

SECOND EDITION



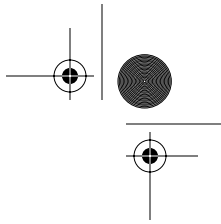
# KNOW YOUR ENEMY

LEARNING ABOUT SECURITY THREATS



CD-ROM  
Included

**The HoneyNet**  
PROJECT



---

# Contents

---

Preface	xix
Foreword	xxvii
<b>PART I</b>	<b>THE HONEYNET</b>
<b>Chapter 1</b>	<b>The Beginning</b>
	The HoneyNet Project
	The Information Security Environment Before HoneyNets
	A Changing Environment: Enter the HoneyPot
	A Growing Group: The HoneyNet Project and GenI HoneyNets
	HoneyNet Challenges
	GenII HoneyNets
	The HoneyNet Research Alliance
	Managing It All: Lessons We've Learned
	Keep It Small
	Make It Fun
	Have Multiple Activities Going on at All Times
	Communicate
	Summary
<b>Chapter 2</b>	<b>HoneyPots</b>
	Definition of HoneyPots
	HoneyPot Advantages and Disadvantages

## CONTENTS

---

Types of Honeyd	21
Low-Interaction Honeyd	21
Low-Interaction Honeyd Example: Honeyd	23
High-Interaction Honeyd	25
High-Interaction Honeyd Example: Symantec Decoy Server	26
Low-Interaction Versus High-Interaction Honeyd	26
Uses of Honeyd	27
Preventing Attacks	28
Detecting Attacks	29
Responding to Attacks	29
Using Honeyd for Research Purposes	30
Summary	30
<b>Chapter 3 Honeyd</b>	<b>33</b>
The Value of a Honeyd	34
The Honeyd Architecture	35
Data Control	37
Data Capture	39
Data Collection	40
Risk	41
Types of Honeyd	44
Summary	45
<b>Chapter 4 GenI Honeyd</b>	<b>47</b>
GenI Honeyd Architecture	48
GenI Options for Data Control	50
GenI Data Control Categories	51
Technology Choices for GenI Data Control	51
GenI Technology in Action	52
GenI Functionality for Data Capture	53
GenI Data Capture Technology Categories	55
Data Capture Technology Review	62
Technology Choices for GenI Data Capture	63
A Complete GenI Honeyd Setup Example	73
Step 1: Obtain and Prepare the Necessary Hardware and Software	76
Step 2: Install and Configure the Firewall Machine	
to Handle Primary Data Control	79
Step 3: Install and Configure the Firewall IDS Machine	
to Handle Primary Data Capture	83

---

**CONTENTS**


---

	Step 4: Install, Configure, and Prepare the Victim (Honeynet) Machine	85
	Step 5: Network the Machines Together and Test the Data Control and Data Capture Systems	88
	Step 6: Connect the Honeynet to the Internet	89
	How It All Works Together: Example Attack Capture	90
	Summary	93
<b>Chapter 5</b>	<b>GenII Honeynets</b>	<b>95</b>
	GenII Honeynet Improvements	95
	GenII Honeynet Architecture	96
	GenII Data Control Overview	97
	GenII Data Capture Overview	98
	GenII Data Control	99
	GenII Data Control Implementation: The Honeywall as a Bridging Gateway	99
	Honeywall Management	101
	IPTables	102
	Snort-Inline and IPTables	106
	The Honeywall Data Control Modes	109
	An Abstract Description of Data Control	118
	Data Capture	120
	Data Capture Layer 1: Firewall Logging	122
	Data Capture Layer 2: IDS	124
	Data Capture Layer 3: Honeypots	128
	GenII Honeynet Deployment	133
	The Topology of the ISLab Honeynet	133
	Honeynet Components	136
	Internet Connection	138
	Honeypots	138
	Remote Syslog Server Honeypot	146
	HNRouter	148
	The Honeywall (Honeynet Gateway)	148
	Summary of the Example ISLab Honeynet Deployment	180
	Summary	180
<b>Chapter 6</b>	<b>Virtual Honeynets</b>	<b>183</b>
	What Is a Virtual Honeynet?	183
	Self-Contained Virtual Honeynets	186
	Hybrid Virtual Honeynets	188

---

## CONTENTS

---

	Possible Implementation Solutions	189
	Option 1: VMware Workstation	190
	Option 2: VMware GSX Server	191
	Option 3: User-Mode Linux	198
	Summary	205
<b>Chapter 7</b>	<b>Distributed Honeynets</b>	<b>207</b>
	What Is a Distributed Honeynet?	208
	Physical Distribution	208
	Honeywall CD-ROM	210
	Deployment Options	211
	Honeypot Farms	212
	The Latency Problem	215
	Setting Up a Honeypot Farm	216
	Technology Review	216
	Honeypot Farm Example Using Linux	218
	Issues Common to All Distributed Honeynets	222
	Summary	223
<b>Chapter 8</b>	<b>Legal Issues</b>	<b>225</b>
	Monitoring Network Users	226
	U.S. Constitutional Provisions	226
	U.S. Statutes	227
	U.S. Contracts and Policies	238
	Laws Outside the U.S.	238
	Crime and the Honeynet	238
	Common Types of Criminal Activity	239
	Protocol for Dealing with Illegal Conduct and Contraband	246
	Entrapment	249
	Do No Harm: Liability to Others	250
	Summary	251
<b>PART II</b>	<b>THE ANALYSIS</b>	<b>253</b>
<b>Chapter 9</b>	<b>The Digital Crime Scene</b>	<b>255</b>
	The Purpose and Value of Data Analysis	255
	Capturing Different Types of Data Within the Honeynet	256
	Firewall Logs	257
	Network Binary Logs	259

## CONTENTS

ASCII SESSION Logs	263
Snort Intrusion Detection Alerts	264
System Logs	268
Keystroke Logs	269
The Multiple Layers of Data Analysis and Their Value	272
Network Forensics	273
Computer Forensics	275
Reverse Engineering	276
Summary	279
<b>Chapter 10 Network Forensics</b>	<b>281</b>
Performing Network Forensics	282
Network Traffic 101	282
The IP Header Through the Analyst's Glasses	283
The TCP Header Through the Analyst's Glasses	285
Capturing and Analyzing Network Traffic	288
Snort Basics	289
A Case Study from the HoneyNet	295
Alerts, One April Morning ...	295
Reconstructing the Attack Session	298
Reconstructing the Rootkit	303
The Follow-Through of the Attack	304
Capturing the IRC Chat	305
Analyzing Nonstandard Protocols	307
Detecting Nonstandard Protocols	307
Common Traffic Patterns for Forensic Analysts	311
The Broadcast Pattern	312
The DNS Reverse Lookup Pattern	313
The Proxy Scanning Pattern	313
The 169.254.x.x Pattern	314
The Traceroute Pattern	314
Passive Fingerprinting	316
A TCP Example of Passive Fingerprinting	318
An ICMP Example of Passive Fingerprinting	320
p0f version 2	324
Summary	325

## CONTENTS

---

<b>Chapter 11</b>	<b>Computer Forensics Basics</b>	<b>327</b>
	Overview	328
	Legal Considerations	329
	The Scientific Method	329
	Data Handling	331
	Key Concepts	332
	Analysis Environment	333
	Hardware Considerations	333
	Linux-Based Analysis System	334
	Linux-Based Analysis Tools	335
	Windows-Based Analysis System	340
	Windows-Based Analysis Tools	341
	Data Acquisition	341
	Concepts	342
	Basic Guidelines	342
	Types of Data	344
	Shutdown Considerations	344
	Acquisition Techniques	344
	Summary	346
<b>Chapter 12</b>	<b>UNIX Computer Forensics</b>	<b>347</b>
	Linux Background	348
	Start-Up	348
	Data Hiding	350
	File Systems	351
	Data Acquisition	357
	Volatile Data Acquisition	357
	Nonvolatile Data Acquisition	359
	Disks and Partitions	363
	The Analysis	366
	Setup	367
	Quick Hits	371
	Filling in the Holes	383
	Readiness Steps	403
	Summary	403
<b>Chapter 13</b>	<b>Windows Computer Forensics</b>	<b>405</b>
	Windows File Systems	406
	FAT Basics	406
	The NTFS File System	408

## CONTENTS

Data Acquisition	412
Volatile Data Acquisition	412
Nonvolatile Data Acquisition	415
Output Options	420
Analysis of the System	422
Establishing Your Setup	422
Viewing the File System Contents	423
Quick Hits	426
Filling in the Holes	430
Analysis with Autopsy and the Sleuth Kit	435
Browsing Files	436
Conducting Keyword Searches	437
File Categorizing	438
File Activity Timelines	439
Recovering Deleted Files	442
Summary	444
<b>Chapter 14 Reverse Engineering</b>	<b>447</b>
Introduction	447
Prerequisites	449
Methods of Analysis	450
Static Analysis	452
Information Gathering	452
Disassembly	456
Symbol Table Regeneration	458
Decompilation Techniques	459
Methodologies for Determining the Order of Decompiling Subroutines	463
Active Analysis	464
Sandboxing the Analysis Environment	464
Black Box Analysis	465
Tracing	466
Antidebugging Tricks	467
Debugging	468
A Walkthrough: The HoneyNet Reverse Challenge	469
Information Gathering	470
Obtaining a Disassembly Listing	473
Decompilation/Analysis	474
Summary	482
Further Reading	483



---

**CONTENTS**

---

<b>Chapter 15</b>	<b>Centralized Data Collection and Analysis</b>	<b>485</b>
	Centralizing Data	486
	Firewall Logs	487
	IDS Logs	489
	tcpdump Logs	490
	System Logs	492
	Keystroke Logs	494
	Data Centralization Summary	496
	The HoneyNet Security Console	497
	Description	497
	Data Correlation Example	497
	Summary	500
<b>PART III</b>	<b>THE ENEMY</b>	<b>503</b>
<b>Chapter 16</b>	<b>Profiling</b>	<b>505</b>
	A Sociological Analysis of the Whitehat/Blackhat Community	506
	Hacker, Cracker, Blackhat, Whitehat: Identity Crisis and the Power of Labels	507
	Motives Within the Community: A Key to Understanding Individuals, Groups, and Their Actions	509
	Section Summary	519
	The Social Structure of the Whitehat/Blackhat Community	520
	Section Summary	530
	“A Bug’s Life”: The Birth, Life, and Death of an Exploit	531
	The Discovery Stage: Finding a Vulnerability	531
	Techniques in Finding Vulnerabilities	532
	The Process of Finding a Vulnerability	533
	The Birth of the Exploit	534
	The Initial Deployment of an Exploit	536
	Exploit Discovery	536
	Parameters that Contribute to Discovery	537
	Life Cycle of an Exploit	538
	A Dangerous Exchange	540
	The Death of an Exploit	541
	Measuring the Risks	541
	Intelligence-Based Information Security: Profiling and Much More	543
	Characteristics of the Event	544
	Consequences of the Event	545

---

CONTENTS

---

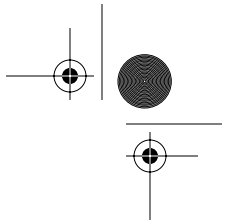
Characteristics of the Blackhat	545
Characteristics of the Target	547
Bringing It All Together	548
Acid Falz	548
IRC Profiling: Another View	551
Summary	556
 <b>Chapter 17 Attacks and Exploits: Lessons Learned</b>	 <b>557</b>
Overview	558
Types of Attacks	558
Active Attacks	560
Who Is Performing Attacks?	562
Common Steps to Exploiting a System	563
Step 1: Active Reconnaissance	563
Step 2: Exploiting the System	565
Step 3: Keeping Access: Backdoors and Trojans	571
Step 4: Covering One's Tracks	572
Summary	574
 <b>Chapter 18 Windows 2000 Compromise and Analysis</b>	 <b>575</b>
Honeypot Setup and Configuration	576
Honeynet Setup and Configuration	576
The Attack Log	578
Day 1: 1 March 2003	578
Day 2: 2 March 2003	581
Day 3: 3 March 2003	582
Day 4: 4 March 2003	584
Day 5: 5 March 2003	587
Attack Log Summary	589
Threat Analysis/Profile	591
Blackhats	591
Warez Traders	592
Carders	592
Spammers	592
Lessons Learned for Defense	593
Lessons Learned About Attackers	593
Summary	594

---

## CONTENTS

---

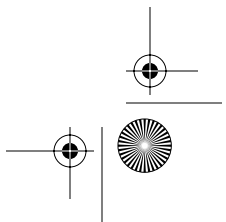
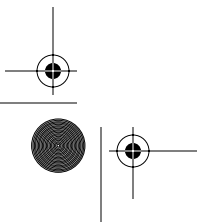
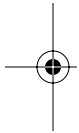
<b>Chapter 19</b>	<b>Linux Compromise</b>	<b>595</b>
	Honeynet Setup and Configuration	596
	Forensics Procedure	597
	Indication of Activity	597
	Evidence Collection	598
	Follow-Through of the Attack	607
	Identifying the Exploits	621
	Examining the Downloaded Packages	624
	The Days After	629
	Event Summary	633
	Summary	634
<b>Chapter 20</b>	<b>Example of Solaris Compromise</b>	<b>635</b>
	Honeynet Setup and Configuration	636
	The Events for Day 1	637
	Detecting the Intrusion	637
	Investigating the Exploit	638
	Reconstructing the Events	644
	Recovering the Intruder's Tools (Day 1)	645
	Recovering the RootKit (Day 1)	646
	Eliminating Competition (Day 1)	650
	Examining IRC Traffic (Day 1)	652
	Locating the Intruder's Denial of Service (DoS) Tool (Day 1)	654
	Day 1 Summary of Events	658
	The Events for Day 3	659
	Examining the DoS Attack (Day 3)	659
	Examining More IRC Traffic (Day 3)	663
	Looking at the SSH Backdoor Access and IPv6 Traffic (Day 3)	666
	The Intruder Setting Up the IPv6 Tunnel (Day 3)	670
	Day 3 Summary of Events	674
	Profiling of the Intruder	674
	Summary	678
<b>Chapter 21</b>	<b>The Future</b>	<b>679</b>
	Distributed Honeynets	680
	Advanced Threats	681
	Insider Threats	681
	Law Enforcement Applications	682



## CONTENTS

---

	Use and Acceptance	682
	Blackhat Response	682
	Summary	683
Appendix A	IPTables Firewall Script	685
Appendix B	Snort Configuration	703
Appendix C	Swatch Configuration	705
Appendix D	Network Configuration Summary	709
Appendix E	Honeywall Kernel Configuration	713
Appendix F	GenII rc.firewall Configuration	717
	Resources and References	721
	About the Authors	737
	Index	743





---

# Foreword

---

## HONEYPOTS AND THE HONEYNET PROJECT

In warfare, information is power. The better you understand your enemy, the more able you are to defeat him. In the war against malicious hackers, network intruders, and the other blackhat denizens of cyberspace, the good guys have surprisingly little information. Most security professionals, even those designing security products, are ignorant of the tools, tactics, and motivations of the enemy. And this state of affairs is to the enemy's advantage.

