Honeypots for Windows

ROGER A. GRIMES

Apress®

Honeypots for Windows

ISBN (pbk): 1-59059-335-9

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Lead Editor: Jim Sumser
Technical Reviewers: Alexzander Nepomnjashiy, Jacco Tunnissen
Editorial Board: Steve Anglin, Dan Appleman, Ewan Buckingham, Gary Cornell, Tony Davis, Jason Gilmore, Chris Mills, Dominic Shakeshaft, Jim Sumser
Assistant Publisher: Grace Wong
Project Manager: Sofia Marchant
Copy Manager: Nicole LeClerc
Copy Editor: Marilyn Smith
Production Manager: Kari Brooks-Copony
Production Editor: Kelly Winquist
Compositors: Kinetic Publishing Services, LLC; Dina Quan
Proofreader: Katie Stence
Indexer: Carol Burbo
Artist: Kinetic Publishing Services, LLC; Dina Quan
Cover Designer: Kurt Krames
Manufacturing Manager: Tom Debolski

Distributed to the book trade in the United States by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013, and outside the United States by Springer-Verlag GmbH & Co. KG, Tiergartenstr. 17, 69112 Heidelberg, Germany.

In the United States: phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders@springer-ny.com, or visit http://www.springer-ny.com. Outside the United States: fax +49 6221 345229, e-mail orders@springer.de, or visit http://www.springer.de.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit http://www.apress.com.

The source code for this book is available to readers at http://www.apress.com in the Downloads section. You will need to answer questions pertaining to this book in order to successfully download the code.

To those who fight the good fight with constant vigilance.

Contents at a Glance

Contents

About the Author

ROGER A. GRIMES is a 17-year computer security industry veteran, full-time teacher, author, and consultant. He is the author of 4 books and more than 150 magazine articles on computer security, specializing in Microsoft Windows security and malware defenses. He is a contributing editor for *Windows IT Pro* and *InfoWorld* magazines. His certifications include CPA, CISSP, CEH, CHFI, TICSA, MCT, MCSE: Security (NT/2000/2003/MVP), Security+, A+, and others. Roger is a frequent presenter at national conferences, including MCP TechMentor, Windows Connections, and SANS, where he is always among the highest rated presenters. Roger has created several courses on advanced Windows security for Microsoft, *Windows IT Pro* magazine, and SANS. His clients have included every branch of the armed forces, Microsoft, VeriSign, Fortune 500 companies, cities, and large public school systems and universities.

About the Technical Reviewers

ALEXZANDER NEPOMNJASHIY is a Microsoft SQL Server database designer for NeoSystems NorthWest, a security services, consulting, and training company. He has more than 11 years of experience in the IT field. His work involves extending and improving clients' corporate ERP systems to manage retail sales data, predict market changes, and calculate trends for future market situations.

JACCO TUNNISSEN has been working in the ISP and security fields since the mid-1990s, mainly focusing on FreeBSD and OpenBSD implementations. Currently, he is "educating the masses" using his web sites, where you can find out all about intrusion detection, honeypots (`http://www.honeypots.net`), incident handling, wireless security, computer forensics, DNS, and BGP routing. In his spare time, he enjoys good food and biking in Rotterdam. Jacco likes working as a technical reviewer for several authors.

Acknowledgments

I wish to thank Apress and my editor Jim Sumser, Sofia Marchant, Marilyn Smith, and StudioB's Neil J. Salkind for seeing the vision for a book like this and putting up with my moving deadlines.

I also want to thank Lance Spitzner, Michael Davis, and Niels Provos, for evangelizing honeypot technology, and answering my many questions. Thanks to Alexzander Nepomnjashiy and Jacco Tunnissen for the excellent technical editing.

Much of this book could not have been written without the previous contributions of The Honeynet Project (`http://project.honeynet.org`), Honeypot: Tracking Hackers (`http://www.tracking-hackers.com`), SANS (`http://www.sans.org`), and the Honeypot mailing list (`http://www.securityfocus.com`).

On a personal note, I would especially like to thank my wife, Tricia, who took care of my every need while I was writing and neglecting her. I could not ask for a better friend and partner.

Introduction

Welcome to the world of honeypots! By reading this book, you are joining a friendly community of like-minded individuals who see honeypots as an important step in preventing and learning about malicious hacking activity.

If you were to ask a honeypot administrator why he uses a honeypot, the first response you would likely get is one or more legitimate business reasons explaining why honeypots are the best tool for the job. We've been trained to respond that way, so we can defend all our time spent managing and learning about honeypots to our bosses and loved ones. But underlying all the official logic is the real reason why most of us get interested in honeypots: the thrill of watching the bad guys expose themselves and their techniques, in an environment explicitly built to exploit them. It's videotaping the thief. It's taking the hacker's favorite tool of social engineering and using it against him. It's hacking the hacker.

