

Hack Attacks Testing

**How to Conduct Your
Own Security Audit**

John Chirillo



WILEY

Wiley Publishing, Inc.

The WILEY *advantage*

Dear Valued Customer,

We realize you're a busy professional with deadlines to hit. Whether your goal is to learn a new technology or solve a critical problem, we want to be there to lend you a hand. Our primary objective is to provide you with the insight and knowledge you need to stay atop the highly competitive and ever-changing technology industry.

Wiley Publishing, Inc., offers books on a wide variety of technical categories, including security, data warehousing, software development tools, and networking—everything you need to reach your peak. Regardless of your level of expertise, the Wiley family of books has you covered.

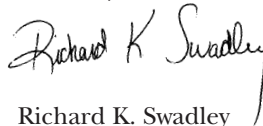
- For Dummies—The *fun* and *easy* way to learn
- The Weekend Crash Course—The *fastest* way to learn a new tool or technology
- Visual—For those who prefer to learn a new topic *visually*
- The Bible—The *100% comprehensive* tutorial and reference
- The Wiley Professional list—*Practical* and *reliable* resources for IT professionals

The book you hold now, *Hack Attacks Testing: How to Conduct Your Own Security Audit*, allows you to perform your own security audit by providing step-by-step guidance on how to build and operate a security analysis/monitoring system. Covering both Windows and UNIX—in a dual boot configuration—the book covers building and operating your own vulnerability analysis system, using only the top-quality tools available today. You'll find these tools on the book's CD-ROM. This book will be very valuable to anyone who needs to regularly conduct network security audits while staying within a limited budget.

Our commitment to you does not end at the last page of this book. We'd want to open a dialog with you to see what other solutions we can provide. Please be sure to visit us at www.wiley.com/compbooks to review our complete title list and explore the other resources we offer. If you have a comment, suggestion, or any other inquiry, please locate the "contact us" link at www.wiley.com.

Finally, we encourage you to review the following page for a list of Wiley titles on related topics. Thank you for your support and we look forward to hearing from you and serving your needs again in the future.

Sincerely,



Richard K. Swadley
Vice President & Executive Group Publisher
Wiley Technology Publishing



Bible

FOR
DUMMIES

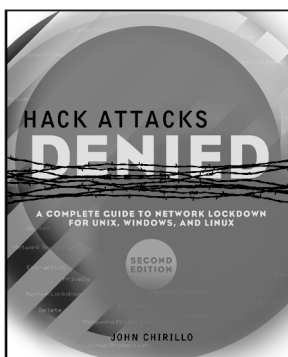


Wiley Publishing, Inc.

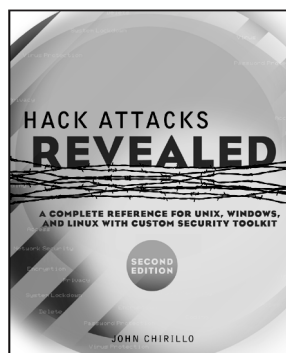
*more information
on related titles*

The Next Level of Hack Attacks Testing

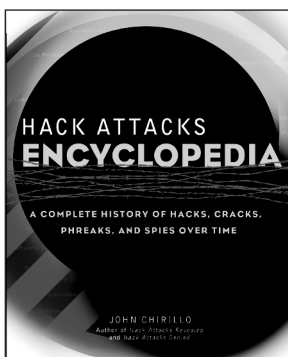
Available from Wiley Publishing



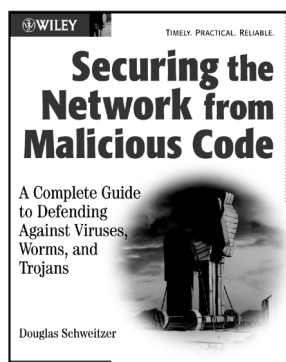
Chirillo/Hack
Attacks Denied 2E
0471232831
Design and fortify
networks against
the latest attacks



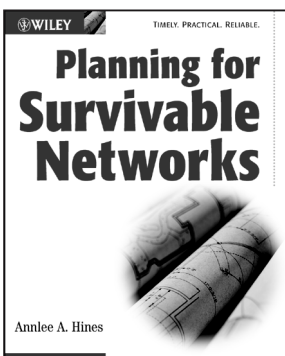
Chirillo/Hack Attacks
Revealed, 2E
0471232823
See network
security through the
hacker's eye