The Honeynet Project was initiated to shine a light into this darkness. This team of researchers has built an entire computer network and completely wired it with sensors. Then it put the network up on the Internet, giving it a suitably enticing name and content, and recorded what happened. (The actual IP address is not published, and changes regularly.) Hackers' actions are recorded as they happen: how they try to break in, when they are successful, what they do when they succeed.

The results are fascinating. A random computer on the Internet is scanned dozens of times a day. The life expectancy, or the time before someone successfully hacks, a default installation of Red Hat 6.2 server is less than 72 hours. A common home user setup, with Windows 98 and file sharing enabled, was hacked





## FOREWORD

---

five times in four days. Systems are subjected to NetBIOS scans an average of 17 times a day. And the fastest time for a server being hacked: 15 minutes after plugging it into the network.

The moral of all of this is that there are a staggering number of people out there trying to break into *your* computer network, every day of the year, and that they succeed surprisingly often. It's a hostile jungle out there, and network administrators that don't take drastic measures to protect themselves are toast.

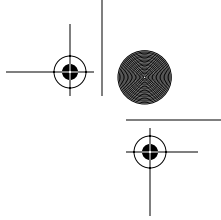
The Honeynet Project is more than a decoy network of computers; it is an ongoing research project into the modus operandi of predatory hackers. The project currently has several honeynets in operation. Want to try this in your own network? Several companies sell commercial, much simpler, versions of what the Honeynet Project is doing. Called "honeypots," they are designed to be installed on an organization's network as a decoy. In theory, hackers find the honeypot and waste their time with it, leaving the real network alone.

This acts as a network alarm. If you are monitoring your network alarms 24/7, or you have a Managed Security Monitoring service, then a honeypot can buy you valuable time to respond to attacks as they happen. The sophisticated attackers will probably avoid the honeypot, but most real-world attackers are amateurs. The key here is real-time monitoring; looking at the log files a week after the fact isn't much use.

For this reason, I am not sold on this as a commercial product. Honeynets and honeypots need to be tended; they're not the kind of product you can expect to work out of the box. Commercial honeypots only mimic an operating system or computer network; they're hard to install correctly and much easier to detect than the Honeynet Project's creations. And the security it buys you is incremental. If you're interested in learning about hackers and how they work, by all means purchase a honeypot and take the time to use it properly. But if you're just interested in protecting your own network, most of the time you'd be better off spending the time on other things.

The Honeynet Project, on the other hand, is pure research. And I am a major fan. The stuff they produce is invaluable, and there's no other practical way to get it.





## FOREWORD

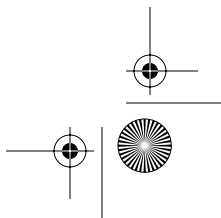
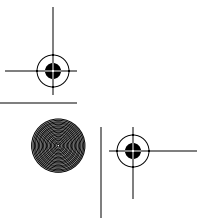
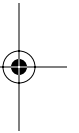
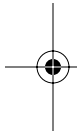
---

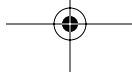
When an airplane falls out of the sky, everyone knows about it. There is a very public investigation, and any airline manufacturer can visit the National Traffic Safety Board and read the multi-hundred page reports on all recent airline crashes. And any airline can use that information to design better aircraft. When a network is hacked, it almost always remains a secret. More often than not, the victim has no idea he's been hacked. If he does know, there is enormous market pressure on him not to go public with the fact. And if he does go public, he almost never releases detailed information about how the hack happened and what the results were.

This paucity of real information makes it much harder to design good security products. This book is a major part of changing that. It talks about how their Honeynet works and how to analyze the data it produces, but is also synthesizes what they've learned so far: the tools, tactics, and motives of the "blackhat community" (i.e., malicious hackers).

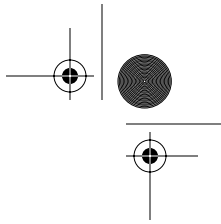
This book is for anyone interested in computer security. Great stuff, and it's all real.

Bruce Schneier  
*<http://www.schneier.com>*









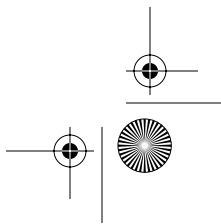
---

# Preface

---

To best defend yourself and to defeat your enemies, you must first understand them: who they are, how they operate, and why. Throughout the ages, countless armies have used this strategy of studying and understanding their enemies in order to defeat them. Just as this strategy was applicable in the days of Julius Caesar, Jan III Sobieski, and Genghis Khan, it can also be applied today in the world of cyberspace. However, whereas enemies of the past may have brandished swords and cannons, today's cyberspace enemies attempt to compromise, steal, or damage information resources using computers and Internet Protocol (IP) packets as their battlefields and weapons.

We all know that computers, networks, software applications, and the Internet have introduced opportunities to the world that no one thought possible. However, as is true with any technology, these same opportunities also carry risks. Whether they are called blackhats, hackers, crackers, disgruntled employees, insiders, or just plain attackers, technology has given these individuals a means to attack almost any resource in the world. While the computer systems and networks we rely on provide us with amazing power, these same systems and networks are static targets: In order to communicate with the rest of the world they must virtually “stay in one spot,” which is a critical vulnerability. Blackhats can launch attacks against these information systems whenever they want, however they want, from wherever they want. In many ways, they have the initiative. No





## PREFACE

---

other technology has held such great potential for constructive purposes while at the same time giving attackers so much power to destroy that same potential. Thus, the Internet has created a global battlefield that spans not only governmental, military, and private enterprise sectors, but also the homes of millions of individual users.

Organizations, businesses, and individual computer owners spend millions of dollars each year to protect their computer resources against these attacks. Virus scanners, firewalls, intrusion detection systems (IDSs), encryption—all of these technologies and techniques are used to protect information systems against attacks. However, the bad guys still succeed, and their success is growing exponentially. One reason for this string of successes is that very few individuals or organizations have taken a step back to better understand who and what the nature of the threats are, how they operate, and why. Only when we are armed with this knowledge, can we better defend against and defeat our enemies.

This book explains the nature of some of these very real threats and gives you the tools and techniques to better learn who your enemies are, how they operate, and why they choose to do so. To do this, we will teach you about “honeynets,” a relatively new security technology made up of networks of systems that are *designed* to be compromised. When attackers break into a honeynet, their every activity, their every keystroke, email, and toolkit is captured, allowing you to see step-by-step how they operate. By learning how to analyze the data honeynets collect, you can better understand who your enemies are and know what you need to do to protect your systems from them.

The first book to discuss honeynets was the first edition of *Know Your Enemy*, written by Honeynet Project members in 2001. This book introduced the concepts of honeynets, how they worked, and how to analyze the information they captured. Since then, radical improvements have been made, not just in honeynet technology, but in deployment concepts and how to analyze the information collected by honeynets. Thus, the second edition of *Know Your Enemy* discusses the advances made since 2001. This new edition covers the older honeynet technologies covered in the first edition—now considered first-generation technologies—in greater detail, offers more examples, and introduces new tools for deploying and maintaining honeynets. Even more exciting, this second edition





discusses new techniques and technologies never published before, including second-generation and distributed honeynets. Most of these new techniques have been tested and deployed by the HoneyNet Project and HoneyNet Research Alliance. The second edition also discusses data analysis in much greater detail, with entire chapters dedicated to Windows forensics, UNIX forensics, reverse engineering, and network forensics. All of this material is based on our experiences, with real-world examples to show you step-by-step all the issues involved.

Perhaps most exciting about the second edition is that each chapter is written by specific members of the HoneyNet Project, HoneyNet Research Alliance, and contributors—people who have developed and deployed the technologies the book discusses in the real world. These are people and organizations who have had their honeynets repeatedly attacked and have learned from their success and failures, and now hope to share their experience with you. We hope you find this book as exciting and fun as we have found our research to be.

## FORMAT OF THE BOOK

The format of this book is very similar to our first edition and is broken down into three main parts:

- **Honeynets, Chapters 1–8:** In the first part, we discuss honeynets—what they are, their value, the different types, and how they work (in excruciating detail). We begin with the history of the HoneyNet Project, then move onto what honeypots and honeynets are, their value, and the issues involved. We then discuss specific honeynet technologies (GenI and GenII) and move on to some more advanced deployments, such as virtual or distributed honeynets.
- **Analysis, Chapters 9–15:** In the second part, we discuss how to analyze the data honeynets collect, including network and disk forensics and data analysis. We attempt to go into as much detail as possible, using real data from a variety of different attacks we have captured.
- **Examples, Chapters 16–20:** In the third part, we cover what we have learned about common threats, using some examples of honeynets we have had compromised.



## PREFACE

---

Finally, in Chapter 21, we finish the book up by discussing the future of this technology, and where it may be headed.

At the end of the book you will find several appendixes detailing configurations and data output from critical tools.

## THE AUDIENCE OF THIS BOOK

Honeynets are used primarily for gathering information on threats. The information they collect has different value to different people, such as identifying insider threats, early warning and prediction, or intelligence gathering on specific new exploits, tools, or threats. This information can also shed light on the attackers themselves, revealing who is launching attacks, how they communicate, and what their motivations are. Thus, this book's target audience is security professionals—individuals who deal with attackers and have to protect their organizations on a daily basis.

Honeynets can capture and analyze information about attackers in both internal and external networks. Thus, in addition to security professionals, other organizations can benefit from this book. Security research organizations and universities can use the material in this book to conduct research on cyber threats using techniques that include content analysis or statistical analysis. Meanwhile, cyber attacks represent a serious threat against the critical information infrastructure of countries and governments, and cyber crime is a new threat law enforcement must deal with on a daily basis, with perpetrators being located all over the globe. Therefore, this book can also help government and law-enforcement organizations better understand and protect themselves against such threats by utilizing honeynets as a tool to identify, counter, and prosecute criminal activity. Military organizations will also find this book valuable, as cyber warfare has become a new, largely not understood, battleground, and honeynets can be deployed as a form of military intelligence. Finally, organizations and legal professionals will find Chapter 8 to be especially interesting, as it is one of the first definitive resources concerning the legal issues of honeynets, written by a member of the United States Department of Justice.





## COMPANION CD-ROM

This book also comes with a companion CD-ROM, providing you with all the tools, materials, source code, and data captures discussed in the book. In addition, this CD provides the documentation, configuration files, and techniques for deploying honeynets, as well as the logs, network captures, and disk images of numerous attacks. Our goal is not just to educate you, but to provide you with the resources you need to gain hands-on experience.

## COMPANION WEB SITE

The book also has a companion Web site (<http://www.honeynet.org/book>) whose purpose is to keep this material updated and to correct any discrepancies or mistakes identified in the book. For example, if any of the URLs mentioned in the book change, the book's Web site will provide you with updated links. In addition, you can visit the Web site to stay up-to-date with the latest in honeynet strategies.

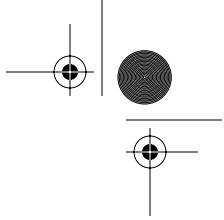
## CHAPTER REFERENCES

At the end of this book you will find a Resources and References section. This section will list, by chapter, all references made by that chapter, and where the reader can find additional information about topics discussed in this book. Examples include Web sites, white papers, and other books.

