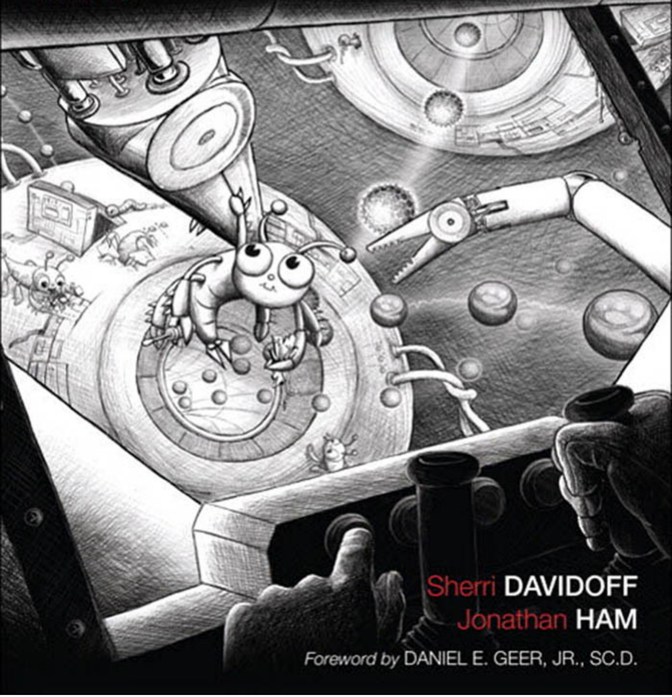
Network Forensics

TRACKING HACKERS THROUGH CYBERSPACE



Sherri DAVIDOFF
Jonathan HAM

Foreword by DANIEL E. GEER, JR., SC.D.

FREE SAMPLE CHAPTER

SHARE WITH OTHERS

www.allitebooks.com

Network Forensics

www.allitebooks.com

This page intentionally left blank

www.allitebooks.com

Network Forensics

Tracking Hackers through Cyberspace

Sherri Davidoff
Jonathan Ham

PRENTICE
HALL

Upper Saddle River, NJ • Boston • Indianapolis • San Francisco
New York • Toronto • Montreal • London • Munich • Paris • Madrid
Capetown • Sydney • Tokyo • Singapore • Mexico City

www.allitebooks.com

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed with initial capital letters or in all capitals.

The authors and publisher have taken care in the preparation of this book, but make no expressed or implied warranty of any kind and assume no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information or programs contained herein.

The publisher offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales, which may include electronic versions and/or custom covers and content particular to your business, training goals, marketing focus, and branding interests. For more information, please contact:

U.S. Corporate and Government Sales
(800) 382-3419
corpsales@pearsontechgroup.com

For sales outside the United States please contact:

International Sales
international@pearson.com

Visit us on the Web: informit.com/ph

Library of Congress Cataloging-in-Publication Data

Davidoff, Sherri.
Network forensics : tracking hackers through cyberspace / Sherri Davidoff, Jonathan Ham.
p. cm.
Includes bibliographical references and index.
ISBN 0-13-256471-8 (hardcover : alk. paper)
1. Computer crimes–Investigation. 2. Computer hackers. 3. Forensic sciences. 4. Computer crimes—Investigation—Case studies. I. Ham, Jonathan. II. Title.
HV8079.C65D348 2012
363.25'968–dc23 2012014889

Copyright © 2012 Pearson Education, Inc.

All rights reserved. Printed in the United States of America. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by an means, electronic, mechanical, photocopying, recording, or likewise. To obtain permission to use material from this work, please submit a written request to Pearson Education, Inc., Permissions Department, One Lake Street, Upper Saddle River, New Jersey 07458, or you may fax your request to (201) 236-3290.

ISBN-13: 978-0-13-256471-7
ISBN-10: 0-13-256471-8
Text printed in the United States on recycled paper at Courier in Westford, Massachusetts.
Second printing, February 2013

www.allitebooks.com

To my mother, father, and sister.
Thank you for teaching me to follow my dreams
and for your endless encouragement and support.

SD

For Charlie and Violet,
and the rest of my family,
whose patience has made this possible.

JH

www.allitebooks.com

“May you find what you seek.”

–*An “ancient curse”; origin unknown.*

www.allitebooks.com

Contents

Foreword **xvii**

Preface **xix**
0.1 The Changing Landscape xix
0.2 Organization xxi
    0.2.1 Part I, “Foundation” xxi
    0.2.2 Part II, “Traffic Analysis” xxii
    0.2.3 Part III, “Network Devices and Servers” xxii
    0.2.4 Part IV, “Advanced Topics” xxiii
0.3 Tools xxiii
0.4 Case Studies xxiii
0.5 Errata xxiv
0.6 Final Notes xxiv

Acknowledgments **xxv**

About the Authors **xxvii**

Part I Foundation **1**

Chapter 1 Practical Investigative Strategies **3**
1.1 Real-World Cases 3
    1.1.1 Hospital Laptop Goes Missing 4
    1.1.2 Catching a Corporate Pirate 6
    1.1.3 Hacked Government Server 7
1.2 Footprints 8
1.3 Concepts in Digital Evidence 9
    1.3.1 Real Evidence 10
    1.3.2 Best Evidence 11
    1.3.3 Direct Evidence 12
    1.3.4 Circumstantial Evidence 12
    1.3.5 Hearsay 13
    1.3.6 Business Records 14
    1.3.7 Digital Evidence 15
    1.3.8 Network-Based Digital Evidence 15

www.allitebooks.com

1.4 Challenges Relating to Network Evidence 16
1.5 Network Forensics Investigative Methodology (OSCAR) 17
  1.5.1 Obtain Information 17
  1.5.2 Strategize 18
  1.5.3 Collect Evidence 19
  1.5.4 Analyze 20
  1.5.5 Report 21
1.6 Conclusion 22

Chapter 2 Technical Fundamentals 23
2.1 Sources of Network-Based Evidence 23
  2.1.1 On the Wire 24
  2.1.2 In the Air 24
  2.1.3 Switches 25
  2.1.4 Routers 25
  2.1.5 DHCP Servers 26
  2.1.6 Name Servers 26
  2.1.7 Authentication Servers 27
  2.1.8 Network Intrusion Detection/Prevention Systems 27
  2.1.9 Firewalls 27
  2.1.10 Web Proxies 28
  2.1.11 Application Servers 29
  2.1.12 Central Log Servers 29
2.2 Principles of Internetworking 30
  2.2.1 Protocols 30
  2.2.2 Open Systems Interconnection Model 31
  2.2.3 Example: Around the World ... and Back 33
2.3 Internet Protocol Suite 35
  2.3.1 Early History and Development of the Internet Protocol Suite 36
  2.3.2 Internet Protocol 37
  2.3.3 Transmission Control Protocol 41
  2.3.4 User Datagram Protocol 43
2.4 Conclusion 44

Chapter 3 Evidence Acquisition 45
3.1 Physical Interception 46
  3.1.1 Cables 46
  3.1.2 Radio Frequency 50
  3.1.3 Hubs 51
  3.1.4 Switches 52
3.2 Traffic Acquisition Software 54
  3.2.1 libpcap and WinPcap 55
  3.2.2 The Berkeley Packet Filter (BPF) Language 55
  3.2.3 tcpdump 59
  3.2.4 Wireshark 64
  3.2.5 tshark 64

www.allitebooks.com

3.2.6 dumpcap 64
3.3 Active Acquisition 65
3.3.1 Common Interfaces 66
3.3.2 Inspection Without Access 70
3.3.3 Strategy 71
3.4 Conclusion 72

Part II Traffic Analysis 73

Chapter 4 Packet Analysis 75
4.1 Protocol Analysis 76
4.1.1 Where to Get Information on Protocols 76
4.1.2 Protocol Analysis Tools 79
4.1.3 Protocol Analysis Techniques 82
4.2 Packet Analysis 95
4.2.1 Packet Analysis Tools 96
4.2.2 Packet Analysis Techniques 99
4.3 Flow Analysis 103
4.3.1 Flow Analysis Tools 105
4.3.2 Flow Analysis Techniques 109
4.4 Higher-Layer Traffic Analysis 120
4.4.1 A Few Common Higher-Layer Protocols 120
4.4.2 Higher-Layer Analysis Tools 129
4.4.3 Higher-Layer Analysis Techniques 131
4.5 Conclusion 133
4.6 Case Study: Ann's Rendezvous 135
4.6.1 Analysis: Protocol Summary 135
4.6.2 DHCP Traffic 136
4.6.3 Keyword Search 138
4.6.4 SMTP Analysis—Wireshark 141
4.6.5 SMTP Analysis—TCPFlow 143
4.6.6 SMTP Analysis—Attachment File Carving 146
4.6.7 Viewing the Attachment 147
4.6.8 Finding Ann the Easy Way 150
4.6.9 Timeline 154
4.6.10 Theory of the Case 155
4.6.11 Response to Challenge Questions 155
4.6.12 Next Steps 157