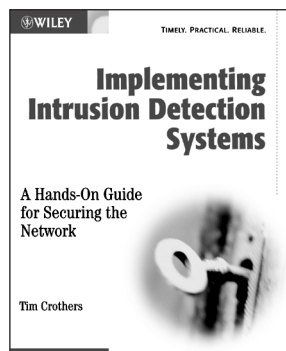
Chirillo/Hack
Attacks
Encyclopedia
0471055891
A complete library
of the texts, files,
and code used by
hackers



Schweitzer/Securing
the Network from
Malicious Code
0764549588
Inoculate your
network against
viruses, worms, and
Trojans



Hines/Planning for
Survivable
Networks
047123284X
Keep your network
safe from security
disasters with a
dependable
recovery strategy



Crothers/
Implementing
Intrusion Detection
Systems
0764549499
A hands-on guide
for securing the
network



Wiley Publishing, Inc. Available at your favorite bookseller or visit www.wiley.com/compbooks

Hack Attacks Testing

**How to Conduct Your
Own Security Audit**

Hack Attacks Testing

**How to Conduct Your
Own Security Audit**

John Chirillo



Wiley Publishing, Inc.

Publisher: Bob Ipsen
Editor: Carol A. Long
Developmental Editor: Janice Borzendowski
Managing Editor: Micheline Frederick
Text Design & Composition: Wiley Composition Services

Designations used by companies to distinguish their products are often claimed as trademarks. In all instances where Wiley Publishing, Inc., is aware of a claim, the product names appear in initial capital or ALL CAPITAL LETTERS. Readers, however, should contact the appropriate companies for more complete information regarding trademarks and registration.

This book is printed on acid-free paper. ♾

Copyright © 2003 by John Chirillo. All rights reserved.

Published by Wiley Publishing, Inc., Indianapolis, Indiana
Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470. Requests to the Publisher for permission should be addressed to the Legal Department, Wiley Publishing, Inc., 10475 Crosspoint Blvd., Indianapolis, IN 46256, (317) 572-3447, fax (317) 572-4447, E-mail: permcoordinator@wiley.com.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Library of Congress Cataloging-in-Publication Data:

ISBN: 0-471-22946-6

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1



Contents

Acknowledgments	xi
Introduction	xv
Part 1 Building a Multisystem Tiger Box	1
Chapter 1: Basic Windows 2000/Windows 2000 Server Installation and Configuration	11
Launching Windows 2000 Server	11
Basic Windows 2000/Windows 2000 Server Configuration	15
Active Directory	16
TCP/IP Customization	40
Domain Name Service	46
Chapter 2 Basic Linux and Solaris Installations and Configurations	53
*NIX Minimum System Requirements (Intel-Based)	53
Installing and Configuring Red Hat Linux	54
Installing and Configuring Solaris 8	64
Installation Completion	69
Chapter 3 Mac OS X Tiger Box Solutions	71
Minimum System Requirements: Step 1	71
Installing Mac OS X: Step 2	72
Installing OS X	72
Upgrading to OS X	73
Installing Developer Tools: Step 3	73
Downloading the Software	73
Installing and Configuring a Port Scanner Infrastructure: Step 4	76
Installing Netscape	81
Enabling the Root Account	81

	Modifying the PATH	82
	Nessus Security Scanner Example Configuration	83
	Logging In with the Client	91
	Conclusion	92
Chapter 4	Installing and Configuring a Testing Target	93
	Minimum Hardware Requirements	93
	Installation Methods	94
	Server Licensing	95
	Server Types	96
	Step-by-Step Installation	97
	Logging In	99
	Optional Services for Your Testing Target	100
	Installing WINS	100
	Setting Preferences for WINS Manager	102
	Configuring a WINS Server	103
	WINS Static Mappings	104
	WINS Database	106
	Installing DNS	106
	DNS Zones, Hosts, and Records	108
	Internet Information Server Step by Step	110
	IIS Installation and Configuration	110
	IIS Administration Utility	111
	Conclusion	120
Part 2	Using Security Analysis Tools for Your Windows-Based Tiger Box Operating System	121
Chapter 5	Cerberus Internet Scanner	135
	System Requirements	136
	Installation	136
	Target Configuration	137
	Vulnerability Scanning	146
	Reporting	147
Chapter 6	CyberCop Scanner	157
	System Requirements	158
	Installation	158
	Initial Configuration and Product Update	159
	Welcome to Update	163
	Setup Configuration Options	164
	Target Configuration	170
	Selecting Modules for a Scan	170
	Vulnerability Scanning	175
	Performing Intrusion Detection System Software Tests	176
	Advanced Software Utilities	179
	CASL	180
	Creating and Sending an Example Packet	182
	Crack	184
	SMB Grind	186