## NETWORK DIAGRAMS

Throughout this book you'll also find network diagrams demonstrating the deployment of honeynets. To help you better understand all the technologies involved, when possible we use different images for different types of systems. Honeynets consist of two different systems: those that you want to be attacked and those you do not. All production systems are illustrated as simple black and white



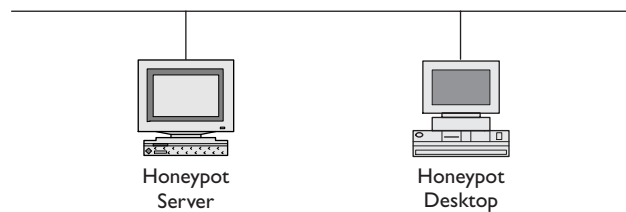


## PREFACE

---



**Figure A** Two black and white production systems deployed on a network. These are systems you do not want to be attacked.



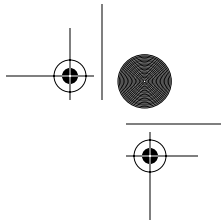
**Figure B** Two shaded honeypots deployed on a network. These are systems you do want to be attacked.

computer objects, as shown in Figure A. These are systems that you *do not* want to be attacked or compromised as they make up the internal architecture of a honeynet or are real-world production systems within an organization. Such systems include firewalls, intrusion detection sensors, and data collection systems.

Systems within honeynets that you *do* want to be attacked are illustrated throughout the book with gray shading going through the system, as shown in Figure B. These systems are referred to as “honeypots.”

## ABOUT THE AUTHORS

As noted earlier, this book was written by members of the HoneyNet Project, HoneyNet Research Alliance, and active contributors. Each chapter was written by the members with the greatest experience in that area. These individuals are security professionals dedicated to learning more about the blackhat community



and sharing the lessons they've learned. Each member brings unique skills and experiences to the table. For example, some members have extensive experience with Windows or UNIX forensics, others in reverse engineering, while still others have expertise in intrusion detection development, firewalls, network architecture, exploit analysis or in fields such as social psychology, statistics, foreign language translation, and profiling. The unique, multidisciplinary approach and expertise of these individuals combine to create an effective team, and we hope a very educational book. You will find the biographies of the authors involved in the creation of each chapter at the end of this book.

## ACKNOWLEDGMENTS

The Honeynet Project, the Honeynet Research Alliance, and this book are the result of the hard work and numerous contributions of the security community. We would like to thank everyone who has helped and contributed to our research. Examples include people volunteering to translate our white papers, people contributing to the Scan of the Month challenges, and developers who have released or tested honeynet-related tools. Unfortunately, we cannot list you all by name, but we know who you are and appreciate your help. Without the community's support and input, our research would have never been possible. Also, we would like to thank the team at Addison-Wesley. Having to deal with one geek writing a book is bad enough. Having to publish for a whole team of dysfunctional geeks is worse. We are especially grateful to Jessica Goldstein, Elizabeth Ryan, Lynda D'Arcangelo, and Shannon Leuma. One last thanks to the security folks at UUnet; Chris, we could not have done it without you!



---

# Index

---

. (dot), data hiding technique, 350, 372–373  
169.254.x.x pattern, 314

## A

ACCEPT action, 103

Accepting packets, 103

ACID (Analysis Console for Intrusion Detection),  
489

Acid Falz, profiling example, 548–551

Ack scans, 287

Active analysis

analysis environment, 464–465

antidebugging tricks, 467–468

black box analysis, 465–466

debugging, 468–469

definition, 450

pros and cons, 451

sandboxing, 464–465

tracing, 466–467

Active attacks, 559–560, 560–561

Active fingerprinting *versus* passive, 317

Active reconnaissance, 563–565

Active users, analyzing, 358–359

Aesthetic jargon, 528

Aleph One, 4

Alerts

IDS, 61–62, 71–73

ISLab example, 177–180

logging, 162–165

network forensics, 295–297

real-time monitoring and alerting, 79

Snort intrusion detection, 264–267

Swatch, 177–180

“An Evening with Bereford,” 5

Analysis Console for Intrusion Detection (ACID),  
489

Analyzing data

*See* computer forensics

*See* data analysis

*See* network forensics

*See* profiling

Anomaly detection, 71–73

Antidebugging tricks, 467–468

Antonomasia, 68

Apache log example, Windows worms, 61

Application-level attacks, 567–568

APUHRP (Azusa Pacific University Honeynet  
Research Project)

attack log, 578–591

attack summary, 589–591

attack timeline, 578–589

attacker profiles, 591–594

blackhats, 591–592

carders, 592

history of, 11

honeynet setup and configuration, 576–578

honeypot setup and configuration, 576

lessons learned, 593–594



## INDEX

- APUHRP, *continued*  
 overview, 575–576  
 spammers, 592–593  
 threat analysis, 591–593  
 warez traders, 592
- Architecture. *See also* data capture; data control.  
 GenI honeynets, 49–50  
 GenII honeynets, 96–97
- Arkin, Ofir, 316, 324
- arp command, 415
- Art jargon, 528
- ASCII SESSION log files, 263–264
- ASCII session logging, 162
- ASR Date: SMART, 340
- Assembly language programming, 483
- Asymmetric routing, 217
- Attackers. *See* hackers.
- Attacks. *See also* exploits.  
 active, 559–560, 560–561  
   analyzing  
     *See* computer forensics  
     *See* data analysis  
     *See* network forensics  
     *See* profiling  
   containing. *See* data control.  
   detecting. *See* alerts; Snort.  
   example scenario, 90–93  
   known, database of, 113–114  
   logging and monitoring  
     *See* data acquisition  
     *See* data capture  
     *See* log files  
     *See* logging  
   passive, 560  
   reconstructing, 608–615  
   types of, 558–561
- Attempted criminal acts, 244–246
- Autopsy case setup, 367–369
- Autopsy Forensic Browser, 337–340, 435–444
- Auto-rooters, 56
- Azusa Pacific University Honeynet Research Project (APUHRP). *See* APUHRP (Azusa Pacific University Honeynet Research Project).
- B**
- Backdoors, 571–572
- Backing up installed honeypots, 197–198
- BackOfficer Friendly, 21–22
- Bad guys. *See* blackhats.
- Balas, Edward, 129
- Banners, 193
- Barnett, Ryan, 194
- bash shell patch, 68
- Basic Honeypot Zone, 135
- Binary logging, 161–162
- Black box analysis, 465–466
- Blackhats  
   definition, 507–509  
   future threats, 682–683  
   profiling, 545–547
- Block pointers, 353–354
- bncs (bouncers), 600–601
- Book reference jargon, 528
- Books and publications. *See also* online resources.  
   “An Evening with Bereford,” 5  
   assembly language programming, 483  
   “Bro: ... Detecting Network Intruders ...”, 71  
   compiler theory, 483  
   *Computer Forensics: Incidence Response Essentials*, 329  
   *The Cuckoo’s Egg*, 5  
   decompilation, 484  
   *Digital Evidence and Computer Crime*, 329  
   exploit coding, 484  
   *Getting Physical with the Digital Investigation Process*, 366  
   *Hacker’s Dictionary*, 507  
   “Honeynet Definitions, Requirements, and Standards,” 37  
   “Honeypot Bandwidth Rate Limitation,” 52  
   “Honeypots: Simple, Cost Effective Detection,” 30  
   *Honeypots: Tracking Hackers*, 31  
   “Honeypotting with VMware: ...”, 194  
   “How to Write Snort Rules,” 289  
   “ICMP Usage in Scanning,” 316  
   “Identifying ICMP Hackery Tools,” 324  
   *Incidence Response and Computer Forensics*, 329  
   *Know Your Enemy*, 10  
   “Know Your Enemy: Motives,” 7  
   “Know Your Enemy: Sebek ...”, 129  
   “Monitoring VMware Honeypots,” 194  
   “Paranoid Penguin: Stealthful Sniffing ... and Logging,” 69

- phrack magazine, 484
  - Practical UNIX and Internet Security*, 4
  - program debugging, 484
  - programming theory, 484
  - Sleuth Kit Informer newsletter, 335
  - “Smashing the Stack for Fun and Profit,” 4
  - “To Build a HoneyPot,” 6
  - “The Use of Honeynets ... Across Large ... Networks,” 35
  - “What are MAC Times?,” 376
  - boot log files, 395
  - Bootable Linux CDs, 360
  - Bootstrapping the honeywall, 153–156
  - Bouncers (bnCs), 600–601
  - Brazilian HoneyNet Project, 11
  - Brenton, Chris, 6
  - Bridge box, 99
  - Bridge utilities, 151
  - Bridging, 99–102, 150–151
  - Bro, 71
  - “Bro: ... Detecting Network Intruders ...”, 71
  - Bro anomaly network IDS, 71–73
  - Broadcast pattern, 312
  - Browsing files, 436–437
  - Buffer overflow, 568
  - Bugs. *See* exploits.
  - burndump, 468
  - burneye, 468
- C**
- Carders, 592
  - Carrier, Brian, 435
  - Case studies. *See* examples.
  - Casey, Eoghan, 329
  - Cause (ideology), hacker motivation, 514–516
  - Chain of Custody, 331
  - Challenge exercises. *See also* examples and case studies.
    - Forensic Challenge, 9
    - history of, 8–9
    - Reverse Challenge, 9
    - Scan of the Month Challenge, 8
    - Scan of the Month Challenge 22, 54
    - Scan of the Month Challenge 28, 80
    - Scan of the Week Challenge, 8
  - Checkpoint Firewall-1, 51–52, 63
  - Cheswick, Bill, 5
  - chkrootkit tool, 358
  - Cluster chains, 406
  - Clusters, 406, 408
  - Command prompt, forensic analysis, 412–413
  - Communication jargon, 528
  - Compiler theory, 483
  - Compiler used, determining, 453
  - Computer forensics, basic. *See also* network forensics; profiling.
    - ASR Date: SMART, 340
    - Autopsy Forensic Browser, 337–340
    - Chain of Custody, 331
    - data acquisition
      - concepts, 342
      - data types, 344
      - dead, 342
      - guidelines, 342–343
      - live, 342
      - netcat tool, 344–345
      - OOV (Order of Volatility), 343
      - shutdown considerations, 344
      - techniques, 344–345
    - data handling, 331–332
    - description, 275–276
    - DFT (ProDiscover Forensics), 341
    - EnCase Forensic, 341
    - file system analysis tools, 335–337
    - FTK (Forensic Toolkit), 341
    - hardware considerations, 333–334
    - hash values, 331–332, 343
    - key concepts, 332–333
    - large file support, 334
    - legal issues, 329
    - Linux-based analysis, 334
    - Linux-based tools, 335–340
    - overview, 328–333
    - preserving the crime scene, 332
    - scientific method, 329–331
    - The Sleuth Kit, 335–337
    - Windows-based analysis, 340–341
    - Windows-based tools, 341
  - Computer Forensics: Incidence Response Essentials*, 329
  - Computer forensics, UNIX. *See also* Greek HoneyNet Project; Solaris compromise.
    - ASR Date: SMART, 340
    - Autopsy Forensic Browser, 337–340

## INDEX

### Computer forensics, *continued*

case studies. *See* Greek HoneyNet Project; Solaris compromise.

#### data acquisition

- active users, 358–359
- chkrootkit tool, 358
- Coroner's Toolkit, 359
- dd tool, 360–361
- dead, 361–363
- device names, identifying, 358
- disk spanning systems, 360
- fdisk tool, 364
- hard disk data, 359–363
- hard disk partitions, 363–366
- hash values, 361
- ils tool, 359
- kstat tool, 358
- live, 362–363
- loopback devices, 363–366
- losetup command, 363–366
- lsuf tool, 358
- mmfs tool, 364–365
- network connections, 358–359
- nonvolatile data, 359–363
- OOV (Order of Volatility), 358
- open files, 358–359
- ps tool, 359
- RAID systems, 360
- running processes, 358–359
- volatile data, 357–359

#### data analysis, detailed

- configuration file analysis, 390–393
- file content analysis, 384–389
- file extensions, 393–394
- file recovery, 396–397
- file types, 393–394
- history files, 393
- keyword searching techniques, 398–402
- lazarus tool, 397
- log files, 394–396
- overview, 383–384
- start-up file analysis, 390–393
- unallocated space, 396–397

#### data analysis, quick hits

- . (dot), data hiding technique, 372–373
- file activity timeline, 376–380
- file integrity verification, 373–376
- hash databases, 380–382

hfind tool, 382

hidden files, 372–373

known bad files, databases of, 380–382

MAC times, 376–380

#### data analysis, setup

- Autopsy case setup, 367–369
- Linux setup, 369
- preparation, 403
- stating the problem, 369–371

#### Linux background

- . (dot), data hiding technique, 350
- block pointers, 353–354
- bootable CDs, 360
- data hiding, 350–351
- direct block pointers, 353–354
- double indirect block pointers, 353–354
- file blocks, 352–353
- file deletion, 355–357
- file fragments, 352–353
- file names, 355
- file systems, 351–357
- file type information, 355
- indirect block pointers, 353–354
- inodes, 353–355
- kernel modules, 349–350
- start-up scripts, 348–349
- store time information, 354–355
- swap space, 357
- triple indirect block pointers, 353–354

#### Linux-based tools, 335–340

#### Computer forensics, Windows

case study. *See* APUHRP (Azusa Pacific University HoneyNet Research Project).