For once, the good guys are in control and winning. With honeypots, we can track the hackers back to their lairs, identify them, and learn and defend against their techniques before they really get a chance to use them. Honeypots can stop hacker attacks, Internet worms, spam, and other acts of maliciousness. Honeypots remove the implicit veil of privacy most hackers think they have when they exploit a computer. If honeypots become as big and mature as most security experts expect them to be, they could put an end to the random, no consequence, hacking that so proliferates our Internet experience today. Honeypots are one of the few offensive plays in the computer security world.

Honeypots no longer need to prove their value. They have been responsible for discovering many new threats (called *zero-day exploits*) before they were widely known and used, including a Samba buffer overflow. Honeypots have been essential in capturing encrypted hacker traffic and instrumental in learning that hackers are using version 6 of the Internet Protocol (IPv6) to tunnel their communications under the very noses of network administrators. But perhaps, the greatest demonstration of a honeypot's value was the recent profiling of an extensive credit card fraud operation (`http://www.honeynet.org/papers/profiles/cc-fraud.pdf`).

Researchers followed the online activities and communications of individuals involved with credit card fraud, also known as *carders*. The honeynet was able to capture an incredibly automated Internet Relay Chat (IRC) network that allowed credit card numbers and identifying information to be stolen by simply typing a few keystrokes. Members of the ring could type in `!cc` to receive one random stolen credit card number. Other commands returned credit card account limits and provided web site links to vulnerable online merchant sites. The honeynet tracked key individuals, mostly located in South Asia and Pacific Rim countries, as they bought credit card numbers from legitimate businesses, hacked credit card security systems, and taught newcomers the ins and outs of carding. The credit card ring, by plying its trade online and out in the open, made their illegal activities appear more as a subculture than a crime. This attracted new participants. Honeypots allowed evidence to be recorded that after-the-fact affidavits, search warrants, and bugged phone conversations couldn't provide. The honeypot has solidified its place as a legitimate computer security defense tool.

Why Another Book on Honeypots?

There are already some excellent books, papers, and web sites on honeypots, so why did I feel compelled to write a book on the subject? In one sentence, it's because the world of computer security, and honeypots in particular, is largely Unix-based. Most of the literature about firewalls, intrusion detection systems (IDSs), and honeypots was written by Unix gurus. Most of the tools are Unix-based and work only on Unix platforms. Even when the tools are ported to Windows, they may talk about Windows and give a few Windows examples, but most of the text and examples are for Unix-based users. It can be very frustrating when you are not a Unix person but still want to learn about computer security and use all the cool tools.

The majority of the world's PCs run one of Microsoft's Windows operation systems. This book was written to fill the large gap for Windows administrators trying to learn about honeypots outside the Unix subculture. I don't give examples of software and exploits that do not occur in the typical Windows environment. When the book discusses mail servers, it will be referring to Microsoft Exchange, not Sendmail. When it refers to web servers, it will be talking about Microsoft Internet Information Services (IIS), not Apache. Yes, I know both of those programs have Windows-based counterparts, but they aren't the norm in a Windows network. This doesn't mean that the knowledge and lessons learned in this book cannot be applied to non-Microsoft environments. The opposite is true. You can take anything covered in this book and easily apply it to Unix, Linux, Macintosh, or any other computer environment. But for once, this is a honeypot resource that targets the Windows administrator. It means the following:

- Honeypot planning and setup will target Windows systems.
- Security tools will be Windows versions.
- Hacking examples will be Windows examples.
- When TCP/IP is discussed, it will be as it applies in Windows.
- When TCP/IP ports are discussed, they will be ports common to Windows.

Honeypots for Windows is a book for Windows-based users and administrators.

Who Is This Book For?

This book is for administrators and users with an intermediate understanding of the Windows operating system and computer security. Readers should have experience with the Windows operating system, the Internet, and Windows-based networking; be able to install and troubleshoot network-related software; and have general understanding of the OSI model. It helps if you're familiar with basic computer security concepts, such as computer worms, buffer overflows, and password cracking. An understanding of Windows security mechanisms will make the book more enjoyable.

A strong understanding of TCP/IP network protocol basics is essential for most honeypot administrators. Although this book will cover the fundamentals needed to understand the material presented, readers should understand the following terms prior to beginning this journey: TCP, UDP, ICMP, stateful, stateless, flags, TCP/IP handshake, packet header, and packet payload.

But even if you're not familiar with the details of all these topics, you should still be able to understand every concept discussed in this book. So, don't panic if you can't name all the TCP header flags off the top of your head, or if don't know the exact meaning of stateful inspection. This book will be of value to people newly interested in computer security and honeypots, as well as to experienced security experts.