	Reporting	188
	Network Map	190
	Output File	191
	Example Report	192
Chapter 7	Internet Scanner	199
	System Requirements	199
	Installation	200
	Starting Internet Scanner for the First Time	200
	Command-Line Option	201
	Target Configuration	202
	Vulnerability Scanning	209
	Scanning from the GUI Mode	209
	Scanning from the Console Mode	210
	Scanning from the Command-Line Mode	211
	Reporting	212
	Sample Report	214
Chapter 8	Security Threat Avoidance Technology Scanner	231
	System Requirements	233
	Installation	233
	Starting STAT Scanner for the First Time	234
	Target Configuration	236
	Target Selection	237
	Vulnerability Selection	238
	Vulnerability Scanning	239
	Command-Line Usage	242
	Vulnerability Display	243
	Reporting	245
	Sample Report	246
Chapter 9	TigerSuite 4.0	257
	Installation	257
	Local Installation Method	258
	Mobile Installation Method	261
	Program Modules	261
	System Status Modules	262
	Hardware Modules	262
	System Status Internetworking Modules	265
	TigerBox Toolkit	269
	TigerBox Tools	269
	TigerBox Scanners	275
	TigerBox Penetrators	277
	TigerBox Simulators	281
	Using the Session Sniffers	283
	PortSpy Communication Sniffer	283
	TigerWipe Active Processes	285
	Practical Application	286
	Tracing Back with TigerSuite	286

Part 3	Using Security Analysis Tools for *NIX and Mac OS X	291
Chapter 10	hping/2	315
	Idle Host Scanning and IP Spoofing	316
	System Requirements	325
	Linux Installation and Configuration	326
	Other Installations	329
	Using hping/2	329
Chapter 11	Nessus Security Scanner	339
	System Requirements	340
	Installation and Configuration	341
	Automatic Installation	346
	Configuring Nessus Security Scanner	347
	Starting the Server Daemon	350
	Additional Notes for Linux and Solaris Users	354
	For Mac OS X Users	355
	Vulnerability Scanning	356
	Plugins	358
	Scan Options	359
	Target Configuration	360
	Reporting	362
Chapter 12	Nmap	371
	System Requirements	373
	Installation and Configuration	373
	Other Installations	380
	For Mac OS X Users	380
	Using Nmap	382
	TCP Scanning	383
	UDP Scanning	384
	Half-Open (Stealth) Scanning	384
	Operating System Fingerprinting	385
	Mixing It Up	391
Chapter 13	SAINT	393
	System Requirements	393
	Installation and Configuration	394
	Vulnerability Scanning with SAINT	398
	SAINT Home	403
	Data Management	403
	Configuration Management	404
	Target Selection	404
	Reporting	407
	Vulnerabilities	408
	Host Information	409
	Severity Levels	410

Using SAINT Remotely	411
The config/passwd File	412
The Command-Line Interface	413
Scheduling Scans Using cron	414
Summing Up	415
Chapter 14 SARA	417
System Requirements	418
Installation and Configuration	418
Advanced Configurations	423
SARA Database Format	424
Vulnerability Scanning	426
Target Configuration and Starting a Scan	429
From the Command Line	431
Reporting	432
Part 4 Vulnerability Assessment	439
Chapter 15 Comparative Analysis	441
Target Network Specifications	441
Windows NT Server 4.0	442
Red Hat Linux 7.3 Professional	444
Sun Solaris 8 SPARC	445
NT and *NIX Auditing Checklists	446
Windows NT System Security Checklist	446
Vulnerability Scanner Results and Comparison	469
What's Next	477
Firewalls and Intrusion Detection System Software	477
Network Monitors	477
Appendix A Linux/Unix Shortcuts and Commands	479
Linux Essential Keyboard Shortcuts	
and Sanity Commands	479
Additional KDE Keyboard Shortcuts	483
System Info	485
File Management	491
Process Control	493
Administration Commands	495
Hard Drive/Floppy Disk Utilities	502
Management of User Accounts and File Permissions	505
Accessing Drives/Partitions	508
Network Administration Tools	509
Appendix B What's on the CD-ROM	513
Index	523



