts. Part I (Internets, XML, and Tools) is comprised of three chapters. Chapter 1, Internets, provides an introduction to XML and the Internet as well as DTDs. Chapter 2, XML, Data, and Documents, explores the nitty-gritty of XML, its architecture, and its features. Chapter 3, XML and SGML Tools, looks at the design capabilities and authoring tools of both SGML and XML, and how to use both in constructing documents and intranets.

Part II (Perl and XML) is comprised of Chapters 4 through 8. Chapter 4, The Desperate Perl Hacker and Internet Applications: Overview, examines Perl architecture in relation to XML. Chapter 5, An XML Bulletin Board, assists the reader in the construction of a bulletin board in XML. Chapter 6, An XML Contact Database, does the same with a contact database. Chapter 7, Structure-Based Search, looks at the sgrep language. Chapter 8, Type Transformation, Import, and Export, examines information transferal from the bulletin board and contact database using Perl and other languages.

Part III, Chapter 9, XML/SGML E-mail, examines setting up email services using XML technology. Part IV, Chapter 10, XML and Java-Parsers and APIs, provides information on XML parsers and application programmer interfaces, including a number of sample XML parsers on the accompanying CD. Finally, Part

V, Chapter 11, Future-Agents and all that, provides both the pros and cons of implementing XML technology, specifically negotiation and language-agent architectures and negotiation problems.

This book is the first complete guide that I have found that assists the reader in building XML Internet applications that can automate and simplify virtually every form of electronic communication. The fundamentals of XML usage and design are given in easy to read step-by-step format. It is an excellent book for the systems designer who wishes to incorporate XML technology in Internet applications, but who is looking for guidance in words and programs. The accompanying CD provides the reader with Java, C++, and Perl source code to construct the XML applications discussed in the book, as well as Sun Microsystem's Java Development Kit and other XML tools. The limitations of HTML are examined with a focus on which issues XML addresses. The authors provide compelling evidence that XML allows the programmer so much more flexibility and power. Having worked somewhat with XML in the Text Encoding Initiative (TEI) metadata standard, I would recommend this book to anyone considering XML as a technology for their intranet and Internet applications.

Dr. Brad Eden (beden@nhmccd.edu (mailto:beden@nhmccd.edu)) is Coordinator of Technical Services/Automated Library Services at North Harris Montgomery Community College District in Houston, Texas.

table of contents ter issues (/lita/publications/archive/ter)

REVIEW OF: John T. Moy. OSPF: Anatomy of an Internet Routing Protocol. Reading, MA: Addison-Wesley, 1998.

by Ray Olszewski

Anyone who has connected a small, Local Area Network (LAN) to the Internet is familiar with a router--a device (actually, a highly specialized computer) that sends IP datagrams from computers on a LAN to the Internet (or some other network external to the LAN) and receives datagrams from the Internet (or other external source) for computers on the LAN. Small, one-router networks are simple to administer--all traffic between hosts on and off the LAN passes through the one gateway interface. Even slightly larger networks, divided into several segments connected by "internal" routers, are only slightly harder to configure; typically, up to a half dozen routers are easily set up and maintained by hand.

Complex networks are another story. Large networks, with many segments and possibly several paths out to the Internet, quickly get too complex to configure and update manually. The many routers on such networks need a standard method to keep track of one another, to learn when new routers are added and old ones are removed or fail. With multiple routes to the Internet, or even between segments, they need rules to decide which route to use to complete any given connection.

OSPF, an acronym for Open Shortest Path First, is a protocol that large, centrally-administered networks (called Autonomous Networks in Internet jargon) can use to share routing information among routers. It provides a standard way for each router on the network to tell the others what ranges of IP addresses it can connect to and how "costly" the connection is. "Cost" here is a technical term and represents some mix of distance, bandwidth, and typical link traffic, with the exact measure defined by the network administrator.

OSPF: Anatomy of an Internet Routing Protocol is a highly technical introduction to this protocol. The author, a member of the IETF (Internet Engineering Task Force) group that created OSPF, provides us with a detailed look at the internals of the protocol, its design strengths and weaknesses, and how it fits into the larger world of Internet routing. Along the way, he provides a good overview of how routing works on the Internet as a whole (there are portions of the Internet that are not Autonomous Systems and so require other routing protocols) and how to troubleshoot and debug OSPF and other routing protocols.

The book provides rich detail in about every area you might wish. Moy begins with an introduction to the basic structure of the Internet and the routing protocols in use prior to OSPF. He then recounts the development of OSPF historically, drawing on his own participation in the IETF group. After that comes the heart of the book: technical descriptions of the structure and operation of OSPF, including its extensions to networks that support multicast.

Finally, the book provides details on installing, configuring, and troubleshooting OSPF. Unfortunately, the actual implementation of OSPF in routers is largely vendor specific, so much of the advice here reduces to: Read your router manual. He does, however, discuss the OSPF implementation in gated (pronounced "gate dee" and short for "gateway daemon"), freely-available routing software for Unix computers, and he describes how to set up and monitor OSPF using the misleadingly-named Simple Network Management Protocol (SNMP). He closes with a comparative review of all the major routing protocols used on the Internet.