Chapter 5 Statistical Flow Analysis 159
5.1 Process Overview 160
5.2 Sensors 161
5.2.1 Sensor Types 162
5.2.2 Sensor Software 163
5.2.3 Sensor Placement 164

www.allitebooks.com

5.2.4 Modifying the Environment 165
5.3 Flow Record Export Protocols 166
5.3.1 NetFlow 166
5.3.2 IPFIX 167
5.3.3 sFlow 167
5.4 Collection and Aggregation 168
5.4.1 Collector Placement and Architecture 169
5.4.2 Collection Systems 170
5.5 Analysis 172
5.5.1 Flow Record Analysis Techniques 172
5.5.2 Flow Record Analysis Tools 177
5.6 Conclusion 183
5.7 Case Study: The Curious Mr. X 184
5.7.1 Analysis: First Steps 185
5.7.2 External Attacker and Port 22 Traffic 186
5.7.3 The DMZ Victim—10.30.30.20 (aka 172.30.1.231) 189
5.7.4 The Internal Victim—192.30.1.101 193
5.7.5 Timeline 194
5.7.6 Theory of the Case 195
5.7.7 Response to Challenge Questions 196
5.7.8 Next Steps 196

Chapter 6 Wireless: Network Forensics Unplugged 199
6.1 The IEEE Layer 2 Protocol Series 201
6.1.1 Why So Many Layer 2 Protocols? 201
6.1.2 The 802.11 Protocol Suite 202
6.1.3 802.1X 212
6.2 Wireless Access Points (WAPs) 214
6.2.1 Why Investigate Wireless Access Points? 214
6.2.2 Types of Wireless Access Points 215
6.2.3 WAP Evidence 218
6.3 Wireless Traffic Capture and Analysis 219
6.3.1 Spectrum Analysis 220
6.3.2 Wireless Passive Evidence Acquisition 221
6.3.3 Analyzing 802.11 Efficiently 222
6.4 Common Attacks 224
6.4.1 Sniffing 224
6.4.2 Rogue Wireless Access Points 225
6.4.3 Evil Twin 227
6.4.4 WEP Cracking 228
6.5 Locating Wireless Devices 229
6.5.1 Gather Station Descriptors 229
6.5.2 Identify Nearby Wireless Access Points 229
6.5.3 Signal Strength 231
6.5.4 Commercial Enterprise Tools 233

6.5.5 Skyhook 233
6.6 Conclusion 235
6.7 Case Study: HackMe, Inc. 236
6.7.1 Inspecting the WAP 236
6.7.2 Quick-and-Dirty Statistics 242
6.7.3 A Closer Look at the Management Frames 248
6.7.4 A Possible Bad Actor 250
6.7.5 Timeline 251
6.7.6 Theory of the Case 252
6.7.7 Response to Challenge Questions 253
6.7.8 Next Steps 255

Chapter 7 Network Intrusion Detection and Analysis 257
7.1 Why Investigate NIDS/NIPS? 258
7.2 Typical NIDS/NIPS Functionality 258
7.2.1 Sniffing 259
7.2.2 Higher-Layer Protocol Awareness 259
7.2.3 Alerting on Suspicious Bits 260
7.3 Modes of Detection 261
7.3.1 Signature-Based Analysis 261
7.3.2 Protocol Awareness 261
7.3.3 Behavioral Analysis 261
7.4 Types of NIDS/NIPSs 262
7.4.1 Commercial 262
7.4.2 Roll-Your-Own 263
7.5 NIDS/NIPS Evidence Acquisition 264
7.5.1 Types of Evidence 264
7.5.2 NIDS/NIPS Interfaces 266
7.6 Comprehensive Packet Logging 267
7.7 Snort 268
7.7.1 Basic Architecture 268
7.7.2 Configuration 269
7.7.3 Snort Rule Language 269
7.7.4 Examples 273
7.8 Conclusion 275
7.9 Case Study: Inter0ptic Saves the Planet (Part 1 of 2) 276
7.9.1 Analysis: Snort Alert 277
7.9.2 Initial Packet Analysis 278
7.9.3 Snort Rule Analysis 279
7.9.4 Carving a Suspicous File from Snort Capture 281
7.9.5 “INFO Web Bug” Alert 283
7.9.6 “Tcp Window Scale Option” Alert 284
7.9.7 Timeline 285
7.9.8 Theory of the Case 286
7.9.9 Next Steps 287

Part III Network Devices and Servers 289

Chapter 8 Event Log Aggregation, Correlation, and Analysis 291
8.1 Sources of Logs 292
    8.1.1 Operating System Logs 292
    8.1.2 Application Logs 300
    8.1.3 Physical Device Logs 302
    8.1.4 Network Equipment Logs 305
8.2 Network Log Architecture 306
    8.2.1 Three Types of Logging Architectures 306
    8.2.2 Remote Logging: Common Pitfalls and Strategies 308
    8.2.3 Log Aggregation and Analysis Tools 309
8.3 Collecting and Analyzing Evidence 311
    8.3.1 Obtain Information 311
    8.3.2 Strategize 313
    8.3.3 Collect Evidence 314
    8.3.4 Analyze 316
    8.3.5 Report 317
8.4 Conclusion 317
8.5 Case Study: L0ne Sh4rk's Revenge 318
    8.5.1 Analysis: First Steps 319
    8.5.2 Visualizing Failed Login Attempts 319
    8.5.3 Targeted Accounts 322
    8.5.4 Successful Logins 323
    8.5.5 Activity Following Compromise 324
    8.5.6 Firewall Logs 325
    8.5.7 The Internal Victim—192.30.1.101 328
    8.5.8 Timeline 330
    8.5.9 Theory of the Case 332
    8.5.10 Response to Challenge Questions 332
    8.5.11 Next Steps 333

Chapter 9 Switches, Routers, and Firewalls 335
9.1 Storage Media 336
9.2 Switches 336
    9.2.1 Why Investigate Switches? 337
    9.2.2 Content-Addressable Memory Table 337
    9.2.3 Address Resolution Protocol 338
    9.2.4 Types of Switches 338
    9.2.5 Switch Evidence 340
9.3 Routers 340
    9.3.1 Why Investigate Routers? 341
    9.3.2 Types of Routers 341
    9.3.3 Router Evidence 343
9.4 Firewalls 344

9.4.1 Why Investigate Firewalls? 344
9.4.2 Types of Firewalls 344
9.4.3 Firewall Evidence 347
9.5 Interfaces 348
9.5.1 Web Interface 348
9.5.2 Console Command-Line Interface (CLI) 349
9.5.3 Remote Command-Line Interface 350
9.5.4 Simple Network Management Protocol (SNMP) 351
9.5.5 Proprietary Interface 351
9.6 Logging 352
9.6.1 Local Logging 352
9.6.2 Simple Network Management Protocol 353
9.6.3 syslog 354
9.6.4 Authentication, Authorization, and Accounting Logging 355
9.7 Conclusion 355
9.8 Case Study: Ann's Coffee Ring 356
9.8.1 Firewall Diagnostic Commands 357
9.8.2 DHCP Server Logs 358
9.8.3 The Firewall ACLs 359
9.8.4 Firewall Log Analysis 360
9.8.5 Timeline 364
9.8.6 Theory of the Case 365
9.8.7 Responses to Challenge Questions 367
9.8.8 Next Steps 367

Chapter 10 Web Proxies 369
10.1 Why Investigate Web Proxies? 369
10.2 Web Proxy Functionality 371
10.2.1 Caching 371
10.2.2 URI Filtering 373
10.2.3 Content Filtering 373
10.2.4 Distributed Caching 374
10.3 Evidence 375
10.3.1 Types of Evidence 375
10.3.2 Obtaining Evidence 376
10.4 Squid 377
10.4.1 Squid Configuration 377
10.4.2 Squid Access Logfile 378
10.4.3 Squid Cache 379
10.5 Web Proxy Analysis 381
10.5.1 Web Proxy Log Analysis Tools 381
10.5.2 Example: Dissecting a Squid Disk Cache 384
10.6 Encrypted Web Traffic 392
10.6.1 Transport Layer Security (TLS) 394
10.6.2 Gaining Access to Encrypted Content 396

10.6.3 Commercial TLS/SSL Interception Tools 400
10.7 Conclusion 401
10.8 Case Study: Inter0ptic Saves the Planet (Part 2 of 2) 402
10.8.1 Analysis: pwny.jpg 403
10.8.2 Squid Cache Page Extraction 405
10.8.3 Squid Access.log File 408
10.8.4 Further Squid Cache Analysis 411
10.8.5 Timeline 415
10.8.6 Theory of the Case 417
10.8.7 Response to Challenge Questions 418
10.8.8 Next Steps 419