Acknowledgments

To be successful, one must surround oneself with the finest people. With that in mind, foremost I would like to thank my wife for her continued support and patience during this book's development. Next, I thank my family and friends for their encouragement and confidence.

I am also grateful to Carol Long, Adaobi Obi, Micheline Frederick, Erica Weinstein, Ellen Reavis, Kathryn Malm, Janice Borzendowski, and anyone else I forgot to mention from John Wiley & Sons.



About the Author

John Chirillo began his computer career at age 12 when, after one year of self-taught education on computers, he wrote a game called Dragon's Tomb. Following the game's publication, thousands of copies were sold to the Color Computer System market. During the next five years, John wrote several other software packages, including The Lost Treasure (a game-writing tutorial), Multimanager (an accounting, inventory, and financial management software suite), Sorcery (an RPG adventure), PC Notes (a GUI used to teach math, from algebra to calculus), Falcon's Quest I and II (a graphical, diction-intensive adventure), and Genius (a complete Windows-based point-and-click operating system). John went on to become certified in numerous programming languages, including QuickBasic, VB, C++, Pascal, Assembler, and Java. John later developed the PC Optimization Kit, which increased the speeds of standard Intel 486 chips by up to 200 percent.

After running two businesses, Software Now and Geniusware, John became a consultant to prestigious companies, where he specialized in performing security and sniffer analyses, as well as LAN/WAN design, implementation, and troubleshooting. During this period, John acquired numerous internetworking certifications, including CCNA, CCDA, CCNP, Intel Certified Solutions Consultant, Compaq ASE Enterprise Storage, Unix, CISSP, and pending CCIE. He is currently a senior internetworking engineer at a technology management company.

John is the author of several security and networking books, including the *Hack Attacks* series from John Wiley & Sons.



Introduction

The objective of this book is to fill a gap found in most books on security: How security examinations can be conducted via illustrations and virtual simulations. Auditing tools with simple graphical user interfaces (GUIs) and automation are becoming increasingly prevalent, and most claim to be the all-inclusive solution for administrators and security consultants to use for their networks' security testing. In practice, however, typically a combination of tools, embraced by the Tiger Box analysis/monitoring system, is necessary for accurate, up-to-date assessments. In a nutshell, a Tiger Box is a system designed to provide the necessary tools designed to reveal potential security weaknesses by discovering, scanning, and in some cases penetrating security vulnerabilities. Covering Windows in addition to Unix- and Linux-flavored (*NIX) dual-boot-configurations, this book explains how to build and operate your own vulnerability analysis system by using exclusively the top-quality and most popular tools available today.

Step by step, the book covers how-to drilldowns for setting up your Tiger Box operating systems, installations, and configurations for some of the most popular auditing software suites. It discusses both common and custom uses, as well as the scanning methods and reporting routines of each. It inspects individual vulnerability scanner results and compares them in an evaluation matrix against a select group of intentional security holes on a target network.

The Companion CD-ROM

If you seek general hands-on experience of most of the scanners discussed in this book, look no further than this book's companion CD-ROM, for it contains an interactive workbook for the text. It covers basic uses of the scanners, some containing interactive reports, so that you can familiarize yourself with their interfaces.

This electronic workbook is designed to introduce scanners as simulations from real uses. For still more experience, simply download product evaluations from the links in each part.

Who Should Read This Book

This book is written to explain how you can perform your own security audits. It contains beginner to advanced uses for which no experience with the tools is necessary. It is intended as a required guide not only for managers, security engineers, network administrators, network engineers, and internetworking engineers but for interested laypeople as well.