This technical book is intended for a limited audience. It is of most interest to anyone involved in developing Internet protocols (both routing and others) and to system and network administrators who want to upgrade their technical skills. For the everyday user of the Internet and even most sysadmins of small networks, the topics covered here are mostly the components of the Internet that are hidden from view by the expertise of Internet Service Providers (ISPs) who handle all the complexity of routing for small domains.

Even for the intended audience, the book suffers somewhat from awkward overall organization. The individual chapters are well written and (after one allows for the need to use a highly technical vocabulary) clear. But the sequence of chapters feels arbitrary, and the many forward references in the text make it difficult to read. He drops an OSPF FAQ (Frequently Asked Questions) chapter into the middle of the book. Though a good overview chapter written in a less demanding style than the others, its placement seems arbitrary and poor.

In recent months, I have come to be impressed by the improved quality of the technical documentation available for various Internet standards and protocols. Once the exclusive province of abstruse RFCs (Requests for Comment, the non-descriptive name for the official document that describes an Internet protocol or standard) and hard-to-decipher Unix man pages (on-line manual), these protocols are now generally documented in well-written print volumes. OSPF: Anatomy of an Internet Protocol continues this tradition. It is not the best I have seen, but neither is it the worst. Reading it will reward any reader with the appropriate level of technical interest.

Ray Olszewski (ray@comarre.com (mailto:ray@comarre.com) or http://www.comarre.com/ray.html (http://www.comarre.com/ray.html)) is a consulting economist and statistician. His work includes development of custom Web-based software to support on-line research. He spent three years as Network Manager at The Nueva School, a private K-8 school in Hillsborough, California.

table of contents ter issues (/lita/publications/archive/ter)

REVIEW OF: Patrick McDermott. Solving the Year 2000 Crisis. Boston: Artech House, 1998.

by John Wynstra

A number of books about the Y2K bug have recently appeared at local book stores. The market for these books is being driven by the fear of the quickly approaching turn of the century and the potential wide-spread computer problems that have been predicted. Some of these books are survival guides to help readers prepare for the "end of the world", some of them are general awareness books for people who just want information, and some of them are technically oriented problem-solving books for software developers and project managers who are faced with the responsibility of eliminating Y2K problems. This book fits in the latter category. It is concerned with software development issues more so than hardware issues. It focuses on "Y2K bugs" in software code- how they got there, how to find them, and what to do about them.

The author starts this book by describing seven symptoms that clarify the problems that can result from using a 2-digit date rather than a 4-digit date in software. The term symptom is almost a misnomer here since these symptoms really define the root cause of the problem and not so much the misbehavior of the software. Symptom 2 states: "2001 - 1999 = 2, but 01 99 = -98"(p. 7). This symptom demonstrates how calculations based on subtracting a two-digit date from another two-digit date that are working fine in 1999 will begin to malfunction in the year 2000. The symptoms are written more for the technically minded individual and provide clues of what to look for in the code. They are concise and cover most of the problems that will appear with the year 2000.

The second part of the book describes seven different approaches to fixing the problems in your software. The solutions are to replace, expand, window, compress, work around, encapsulate, and abandon. The author dedicates a chapter to describing each solution's pros and cons. In addition, the author explains when the appropriate time is to use each solution. The solutions to abandon, replace, or find a work around are not fixes to the software, but advice to find a way to stop using the affected software. The other solutions are different approaches to the same problem and not different solutions for different problems. Here the author includes much good information about why one solution may be more appropriate than another after taking all things into consideration, including time and cost.

Part three is about the various people that may need to be involved in fixing Y2K problems. Among topics covered in this section are outsourcing, the labor market, and consultants. The author brings up many important considerations about getting and paying for qualified workers for a Y2K project. This section includes an interesting analysis of the number of programmers world-wide listed by the languages they program in and compared to the number of man-hours estimated to fix Y2K problems in each language. Some languages are more effected than others and will have less programmers available to solve the problems. This section is particularly relevant for a project manager or business manager who has to understand the costs involved in proceeding with a Y2K project.

Part four is about developing and managing a Y2K project. The basic project has 6 phases: Assessment, Strategy, Repair, Test, Implement, and Postimplementation. This is one project that has an unchangeable deadline. Based on the fact that most major projects come in over budget and late, the author recommends preparing a triage strategy for determining which systems get fixed first.

Most of the book up to this point is concerned with the technical aspect of this problem; part five is a change of pace with it's focus on the business perspective. This section is for "the business user who is not a computer expert and does not want to be." (p. 229) This section contains basic concepts and includes a chapter about desktop PCS and how they may be affected by the turn of the century. The author includes a chapter at the end of this section about performing Y2K testing to verify and validate the effectiveness of software fixing that has been done. The author considers this testing phase to be one of the most important phases and potentially the most time consuming.

Overall this is a well-written and relevant book. The author is thorough in covering known problems and solutions related to software development. The author has good insight into making solution selection decisions, creating a triage strategy, and developing and managing a full-scale project. He also includes a number of appendices with code segments in Cobol, C++, Fortran, and Visual Basic that can be used to fix Y2K bugs. The book is written from a technical point of view for the most part and is recommended reading for anyone involved in fixing Y2K problems in software.

John Wynstra (wynstra@uni.edu (mailto:wynstra@uni.edu)) is the Library Systems Project Manager at the University of Northern Iowa.

table of contents ter issues (/lita/publications/archive/ter)

About TER