cluster chains, 406

clusters, 406, 408

#### data acquisition

- arp command, 415
- command prompt, 412–413
- Cygwin tool, 413
- dd tool, 415, 417–420
- drive enumeration, 416–417
- first data, 414
- fport tool, 415
- hard disks, wiping clean, 416
- memory information, 415
- netcat tool, 421–422
- netstat tool, 415
- network information, 414–415

- nonnative Windows tools, 413
- nonvolatile data, 415–420
- output options, 420–422
- process information, 414
- PsInfo utility, 414
- PsList utility, 414
- raw image acquisition, 416–417
- sterilizing media, 416
- UnxUtils, 413
- volatile data, 412–415
- volume\_dump program, 416
- wipe program, 416
- data analysis, detailed
  - Autopsy Forensic Browser, 435–444
  - browsing files, 436–437
  - file activity time lines, 439–442
  - file categorizing, 438–439
  - grep command, 430–431
  - hex-based searches, 431
  - keyword searches, 437
  - keyword searching, 430–431
  - kregedit program, 434
  - MAC times, 439–442
  - memory searches, 431–433
  - ntreg tool, 435
  - recovering deleted files, 442–444
  - regedit32.exe program, 434
  - registry analysis, 433–435
  - The Sleuth Kit, 435–444
  - sorter command, 438
  - Sorter tool, 438
  - sorting files, 438–439
  - strings command, 430–431
- data analysis, quick hits
  - file deletion, 427–430, 442–444
  - IIS (Internet Information Server) log files, 426
  - Internet Explorer, 426–427
  - log files, 426
  - Recycle Bin, 427–430
- data analysis, setup
  - analysis environment, 422–423
  - mounting local image files, 423
  - mounting remote shares, 424
  - read-only restrictions, 424–425
  - Samba program, 425
  - viewing file system contents, 423–425
  - virtual hardware write blockers, 424–425
- FAT file system, 406–408
- FAT12 file system, 406–407
- FAT16 file system, 406–407
- FAT32 file system, 406
- file names, 407–408, 410
- file size limitations, 407, 408
- file systems, 406–411
- file timestamps, 409
- meta-data, 407, 408–409
- MFT (Master File Table), 408–409
- nonresident MFT records, 409
- NTFS file system, 408–411
- reserved files, 410–411
- resident MFT records, 409
- sectors, 406
- Windows-based forensic analysis, 340–341
- Computer Fraud and Abuse Act, 239–246
- “Computer trespasser” exception, 236
- Confederated distributed honeynets, 211–212
- Configuration file analysis, 390–393
- Connect scans, 287
- Connection blocking, 51–52
- Connection limiting, 51
- Connection Rate Limiting Mode (CRLM), setting, 152
- Connection tracking, 105–106. *See also* stateful inspection.
- “Consent of a party” exception, 235–236
- Contraband, 246–249
- Copying data
  - See* computer forensics
  - See* data acquisition
  - See* data capture
  - See* data collection
  - See* log files
  - See* logging
  - See* network forensics
- Coroner’s Toolkit, 359
- Covering one’s tracks, 572–574
- Covert monitoring, 60
- Crackers, definition, 507–509
- Crimes by juveniles, 246
- Criminal activity
  - legal issues. *See* legal issues, criminal activity.
  - risk of, 43
- CRLM (Connection Rate Limiting Mode), setting, 152
- cron log files, 395

## INDEX

---

*The Cuckoo's Egg*, 5

Cygwin tool, 413

### D

Damage, limiting. *See* data control.

Data acquisition, basics

*See also* computer forensics

*See also* data capture

*See also* data collection

*See also* log files

*See also* logging

*See also* network forensics

concepts, 342

data types, 344

dead, 342

guidelines, 342–343

live, 342

netcat tool, 344–345

OOV (Order of Volatility), 343

shutdown considerations, 344

techniques, 344–345

Data acquisition, UNIX

active users, 358–359

chkrootkit tool, 358

Coroner's Toolkit, 359

dd tool, 360–361

dead, 361–363

device names, identifying, 358

disk spanning systems, 360

fdisk tool, 364

hard disk data, 359–363

hard disk partitions, 363–366

hash values, 361

ils tool, 359

kstat tool, 358

live, 362–363

loopback devices, 363–366

losetup command, 363–366

lsot tool, 358

mmls tool, 364–365

network connections, 358–359

nonvolatile data, 359–363

OOV (Order of Volatility), 358

open files, 358–359

ps tool, 359

RAID systems, 360

running processes, 358–359

volatile data, 357–359

Data acquisition, Windows

arp command, 415

command prompt, 412–413

Cygwin tool, 413

dd tool, 415, 417–420

drive enumeration, 416–417

first data, 414

fport tool, 415

hard disks, wiping clean, 416

memory information, 415

netcat tool, 421–422

netstat tool, 415

network information, 414–415

nonnative Windows tools, 413

nonvolatile data, 415–420

output options, 420–422

process information, 414

PsInfo utility, 414

PsList utility, 414

raw image acquisition, 416–417

sterilizing media, 416

UnxUtils, 413

volatile data, 412–415

volume\_dump program, 416

wipe program, 416

Data analysis. *See also* computer forensics; network forensics; profiling.

layers

computer forensics, 275–276

network forensics, 273–275

reverse engineering, 276–279

types of data

ASCII SESSION logs, 263–264

firewall logs, 257–259

keystroke logs, 269–272

network binary logs, 259–262

Snort intrusion detection alerts,

264–267

Data capturesystem logs, 268–269

*See also* data acquisition

*See also* data collection

*See also* log files

*See also* logging

*See also* Sebek

*See also* Snort

definition, 36

description, 39

encryption, 39

- GenI example, 83–85, 88–89
- guidelines for, 40
- Data capture, firewalls
  - Checkpoint Firewall-1, 63
  - example, 83–85
  - GenI honeynets, 63–64
  - Linux IPTables Firewall, 63
  - network transaction recording, 63–64
  - OpenBSD PF Firewall, 63
- Data capture, GenI honeynets
  - anomaly detection, 71–73
  - auto-rooters, 56
  - Bro anomaly network IDS, 71–73
  - covert monitoring, 60
  - disk images, 54–55
  - DoS (denial of service) traffic, 57
  - encryption, 58–59
  - flooding, 58–59
  - honeynet attackers, 57
  - host activity recording, 59–61, 65–70
  - IDS alerts, 61–62, 71–73
  - inbound activity, 56–57
  - keystroke logging, 68
  - layered, 53–55, 62–63
  - malicious software, 57
  - multiple intruders, 58
  - mystery traffic, 57
  - network traffic recording, 58–59, 64–65
  - network transaction recording, 55–58, 63–64
  - outbound activity, 57–58
  - previous owner traffic, 57
  - repeat visitors, 56–57
  - script kiddies, 56
  - session tracking, 71–73
  - setup example, 83–85, 88–89
  - spam, 57
  - system software, 58
  - tcpdump, 64–65
  - technology categories, 55–62
  - technology choices, 63–73
  - tools, 53
  - traffic dumps, 59
  - worms, 56
- Data capture, GenII honeynets
  - layers
    - firewall logging, 122–124
    - flow diagram, 121
    - honeypots, 128–133
- IDSs (intrusion detection systems), 124–128
  - intrusion detection, 125
  - network traffic sniffing, 125
  - overview, 120–122
- overview, 98–99
- tools
  - keystroke logger, 129–132
  - Sebek, 129–132
  - system logs, 132–133
- Data collection, 40–41. *See also* data acquisition; data capture; distributed honeynets.
- Data control
  - definition, 36
  - description, 37–38
- Data control, GenI honeynets
  - connection blocking, 51
  - connection limiting, 51
  - examples
    - setting up, 79–83
    - technologies, 52–53
    - testing, 88–89
  - firewalls
    - Checkpoint Firewall-1, 51–52
    - configuring, 49
    - connection blocking, 51–52
    - example, 79–83
    - versus* gateways, 48–49
    - IPTables Firewall, 51, 52
    - Linux IPTables Firewall, 51
    - OpenBSD PF Firewall, 51–52
    - rc.firewall, 51–52
  - guidelines for, 38–39
  - setup example, 79–83, 88–89
  - technology choices, 51–52
- Data control, GenII honeynets
  - bridge box, 99
  - bridging, 99
  - bridging gateways, 99–102
  - description, 118–120
  - Ethernet frames, definition, 99
  - firewalls
    - honeypots, 99–102, 109–118
    - IPTables, 102–106
  - forwarding Ethernet frames, 99–102
  - honeypots
    - implementing, 99–102
    - managing, 101–102
    - revealing, 100

## INDEX

### Data control, *continued*

- honeywalls, data control modes
  - control layers, 109
  - CRLM (Connection Rate Limiting Mode), 110–113
  - IPS layer, 109
  - limiting connection rates, 110–113
  - malicious packets, dropping, 110, 113–116
  - malicious packets, replacing, 110, 116–118
  - network gateway layer, 109
  - PDM (Packet Drop Mode), 110, 113–116
  - PRM (Packet Replace Mode), 110, 116–118
- IPTables, 102–106
- netfilter, 102
- network filtering, 102–106
- packets
  - accepting, 103
  - acting on, 102–106
  - forwarding, 99–102
  - hiding, 130
  - logging, 103
  - malicious, dropping, 110, 113–116
  - malicious, replacing, 110, 116–118
  - queuing, 103
  - rejecting, 103, 108–109
  - from Sebek clients, 130
  - selecting, 102–106
  - TARGETS, 102–106
- packets, filtering with
  - honeywalls, 99–102
  - IPTables, 102–106, 106–109
  - Snort-Inline, 106–109
- rules
  - IPTables, 104–106
  - pseudo-rules, 118–120
  - Snort-Inline, 107–109, 116
- STP (spanning tree protocol), 99–102

Data control modes, GenII honeynets

- control layers, 109
- CRLM (Connection Rate Limiting Mode), 110–113
- IPS layer, 109
- limiting connection rates, 110–113
- malicious packets, dropping, 110, 113–116
- malicious packets, replacing, 110, 116–118
- network gateway layer, 109
- PDM (Packet Drop Mode), 110, 113–116
- PRM (Packet Replace Mode), 110, 116–118

Data hiding, 350–351

Data sanitization, 222

### Databases

- hash values, 375, 380–382
- known attacks, 113–114
- log files. *See* HSC (HoneyNet Security Console); log files, centralizing.
- Solaris Fingerprint Database, 375

dd tool, 360–361, 415, 417–420

de Haas, Job, 10

Dead data acquisition, 342, 361–363

Debugging

- See also* computer forensics
- See also* data capture
- See also* log files
- See also* logging
- See also* network forensics
- See also* profiling
- antidebugging tricks, 468
- books and publications, 484

Deception Toolkit, 28

### Decompilation

- books and publications, 484
- example, 474–481
- order of, 463–464
- techniques, 459–463

Decoy networks. *See* honeynets.

### Deleted files

- forensic analysis, 355–357, 427–430, 442–444
- recovering, 442–444

Deleting directories, 60

Denial of service (DoS). *See* DoS (denial of service).

Deploying exploits, 536

Derogatory jargon, 526

Detecting attacks. *See* alerts; Snort.

Detecting honeynets, 42, 193

Device names, identifying, 358

DFT (ProDiscover Forensics), 341

*Digital Evidence and Computer Crime*, 329

Dike, Jeff, 185

Direct block pointers, 353–354

Directories, deleting, 60

Disabling the honeynet, risk of, 43

Disassembler, fooling, 457–458

Disassembly, 456–458, 473

Discovering exploits, 536–538

Disk spanning systems, forensic analysis, 360

Disks. *See* hard disks.



**Distributed honeynets***See also* data collection*See also* GenI honeynets*See also* GenII honeynets*See also* honeynets*See also* virtual honeynets

confederated model, 211–212

creating a honeynet gateway, 210–211

data loss, 223

data sanitization, 222

definition, 208

deployment drawbacks, 212

deployment options, 211–212

federal model, 212

future of, 680

**honeypot farms**

asymmetric routing, 217

configuring the Ethernet, 222

definition, 212

example, 218–222

hot-zoning, 213–214

IP tunnels, 216, 219–220

latency problem, 215–216

NAT (Network Address Translation), 218, 221

packet mangling, 217–218, 220–221

policy-based routing, 217, 220

pros and cons, 214

protecting production hosts, 213–214

sample diagram, 213

setting up, 216–218

technology choices, 216–218

VLANs (virtual LANS), 217

honeywall CD-ROM, 210–211

issues, 222–223

physical distribution, 208–212

RTT (Round Trip Time), 215

session encryption, 223

size issues, 223

time synchronization, 222–223

time zone synchronization, 41

Dittrich, David, 9, 10

DNS reverse lookup pattern, 313

**DoS (denial of service)**

attack analysis, 659–663

description, 571

legal issues, 240–242

tool detection, 654–658

traffic, capturing, 57

Dot (.), data hiding technique, 350, 372–373

Double indirect block pointers, 353–354

Drive enumeration, 416–417

DROP action, 103

drop action, 108–109

Dynamic linking, 453

**E**

Ego, hacker motivation, 513–514

Elevation of privileges attack, 570

Embedded strings, identifying, 453

EnCase Forensic, 341

Encrypted blackhat connections, 122

**Encryption**

and data capture, 39, 58–59

distributed honeynets, 223

packets, and network forensics, 304

Enemy. *See* hackers.