Part IV Advanced Topics 421

Chapter 11 Network Tunneling 423
11.1 Tunneling for Functionality 423
11.1.1 Background: VLAN Trunking 424
11.1.2 Inter-Switch Link (ISL) 424
11.1.3 Generic Routing Encapsulation (GRE) 425
11.1.4 IPv6 over IPv4 with Teredo 425
11.1.5 Implications for the Investigator 426
11.2 Tunneling for Confidentiality 427
11.2.1 Internet Protocol Security (IPsec) 427
11.2.2 Transport Layer Security (TLS) and Secure Socket Layer (SSL) 428
11.2.3 Implications for the Investigator 430
11.3 Covert Tunneling 430
11.3.1 Covert Tunneling Strategies 430
11.3.2 TCP Sequence Numbers 430
11.3.3 DNS Tunnels 431
11.3.4 ICMP Tunnels 432
11.3.5 Example: ICMP Tunnel Analysis 434
11.3.6 Implications for the Investigator 438
11.4 Conclusion 439
11.5 Case Study: Ann Tunnels Underground 441
11.5.1 Analysis: Protocol Statistics 442
11.5.2 DNS Analysis 443
11.5.3 Quest for Tunneled IP Packets 446
11.5.4 Tunneled IP Packet Analysis 451
11.5.5 Tunneled TCP Segment Analysis 454
11.5.6 Timeline 456
11.5.7 Theory of the Case 456
11.5.8 Response to Challenge Questions 458
11.5.9 Next Steps 459

Chapter 12 Malware Forensics **461**
12.1 Trends in Malware Evolution 462
  12.1.1 Botnets 462
  12.1.2 Encryption and Obfuscation 463
  12.1.3 Distributed Command-and-Control Systems 465
  12.1.4 Automatic Self-Updates 469
  12.1.5 Metamorphic Network Behavior 472
  12.1.6 Blending Network Activity 477
  12.1.7 Fast-Flux DNS 479
  12.1.8 Advanced Persistent Threat (APT) 480
12.2 Network Behavior of Malware 484
  12.2.1 Propagation 485
  12.2.2 Command-and-Control Communications 487
  12.2.3 Payload Behavior 490
12.3 The Future of Malware and Network Forensics 491
12.4 Case Study: Ann's Aurora 492
  12.4.1 Analysis: Intrusion Detection 492
  12.4.2 TCP Conversation: 10.10.10.10:4444–10.10.10.70:1036 495
  12.4.3 TCP Conversations: 10.10.10.10:4445 502
  12.4.4 TCP Conversation: 10.10.10.10:8080–10.10.10.70:1035 508
  12.4.5 Timeline 513
  12.4.6 Theory of the Case 514
  12.4.7 Response to Challenge Questions 515
  12.4.8 Next Steps 516

Afterword **519**

Index **521**

This page intentionally left blank

Foreword

My great-grandfather was a furniture maker. I am writing this on his table, sitting in his chair. His world was one of craft, “the skilled practice of a practical occupation.”[1] He made furniture late in life that was in superficial respects the same as that which he made earlier, but one can see his craft advance.

Cybersecurity’s hallmark is its rate of change, both swift incremental change and the intermittent surprise. In the lingo of mathematics, the cybersecurity workfactor is the integral of a brisk flux of step functions punctuated by impulses. My ancestor refined his craft without having to address a change in walnut or steel or linseed. The refinement of craft in cybersecurity is not so easy.

Forensics might at first seem to be a simple effort to explain the past, and thus an affectation. It is not, and the reason is complexity. Complexity is cumulative and, as the authors say at the outset, enough has accumulated that it is impossible to know everything about even a *de minimus* network. Forensics’ purpose, then, is to discover meaningful facts in and about the network and the infrastructure that were not previously known. Only after those facts are known is there any real opportunity to improve the future.

Forensics is a craft. Diligence can and does improve its practice. The process of forensic discovery is dominated by ruling out potential explanations for the events under study. Like sculpture, where the aim is to chip away all the stone that doesn’t look like an elephant, forensics chips away all the ways in which what was observed didn’t happen. In the terms popularized by EF Schumacher, forensics is a convergent problem where cybersecurity is a divergent one; in other words, as more effort is put into forensics, the solution set tends to converge to one answer, an outcome that does not obtain for the general cybersecurity problem.

Perhaps we should say that forensics is not a security discipline but rather an insecurity discipline. Security is about potential events, consistent with Peter Bernstein’s definition: “Risk is simply that more things can happen than will.” Forensics does not have to induce all the possibilities that accumulated complexity can concoct, but rather to deduce the path by which some part of the observable world came to be as it is. Whereas, in general, cybersecurity the offense has a permanent structural advantage, in forensics it is the defense that has superiority.

That forensics is a craft and that forensics holds an innate strategic advantage are factual generalities. For you, the current or potential practitioner, the challenge is to hone your craft to where that strategic advantage is yours—not just theoretically but in operational reality. For that you need this book.

It is the duty of teachers to be surpassed by their students, but it is also the duty of the student to surpass their teacher. The teachers you have before you are very good; surpassing them will be nontrivial. In the end, a surpassing craft requires knowing what parts of your then current toolbox are eternal and which are subject to the obsolescence that comes with progress. It is likewise expeditious to know what it is that you don't know. For that, this book's breadth is directly useful.

---

[1] “WordNet Search—3.1,” Princeton University, 2011. http://wordnetweb.princeton.edu/perl/webwn.

Because every forensics investigation is, in principle, different, the tools that will be needed for one study may well be a different set from those needed for another study. The best mechanics have all the specialized tools they can need, but may use a few tools far more than others. A collection of tools is only so good as your knowledge of it as a collection of tools, not necessarily that you've used each tool within the last week. Nicholas Taleb described the library of Umberto Eco as an anti-library that "...should contain as much of what you do not know as your financial means, mortgage rates, and the real-estate market allows you to put there."

You, dear reader, hold just such an anti-library of forensics in your hand. Be grateful, and study hard.

Daniel E. Geer, Jr., Sc.D.

Preface

Every day, more bits of data flow across the Internet than there are grains of sand on all the beaches in the world. According to the Cisco Visual Networking Index, the total global IP traffic for 2011 was forecast to be approximately 8.4 * 10^{18} bits per day. Meanwhile, mathematicians at the University of Hawaii have estimated the number of grains of sand on all the beaches in the world to be approximately 7.5 * 10^{18} grains. According to Cisco, global IP traffic is expected to increase at an annual growth rate of 32% per year, so by the time you read this, the number of bits of data flowing across the Internet *every day* may have *far* exceeded the estimated number of grains of sand on all the beaches in the world.[2, 3, 4]

Of course, these estimates are very rough, because in both cases the systems involved are far larger and more complex than humanity has the tools to quantify. The Internet has long since passed the point where we can fully analyze and comprehend its workings. We can understand bits and pieces of it and we can make broad generalizations; but the fact is that we humans have already created a monster far more powerful and complex than we can ever fully understand.

In this environment a new, endless field of study has emerged: network forensics. Forensics, in general, is “the application of scientific knowledge to legal problems, especially scientific analysis of physical evidence, as from a crime scene.” Network forensics therefore refers to the scientific study of network-based evidence, commonly applied to legal questions. Of course, network forensics is a field of study independent of any specific legal case, and many of the scientific advances, tools, and techniques developed for the purposes of legal investigation can also be used for social study, historical analysis, and scientific exploration of network environments. In this book, we have endeavored to provide a technical foundation that will be practically useful not just for professional network forensic analysts conducting legal investigations, but also for students, independent researchers, and all those who are curious.

0.1 The Changing Landscape

The Internet is constantly changing. As new features are developed in hardware and software, new protocols are implemented to reflect those changes and old protocols are adapted and revised to suit the latest technology. In the past decade, we have seen the emergence of

---

[2] Cisco estimates the total global IP traffic for 2011 at 28,023 petabytes per month. Dividing this by 30 days in one month, we get approximately 8.4 * 10^{18} bits each day.

[3] “Networking Solutions,” *Cisco*, http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360_ns827_Networking_Solutions_White_Paper.html.

[4] Howard McAllister, “Grains of Sand on the World’s Beaches,” 21st Century Problem Solving, http://www.hawaii.edu/suremath/jsand.html.

protocols for distributed peer-to-peer video chat systems, protocols for conducting surgery on people from thousands of miles away, and protocols for driving robots halfway around the world.

Network forensics can seem daunting to investigators who are familiar with traditional filesystem forensics. There are a relatively small number of filesystem formats in widespread use, compared with hundreds of network protocols. On Windows systems, it is common to find FAT32 or NTFS filesystems. On UNIX/Linux systems, it is common to find ext2, 3, 4, ZFS, or HFS Plus filesystems. In contrast, on any given network you may find Ethernet, 802.11b/g/n, ARP, DHCP, IPv4, IPv6, TCP, UDP, ICMP, TLS/SSL, SMTP, IMAP, DNS, HTTP, SMB, FTP, RTP, and many other protocols.

On the Internet, there is no guarantee that any protocol you encounter will match the documented specifications, or that there are any published specifications in the first place. Furthermore, implementations of protocols change frequently. Manufacturers are not required to adhere to any standards, and so they implement the protocol as suits them best for any particular revision of software or hardware.