Building a Multisystem Tiger Box

Within the International Information Systems Security Certification Consortium's Common Body of Knowledge domains, vulnerability scanning and penetration testing are positioned as part of problem identification auditing for network defense testing against techniques used by intruders. In other words, regularly scheduled security audits should be practiced, especially in regard to safeguarding the assets of all enterprises, from the very large to the small office/home office. An effective security implementation is composed of several life cycle components, including security policies, perimeter defenses, and disaster recovery plans, to name a few; however, auditing the effectiveness of security controls is critical.

This book is intended to serve as a general how-to "cookbook" in regard to discovery, vulnerability, and penetration testing. With that in mind, let's begin by reviewing the National Institute of Security Technology (NIST) list of the eight major elements of computer security:

1. Computer security should support the mission of the organization.
2. Computer security is an integral element of sound management.
3. Computer security should be cost-effective.
4. Computer security responsibilities and accountability should be made explicit.
5. System owners have computer security responsibilities outside their own organizations.

6. Computer security requires a comprehensive and integrated approach.
7. Computer security should be periodically reassessed.
8. Computer security is constrained by societal factors.

Whether or not all of the security controls or elements are in place, an analysis can help provide a solid grasp of how your security solution will protect critical systems and data. Networks, including those not connected to the Internet, may have security breaches and other areas that, if not addressed, can invite undesired access to confidential data. The principal mission of this book is to identify the most popular assessment tools, illustrate and virtually simulate their *modus operandi* for local and remote assessments, and then report our findings and document our corrective procedures.

NOTE This text attempts to adhere to the InfoSec Criteria and Methods of Evaluations of Information Systems, specifically, Information Technology Security Evaluation Criteria for effective assessment of a target of evaluation (TOE) against the following approaches: (1) the suitability of the TOE's security-enforcing functions to counter the threats to the security of the TOE identified in the security target; (2) the ability of the TOE's security-enforcing functions and mechanisms to bind in a way that is mutually supportive and that provides an integrated and effective whole; (3) the ability of the TOE's security mechanisms to withstand direct attack; (4) whether known security vulnerabilities in the construction and the operation of the TOE could, in practice, compromise the security of the TOE; and (5) that the TOE cannot be configured or used in a manner that is insecure but that an administrator or end user of the TOE would reasonably believe to be secure.

Seven Phases of Analysis

Whether your home or business is newly connected to the Internet or you have long had your Internet connectivity and/or network infrastructure in place, an analysis can help determine whether you are sufficiently protected from intrusion. The typical guidelines for performing a security analysis are to develop a plan, perform the audit, and then report your findings. This section proposes the common assessment phases of a detailed security audit. We'll cover the following:

- *Site scans*, to test port and application layer against internal defenses.
- *Remote audits*, to test against external services—for example, Internet service provider (ISP) hosting, servers, and conduits.
- *Penetration tests*, to test Internet security and validate current risks. You should be responsible to clearly articulate the specific objectives, requirements, and timeframes associated with the testing, and exercise due care to ensure that data and systems are not damaged by the testing, that the target site is notified

of any vulnerabilities created during testing, and that testing is stopped immediately at the request of the site.

- *Internet protocol (IP), mail spoof, and spam tests*
- *Dial-up audit*, to ensure remote access connectivity security for products such as PC Anywhere, Reachout, and/or Citrix.

An external audit should be performed remotely, that is, off-site or from outside any perimeter defense, such as a firewall. This should be first performed blind, that is to say, without detailed infrastructure knowledge.

Following this first phase, a knowledgeable penetration test will determine the extent and risk (if any) of an external attack. This audit is valuable for testing the configuration of perimeter security mechanisms, the respective Web, File Transfer Protocol (FTP), e-mail, and other services. This scan and simulated attack are done remotely over the Internet. Preferably, this phase should be performed with limited disclosure (blind to all but select management) as an unscheduled external penetration assessment.

Many times penetration tests should be limited to passive probes so as not to cause any manner of disruption to business. Optionally, penetration tests may include the attack and evaluation of modem dial-ups and physical security, which may be accomplished by a method known as *wardialing*, a procedure used to scan and detect misconfigured dial-ups and terminal servers, as well as rogue and/or unauthorized modems.