Entertainment, hacker motivation, 512–513

Entrance to social group, hacker motivation, 516–517

Entrapment, 249–250

**Erasing**

hard disks, 416

the victim machine, 86

Ethereal sniffer, 289

**Ethernet**

configuring for honeypot farms, 222

frames, definition, 99

packets, forwarding, 99–102

Eventlog to syslog utility, 140–141

Examples and case studies. *See also* challenge

exercises; GenI honeynets, setup example.

Apache log, Windows worms, 61

GenII honeynet deployment. *See* ISLab example, GenII deployment.

high-interaction honeypots, 25–27

Honeyd, 23–25

honeynet deployment. *See* ISLab example, GenII deployment.

honeypot farms, 218–222

HSC (HoneyNet Security Console), 497–500

low-interaction honeypots, 23–25

monitoring network users, legal issues, 231–235

passive fingerprinting, 318–324

profiling, 548–556

reverse engineering. *See* HoneyNet Reverse Challenge.

Snort, network forensics, 295–298



## INDEX

- Ethernet, *continued*
  - Snort-based IDSs (intrusion detection systems), 126–128
  - Symantec Decoy Server, 25–27
  - system log, statd attack, 60–61
  - UNIX forensics. *See* Greek HoneyNet Project; Solaris compromise.
  - Windows forensics. *See* APUHRP (Azusa Pacific University HoneyNet Research Project).
- Executable file formats, identifying, 452
- Exploits. *See also* attacks.
  - analyzing
    - See* computer forensics
    - See* data analysis
    - See* network forensics
    - See* profiling
  - birth, 534–536
  - books and publications, 484
  - common steps
    - active reconnaissance, 563–565
    - application-level attacks, 567–568
    - backdoors, 571–572
    - buffer overflow, 568
    - covering one's tracks, 572–574
    - DoS (denial of service), 571
    - elevation of privileges, 570
    - exploiting standardized installation procedures, 569
    - gaining access, 566
    - misconfiguration attacks, 570
    - operating system attacks, 566–567
    - sample program attacks, 569
    - scripts, 569
    - trojans, 571–572
  - containing. *See* data control.
  - corrupting, 540–541
  - death, 541
  - definition, 531
  - deploying, 536
  - detecting. *See* alerts; Snort.
  - discovering, 536–538
  - example, 638–644
  - life cycle, 538–539
  - logging and monitoring
    - See* data acquisition
    - See* data capture
    - See* log files
    - See* logging
  - online resources, 484
  - risk analysis, 541–542
  - source code auditing, 532
  - vulnerability, 531–534
- F**
  - False negatives, 19
  - False positives, 19, 124
  - Farmer, Dan, 9, 335, 376
  - FAT file system, 406–408
  - FAT12 file system, 406–407
  - FAT16 file system, 406–407
  - FAT32 file system, 406
  - fdisk tool, 364
  - Federal distributed honeynets, 212
  - File blocks, forensic analysis, 352–353
  - File extension analysis, 393–394
  - File names
    - FAT, 407–408
    - forensic analysis, 355
    - NTFS, 410
  - File systems
    - analysis tools, 335–337
    - contents, viewing, 423–425
    - forensic analysis, 351–357
    - UML (User-Mode Linux), 201
    - Windows, 406–411
  - Files
    - activity timeline analysis, 376–380, 439–442
    - browsing, 436–437
    - categorizing, 438–439
    - content analysis, 384–389
    - deleted, forensic analysis, 355–357, 427–430, 442–444
    - deleted, recovering, 442–444
    - integrity verification, 373–376
    - recovering, 396–397
    - size limitations, 407–408
    - sorting, 438–439
    - timestamps, 409
    - type analysis, 355, 393–394
  - Filtering
    - packets with
      - honeywalls, 99–102
      - IPTables, 102–106, 106–109
      - Snort-Inline, 106–109
      - Snort output, 291
    - FIN scans, 287

- Fingerprinting
    - UML (User-Mode Linux), 200
    - virtual honeynets, 185
    - VMware GSX Server, 193
  - Fingerprints, default, 204
  - FIRE, 360
  - Firewall log files
    - centralizing, 487–489
    - description, 257–259
    - example, 598–600
  - Firewall logging
    - GenII honeynets, 122–124
    - ISLab example, 153
  - Firewall SQL Import Script (FISQ), 488
  - Firewalls
    - data capture
      - Checkpoint Firewall-1, 63
      - example, 83–85
      - Linux IPTables Firewall, 63
      - network transaction recording, 63–64
      - OpenBSD PF Firewall, 63
    - data control
      - Checkpoint Firewall-1, 51–52
      - configuring, 49
      - connection blocking, 51–52
      - example, 79–83
      - versus* gateways, 48–49
      - IPTables Firewall, 51, 52
      - Linux IPTables Firewall, 51
      - OpenBSD PF Firewall, 51–52
      - rc.firewall, 51–52
    - definition, 17
    - GenI honeynets
      - Checkpoint Firewall-1, 51–52
      - configuring, 49
      - connection blocking, 51–52
      - versus* gateways, 48–49
      - IPTables Firewall, 51, 52
      - Linux IPTables Firewall, 51
      - OpenBSD PF Firewall, 51–52
      - rc.firewall, 51–52
    - GenII honeynets, 99–106, 109–118
    - hardening, 80
    - history of, 4
    - TIS Firewall Toolkit, 4
  - FISQ (Firewall SQL Import Script), 488
  - Flooding, 58–59
  - Forensic Challenge, 9
  - Forensic Toolkit (FTK), 341
  - Forensics. *See* computer forensics; network forensics.
  - Forwarding packets, 99–102
  - Founding members, 6, 7–8
  - Fourth Amendment issues, 226–227
  - fport tool, 415
  - Frames, definition, 99
  - FTK (Forensic Toolkit), 341
  - Future of honeynets
    - advanced threats, 681
    - blackhat response, 682–683
    - distributed honeynets, 680
    - insider threats, 681–682
    - law enforcement applications, 682
    - use and acceptance, 682
  - Fyodor's Nmap Security Scanner, 316
- G**
- Garfinkel, Simson, 4
  - Garner, George
    - dd tool, 415, 417–420
    - volume\_dump program, 416
    - wipe tool, 416
  - Gateways *versus* firewalls, 48–49
  - Gathering information. *See* information gathering.
  - GenI honeynets
    - See also* distributed honeynets
    - See also* GenII honeynets
    - See also* honeynets
    - See also* virtual honeynets
    - architecture, 49–50
    - See also* data capture, GenI
    - See also* data control, GenI
    - history of, 7–8
    - uses for, 44
  - GenII honeynets, setup example
    - attack scenario, 90–93
    - erasing the victim machine, 86
    - firewall machine, data capture, 83–85
    - firewall machine, data control, 79–83
    - firewall machine, hardening, 80
    - hardware requirements, 76–78
    - honeynet machine, 85–88
    - Internet, connecting to, 89
    - Internet connection hardware, 77–78
    - Linux operating system, 78
    - networking, 88–89

## INDEX

- GenI honeynets, *continued*
    - overview, 73–76
    - passwordless authentication, 80
    - process summary, 76
    - sniffer, deploying on the firewall, 80
    - sniffing the honeynet, 83–85
    - Snort network IDS, 78–79
    - software requirements, 78–79
    - stealth interface, 83–85
    - Swatch, 79
    - systems hardware, 76–77
    - utilities, 79
    - victim machine, 85–88
  - GenII honeynets
    - See also* distributed honeynets
    - See also* GenI honeynets
    - See also* honeynets
    - See also* virtual honeynets
    - architecture, 96–97. *See also* data capture, GenII; data control, GenII.
    - deployment example. *See* ISLab example, GenII deployment.
    - encrypted blackhat connections, 122
    - history of, 10
    - improvements over GenI, 95–96, 122, 128–129
    - overview, 98–99
    - uses for, 44–45
  - Getting Physical with the Digital Investigation Process*, 366
  - Glazer, J.D., 6Gettysburg, Civil War battle, 5
  - GMT (Greenwich Mean Time), synchronizing to, 40, 41
  - Good guys. *See* whitehats.
  - Grannick, Jennifer, 8
  - Grayhats, definition, 508
  - Greek Honeynet Project. *See also* Solaris compromise.
    - event summary, 633–634
    - forensic procedure
      - attack follow-through, 607–621
      - blackhat activity, 615–621
      - bncs (bouncers), 600–601
      - evidence collection, 598–607
      - examining downloaded packages, 624–629
      - firewall logs, 598–600
      - identifying exploits, 621–624
      - indication of activity, 597–598
      - locating the attack session, 608–615
      - ptrace vulnerability, 623–624
      - reconstructing the attack session, 608–615
    - Sebek logs, 606–607
    - Snort alerts, 600–601
    - Snort session files, 602–606
    - SSL vulnerability, 622–623
    - system logs, 607
  - history of, 11
  - honeynet setup and configuration, 596–597
  - overview, 595–596
  - post-attack analysis, 629–633
  - Greenwich Mean Time (GMT), synchronizing to, 40, 41
  - grep command, 430–431
- ## H
- Hacker wargames sites, 484
  - Hackers. *See also* profiling.
    - bad guys. *See* blackhats.
    - blackhats, definition, 507–509
    - carders, 592
    - crackers, definition, 507–509
    - definition, 507–509
    - good guys. *See* whitehats.
    - grayhats, definition, 508
    - Jargon File, 509, 526–530
    - motivation
      - cause (ideology), 514–516
      - ego, 513–514
      - entertainment, 512–513
      - entrance to social group, 516–517
      - hacktivism, 514
      - MEECES (Money, Entertainment, Ego, Cause, Entrance, Status), 509–510
      - MICE (Money, Ideology, Compromise, Ego), 509
      - money, 510–512
      - status, 517–519
    - script kiddies, 562
    - social structure
      - aesthetic jargon, 528
      - art jargon, 528
      - book reference jargon, 528
      - characteristics of, 521–522
      - communication jargon, 528
      - derogatory jargon, 526
      - external influences, 524–525
      - history jargon, 527

- humor jargon, 527
- jargon, thematic categories, 526–530
- magic/religion jargon, 527
- mapping the structure, 525–530
- measure jargon, 528
- meritocracy, 522–524
- metasyntactic jargon, 528
- popular reference jargon, 527
- recreation jargon, 528
- self-reference jargon, 527
- social control jargon, 527
- social function jargon, 528
- status jargon, 527
- studying the community, 520–521
- symbol jargon, 528
- technical jargon, 526
- spammers, 592–593
- warez traders, 592
- whitehats, definition, 507–509
- Hacktivism, hacker motivation, 514
- Hard disks
  - erasing, 416
  - forensic analysis, 359–363
  - forensic images, 359–363
  - images, capturing, 54–55
  - partitions, forensic analysis, 363–366
  - wiping clean, 416
- Hardened Honeygot Zone, 135
- Hash Keeper, 375
- Hash values
  - databases of, 375, 380–382
  - examples of, 343
  - online resources, 361
  - uses for, 331–332
  - verifying file integrity, 374
- Heiser, Jay, 329
- Hexadecimal data display, 293–294
- hfind tool, 382
- Hidden files, finding, 372–373
- Hiding
  - packets, 130
  - UML kernel data, 200
- High-interaction honeypots, 25–27
- High-level language characteristics, identifying, 454–456
- High-level language used, identifying, 453
- History files, 393
- History jargon, 527
- HNRouter, 148
- Honeyd, 23–25, 317
- Honeynet Administration Zone, 136
- "Honeynet Definitions, Requirements, and Standards," 37
- Honeynet Project, history of
  - business plan, 12–15
  - challenge exercises, 8–9
  - communication, 14–15
  - founding members, 6, 7–8
  - group size, 12–13
  - Honeynet Research Alliance, 10–12
    - before honeynets, 4–5
    - honeynets, advent of, 7–8
    - honeynets, GenI, 7–8
    - honeynets, GenII, 10
    - honeypots, advent of, 6–7
    - keeping it fun, 13
    - management strategy, 12–15
    - military influence, 4–5
    - multitasking, 13–14
    - security community support, 10–12
    - tools and technique development, 11–12
- Honeynet Project Tools page, 52
- Honeynet Research Alliance, 10–12
- Honeynet Reverse Challenge. *See also* reverse engineering.
  - analysis, 474–481
  - decompilation, 474–481
  - disassembly, 473
  - history of, 9
  - information gathering, 470–473
  - overview, 469–470
- Honeynet Security Console (HSC), 497–500. *See also* log files.
- Honeynets
  - architecture, 35–41
  - benefits of, 34–35
  - central data collection. *See* distributed honeynets.
  - components, 136–138
  - containing attacks. *See* data control.
  - detection by intruders. *See* fingerprinting; latency problem.
  - future of
    - advanced threats, 681
    - blackhat response, 682–683
    - distributed honeynets, 680
    - insider threats, 681–682