Sometimes protocols are developed before their time, and the applications that are built on top of them have not matured to the point where they can support all the features of the protocol. In the interim, the protocol or specific fields within it may go unused, or may be repurposed by vendors, standards committees, or hackers for other uses. Sometimes protocols are replaced because the environment has changed so much that the old protocols no longer work as intended. A perfect example of this is IPv4, which worked well in its original, relatively small environment. IPv4 is designed with 32-bit fields to store the source and destination addresses, which accomodates 2^{32}, or approximately 4.3 billion unique addresses. However, large segments of this address space were allocated to different organizations in the early years of the Internet, when there were relatively few users. Now that over a billion people have connected to the Internet, the 32-bit address space is too limited to accommodate the demand. As a result, IPv6 was developed with a far larger 128-bit address space (2^{128}, or 3.4×10^{38} unique addresses). With it have emerged even more protocols, such as Teredo (which is designed to tunnel IPv6 traffic over IPv4-only networks).

Forensics tools also go through changes and revisions as the protocols change. A tool made in 2010 may not correctly interpret a packet capture from 2002, or vice versa. Sometimes the errors can be very subtle, perhaps even undetectable. It is very important for investigators to understand how forensic tools work, and be capable of going down to the lowest layers to verify findings. Network forensics professionals must be highly skilled, highly motivated, and have a great deal of expertise because you can't always rely on tools that other people have written in order to correctly interpret results and perhaps even testify in court.

Compounding these issues is the overwhelming variety of network devices, including routers, switches, application servers, and more. Each system on any given network may have unique configuration, interface, and capabilities. It is not possible for investigators to be familiar with all network devices—or even a significant percentage—including current and past makes and models. Instead, network investigators must be prepared to research and learn about equipment on the fly, while at the same time managing an investigation and projecting an air of confidence. It's a fine balancing act.

Tracking down devices to examine them in the first place can be difficult or even impossible. Anonymity has been a hallmark of the Internet since its early days. While it may be possible to track an IP address down to a remote ISP, it is often impossible to wrangle any further identifying details out of a third party—particularly if they are located in a foreign country with lax information security laws. Even when the device is located inside your own organization, tracking it down depends on the quality of network documentation and logs, which are often not granular enough to suit investigative needs. With the rise of mobile networks, tracking down devices often feels like a game of hide-and-seek, where the mobile user (perhaps even unknowingly) has the upper hand.

The point is that the Internet functions as an ecosystem. It is not controlled by central forces or "designed" in the way one designs a car. When you examine network traffic, there is no telling what you may encounter or whether your tools will properly parse the specific versions of the protocols in your packet capture. When you gather evidence from network devices or reconfigure them, you may have to research the specific make and model to properly understand the interfaces and the sources of evidence. When you track down systems, you may have to wander all over creation chasing a mobile device, or call dozens of ISP contacts and law enforcement officials in multiple countries to pinpoint the source.

There is no specification to which manufacturers are universally bound to adhere, no set of rules that users around the globe must follow, and no manual that can tell you precisely how to conduct your investigation.

0.2 Organization

This book is designed to provide you with a broad overview of the most important topics in network forensics. It is divided into four parts: "Foundation," "Traffic Analysis," "Network Devices and Servers," and "Advanced Topics."

0.2.1 Part I, "Foundation"

Part I, "Foundation," covers the basic concepts of evidence handling, networking, and evidence acquisition. This provides a foundation for more advanced topics, which we cover later in the book. In addition to the topics in these chapters, we strongly recommend that all readers have a good understanding of TCP/IP networking. *TCP/IP Illustrated* by W. Richard Stevens is a fantastic book that we highly recommend as a reference.

Part I includes the following chapters:

- Chapter 1, "Practical Investigative Strategies," presents a myriad of challenges faced by network forensic investigators, introduces important concepts in digital evidence, and lays out a methodology for approaching network-based investigations.
- Chapter 2, "Technical Fundamentals," provides a technical overview of common networking components and their value for the forensic investigator, and presents the concept of protocols and the OSI model in the context of network forensic investigations.
- Chapter 3, "Evidence Acquisition," dives into passive and active evidence acquisition, including hardware and software used for sniffing traffic, as well as strategies for actively collecting evidence from network devices.

0.2.2 Part II, “Traffic Analysis”

Part II, “Traffic Analysis,” discusses the many ways that investigators can analyze network traffic. We begin with packet analysis, from examination of protocol headers to payload extraction and reconstruction. Since flow record data retention is becoming commonplace, we subsequently devote a full chapter to statistical flow record analysis. This is followed by an in-depth look at wireless networks and the 802.11 protocol suite. Finally, we discuss network intrusion detection and prevention systems, which are designed to analyze traffic in real time, produce alerts, and in some cases capture packets on the fly.

Part II includes the following chapters:

- Chapter 4, “Packet Analysis,” is a comprehensive study of protocols, packets, and flows, and methods for dissecting them.
- Chapter 5, “Statistical Flow Analysis,” presents the increasingly important field of statistical flow record collection, aggregation, and analysis.
- Chapter 6, “Wireless: Network Forensics Unplugged,” discusses evidence collection and analysis of wireless networks, specifically focusing on the IEEE 802.11 protocol suite.
- Chapter 7, “Network Intrusion Detection and Analysis,” is a review of network intrusion prevention and detection systems, which are specifically designed to produce security alerts and supporting evidence.

0.2.3 Part III, “Network Devices and Servers”

Part III, “Network Devices and Servers,” covers evidence acquisition and analysis from all kinds of network devices. We begin by discussing event log collection and examination, including challenges and benefits relating to different types of logging architectures. Next, we specifically talk about forensic investigation of switches, routers, and firewalls, which make up the backbone of our networks. Since web proxies have exploded in popularity, and often contain enormously valuable evidence, we close with a detailed discussion of web proxy evidence collection and analysis.

Part III includes the following chapters:

- Chapter 8, “Event Log Aggregation, Correlation, and Analysis,” discusses collection and analysis of logs from various sources, including operating systems of servers and workstations (such as Windows, Linux, and UNIX), applications, network equipment, and physical devices.
- Chapter 9, “Switches, Routers, and Firewalls,” studies the evidence that can be gathered from different types of network equipment and strategies for collecting it, depending on available interfaces and level of volatility.
- Chapter 10, “Web Proxies,” reviews the explosion of web proxies and how investigators can leverage these devices to collect web surfing histories and even cached copies of web objects.

0.2.4 Part IV, "Advanced Topics"

Part IV, "Advanced Topics," includes a discussion of two of the most fascinating topics in network forensics: network tunneling and malware. We review legitimate and covert network tunnels and discuss investigative strategies for dealing with different types of tunnels. To close, we review a history of malware and the corresponding impact on forensic analysis, including the evolution of command-and-control channels, botnets, IDS/IPS evasion, and the advanced persistent threat (APT).

Part IV includes the following chapters:

- Chapter 11, "Network Tunneling," discusses both legitimate and covert network tunnels, methods for recognizing tunnels, and strategies for recovering evidence from tunneled traffic.
- Chapter 12, "Malware Forensics," is a condensed history of malware development, including the evolution of command-and-control channels, botnets, IDS/IPS evasion, and the advanced persistent threat. Along the way, we discuss how malware has changed—and been changed by—forensic investigations.

0.3 Tools

This book is designed to be accessible to a wide audience, and to teach you the fundamental principles and techniques of network forensics. There are many commercial, point-and-click tools that you can also use to reach the same answers, and we briefly touch on a few of those in this book. However, our focus is on tools that are freely available and that can be used to illustrate fundamental techniques. In this way, we hope to give you the ability to understand how forensic tools work at a low level, verify results gleaned from automated tools, and make educated decisions when selecting tools for your investigations.

0.4 Case Studies

Each of the chapters in Parts II, III, and IV includes a detailed case study designed to showcase the tools and techniques discussed in the chapter. You can download the evidence files and practice dissecting them on your own forensic workstation.

The case study evidence files are located here:

`http://lmgsecurity.com/nf/`

They are freely available for your personal use.

0.5 Errata

In any technical book of this size and density, there are bound to be a few errors. We will maintain a list of errata at:

`http://lmgsecurity.com/nf/errata`

If you find an error, we would like to know about it. Please email us at errata@lmgsecurity.com. Check the web site before emailing to make sure the error is not already listed.

0.6 Final Notes

This book is a labor of love. Each chapter required countless hours of research, discussion, debate, and writing. To create the case studies and corresponding packet captures, we first built a laboratory with the equivalent of a small business-sized network, configured and reconfigured it for each exercise, wrote each scenario, and then ran the scenarios over and over again until we got them all exactly right.

It's impossible to count all the late nights and early mornings, flipped circuit breakers and dead hard drives, warm beers and cold pizzas that went into this book. Even though this book is hundreds of pages, we feel that we have only scratched the surface of the very deep field of network forensics. We learned an enormous amount from all the effort, and we hope you do, too.

Acknowledgments

This book would not exist without the support of two widely respected security professionals: Rob Lee and Ed Skoudis. Three years ago, Rob Lee tapped us to create the network forensics curriculum for the SANS Institute. This was the first time that we pooled our joint knowledge on the subject, and actually gave that body of work a name. Since that time, Rob has continued to act as a mentor and friend, constantly pushing us to improve our work, incorporate feedback, and extend the limits of our knowledge. Rob: Thank you for your high standards, your open and honest feedback, and most of all for believing in us. We could not have accomplished this without you.