When audits are aimed at Web sites, source code audits of the common gateway interface (CGI), Java, JavaScript, and ActiveX should be performed. As audits are being performed, a detailed, time-stamped log should be maintained of all actions. This log will be used in further testing against current station logging facilities by comparing audit logs and target site logs. Most important, if you perform an audit for reasons other than personal, you should initiate it only upon gaining written permission on company letterhead from the appropriate company officer.

Security audits should be performed regularly. Based on the techniques, tools, and software evaluated in books such as *Hack Attacks Revealed, Second Edition*, a good analysis can be divided into seven phases.

Phase 1: Blind Testing

In blind, or remote, testing, one lacks detailed knowledge of the target infrastructure.

Site Scan

The site scan includes the following:

- Network discovery
- Port scan of all ports identified during the discovery
- Application scan to identify system services as they pertain to discovered ports
- Throughput scans for port utilization levels to identify vulnerabilities
- Documentation

Remote Audit

During a remote audit, one does the following:

- Tests the configuration, stability, and vulnerabilities of perimeter defenses, external ISP services, and any other network services acting as conduits through a firewall or proxy
- Provides documentation

Penetration Tests

During penetration tests, one does the following:

- Attacks and evaluates the physical security, with intent to penetrate, of all items that were identified during the site scan and remote audit
- Audits the source code for CGI, JavaScript, and ActiveX
- Initiates Object Database Connectivity (ODBC) calls from customer-identified databases
- Performs IP flood tests
- Initiates standard Windows NT, Novell NetWare, and Unix IOS cracks
- Carries out Domain Name Service (DNS) spoofing
- Initializes sniffer-passive probes to capture traffic
- Prepares documentation

IP, Mail Spoof, and Spam Tests

During IP, mail spoof, and spam tests, one does the following:

- Performs penetration attacks to drive infrastructure equipment into making damaging statements and/or releasing sensitive information (e.g., password keys)
- Tests the ability to forge e-mail and control any Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP3), and Internet Message Access Protocol Version 4 (IMAP4) server that utilizes the customer's expensive bandwidth for sending external mail blasts
- Prepares documentation

Phase 2: Knowledgeable Penetration

In knowledgeable penetration testing, one has knowledge of the target infrastructure. This testing involves the following:

- IP and Internetwork Packet Exchange (IPX) addressing schemes
- Protocols

- Network/port address translation schemes
- Dial-up information (e.g., users, dial-up numbers, and access methods)
- Internetworking operating system configurations
- Privileged access points
- Detailed external configurations (e.g., ISP and Web hosting)
- Documentation
- Site scan, which includes the following:
 - Network discovery
 - Port scan of all ports identified during the discovery
 - Application scan to identify system services as they pertain to discovered ports
 - Throughput scans of port utilization levels to identify vulnerabilities
 - Documentation
- Remote audit, in which one does the following:
 - Tests the configuration, stability, and vulnerabilities of perimeter defenses, external ISP services, and any other network services acting as conduits through a firewall or proxy
 - Prepares documentation
- Penetration tests, in which one does the following:
 - Attacks and evaluates the physical security of, with intent to penetrate, all items that were identified during the site scan and remote audit
 - Audits the source code for CGI, JavaScript, and ActiveX
 - Initiates ODBC captures (databases)
 - Performs IP flood tests
 - Initiates standard Windows NT, Novell NetWare and Unix IOS cracks
 - Carries out DNS spoofing
 - Initializes sniffer-passive probes to capture traffic
 - Prepares documentation
- IP, mail spoof, and spam tests, in which does the following:
 - Performs penetration attacks to coerce infrastructure equipment into making damaging statements and/or releasing sensitive information (e.g., passwords)
 - Tests the ability to forge e-mail and control any SMTP, POP3, and IMAP4 server that uses the customer's expensive bandwidth for sending external mail blasts
 - Prepares documentation

Phase 3: Internet Security and Services

During phase 3, penetration tests are conducted. They include the following:

- Attacks and evaluates the physical security of, with intent to penetrate, all items that were identified during the site scan and remote audit
- Audits the source code for CGI, JavaScript, and ActiveX
- Initiates ODBC calls from customer-identified databases
- Performs IP, Hypertext Transfer Protocol (HTTP), and Internet Control Message Protocol (ICMP) flood tests
- Carries out DNS spoofing
- Prepares documentation