## INDEX

### Honeynets, *continued*

- law enforcement applications, 682
- use and acceptance, 682
- history of, 7–8
- logging. *See* data capture; log files; logging.
- monitoring. *See* data capture.
- monitoring several at once. *See* data collection; distributed honeynets.
- multiple on a single computer. *See* virtual honeynets.
- multiple OSs on a single computer. *See* virtual honeynets.
- risks
  - criminal activity, 43
  - customization, 44
  - detection of the honeynet, 42
  - disabling the honeynet, 43
  - harm to a system, 42
  - human monitoring, 43
  - mitigating, 43–44
  - violation, 43
- time zone synchronization, 40
- types of. *See* distributed honeynets; GenI honeynets; GenII honeynets; virtual honeynets.
- “Honeypot Bandwidth Rate Limitation,” 52
- Honeypot farms
  - asymmetric routing, 217
  - configuring the Ethernet, 222
  - definition, 212
  - example, 218–222
  - hot-zoning, 213–214
  - IP tunnels, 216, 219–220
  - latency problem, 215–216
  - NAT (Network Address Translation), 218, 221
  - packet mangling, 217–218, 220–221
  - policy-based routing, 217, 220
  - pros and cons, 214
  - protecting production hosts, 213–214
  - sample diagram, 213
  - setting up, 216–218
  - technology choices, 216–218
  - VLANs (virtual LANS), 217
- HoneyPot Proc FS (hppfs), 200
- Honeypots
  - backing up, 197–198
  - BackOfficer Friendly, 21–22
  - Deception Toolkit, 28
  - definition, 17–18
  - detecting attacks, 29
  - false negatives, 19
  - false positives, 19, 124
  - high interaction, 25–27
  - history of, 6–7
  - infrastructure, 18
  - and IPv6, 20
  - ISLab example
    - configuration, 139
    - eventlog to syslog utility, 140–141
    - keystroke logging, 141–148
    - remote syslog server, 146–148
    - Sebek, 141–146
    - Syslogd, 139–140, 146–148
    - system events logging, 139–141
  - layers, GenII honeynets, 128–133
  - low interaction, 21–25, 26–27
  - preventing attacks, 28
  - pros and cons, 19–21
  - as research tools, 30
  - responding to attacks, 29–30
  - risks, 20–21
  - Specter, 21–22
  - types of, 21–27
  - uses of, 27–30
- “Honeypots: Simple, Cost Effective Detection,” 30
- Honeypots: Tracking Hackers*, 31
- “Honeypotting with VMware: ...”, 194
- Honeytokens, 18
- Honeywall CD-ROM, 210–211
- Honeywalls
  - definition, 36
  - filtering packets, 99–102
  - kernel configuration, 713–715
- Honeywalls, GenII honeynets
  - data control modes
    - control layers, 109
    - CRLM (Connection Rate Limiting Mode), 110–113
    - IPS layer, 109
    - limiting connection rates, 110–113
    - malicious packets, dropping, 110, 113–116
    - malicious packets, replacing, 110, 116–118
    - network gateway layer, 109
    - PDM (Packet Drop Mode), 110, 113–116
    - PRM (Packet Replace Mode), 110, 116–118
  - firewalls, 99–102, 109–118

- implementing, 99–102
  - IPTables, 102–106
  - managing, 101–102
  - revealing, 100
  - Honeywalls, ISLab example
    - alerting, 177–180
    - alerts logging, 162–165
    - ASCII session logging, 162
    - binary logging, 161–162
    - bootstrapping the honeywall, 153–156
    - bridge utilities, 151
    - bridging capability, 150–151
    - CRLM (Connection Rate Limiting Mode), setting, 152
    - data capture, Sebek, 169–176
    - data capture, Snort, 161–169
    - firewall logging, 153
    - installing and configuring, 152–156
    - IPTable updates, 151–152
    - PDM (Packet Drop Mode), disabling, 153
    - PRM (Packet Replace Mode), disabling, 153
    - remote management decisions, 153
    - Sebek traffic, 153
    - sniffing network traffic, 161–165
    - Snort log size, 161–165
    - Snort-Inline data control, 156–162
    - Swatch, 177–180
    - tools and utilities, 150–152
  - Host activity recording, 59–61, 65–70
  - Host system, definition, 184
  - Hot-zoning, 213–214
  - “How to Write Snort Rules,” 289
  - hppfs (HoneyPot Proc FS), 200
  - HSC (Honeynet Security Console), 497–500. *See also* log files.
  - Humor jargon, 527
  - Hybrid virtual honeynets, 188–189
- I**
- ICMP Echo Request Data Payload Content, 322
  - ICMP Echo Request Datagram Size, 322
  - ICMP Echo Request Timestamp, 322
  - ICMP Identification Number Used, 322
  - ICMP packet signatures, 322
  - ICMP Sequence Numbers, 322
  - “ICMP Usage in Scanning,” 316
  - “Identifying ICMP Hackery Tools,” 324
  - Ideology (cause), hacker motivation, 514–516
  - IDS alerts, 61–62, 71–73
  - IDS logs, centralizing, 489–490
  - IDSs (intrusion detection systems), 17, 124–128
  - IIS (Internet Information Server) log files, 426
  - ils tool, 359
  - Inbound activity, monitoring, 56–57
  - Incidence Response and Computer Forensics*, 329
  - Indirect block pointers, 353–354
  - Information gathering
    - attackers. *See* profiling.
    - attacks
      - See* computer forensics
      - See* data acquisition
      - See* data capture
      - See* data collection
      - See* log files
      - See* logging
      - See* network forensics
    - description, 452–456
    - example, 470–473
  - Information security. *See* data control.
  - Information security, history of. *See* Honeynet Project, history of.
  - inodes, forensic analysis, 353–355
  - Internet connection hardware, 77–78
  - Internet connections, 89, 138
  - Internet Explorer, forensic analysis, 426–427
  - Internet Information Server (IIS) log files, 426
  - Internet resources. *See* online resources.
  - Intruders. *See* hackers.
  - Intrusion detection. *See* alerts; Snort.
  - Intrusion detection systems (IDSs), 17, 124–128
  - Intrusions, legal issues, 242–243
  - IP headers, 283–285
  - IP Stack simulation, 317
  - IP tunnels, 216, 219–220
  - ippersonality patch, 317
  - IPTables
    - connection tracking, 105–106
    - filtering packets, 102–106, 106–109
    - GenII honeynets, 102–106
    - packet filtering, 102–105
    - rules, 104–106
    - stateful inspection, 105–106
    - updates, 151–152
  - IPTables Firewall
    - data control, 51, 52
    - script code, 685–702

---

**INDEX**

---

- IPv6
  - and honeypots, 20
  - traffic analysis, 666–670
  - tunnel setup, 670–674
- IRC chats
  - capturing, 305
  - extracting from tcpdump files, 305
  - profiling from, 551–556
  - traffic examination, 652–654, 663–666
- ISLab example, GenII deployment
  - Basic Honeypot Zone, 135
  - Hardened Honeypot Zone, 135
  - HNRouter, 148
  - HoneyNet Administration Zone, 136
  - honeynet components, 136–138
  - honeypots
    - configuration, 139
    - eventlog to syslog utility, 140–141
    - keystroke logging, 141–148
    - remote syslog server, 146–148
    - Sebek, 141–146
    - Syslogd, 139–140, 146–148
    - system events logging, 139–141
  - honeywall
    - alerting, 177–180
    - alerts logging, 162–165
    - ASCII session logging, 162
    - binary logging, 161–162
    - bootstrapping the honeywall, 153–156
    - bridge utilities, 151
    - bridging capability, 150–151
    - CRLM (Connection Rate Limiting Mode), setting, 152
    - data capture, Sebek, 169–176
    - data capture, Snort, 161–169
    - firewall logging, 153
    - installing and configuring, 152–156
    - IPTable updates, 151–152
    - PDM (Packet Drop Mode), disabling, 153
    - PRM (Packet Replace Mode), disabling, 153
    - remote management decisions, 153
    - Sebek traffic, 153
    - sniffing network traffic, 161–165
    - Snort log size, 161–165
    - Snort-Inline data control, 156–162
    - Swatch, 177–180
    - tools and utilities, 150–152
- Internet connections, 138
  - Public Internet Zone, 135
  - topology, 133–136, 137
- J**
  - Jargon, thematic categories, 526–530
  - Jargon File
    - definition, 509
    - thematic analysis, 526–530
  - Journal of Digital Evidence, 366
- K**
  - Kernel modules, forensic analysis, 349–350
  - Keystroke logging
    - centralizing, 494–496
    - forensic analysis, 269–272
    - ISLab example, 141–148
    - UML (User-Mode Linux), 199
    - vulnerability, 68
  - Keyword searching techniques, 398–402
  - Knoppix, 360
  - Knoppix STD, 360
  - Know Your Enemy*, 10
  - “Know Your Enemy: Motives,” 7
  - “Know Your Enemy: Sebek ...”, 129
  - Known bad files, databases of, 380–382
  - Known Goods, 375
  - kregedit program, 434
  - Kruse, Warren, 329
  - kstat tool, 358
  - Kurtz, George, 8
- L**
  - Large file support, 334
  - lastlog log files, 395
  - Latency problem, 215–216
  - Law enforcement involvement, 247–248
  - Laws. *See also* legal issues.
    - outside the U.S., 238
    - Pen Register, Trap and Trace Devices statute, 236–238
    - U.S. Constitution issues, 226–227
    - U.S. contracts and policies, 238
    - U.S. statute issues, 227–238
    - USA Patriot Act, 236
    - Wiretap Act, 228–236
  - Layered data capture, 53–55, 62–63



- Layers, data analysis
  - computer forensics, 275–276
  - network forensics, 273–275
  - reverse engineering, 276–279
- Lazarus tool, 397
- Legal issues
  - computer forensics, 329
  - criminal activity
    - attempted criminal acts, 244–246
    - Computer Fraud and Abuse Act, 239–246
    - contraband, 246
    - crimes by juveniles, 246
    - DoS (denial of service) attacks, 240–242
    - entrapment, 249–250
    - informing victims, 248–249
    - intrusions, 242–243
    - law enforcement involvement, 247–248
    - liability to others, 250–251
    - malicious code, 240–242
    - network crimes, 239–246
    - protected computer, definition, 239
    - protecting other systems, 248
    - protocols for dealing with, 246–249
    - threatening computer damage, 244
    - trafficking in passwords, 244
    - unauthorized access, 243
  - monitoring network users
    - “computer trespasser” exception, 236
    - “consent of a party” exception, 235–236
    - examples, 231–235
    - Fourth Amendment issues, 226–227
    - laws outside the U.S., 238
    - Pen Register, Trap and Trace Devices statute, 236–238
    - privacy laws. *See* Pen Register, Trap and Trace Devices statute; Wiretap Act.
    - “provider protection” exception, 229–231
    - reasonable expectation of privacy, 227
    - search warrants, 226–227
    - U.S. Constitution issues, 226–227
    - U.S. contracts and policies, 238
    - U.S. statute issues, 227–238
    - USA Patriot Act, 236
    - Wiretap Act, 228–236
- Levy, Elias, 8
- Liability to others, 250–251
- Limiting damage. *See* data control.
- Linux. *See also* computer forensics, UNIX.
  - bootable CDs, 360
  - case study. *See* Greek HoneyNet Project.
  - IPTables Firewall, 51, 63
  - setup for forensic analysis, 369
- Linux-based forensic analysis, 334
- Linux-based forensic analysis tools, 335–340
- Live data acquisition, 342, 362–363
- LOG action, 103
- Log files. *See also* HSC (HoneyNet Security Console); monitoring; Sebek.
  - ASCII SESSION, 263–264
  - boot, 395
  - centralizing
    - firewall logs, 487–489
    - IDS logs, 489–490
    - keystroke logs, 494–496
    - overview, 486
    - system logs, 492–494
    - tcpdump logs, 490–492
  - cron, 395
  - firewall, 257–259, 598–600
  - forensic analysis, 394–396, 426
  - IIS (Internet Information Server), 426
  - keystroke, 269–272
  - lastlog, 395
  - maillog, 395
  - messages, 395
  - real-time monitoring and alerting, 79network
    - binary, 259–262
  - Sebek, 606–607
  - secure, 395
  - Snort intrusion detection alerts, 264–267
  - system, 268–269, 607
  - system logs, 607
  - wtmp/wtmpx, 395
  - xferlog, 395
- Logging. *See also* monitoring.
  - alerts, 162–165
  - ASCII sessions, 162, 168
  - attacks. *See* data capture; log files; logging.
  - binary, 161–162, 167
  - firewalls, 122–124, 153
  - honeynets. *See* data capture; log files; logging.
  - keystrokes
    - centralizing, 494–496
    - forensic analysis, 269–272



## INDEX

### Logging. *continued*

- ISLab example, 141–148
- UML (User-Mode Linux), 199
- vulnerability, 68
- packets, 103
- "Paranoid Penguin: Stealthful Sniffing ... and Logging," 69
- system events, 139–141
- TTY, 199
- Loopback devices, 363–366
- losetup command, 363–366
- Low-interaction honeypots, 21–25, 26–27
- lsuf tool, 358