Ed, in turn, encouraged us to write this book and kindly took the time to introduce us to his editor. His advice on the process proved invaluable. Thank you, Ed, for all of your help and support. We are forever grateful.

Thanks to all the terrific staff at Pearson who put so much time and effort into producing this book, especially our wonderful editor, Chris Guzikowski, as well as Jessica Goldstein, Raina Chrobak, Julie Nahil, Olivia Basegio, Chris Zahn, Karen Gettman, Chuti Prasertsith, John Fuller, and Elizabeth Ryan. A very special thanks as well to the great team at Laserwords for producing this book, especially Patty Donovan for her patience and kindness.

Many thanks to Jonah Elgart for creating the awesome cover illustration. We deeply appreciate the work of Dr. Dan Geer, who kindly wrote the foreword for this book. We are very grateful to our friends and colleagues who conducted technical reviews of the book: Michael Ford, Erik Hjelmvik, Randy Marchany, Craig Wright, and Joshua Wright. Their perspectives and attention to detail made this book infinitely better.

We would like to give a shout-out to our fantastic crew at LMG Security, particularly: Eric Fulton, Jody Miller, Randi Price, Scott Fretheim, David Harrison, and Diane Byrne. Each of them devoted many hours to helping us with network forensics research and curriculum. Eric Fulton was instrumental in developing several of the ForensicsContest.com puzzles, which formed the basis for some of the case studies (in particular, “HackMe” and “Ann’s Aurora”). Jody Miller became our He-Man when he swooped in and conquered the evil forces of Skeletor—*ahem*, we mean, formatted all of the footnotes for the book (nearly 500!).

Thanks to our friends, colleagues, and mentors who have taught us so much over the years: Shane Vannatta, Marsha and Bill Dahlgren, Pohl Longsine, Gary Longsine, John Strand, Michael P. Kenny, Gary and Pue Williams, the good folks at Modwest, Mike Poor, Kevin Johnson, Alan Ptak, Michael Grinberg, Sarah and Kerry Metlen, Anissa Schroeder, Bradley Coleman, Blake Brasher, Stephanie Henry, Nadia Madden and Jon McElroy, Clay Ward, the MIT Student Information Processing Board (SIPB), Wally Deschene, Steven and Linda Abate, Karl Reinhard, Brad Cool, Nick Lewis, Richard Souza, Paul Asadoorian, Larry Pesce, George Bakos, Johannes Ullrich, Paul A. Henry, Rick Smith, Guy Bruneau, Lenny Zeltser, Eric Cole, Judy Novak, Alan Tu, Fabienne van Cappel, Robert C de Baca, Mark Galassi, and Dan Starr.

Special thanks to the staff and faculty of the SANS Institute, in particular: Steven and Kathy Northcutt; Deb Jorgensen; Katherine Webb Calhoon; Lana Emery; Kate Marshall; Velda Lempka; Norris Campbell; and Lynn Lewis.

We would also like to thank everyone who contributed to ForensicsContest.com, whether by creating tools and writeups, submitting comments, or just playing the games. We have learned so much from all of you!

Thanks to our wonderful families for their encouragement and support while we were writing this book, especially Sheila Temkin Davidoff, E. Martin Davidoff, Philip and Lynda Ham, Barbara and Larry Oltmanns, Laura Davidoff, Michele, Kirk, and Naomi Robertson, Latisha, Mike, Makenna, and Braelyn Monnier, Chad, Amy, and Brady Rempel, Sheryl and Tommy Davidoff, Jonathan and Stefanie Davidoff, Jill and Jake Dinov, Jamie and Adam Levine, Annabelle Temkin, Norman and Eileen Shoenfeld, Brian and Marie Shoenfeld, and Debbie Shoenfeld.

Thanks to our little cat, Shark, who stayed curled up by our side for hundreds of hours as we wrote.

Most important of all: Thanks to our two daughters, Charlie and Violet. This book is for you.

About the Authors

Sherri Davidoff has over a decade of experience as an information security professional, specializing in penetration testing, forensics, social engineering testing, and web application assessments. She has consulted for a wide variety of industries, including banking, insurance, health care, transportation, manufacturing, academia, and government. Sherri is a course author for the SANS Institute and has published many articles on privacy and security. She is a GIAC-certified forensic examiner (GCFA) and penetration tester (GPEN), and holds her degree in computer science and electrical engineering from MIT.

Jonathan Ham specializes in large-scale enterprise security issues, from policy and procedure, to scalable prevention, detection, and response techniques. He's been commissioned to teach NCIS investigators how to use Snort, performed packet analysis from a facility more than 2,000 feet underground, taught intrusion analysis to the NSA, and chartered and trained the CIRT for one of the largest U.S. civilian Federal agencies. Jonathan has helped his clients achieve greater success for over fifteen years. He is a Certified Instructor with the SANS Institute, and regularly teaches courses on network security.

Sherri and Jonathan are principals of the security consulting firm, LMG Security. They are married and live in Montana, where they spend their days cracking jokes about TCP/IP and raising their two beautiful daughters, Charlie and Violet.

Source: Photo by Jonathan Qualben Photography, jonathanqualben.com.

This page intentionally left blank

www.allitebooks.com

Chapter 1

Practical Investigative Strategies

> *"A victorious army first wins and then seeks battle; a defeated army first battles and then seeks victory." –Sun Tsu,* The Art of War[1]

Evidence scattered around the world. Not enough time. Not enough staff. Unrealistic expectations. Internal political conflicts. Gross underestimation of costs. Mishandling of evidence. Too many cooks in the kitchen. Network forensic investigations can be tricky. In addition to all the challenges faced by traditional investigators, network forensics investigators often need to work with unfamiliar people in different countries, learn to interact with obscure pieces of equipment, and capture evidence that exists only for fleeting moments. Laws surrounding evidence collection and admissibility are often vague, poorly understood, or nonexistent. Frequently, investigative teams find themselves in situations where it is not clear who is in charge, or what the team can accomplish.

An organized approach is key to a successful investigation. While this book is primarily designed to explore technical topics, in this chapter, we touch on the fundamentals of investigative management. This is particularly important because network forensics investigations often involve coordination between multiple groups and evidence that may be scattered around the globe.

We begin by presenting three cases from different industries in order to give you some examples of how network forensics is used to support investigations in the real world. Next, we explore the fundamentals of evidence collection and distinctions that are made between different types of evidence in many courts. We discuss the challenges specific to network-based evidence, such as locating evidence on a network and questions of admissibility. Finally, we present the OSCAR investigative methodology, which is designed to give you an easy-to-remember framework for approaching digital investigations.

1.1 Real-World Cases

How is network forensics used in real life? In this section, we present three cases:

- Hospital Laptop Goes Missing
- Catching a Corporate Pirate
- Hacked Government Server

---

[1] Sun Tsu (Author), Lionel Giles (Translator), *The Art of War,* El Paso Norte Press, 2005.

These cases have been chosen to provide examples of common IT security incidents and illustrate how network forensics is frequently used. Although these cases are based on real-life experiences, they have been modified to protect the privacy of the organizations and individuals involved.

1.1.1 Hospital Laptop Goes Missing

A doctor reports that her laptop has been stolen from her office in a busy U.S. metropolitan hospital. The computer is password-protected, but the hard drive is not encrypted. Upon initial questioning, the doctor says that the laptop may contain copies of some patient lab results, additional protected health information (PHI) downloaded from email attachments, schedules that include patient names, birth dates, and IDs, notes regarding patient visits, and diagnoses.

1.1.1.1 Potential Ramifications

Since the hospital is regulated by the United States' Health Information Technology for Economic and Clinical Health (HITECH) Act and Health Insurance Portability and Accountability Act (HIPAA), it would be required to notify individuals whose PHI was breached.[2] If the breach is large enough, it would also be required to notify the media. This could cause significant damage to the hospital's reputation, and also cause substantial financial loss, particularly if the hospital were held liable for any damages caused due to the breach.

1.1.1.2 Questions

Important questions for the investigative team include:

1. Precisely when did the laptop go missing?
2. Can we track down the laptop and recover it?
3. Which patient data was on the laptop?
4. How many individuals' data was affected?
5. Did the thief leverage the doctor's credentials to gain any further access to the hospital network?

1.1.1.3 Technical Approach

Investigators began by working to determine the time when the laptop was stolen, or at least when the doctor last used it. This helped establish an outer bound on what data *could* have been stored on it. Establishing the time that the laptop was last in the doctor's possession also gave the investigative team a starting point for searching physical surveillance footage and access logs. The team also reviewed network access logs to determine whether the laptop was subsequently used to connect to the hospital network after the theft and, if so, the location that it connected from.

---

[2] "HITECH Breach Notification Interim Final Rule," U.S. Department of Health and Human Services, http://www.hhs.gov/ocr/privacy/hipaa/understanding/coveredentities/breachnotificationifr.html.

There are several ways investigators could try to determine what time the laptop went missing. First, they could interview the doctor to establish the time that she last used it, and the time that she discovered it was missing. Investigators might also find evidence in wireless access point logs, Dynamic Host Control Protocol (DHCP) lease assignment logs, Active Directory events, web proxy logs (if there is an enterprise web proxy), and of course any sort of laptop tracking software (such as Lojack for Laptops) that might have been in use on the device.