Phase 4: Dial-up Audit

During a dial-up audit, one does the following:

- Utilizes wardialing to scan for and detect misconfigured dial-ups, and terminal servers (e.g., PCAnywhere, Reachout, and Citrix), as well as any rogue or unauthorized desk modems
- Documents procedures

Phase 5: Local Infrastructure Audit

The local infrastructure audit is a compilation of each section report as a deliverable. It includes the following:

User Problem Report. Includes issues such as slow boot times, file/print difficulty, low bandwidth availability, and spontaneous connection terminations.

Composition of Traffic by Protocol Family. A percentage breakdown by protocol, utilized during the capture period. Each frame is categorized into protocol families. A frame to which more than one protocol applies is categorized according to the highest protocol analyzed. Thus, for example, a Transmission Control Protocol/Internet Protocol (TCP/IP) frame encapsulated within frame relay would be categorized as TCP/IP; all the bytes in the frame would be counted as part of the TCP/IP percentage.

Network Segments/Stations versus Symptoms. A breakdown of the network stations and symptoms found. This breakdown includes the number of errors or symptoms per network. Symptoms that might be detected include the following:

- *Frame freezes*, which indicate a hung application or inoperative station.
- *File retransmission*, which indicates that an entire file or a subset of a file has been retransmitted and is generally caused by an application that does not use the network efficiently.

- *Low throughput*, the calculation of which is based on the average throughput during file transfers.
- *Redirected host*, which indicates that stations are receiving an ICMP redirect message sent by a router or gateway to inform stations that a better route exists or that a better route is not available.

Bandwidth Utilization. Indicates the total bandwidth utilized by stations during the analysis session. From this data, recommendations can be made to increase throughput and productivity.

Phase 6: Wide Area Network Audit

The wide area network (WAN) audit is a compilation of each section report as a deliverable. This compilation incorporates the following:

Internetworking Equipment Discovery. An inventory of current internetworking hardware, including switches, routers, firewalls, and proxies.

Alarms and Thresholds. This function tracks all HTTP, FTP, POP3, SMTP, and Network News Transfer Protocol (NNTP) traffic, as well as custom-defined-site access information, in real time. Other monitored access information includes, in summary form, network load, number and frequency of each user's access, and rejected attempts.

Alarm/Event Logging. Excerpts from the actual log files during the analysis session.

Phase 7: Reporting

The reporting phase is a compilation of each section report as a deliverable. It includes the following:

- Detailed documentation of all findings
- Diagrams or screenshots of each event
- Recommended defense enhancement based on Tiger Team techniques
- List of required or optional enhancements to vulnerabilities in immediate danger

The deliverables for your security analysis should incorporate all the functions outlined in the project review of your analyses phases. Each deliverable should be in the form of a detailed report, divided into parts such as scans, spoofs, spams, floods, audits, penetrations, discoveries, network information, system information, vulnerability assessment, and recommendations for increased network security (required and optional). Time should be allotted for organizing the findings, as doing so will facilitate subsequent remediation steps. You should incorporate findings from vulnerability scanners, such as the Network Associates Inc. (NAI) CyberCop Scanner or Nessus Security Scanner, into the report as well. We'll talk more about these and other scanners later in this book.

Unleashing the Power of Windows, Linux, and Solaris

Before we discuss the specifics of vulnerability and penetration assessment, we'll take a moment to review the minimum requirements and construction of our testing system, or *Tiger Box*. Tiger terminology was derived from a team of security experts. Originally, a *Tiger Team* was a group of paid professionals whose purpose was to penetrate perimeter security and test or analyze the internal security policies of corporations. These people penetrated the security of computer systems, phone systems, safes, and so on, to help companies assess the effectiveness of their security systems and learn how to efficiently revamp their security policies.