### M

- MAC times, 376–380, 439–442
- Magic/religion jargon, 527
- maillog log files, 395
- Malicious code, 57, 240–242
- Malicious packets, 110, 113–118
- Mandia, Kevin, 329
- Manuals. *See* books and publications.
- Master File Table (MFT), 408–409
- McMillen, Rob, 152
- Measure jargon, 528
- MEECES (Money, Entertainment, Ego, Cause, Entrance, Status), 509–510
- Memory information, forensic analysis, 415
- Mendel, Dion, 9
- Meritocracy, hacker community, 522–524
- messages log files, 395
- Meta-data, 407, 408–409
- Metasyntactic jargon, 528
- Mexico Honeynet Project, 11
- MFT (Master File Table), 408–409
- MICE (Money, Ideology, Compromise, Ego), 509
- Military influence on Honeynet Project, 4–5
- Misaligned code, 457–458
- Misconfiguration attacks, 570
- Mitnick, Kevin, 64
- mmls tool, 364–365
- Money, Entertainment, Ego, Cause, Entrance, Status (MEECES), 509–510
- Money, hacker motivation, 510–512
- Money, Ideology, Compromise, Ego (MICE), 509
- Monitoring. *See also* log files; logging.
  - covert, 60
  - honeynet attackers, 57

### honeynets

- See* data capture
- See* data collection
- See* log files
- See* logging
- malicious software, 57
- multiple intruders, 58
- mystery traffic, 57
- network users, legal issues. *See* legal issues, monitoring network users.
- outbound activity, 57–58
- previous owner traffic, 57
- real-time monitoring and alerting, 79
- repeat visitors, 56–57
- risks, 43
- script kiddies, 56
- spam, 57
- system software, 58
- worms, 56
- "Monitoring VMware Honeypots," 194
- Mounting
  - local image files, 423
  - remote shares, 424
- Multiple dimensional arrays, 455
- Multiple intruders, monitoring, 58
- Mystery traffic, monitoring, 57

### N

- NAT (Network Address Translation), 218, 221
- National Institute of Standards and Technology (NIST), 375
- National Software Reference Library (NSRL), 375
- netcat tool, 344–345, 421–422
- netfilter, 102
- netstat tool, 415
- Network Address Translation (NAT), 218, 221
- Network forensics. *See also* computer forensics; profiling.
  - description, 273–275
  - limitation, 304
  - packet encryption, 304
  - protocols, nonstandard, 307–311
  - protocols, standard, 283
  - uses for, 282
- Network forensics, example
  - alerts, 295–297
  - attack follow-through, 304–305
  - capturing IRC chats, 305–307

- log analysis, 297–298
  - rootkit reconstruction, 303–304
  - session reconstruction, 298–304
  - Network forensics, Snort
    - command line options, 291
    - example, 295–298
    - filtering output, 291
    - hexadecimal data display, 293–294
    - “How to Write Snort Rules,” 289
    - output message details, 290–291
    - packets, capturing, 292–293
    - packets, inspecting, 293–294
    - rootkit reconstruction, 303–304
    - session reconstruction, 294–295, 298–304
    - traffic analysis, 289–295
    - verbose option, 290–291
  - Network forensics, traffic analysis
    - 169.254.x.x pattern, 314
    - Ack scans, 287
    - broadcast pattern, 312
    - capture and analysis, 288–295
    - common patterns, 311–324
    - connect scans, 287
    - DNS reverse lookup pattern, 313
    - Ethereal sniffer, 289
    - FIN scans, 287
    - IP headers, 283–285
    - open ports, determining, 287
    - passive fingerprinting
      - versus* active, 317
      - description, 316–317
      - ICMP example, 320–324
      - ICMP packet signatures, 322
      - p0f tool, 324
      - pros and cons, 317
      - TCP example, 318–320
    - port scans, identifying, 287
    - proxy scanning pattern, 313–314
    - SYN scans, 287
    - TCP headers, 285–286, 288
    - tcpdump, 289
    - traceroute pattern, 314–316
  - Networks. *See also* network forensics.
    - binary log files, 259–262
    - configuration summary, 709–711
    - connections, forensic analysis, 358–359
    - crimes, 239–246
    - decoy. *See* honeynets.
    - filtering, 102–106
    - intrusion detection tools. *See* Snort.
    - traffic analysis. *See* network forensics, traffic analysis.
    - traffic capture tools. *See* tcpdump.
    - traffic recording, 58–59, 64–65
    - traffic sniffing, 125
    - transaction recording, 55–58, 63–64
  - ngrep tool, 63
  - ngrep-like tool, 70
  - NIST (National Institute of Standards and Technology), 375
  - Nonresident MFT records, 409
  - Nonstandard protocols, network forensics, 307–311
  - Nonvolatile data acquisition
    - UNIX forensics, 359–363
    - Windows forensics, 415–420
  - NSRL (National Software Reference Library), 375
  - NTFS file system, 408–411
  - ntreg tool, 435
- 
- 169.254.x.x pattern, 314
  - Online resources. *See also* books and publications.
    - ACID (Analysis Console for Intrusion Detection), 489
    - antidebugging tricks, 468
    - ASR Date: SMART, 340
    - Autopsy Forensic Browser, 339
    - bash shell patch, 68
    - Bro, 71
    - “Bro: ... Detecting Network Intruders ...”, 71
    - burndump, 468
    - burneye, 468
    - chkrootkit tool, 358
    - computer crime laws, 239
    - computer crime prosecutions, 239
    - Computer Fraud and Abuse Act, 239
    - Coroner's Toolkit, 359
    - Cygwin tool, 413
    - data capture tools, 53
    - dd tool, 415
    - DFT (ProDiscover Forensics), 341
    - EnCase Forensic, 341
    - exploit coding, 484
    - FIRE, 360
    - FISQ (Firewall SQL Import Script), 488
    - FTK (Forensic Toolkit), 341

## INDEX

### Online resources, *continued*

Fyodor's Nmap Security Scanner, 316  
*Getting Physical with the Digital Investigation Process*, 366  
 hacker wargames sites, 484  
 hash databases, 382  
 Hash Keeper, 375  
 hash value databases, 375  
 hash values, 361  
 Honeyd, 317  
 HoneyNet Project Tools page, 52  
 "How to Write Snort Rules," 289  
 HSC (HoneyNet Security Console), 497  
 "ICMP Usage in Scanning," 316  
 "Identifying ICMP Hackery Tools," 324  
 IP Stack simulation, 317  
 ippersonality patch, 317  
 IPTables, 102  
 IRC chat capture, 305  
 Journal of Digital Evidence, 366  
 Knoppix, 360  
 Knoppix STD, 360  
 "Know Your Enemy: Sebek ...", 129  
 Known Goods, 375  
 kstat tool, 358  
 Linux bootable CDs, 360  
 log files, real-time monitoring and alerting, 79  
 loopback devices, 366  
 lsof tool, 358  
 netcat tool, 345  
 netfilter, 102  
 ngrep tool, 63  
 ngrep-like tool, 70  
 NIST (National Institute of Standards and Technology), 375  
 NSRL (National Software Reference Library), 375  
 "Paranoid Penguin: Stealthful Sniffing ... and Logging," 69  
 Penguin Sleuth Kit, 360  
 phrack magazine, 484  
 PLAC, 360  
 PsInfo utility, 414  
 PsList utility, 414  
 Red Hat Linux, 78  
 Scan of the Month Challenge 22, 54  
 Scan of the Month Challenge 28, 80  
 The Sleuth Kit, 337  
 Sleuth Kit Informer newsletter, 335

Snort NIDS, 53  
 SnortConfig tool, 116  
 Solaris Fingerprint Database, 375  
 Swatch, 79  
 tcpdump, 53  
 UNFBurninHell, 468  
 UnxUtils, 413  
 U.S. Air Force Office of Special Investigations, 397  
 "What are MAC Times?," 376  
 OOV (Order of Volatility), 343, 358  
 Open files, forensic analysis, 358–359  
 Open ports, determining, 287  
 OpenBSD PF Firewall, 51–52, 63  
 Operating system attacks, 566–567  
 Order of Volatility (OOV), 343, 358  
 Outbound activity, monitoring, 57–58  
 Owning a system, 57

## P

p0f tool, 324  
 Packet Drop Mode (PDM), disabling, 153  
 Packet mangling, 217–218, 220–221  
 Packet Replace Mode (PRM), disabling, 153  
 Packets. *See also* TARGETS.  
   capturing, 292–293  
   encryption, and network forensics, 304  
   inspecting, 293–294  
 Packets, GenII honeynets  
   accepting, 103  
   acting on, 102–106  
   filtering with  
     honeywalls, 99–102  
     IPTables, 102–106, 106–109  
     Snort-Inline, 106–109  
   forwarding, 99–102  
   hiding, 130  
   logging, 103  
   malicious, dropping, 110, 113–116  
   malicious, replacing, 110, 116–118  
   queuing, 103  
   rejecting, 103, 108–109  
   from Sebek clients, 130  
   selecting, 102–106  
   TARGETS, 102–106  
 Papers. *See* books and publications.  
 "Paranoid Penguin: Stealthful Sniffing ... and Logging," 69

- Passive attacks, 560
  - Passive fingerprinting
    - versus* active, 317
    - description, 316–317
    - ICMP example, 320–324
    - ICMP packet signatures, 322
    - p0f tool, 324
    - pros and cons, 317
    - TCP example, 318–320
  - Passwordless authentication, 80
  - Passwords, trafficking in, 244
  - Patriot Act, 236
  - PDM (Packet Drop Mode), disabling, 153
  - Pen Register, Trap and Trace Devices statute, 236–238
  - Penguin Sleuth Kit, 360
  - Pepe, Matt, 329
  - phrack magazine, 484
  - PLAC, 360
  - Policy-based routing, 217, 220
  - Popular reference jargon, 527
  - Port scans, identifying, 287
  - Practical UNIX and Internet Security*, 4
  - Preventing attacks. *See* data control; firewalls.
  - Previous owner traffic, monitoring, 57
  - Privacy, legal issues. *See* legal issues, monitoring network users.
  - PRM (Packet Replace Mode), disabling, 153
  - Problem statement, 369–371
  - Processes, forensic analysis, 358–359, 414
  - ProDiscover Forensics (DFT), 341
  - Profiling
    - Acid Falz example, 548–551
    - blackhat characteristics, 545–547
    - event characteristics, 544–545
    - event consequences, 545
    - example, 674–678
    - with IRC, 551–556
    - overview, 543–544
    - target characteristics, 547–548
  - Program debugging. *See* debugging.
  - Programming theory, 484
  - Prosise, Chris, 329
  - Protected computer, definition, 239
  - Protocols
    - dealing with criminal activity, 246–249
    - network forensics, 283, 307–311
    - “Provider protection” exception, 229–231
  - Provos, Niels, 23
  - Proxy scanning pattern, 313–314
  - ps tool, 359
  - Pseudo-rules, 118–120
  - PsInfo utility, 414
  - PsList utility, 414
  - ptrace vulnerability, 623–624
  - Public Internet Zone, 135
  - Publications. *See* books and publications.
- Q**
- QUEUE action, 103
  - Queuing packets, 103
- R**
- RAID systems, forensic analysis, 360
  - Ranum, Marcus, 4
  - Raw image acquisition, 416–417
  - rc.firewall
    - configuration, 717–719
    - data control, 51–52
    - Snort-Inline, 160
  - Read-only restrictions, forensic environment, 424–425
  - Reasonable expectation of privacy, 227
  - Reconnaissance, 563–565
  - Reconstructing
    - rootkits, 303–304
    - sessions, 294–295
  - Recreation jargon, 528
  - Recycle Bin, forensic analysis, 427–430
  - Red Hat Linux, 78
  - Reed, Darren, 8
  - Reference material. *See* books and publications; online resources.
  - regedit32.exe program, 434
  - Registry analysis, 433–435
  - REJECT action, 103
  - reject action, 108–109
  - Rejecting packets, 103, 108–109
  - Remote management decisions, 153
  - Remote syslog server, 146–148
  - Repeat visitors, monitoring, 56–57
  - Reserved files, 410–411
  - Resident MFT records, 409
  - Responding to attacks, 29–30
  - Reverse Challenge. *See* Honeynet Reverse Challenge.