Enterprise wireless access point (WAP) logs can be especially helpful for determining the physical location in the facility where a mobile device was most recently connected, and the last time it was connected. In order to ensure uniform availability of wireless networks, enterprises typically deploy a fleet of WAPs that all participate in the same network. Although they appear to the end user as a single wireless network, network operators can view which mobile devices were connected to specific access points throughout the building. Some enterprises even have commercial software that can graphically represent the movement of wirelessly connected devices as they traverse the physical facility. If the laptop was still connected to the hospital's wireless network as the thief exited the building, investigators might be able to use wireless access point logs to show the path that the thief navigated as he or she exited the building. This might also be correlated with video surveillance logs or door access control logs.

Once investigators established an approximate time of theft, they could narrow down the patient information that might have been stored on the system. Email logs could reveal when the doctor last checked her email, which would place an outer bound on the emails that could have been replicated to her laptop. These logs might also reveal which attachments were downloaded. More importantly, the hospital's email server would have copies of all of the doctor's emails, which would help investigators gather a list of patients likely to have been affected by the breach. Similarly, hospital applications that provide access to lab results and other PHI might contain access logs, which could help investigators compile a list of possible data breach victims.

There might be authentication logs from Active Directory domain controllers, VPN concentrators, and other sources that indicate the laptop was used to access hospital resources even after the theft. If so, this might help investigators track down the thief. Evidence of such activities could also indicate that additional information was compromised, and that the attacker's interests went beyond merely gaining a new laptop.

1.1.1.4 Results

Leveraging wireless access point logs, the investigative team was able to pinpoint the time of the theft and track the laptop through the facility out to a visitor parking garage. Parking garage cameras provided a low-fidelity image of the attacker, a tall man wearing scrubs, and investigators also correlated this with gate video of the car itself as it left the lot with two occupants. The video was handed to the police, who were able to track the license plate. The laptop was eventually recovered amongst a stack of stolen laptops.

The investigative team carefully reviewed VPN logs and operating system logs stored on the central logging server and found no evidence that the doctor's laptop was used to attempt any further access to hospital IT resources. Hard drive analysis of the recovered laptop showed no indication that the system had been turned on after the theft. After extensive consultation with legal counsel, hospital management concluded that patient data had ultimately not been leaked.

In response to the incident, the hospital implemented full-disk encryption for all laptop hard drives, and deployed physical laptop locking mechanisms.

1.1.2 Catching a Corporate Pirate

GlobalCorp, Inc., has a centrally managed intrusion detection system, which receives alerts from sites around the world. Central security staff notice an alert for peer-to-peer (P2P) file-sharing, and on closer inspection see filename references to movies that are still in theaters. Fearing legal ramifications of inaction, they investigate.

1.1.2.1 Potential Ramifications

Management at GlobalCorp, Inc., were highly concerned that an employee was using the company network for trafficking in pirated intellectual property. If this activity were detected, the owner of the intellectual property might sue the company. This case occurred in 2003, at the height of Digital Millennium Copyright Act (DMCA) fervor, and it was assumed that if an individual within the company was illicitly trading pirated music or movies, then it could place the company at risk of costly legal battles.

1.1.2.2 Questions

Important questions for the investigative team include:

1. Where is the source of the P2P traffic physically located?
2. Which user is initiating the P2P traffic?
3. Precisely what data is being shared?

1.1.2.3 Technical Approach

Using the IP address from the IDS alerts, investigators identified the physical site that was the source of the traffic. In order to specifically identify the client system, its location, and primary user, investigators worked with local network management staff.

Meanwhile, intrusion analysts in the central office began capturing all of the P2P-related packets involving the IP address in question. The local facility confirmed that this IP address was part of a local DHCP pool on the wired local area network (LAN). Intrusion analysts reviewed DHCP lease assignment logs for relevant time periods, and recovered the media access control (MAC) address associated with the suspicious activity. From the MAC address investigators identified the manufacturer of the network card (Dell, in this case).

In order to trace the IP address to a specific office, local networking staff logged into switches and gathered information mapping the IP address to a physical port. The physical port was wired to a cubicle occupied by an email system administrator. Investigators entered his office after hours one evening and recovered his desktop for forensic analysis.

Upon examination, however, it was clear that the confiscated desktop was not the source of the P2P activity. The MAC address of the network card in the confiscated system (a Hewlett-Packard desktop) was not consistent with the MAC address linked to the suspicious activity. Subsequent analysis of the company's email server produced evidence that the suspect, an email system administrator, had leveraged privileged access to read emails of key networking staff involved in the investigation.

Local networking staff took caution to communicate out-of-band while coordinating the remainder of the investigation. Investigators conducted a thorough search of the premises for a system with the MAC address implicated. The matching desktop was eventually found in the desktop staging area, buried in a pile of systems queued for reimaging.

1.1.2.4 Results

Network forensic analysts examined full packet captures grabbed by the IDS, and were ultimately able to carve out video files and reconstruct playable copyrighted movies that were still in theaters. Hard drive analysis of the correct desktop produced corroborating evidence that the movies in the packet capture had been resident on the hard drive. The hard drive also contained usernames and email addresses linking the hard drive and associated network traffic with the suspect.

Case closed!

1.1.3 Hacked Government Server

During a routine antivirus scan, a government system administrator was alerted to suspicious files on a server. The files appeared to be part of a well-known rootkit. The server did not host any confidential data other than password hashes, but there were several other systems on the local subnet that contained Social Security numbers and financial information of thousands of state residents who had filed for unemployment assistance. The administrative account usernames and passwords were the same for all servers on the local subnet.

1.1.3.1 Potential Ramifications

State laws required the government to notify any individuals whose Social Security numbers were breached. If the servers containing this sensitive information were hacked, the state might be required to spend large amounts of money to send out notifications, set up hotlines for affected individuals, and engage in any resulting lawsuits. In addition, disclosure of a breach might damage the careers of high-ranking elected state officials.

1.1.3.2 Questions

Important questions for the investigative team include:

- Was the server in question truly compromised?
- If so, how was the system exploited?
- Were any other systems on the local network compromised?
- Was any confidential information exported?

1.1.3.3 Technical Approach

The server in question appeared to contain files with names that fit the pattern for a well-known rootkit. Investigators began by examining these files and concluded that they were, indeed, malicious software. The rootkit files were found in the home directory of an old local administrator account that staff had forgotten even existed.

Investigators found that the local authentication logs had been deleted. Fortunately, all servers on the subnet were configured to send logs to a central logging server, so instead investigators reviewed Secure Shell (SSH) logs from the central logging server that were associated with the account. From the SSH logs, it was clear that the account had been the target of a brute-force password-guessing attack. Investigators used visualization tools to identify the times that there were major spikes in the volume of authentication attempts. A subsequent password audit revealed that the account's password was very weak.

The SSH logs showed that the source of the brute-force attack was a system located in Brazil. This was surprising to IT staff because according to network documentation the perimeter firewall was supposed to be configured to block external access to the SSH port of servers on the subnet under investigation. Investigators gathered copies of the current, active firewall configuration and found that it did not match the documented policy—in practice, the SSH port was directly accessible from the Internet. Subsequently, investigators analyzed firewall logs and found entries that corroborated the findings from the SSH logs.

When one system in the environment is compromised, there is a significant probability that the attacker may use credentials from that system to access other systems. IT staff were concerned that the attacker might have used the stolen account credentials to access other systems on the local subnet.

Fortunately, further analysis of the server hard drive indicated that the attacker's access was short-lived; the antivirus scan had alerted on the suspicious files shortly after they were created. Investigators conducted a detailed analysis of authentication logs for all systems on the local subnet, and found no other instances of suspicious access to the other servers. Furthermore, there were no records of logins using the hacked account on any other servers. Extensive analysis of the firewall logs showed no suspicious data exportation from any servers on the local subnet.

1.1.3.4 Results

Investigators concluded that the server under investigation was compromised but that no other systems on the local subnet had been exploited and no personal confidential information had been breached. To protect against future incidents, the state IT staff corrected the errors in the firewall configuration and implemented a policy in which firewall rules were audited at least twice per year. In addition, staff removed the old administrator account and established a policy of auditing all server accounts (including privileges and password strength) on a quarterly basis.

1.2 Footprints

When conducting network forensics, investigators often work with live systems that cannot be taken offline. These may include routers, switches, and other types of network devices, as well as critical servers. In hard drive forensics, investigators are taught to minimize system modification when conducting forensics. It is much easier to minimize system modification when working with an offline copy of a write-protected drive than with production network equipment and servers.

In network forensics, investigators also work to minimize system modification due to forensic activity. However, in these cases investigators often do not have the luxury of an offline copy. Moreover, network-based evidence is often highly volatile and must be collected through active means that inherently modify the system hosting the evidence. Even when investigators are able to sniff traffic using port monitoring or tapping a cable, there is always some impact on the environment, however small. This impact can sometimes be minimized through careful selection of acquisition techniques, but it can never be eliminated entirely.