More recently, however, a Tiger Team has come to be known as any official inspection or special operations team that is called in to evaluate a security problem. A subset of Tiger Teams comprises professional hackers and crackers who test the security of computer installations by attempting remote attacks via networks or via supposedly secure communication channels. In addition, Tiger Teams are also called in to test programming code integrity. Many software development companies outsource a tiger team to perform stringent dynamic code testing before putting their software on the market. Tiger Teams use what's coined a Tiger Box to provide the necessary tools for revealing potential security weaknesses. A Tiger Box contains tools designed to discover, scan, and in some cases penetrate security vulnerabilities.

The central element of a Tiger Box is the operating system foundation. A first-rate Tiger Box is configured in a multiple-boot configuration setting that includes *NIX and Microsoft Windows operating systems. Currently, Tiger Box utilities for Windows operating systems are not as popular as those for *NIX, but Windows is becoming more competitive in this regard. Originally developed at AT&T Bell Laboratories, Unix, as you probably know, is a powerful operating system used by scientific, engineering, and academic communities. By its nature, Unix is a multiuser, multitasking environment that is both flexible and portable and offers e-mail, networking, programming, text processing, and scientific capabilities. Over the years, two major forms of Unix have evolved, each with numerous vendor variants: AT&T Unix System V and Berkeley Software Distribution (BSD) Unix, developed at the University of California at Berkeley. In addition, to Sun Microsystems Solaris, is Linux, a trendy Unix variant, that is commonly configured on a Tiger Box. Linux offers direct control of the OS command line, including custom code compilation for software stability and flexibility. Linux is customized, packaged, and distributed by many vendors, including the following:

RedHat Linux (www.redhat.com)

Slackware (www.slackware.org)

Debian (www.debian.org)

TurboLinux (www.turbolinux.com)

Mandrake (www.linux-mandrake.com)

SuSE (www.suse.com)

Trinux (www.trinux.org)
MkLinux (www.mklinux.org)
LinuxPPC (www.linuxppc.org)
SGI Linux (www.oss.sgi.com/projects/sgilinux11)
Caldera OpenLinux (www.caldera.com)
Corel Linux (www.linux.corel.com)
Stampede Linux (www.stampede.org)

Tiger Box Components

Step-by-step guidelines for installing and configuring your Tiger Box operating systems are given in Part I. If you are technically savvy and/or if you already have a Tiger Box operating system installed and configured with your Windows and/or *NIX operating systems, you can simply move on to Part II.

Referring back, now, to the multiple operating system proposition: A multiple-boot configuration makes it easy to boot different operating systems on a single Tiger Box. (Note, for simplicity the Windows complement should be installed and configured prior to *NIX.) As of this writing, the Windows versions that are most stable and competent include Windows 2000, Windows 2000 Professional, and Windows 2000 Server. The *NIX flavor regarded as the most flexible and supportive is Red Hat Linux (www.redhat.com) version 7.3/8, and/or Sun Microsystems Solaris 8 (www.sun.com/software/solaris/). The good news is that with the exception of the Microsoft operating system, you can obtain the Linux and Solaris binaries at no charge.

Incidentally, if multiboot third-party products seem to rub you the wrong way, the Red Hat installation, among other variants, offers the option of making a boot disk that contains a copy of the installed kernel and all modules required to boot the system. The boot disk can also be used to load a rescue disk. When it is time to execute Windows, simply reboot the system minus the boot disk, or when you use Linux, simply reboot the system with the boot disk. Inexperienced users may benefit from using a program such as BootMagic (www.powerquest.com/products/index.html) by PowerQuest Corporation for hassle-free, multiple-boot setup with a graphical interface.

Minimum System Requirements

Hardware requirements depend on the intended use of the Tiger Box, such as whether the system will be used for exploit and script programming and whether the system will be used for a network service. Currently, the minimum requirements, to accommodate most scenarios, include the following:

Processor(s). Pentium II+.

RAM. 128 MB.

HDD. 10 GB.

Video. Support for at least a 1,024 × 768 resolution at 16,000 colors.

Network. Dual network interface cards (NICs), at least one of which supports the passive or so-called promiscuous mode. (When an interface is in the promiscuous mode, you would explicitly ask to receive a copy of all packets, regardless of whether they are addressed to the Tiger Box.)

Other. Three-button mouse, CD-ROM, and floppy disk drive.

Part I begins by stepping you through the installation and configuration of a Windows 2000 and Server Tiger Box operating system.