## INDEX

Reverse engineering. *See also* Honeynet Reverse Challenge.

active analysis

- analysis environment, 464–465
- antidebugging tricks, 467–468
- black box analysis, 465–466
- debugging, 468–469
- definition, 450
- pros and cons, 451
- sandboxing, 464–465
- tracing, 466–467

data analysis layer, 276–279

definition, 447

example. *See* Honeynet Reverse Challenge.

locating code weak spots, 532

methods, 450–452

prerequisites, 448–450

Reverse Challenge, 9

static analysis

- compiler used, 453
- decompilation, example, 474–481
- decompilation, order of, 463–464
- decompilation, techniques, 459–463
- definition, 450
- disassembly, description, 456–458
- disassembly, example, 473
- dynamic linking, 453
- embedded strings, 453
- executable file formats, 452
- fooling the disassembler, 457–458
- high-level language characteristics, 454–456
- high-level language used, 453
- information gathering, description, 452–456
- information gathering, example, 470–473
- misaligned code, 457–458
- multiple dimensional arrays, 455
- pros and cons, 451
- static linking, 453
- string representations, 454–455
- subroutine calling conventions, 455–456
- symbol table regeneration, 458–459
- target architecture/platform, 453

uses for, 448

Reymond, Eric, 507

Risks

- containing. *See* data control.
- criminal activity, 43
- customization, 44

detection of the honeynet, 42

disabling the honeynet, 43

exploits, 541–542

harm to a system, 42

honeypots, 20–21

human monitoring, 43

mitigating, 43–44

violation, 43

virtual honeynets, 185

rm command, 60

Roesch, Marty, 6, 7

Rootkit

chkrootkit tool, 358

reconstruction, 303–304

recovery, 646–650

Round Trip Time (RTT), 215

RST packet flag, 64

RTT (Round Trip Time), 215

Rules

GenII honeynets

IPTables, 104–106

pseudo-rules, 118–120

Snort-Inline, 107–109, 116

IPTables, 104–106

## S

Samba program, 425

Sample program attacks, 569

Scan of the Month Challenge, 8

Scan of the Month Challenge 22, 54

Scan of the Month Challenge 28, 80

Scan of the Week Challenge, 8

Schneier, Bruce, 8

Scientific method, 329–331

Script kiddies, monitoring, 56

Scripts, 569

sdop action, 108–109

Search warrants, 226–227

Searches

hex-based, 431

keyword, 430–431, 437

memory, 431–433

Sebek. *See also* Snort.

configuring, 142–144

installing, 142–144

ISLab example, 141–146

keystroke logs, 269–272

logs, example, 606–607

- packets from clients, 130
  - running, 145
  - source for, 142
  - testing, 145–146
  - traffic, example, 153
- Sectors, 406
- secure log files, 395
- Security community, 10–12
- Security of information. *See* data control.
- Security of information, history of. *See* Honeynet Project, history of.
- Seifried, Kurt, 194
- Selecting packets, 102–106
- Self-contained virtual honeynets, 186–187
- Self-reference jargon, 527
- Session reconstruction, 294–295, 298–304
- Session tracking, 71–73
- Shah, Saumil, 7
- Shimomura, Tsutomu, 64
- Shutdown considerations, forensic analysis, 344
- skas mode, 200
- Skoudis, Ed, 6
- The Sleuth Kit, 335–337, 435–444
- Sleuth Kit Informer newsletter, 335
- Sniffer, deploying on the firewall, 80"Smashing the Stack for Fun and Profit," 4
- Sniffing network traffic, 83–85, 161–165
- Snort. *See also* Sebek.
  - alerts, example, 600–601
  - alerts, logging, 162–165
  - ASCII session logging, 162, 168
  - binary logging, 161–162, 167
  - configuration, 703
  - data capture example, 72
  - example, 78–79
  - intrusion detection alerts log files, 264–267
  - ISLab example, 161–169
  - log size, 161–165
  - network forensics. *See* network forensics, Snort.
  - session files, example, 602–606
  - sniffing network traffic, 161–165
- Snort NIDS, 53
- snort\_fast file, 168
- snort\_full file, 168
- Snort-Inline
  - data control, 156–162
  - database of known attacks, 113–114
  - filtering packets, 106–109
  - and rc.firewall, 160
  - rules, 107–109, 116
  - rules, GenII honeynets, 107–109, 116
  - running at system restart, 160
- Social control jargon, 527
- Social function jargon, 528
- Solaris compromise. *See also* Greek Honeynet Project.
  - event timeline
    - Day 1 event summary, 658–659
    - Day 2 event summary, 659
    - Day 3 event summary, 674
  - DoS (denial of service) attack analysis, 659–663
  - DoS (denial of service) tool detection, 654–658
  - eliminating competition, 650–652
  - event reconstruction, 644–645
  - exploit investigation, 638–644
  - intruder tool recovery, 645–646
  - intrusion detection, 637–638
  - IPv6 traffic analysis, 666–670
  - IPv6 tunnel setup, 670–674
  - IRC traffic examination, 652–654, 663–666
  - rootkit recovery, 646–650
  - SSH backdoor access detection, 666–670
  - honeynet setup and configuration, 636–637
  - intruder profile, 674–678
  - overview, 635
- Solaris Fingerprint Database, 375
- sorter command, 438
- Sorter tool, 438
- Source code auditing, 532
- South Florida Honeynet Project, 11
- Spafford, Gene, 4
- Spam, monitoring, 57
- Spammers, 592–593
- Spanning tree protocol (STP), 99–102
- Specter, 21–22
- Spitzner, Lance, 6, 51
- SSH backdoor access detection, 666–670
- SSL vulnerability, 622–623
- Standard protocols, network forensics, 283
- Start-up file analysis, 390–393
- Start-up scripts, 348–349
- statd attack, system log example, 60–61
- Stateful inspection, 105–106. *See also* connection tracking.

## INDEX

- Static analysis
    - compiler used, 453
    - decompilation, example, 474–481
    - decompilation, order of, 463–464
    - decompilation, techniques, 459–463
    - definition, 450
    - disassembler, fooling, 457–458
    - disassembly, description, 456–458
    - disassembly, example, 473
    - dynamic linking, 453
    - embedded strings, 453
    - executable file formats, 452
    - high-level language characteristics, 454–456
    - high-level language used, 453
    - information gathering, description, 452–456
    - information gathering, example, 470–473
    - misaligned code, 457–458
    - multiple dimensional arrays, 455
    - pros and cons, 451
    - static linking, 453
    - string representations, 454–455
    - subroutine calling conventions, 455–456
    - symbol table regeneration, 458–459
    - target architecture/platform, 453
  - Static linking, 453
  - Status, hacker motivation, 517–519
  - Status jargon, 527
  - Stealth interface, 83–85
  - Sterilizing media, 416
  - Stoll, Cliff, 5
  - Store time information, 354–355
  - STP (spanning tree protocol), 99–102
  - Streams, definition, 259
  - String representations, identifying, 454–455
  - strings command, 430–431
  - Subroutine calling conventions, 455–456
  - Swap space, forensic analysis, 357
  - Swatch
    - configuration, 705–707
    - ISLab example, 177–180
    - online source for, 79
    - uses for, 79
  - Symantec Decoy Server example, 25–27
  - Symbol jargon, 528
  - Symbol table regeneration, 458–459
  - SYN scans, 287
  - Syslogd, 139–140, 146–148
  - System events logging, 139–141
  - System log example, statd attack, 60–61
  - System log files
    - centralizing, 492–494
    - example, 607
    - forensic analysis, 268–269
    - Linux compromise, 607
  - System software, monitoring, 58
- T**
- Target architecture/platform, identifying, 453
  - TARGETS, 102–106. *See also* packets.
  - TCP headers, analyzing, 285–286, 288
  - tcpdump
    - GenI data capture, 64–65
    - logs, centralizing, 490–492
    - online source for, 53
    - traffic analysis, 289
  - Technical jargon, 526
  - Threatening computer damage, 244
  - Time synchronization, 40, 41, 222–223
  - TIS Firewall Toolkit, 4
  - “To Build a Honeypot,” 6
  - Tools and utilities. *See also* HSC (Honeynet Security Console).
    - ACID (Analysis Console for Intrusion Detection), 489
    - alerting. *See* Swatch.
    - antidebugging tricks, 467–468
    - arp command, 415
    - Autopsy Forensic Browser, 337–340, 435–444
    - burndump, 468
    - burneye, 468
    - chkrootkit, 358
    - computer forensics
      - file system analysis tools, 335–337
      - FTK (Forensic Toolkit), 341
      - Linux-based tools, 335–340
      - netcat tool, 344–345
      - Windows-based tools, 341
    - Coroner's Toolkit, 359
    - Cygwin, 413
    - data capture
      - See also* log files
      - See also* logging
      - See also* Snort
      - See also* tcpdump



- keystroke logger, 129–132
    - Sebek, 129–132
    - system logs, 132–133
  - dd, 415, 417–420
  - Deception Toolkit, 28
  - DFT (ProDiscover Forensics), 341
  - disassemblers, 456–458
  - EnCase Forensic, 341
  - Ethereal sniffer, 289
  - file system analysis, 335–337
  - FISQ (Firewall SQL Import Script), 488
  - fport, 415
  - FTK (Forensic Toolkit), 341
  - Fyodor's Nmap Security Scanner, 316
  - grep command, 430–431
  - Honeyd, 23–25, 317
  - Honeynet Project Tools page, 52
  - honeywalls, ISLab example, 150–152
  - IDA Pro, 456–458
  - "Identifying ICMP Hackery Tools," 324
  - keystroke logger, 129–132
  - kregedit program, 434
  - kstat, 358
  - lsof, 358
  - ndisasm, 456–458
  - netcat, 421–422
  - netstat, 415
  - network intrusion detection. *See* Snort.
  - network traffic capture. *See* tcpdump.
  - ngrep, 63
  - ngrep-like, 70
  - nonnative Windows, 413
  - ntreg, 435
  - objdump, 456–458
  - OpenBSD PF Firewall, 51–52
  - p0f, 324
  - passive fingerprinting, 324
  - Penguin Sleuth Kit, 360
  - PsInfo utility, 414
  - PsList utility, 414
  - regedit32.exe program, 434
  - Samba program, 425
  - Sebek, 129–132
  - The Sleuth Kit, 335–337, 435–444
  - SnortConfig, 116
  - Solaris Fingerprint Database, 375
  - Sorter, 438
  - sorter command, 438
  - strings command, 430–431
  - Swatch, 79
  - system logs, 132–133
  - TIS Firewall Toolkit, 4
  - UNFBurninHell, 468
  - UNIX tools for Windows, 413
  - UnxUtils, 413
  - volume\_dump program, 416
  - WinDasm, 456–458
  - wipe program, 416
  - Topology, 133–136, 137
  - Traceroute pattern, 314–316
  - Tracing, 466–467
  - Traffic analysis. *See* network forensics, traffic analysis.
  - Traffic dumps, 59
  - Trafficking in passwords, 244
  - Triple indirect block pointers, 353–354
  - Trojans, 571–572
  - TTY logging, 199
- U**
- UML (User-Mode Linux)
    - building, 200–205
    - confirming setup, 202–205
    - features, 199–200
    - file system, 201
    - fingerprinting, 200
    - fingerprints, default, 204
    - hiding UML kernel data, 200
    - hpps (HoneyPot Proc FS), 200
    - installing, 200–205
    - keystroke logging, 199
    - pros and cons, 198–199
    - skas mode, 200
    - TTY logging, 199
  - Unallocated space analysis, 396–397
  - Unauthorized access, 243
  - UNFBurninHell, 468
  - University of Texas Honeynet Project, 11
  - UNIX forensics. *See* computer forensics, UNIX.
  - UnxUtils, 413
  - U.S. Air Force Office of Special Investigations, 397
  - U.S. Constitution, legal issues, 226–227
  - U.S. contracts and policies, legal issues, 238



---

**INDEX**

---

- U.S. statutes, legal issues, 227–238
- USA Patriot Act, 236
- “The Use of Honeynets ... Across Large ... Networks,” 35
- User-Mode Linux (UML). *See* UML (User-Mode Linux).
- Users, analyzing, 358–359
- V**
  - Venema, Wietse, 9, 335
  - verbose option, Snort, 290–291
  - Victim machines. *See* honeynets; honeypots.
  - Violation of a system, risk of, 43
  - Virtual hardware write blockers, 424–425
  - Virtual honeynets
    - See also* distributed honeynets
    - See also* GenI honeynets
    - See also* GenII honeynets
    - See also* honeynets
    - classic/virtual hybrid, 188–189
    - description, 183–186
    - fingerprinting, 185
    - implementation options, 190–191
    - limitations, 185
    - pros and cons, 185
    - risks, 185
    - self contained, 186–187
  - UML (User-Mode Linux)
    - building, 200–205
    - confirming setup, 202–205
    - features, 199–200
    - file system, 201
    - fingerprinting, 200
    - fingerprints, default, 204
    - hiding UML kernel data, 200
    - hppfs (HoneyPot Proc FS), 200
    - installing, 200–205
    - keystroke logging, 199
    - pros and cons, 198–199
    - skas mode, 200
    - TTY logging, 199
  - VMware ESX Server, 192–193
  - VMware GSX Server
    - backing up installed honeypots, 197–198
    - banners, 193
    - building a virtual honeynet, 194–198
    - detection, 193
    - features, 193–194
    - fingerprinting, 193
    - installing VMware tools, 196–197
    - issues, 193–194
    - pros and cons, 191–192
    - resetting a virtual machine, 194
    - suspending a virtual machine, 194
  - VMware Workstation, 190–191
  - Virtual LANS (VLANs), 217
  - Virtual machines, 194
  - Vision, Max, 7, 305
  - VLANs (virtual LANS), 217
  - VMware ESX Server, 192–193
  - VMware GSX Server
    - backing up installed honeypots, 197–198
    - banners, 193
    - building a virtual honeynet, 194–198
    - detection, 193
    - features, 193–194
    - fingerprinting, 193
    - installing VMware tools, 196–197
    - issues, 193–194
    - pros and cons, 191–192
    - resetting a virtual machine, 194
    - suspending a virtual machine, 194
  - VMware Workstation, 190–191
  - Volatile data acquisition, 357–359, 412–415
  - volume\_dump program, 416
  - Vulnerability, 531–534
- W**
  - warez traders, 592
  - Web resources. *See* online resources.
  - West Point Honeynet Project, 11
  - “What are MAC Times?”, 376
  - Whitehats, definition, 507–509
  - Windows forensics. *See* computer forensics, Windows.
  - Windows worms, log example, 61
  - wipe program, 416
  - Wiretap Act, 228–236
  - Worms, 56, 61
  - wtmp/wtmpx log files, 395
- X**
  - xferlog log files, 395
- Z**
  - Zalewski, Michal, 324