Every interaction that an investigator has with a live system modifies it in some way, just as an investigator in real life modifies a crime scene simply by walking on the floor. We use the term “footprint” throughout this book to refer to the impact that an investigator has on the systems under examination.

You will always leave a footprint. Often, the size of the footprint required must be weighed against the need for expediency in data collection. Take the time to record your activities carefully so that you can demonstrate later that important evidence was not modified. Always be conscious of your footprint and tread lightly.

1.3 Concepts in Digital Evidence

What is evidence? The *Compact Oxford English Dictionary* defines “evidence” as:[3]

> evidence (noun)
>
> 1. information or signs indicating whether a belief or proposition is true or valid.
> 2. information used to establish facts in a legal investigation or admissible as testimony in a law court.

In this book, we are concerned with both of the above definitions. Our goal in many investigations is to compile a body of evidence suitable for presentation in court proceedings (even if we hope never to end up in court!). Both relevance to the case and admissibility are important, but the first goal is to ascertain the facts of the matter and understand truly and correctly what has transpired.

Consequently, we define “evidence” in the broadest sense as any observable and recordable event, or artifact of an event, that can be used to establish a true understanding of the cause and nature of an observed occurrence.

Of course, it’s one thing to be able to reconstruct and understand the events that comprise an occurrence, and yet another to be able to demonstrate that in such a way that victims can be justly compensated and perpetrators justly punished within our legal framework. Within this system there are a few categories of evidence that have very specific meanings:

---

[3] “Oxford Dictionaries Online—English Dictionary and Language Reference,” *Oxford Dictionaries*, http://www.askoxford.com/concise_oed/evidence?view=uk.

- Real
- Best
- Direct
- Circumstantial
- Hearsay
- Business Records

We'll take each of these in turn and discuss their nature and relative usefulness and importance. Due to the rising popularity of electronic communications systems, we also include the following general categories of evidence:

- Digital
- Network-Based Digital

In this book, our discussion of evidence is based on the United States common law system and the U.S. Federal Rules of Evidence (FRE).[4] Many of these concepts may be similar in your jurisdiction, although we also recommend that you familiarize yourself with the rules specific to your region of the world.

1.3.1 Real Evidence

What is "real" evidence? "Real evidence" is roughly defined as any physical, tangible object that played a relevant role in an event that is being adjudicated. It is the knife that was pulled from the victim's body. It is the gun that fired the bullet. It is the physical copy of the contract that was signed by both parties. In our realm it is also the physical hard drive from which data is recovered, and all the rest of the physical computer components involved.

Real evidence usually comprises the physicality of the event, and as such is often the most easily presented and understood element of a crime. Human beings understand tangible objects much more readily than abstract concepts, such as data comprised of ones and zeros (which are themselves comprised of the presence or absence of magnetization on microscopic bits of a spinning platter). Unless the hard drive was used as a blunt object in an assault, and as a consequence is covered in identifiable traces of blood and hair follicles (DNA is real evidence too), the judge or jury may have a difficult time envisioning the process through which the evidence reached its current state and was preserved.

---

[4] Committee on the Judiciary House (US) and US House Committee on the Judiciary, *Federal Rules of Evidence (December 2011)* (Committee on the Judiciary, 2011), http://judiciary.house.gov/hearings/printers/112th/evidence2011.pdf.

Examples of “real evidence” can include:

- The murder weapon
- The fingerprint or footprint
- The signed paper contract
- The physical hard drive or USB device
- The computer itself—chassis, keyboard, and all

1.3.2 Best Evidence

“Best evidence” is roughly defined as the best evidence that can be produced in court. The FRE states, “To prove the content of a writing, recording, or photograph, the *original* writing, recording, or photograph is required, except as otherwise provided in these rules or by Act of Congress” [emphasis added].[5] If the original evidence is not available, then alternate evidence of its contents may be admitted under the “best evidence rule.” For example, if an original signed contract was destroyed but a duplicate exists, then the duplicate may be admissible. However, if the original exists and could be admitted, then the duplicate would not suffice.

Our favorite illustration of the “best evidence rule” comes from Dr. Eric Cole, as presented in his SANS courses: Imagine that a helicopter and a tractor trailer collide on a bridge. Real evidence in this case would be the wreckage, but there is no hope of bringing the real evidence into depositions, much less in front of a jury. In such a case the photographs of the scene comprise the best records that can be brought to court. They will have to suffice, and most often do.

Forensic analysts, lawyers, and jurors have questioned what constitutes “original” evidence in the case of digital evidence. Fortunately, the FRE explicitly addresses this issue, as follows:[6]

> An “original” of a writing or recording means the writing or recording itself or any counterpart intended to have the same effect by the person who executed or issued it. For electronically stored information, “original” means any printout—or other output readable by sight—if it accurately reflects the information. An “original” of a photograph includes the negative or a print from it. (e) A “duplicate” means a counterpart produced by a mechanical, photographic, chemical, electronic, or other equivalent process or technique that accurately reproduces the original.

In other words, a printout from a computer hard drive that accurately reflects the data would normally be considered “original” evidence.

With network forensics, the bits and bytes being presented have been recorded and may be treated in the same way as a photograph of an event. It is as though we’ve photographed the bullet as it traveled through the air. The difference is that network forensic investigators can often reconstruct a forensically identical copy of the entire bullet from the snapshot, rather than just presenting a grainy photograph from which legal teams hope to divine trajectories, masses, and the sending barrel's rifling.

---

[5] Committee on the Judiciary House (US) and US House Committee on the Judiciary, *Federal Rules of Evidence*, 25.

[6] *Ibid.*

Examples of "best evidence" include:

- A photo of the crime scene
- A copy of the signed contract
- A file recovered from the hard drive
- A bit-for-bit snapshot of a network transaction

1.3.3 Direct Evidence

"Direct evidence" is the testimony offered by a direct witness of the act or acts in question. There are lots of ways that events can be observed, captured, and recorded in the real world, and our court systems try to accommodate most of these when there is relevant evidence in question. Of course, the oldest method is the reportable observation of a fellow human being. This human testimony is classified as "direct evidence," and it remains some of the most utilized forms of evidence, even if it is often disputed and unreliable.

Direct evidence is usually admissible, so long as it's relevant. What other people witnessed can have a great impact on a case.

Examples of "direct evidence" can include:

- "I saw him stab that guy."
- "She showed me an inappropriate video."
- "I watched him crack passwords using John the Ripper and a password file he shouldn't have."
- "I saw him with that USB device."

1.3.4 Circumstantial Evidence

In contrast to "direct evidence," "circumstantial evidence" is evidence that does not directly support a specific conclusion. Rather, circumstantial evidence may be linked together with other evidence and used to deduce a conclusion.

Circumstantial evidence is important for cases involving network forensics because it is "the primary mechanism used to link electronic evidence and its creator."[7] Often, circumstantial evidence is used to establish the author of emails, chat logs, or other digital evidence. In turn, authorship verification is necessary to establish authenticity, which is required for evidence to be admissible in court. The DoJ elaborates:[8]

---

[7] Scott M. Giordano, "Electronic Evidence and the Law," *Information Systems Frontiers* 6, no. 2 (June 1, 2004): 165.

[8] H. Marshall Jarrett, Director, EOUSA, "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations," 203.

> [D]istinctive characteristics like email addresses, nicknames, signature blocks, and message contents can prove authorship, at least sufficiently to meet the threshold for authenticity ... For example, in United States v. Simpson, 152 F.3d 1241 (10th Cir. 1998), prosecutors sought to show that the defendant had conversed with an undercover FBI agent in an Internet chat room devoted to child pornography. The government offered a printout of an Internet chat conversation between the agent and an individual identified as 'Stavron' and sought to show that 'Stavron' was the defendant ... 'Stavron' had told the undercover agent that his real name was 'B. Simpson,' gave a home address that matched Simpson's, and appeared to be accessing the Internet from an account registered to Simpson. Further, the police found records in Simpson's home that listed the name, address, and phone number that the undercover agent had sent to 'Stavron.' Accordingly, the government had provided evidence sufficient to support a finding that the defendant was 'Stavron,' and the printout was properly authenticated.

Examples of "circumstantial evidence" can include:

- An email signature
- A file containing password hashes on the defendant's computer
- The serial number of the USB device

1.3.5 Hearsay

"Hearsay" is the label given to testimony offered second-hand by someone who was not a direct witness of the act or acts in question. It is formally defined by the FRE as "a statement, other than one made by the declarant while testifying at the trial or hearing, offered in evidence to prove the truth of the matter asserted." This includes the comments of someone who may have direct knowledge of an occurrence, but who is unable or unwilling to deliver them directly to the court. Hearsay is generally not ruled admissible, unless it falls into the category of an exception as listed in the FRE (Rules 803 and 804).

Digital evidence can be classified as hearsay if it contains assertions created by people. The U.S. Department of Justice cites "a personal letter; a memo; bookkeeping records; and records of business transactions inputted by persons" as examples of digital evidence that would be classified as hearsay.[9]

However, digital evidence that is generated by a fully automated process with no human intervention is generally *not* considered heresay. The Department of Justice explains:[10]

> Computer-generated records that do not contain statements of persons therefore do not implicate the hearsay rules. This principle applies both to records generated by a computer without the involvement of a person (e.g., GPS tracking records) and to computer records that are the result of human conduct other than assertions (e.g., dialing a phone number or punching in a PIN at an ATM).