Readers without a firm foundation in these fundamentals should consider a quick refresher with a TCP/IP protocol reference. There are several good books on the TCP/IP protocol, and here are some online references:

- Webopedia's TCP/IP page: `http://www.webopedia.com/TERM/T/TCP_IP.html`
- An excellent TCP/IP Reference by Cisco: `http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ip.pdf`
- About.com's Computer Networking Guide to TCP/IP: `http://compnetworking.about.com/cs/basictcpip`
- Wikipedia Internet Protocol Suite reference: `http://www.wikipedia.org/wiki/Internet_protocol_suite`
- Internet Engineering Task Force (IETF) references: `http://www.ietf.org`

Note On the IETF web site, at a minimum, read the following Request For Comments (RFCs): 791-IP, 792-ICMP, 768-UDP, and 793-TCP. RFCs are very wordy and long, and a bit like reading IRS tax code, but taking the time to read them will allow you to understand the TCP/IP protocol suite in detail.

What's In This Book?

The book has twelve chapters organized into three main parts.

By the time you get through reading this book, you should have an excellent understanding of honeypots in a Windows environment.

Note Many of the tools covered in this book are Windows ports of open-source Unix tools, like Honeyd, WinPcap, and Snort. All of these tools have been tested on Windows 98 and later Microsoft platforms, but most have been optimized for Windows 2000 and above. Menu options and screenshots were done on Windows 2000 and XP Professional computers, but most commands and screens are identical, no matter which Microsoft operating system you use. Every effort has been made to verify that all commands and utilities work across all current versions of Windows. Exceptions are noted when known.

Part One: Honeypots in General

Part One covers honeypot theory and topics common to all honeypots, along with the particular configuration requirements of a Windows-based environment.

Chapter 1 explains general honeypot theory and reasons to use honeypots. It discusses the main honeypot types, along with advantages and disadvantages of each choice. The chapter also covers hacking basics, such as attack model types and fingerprinting. Understanding the different hacking threats is essential to setting up and using a honeypot.

Chapter 2 describes the general setup and deployment of a honeypot, as well as how to attract hackers to it. Topics include how to decide where to place a honeypot and why. It covers the physical deployment issues involved in placing a honeypot, including hardening the host and configuring your network to route hacking traffic to your honeypot. It includes details on the problems introduced on switched networks and how to correctly configure your routing tables.

Part Two: Windows Honeypots

Part Two provides a detailed lesson in configuring and using Windows-based honeypots. Using an emulated honeypot in a Windows environment takes special consideration to make it appear as a Windows-based host. This means it should have the normal Windows ports open, run the normal Windows services, and respond in a predictable way. Chapter 3 defines normal behaviors, ports, and services on a Windows host, and tells you how to emulate them on a honeypot.

Chapter 4 describes using a real Windows operating system as a honeypot. It reveals what is the best Windows version to attract malicious hackers and presents hardening tips you can use to minimize compromise damage.

Chapters 5 through 7 focus on Honeyd, the most popular honeypot software in use today. Chapter 5 covers how to download and install Honeyd. Honeyd is a fantastic free tool, but like many other open-source programs, not particularly easy to configure. Chapter 6 begins deciphering the Honeyd configuration and provides several sample configuration files that you can adapt for your own needs. Chapter 7 explains how to use service scripts, which allow Honeyd to mimic basic applications, such as FTP, telnet, and IIS. Service scripts are very important in making a honeypot look like a real system.

Honeyd is the most popular and versatile honeypot software in use today, but it isn't the easiest to use. In Chapter 8, we explore six other Windows-based honeypots with front-end graphical user interfaces that make for a more pleasant user experience. Each of these honeypots excels at different goals. The honeypots are Back Officer Friendly, LaBrea, SPECTER, KFSensor, PatriotBox, and Jackpot.

Part Three: Honeypot Operations

Part Three discusses a range of topics related to getting the most out of your honeypot.

Using a network traffic analyzer and understanding how to recognize and decode malicious network traffic is essential to honeypot operations. Chapter 9 discusses how to install and use various tools for analyzing network traffic. It begins with network protocol basics, reviewing the OSI model and TCP/IP suite, and then focuses on using Snort and Ethereal.

Chapter 10 covers the very important issues of monitoring, logging, alerting, and reporting. It discusses how to set up an alert system, how and what to log, and what reports you need to generate.

Honeypots can quickly gather copious amounts of information—sometimes an overwhelming amount. The ultimate success of your honeypot is determined by how well you interpret the attack evidence. Chapter 11 discusses techniques to use in the forensics analysis of your honeypot data.

Chapter 12 discusses analyzing malicious code by disassembling it. For new programmers, this involves learning assembly language, learning how to disassemble executables, and learning about malicious coding in general. Becoming a disassembler is not for the faint of heart, but with a moderate amount of effort and practice, it can reveal malware functions that cannot be found any other way.

The sample files presented in this book, as well as other related files, are available from the Downloads area of the Apress web site (`http://www.apress.com`). You can direct any technical questions or concerns to me at roger@banneretcs.com.