---

[9] H. Marshall Jarrett, Director, EOUSA, "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations," 192.

[10] H. Marshall Jarrett, Director, EOUSA, "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations," 193.

In some cases, courts have admitted digital evidence using the "business records" exception of the hearsay rule, which we discuss further in the next section. However, the Department of Justice points out that in these cases, the courts overlooked the question of whether the digital evidence should have been classified as hearsay in the first place. "Increasingly ... courts have recognized that many computer records result from a process and are not statements of persons—they are thus not hearsay at all."[11]

Examples of "hearsay" can include:

- "The guy told me he did it."
- "He said he knew who did it, and could testify."
- "I saw a recording of the whole thing go down."
- A text file containing a personal letter

1.3.6 Business Records

Business records can include any documentation that an enterprise routinely generates and retains as a result of normal business processes, and that is deemed accurate enough to be used as a basis for managerial decisions. The FRE specifically exempts business records from the rule that hearsay is inadmissible, stating that:[12]

> The following are not excluded by the rule against hearsay, regardless of whether the declarant is available as a witness: [...] A record of an act, event, condition, opinion, or diagnosis if ... the record was kept in the course of a regularly conducted activity of a business, organization, occupation, or calling, whether or not for profit ...

This can include everything from email and memos to access logs and intrusion detection system (IDS) reports. There may be legally mandated retention periods for some of this data. Other records may be subject to internal retention and/or destruction policies. The bottom line is that if the records are seen as accurate enough by the enterprise that they are the basis for managerial decision making, then the courts usually deem them reliable enough for a proceeding.

Digital evidence has been admitted under the "business records" exception to hearsay many times, although in some cases this was erroneous. The Department of Justice points out that "courts have mistakenly assumed that computer-generated records are hearsay without recognizing that they do not contain the statement of a person."

Examples of "business records" can include:

- Contracts and other employment agreements
- Invoices and records of payment received

---

[11] H. Marshall Jarrett, Director, EOUSA, "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations," 191.

[12] Committee on the Judiciary House (US) and US House Committee on the Judiciary, *Federal Rules of Evidence*, 17.

- Routinely kept access logs
- /var/log/messages

1.3.7 Digital Evidence

"Digital evidence" is any documentation that satisfies the requirements of "evidence" in a proceeding, but that exists in electronic digital form. Digital evidence may rest in microscopic spots on spinning platters, magnetized to greater or lesser degrees in a somewhat nonvolatile scheme, but regardless, unintelligible except through multiple layers of abstraction and filesystem protocols. In other cases, digital evidence may be charges held in volatile storage, which dissipate within seconds of a loss of power to the system. Digital evidence may be no more tangible, nor permanent, than pulses of photons, radio frequency waves, or differential levels of voltage on copper wires.

Naturally, digital evidence poses challenges for investigators seeking to preserve it and attorneys seeking to admit it in court. In order for evidence to be admissible in United States federal courts, digital evidence must adhere to the same standards as other types of evidence: it must be deemed relevant to the case and authentic. "The standard for authenticating computer records is the same as for authenticating other records . . . ," wrote the U.S. Department of Justice (DoJ) in 2009. "Importantly, courts have rejected arguments that electronic evidence is inherently unreliable because of its potential for manipulation. As with paper documents, the mere possibility of alteration is not sufficient to exclude electronic evidence. Absent specific evidence of alteration, such possibilities go only to the evidence's weight, not admissibility."[13]

Examples of "digital evidence" include:

- Emails and IM sessions
- Invoices and records of payment received
- Routinely kept access logs
- /var/log/messages

1.3.8 Network-Based Digital Evidence

"Network-based digital evidence" is digital evidence that is produced as a result of communications over a network. The primary and secondary storage media of computers (e.g., the RAM and hard drives) tend to be fruitful fodder for forensic analysis. Due to data remanence, persistent storage can retain forensically recoverable and relevant evidence for hours, days, even years beyond file deletion and storage reuse. In contrast, network-based digital evidence can be extremely volatile. Packets flit across the wire in milliseconds, vanish from switches in the blink of an eye. Web sites change depending on from where they're viewed and when.

The requirements for admissibility of network-based digital evidence are murky. Often, the source that generated the evidence is not obtainable or cannot be identified. When the evidence is a recording of a chat log, blog posting, or email, the identity of the parties in the conversation (and therefore the authors of the statements) may be difficult to prove. When the evidence is a web site, the litigant may need to provide supporting evidence to demonstrate that the image presented in court is what actually existed at the time and location that it was supposedly viewed. For example, "[s]everal cases have considered what foundation is necessary to authenticate the contents and appearance of a website at a particular time. Print-outs of web pages, even those bearing the URL and date stamp, are not self-authenticating.... Thus, courts typically require the testimony of a person with knowledge of the website's appearance to authenticate images of that website."[14]

---

[13] H. Marshall Jarrett, Director, EOUSA, "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations" (Office of Legal Education Executive Office for United States Attorneys, 2009), 198–202, http://www.justice.gov/criminal/cybercrime/ssmanual/ssmanual2009.pdf.

There is little case precedent on the admissibility of network packet captures. Depending on the method of capture and the details of the case, packet captures of network traffic may be treated as *recordings* of events, similar to a taped conversation.

Examples of "network-based digital evidence" can include:

- Emails and IM sessions
- Browser activity, including web-based email
- Routinely kept packet logs
- /var/log/messages

1.4 Challenges Relating to Network Evidence

Network-based evidence poses special challenges in several areas, including acquisition, content, storage, privacy, seizure, and admissibility. We will discuss some common challenges below.

- **Acquisition** It can be difficult to locate specific evidence in a network environment. Networks contain so many possible sources of evidence—from wireless access points to web proxies to central log servers—that sometimes pinpointing the correct location of the evidence is tricky. Even when you do know where a specific piece of evidence resides, you may have difficulty gaining access to it for political or technical reasons.
- **Content** Unlike filesystems, which are designed to contain all the contents of files and their metadata, network devices may or may not store evidence with the level of granularity desired. Network devices often have very limited storage capacity. Usually, only selected metadata about the transaction or data transfer is kept instead of complete records of the data that traversed the network.
- **Storage** Network devices commonly do not employ secondary or persistent storage. As a consequence, the data they contain may be so volatile as to not survive a reset of the device.
- **Privacy** Depending on jurisdiction, there may be legal issues involving personal privacy that are unique to network-based acquisition techniques.

---

[14] H. Marshall Jarrett, Director, EOUSA, "Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations," 204.

- **Seizure** Seizing a hard drive can inconvenience an individual or organization. Often, however, a clone of the original can be constructed and deployed such that critical operations can continue with limited disruption. Seizing a network device can be much more disruptive. In the most extreme cases, an entire network segment may be brought down indefinitely. Under most circumstances, however, investigators can minimize the impact on network operations.
- **Admissibility** Filesystem-based evidence is now routinely admitted in both criminal and civil proceedings. As long as the filesystem-based evidence is lawfully acquired, properly handled, and relevant to the case, there are clear precedents for authenticating the evidence and admitting it in court. In contrast, network forensics is a newer approach to digital investigations. There are sometimes conflicting or even nonexisting legal precedents for admission of various types of network-based digital evidence. Over time, network-based digital evidence will become more prevalent and case precedents will be set and standardized.

1.5 Network Forensics Investigative Methodology (OSCAR)

Like any other forensic task, recovering and analyzing digital evidence from network sources must be done in such a way that the results are both reproducible and accurate. In order to ensure a useful outcome, forensic investigators should perform our activities within a methodological framework. The overall step-by-step process recommended in this book is as follows:

- **O**btain information
- **S**trategize
- **C**ollect evidence
- **A**nalyze
- **R**eport

We refer to this methodology as "OSCAR," and walk through each of these steps in the following section.

1.5.1 Obtain Information

Whether you're law enforcement, internal security staff, or a forensic consultant, you will always need to do two things at the beginning of an investigation: obtain information about the incident itself, and obtain information about the environment.

1.5.1.1 The Incident

Usually you will want to know the following things about the incident:

- Description of what happened (as is currently known)
- Date, time, and method of incident discovery

- Persons involved
- Systems and data involved
- Actions taken since discovery
- Summary of internal discussions
- Incident manager and process
- Legal issues
- Time frame for investigation/recovery/resolution
- Goals

This list is simply a starting point, and you will need to customize it for each incident.

1.5.1.2 The Environment

The information you gather about the environment will depend on your level of familiarity with it. Remember that every environment is constantly changing, and complex social and political dynamics occur during an incident. Even if you are very familiar with an organization, you should always take the time to understand how the organization is responding to this particular incident, and clearly establish who needs to be kept in the loop. Usually you will want to know the following things about the environment:

- Business model
- Legal issues
- Network topology (request a network map, etc. if you do not have one)
- Available sources of network evidence
- Organizational structure (request an organizational chart if you do not have one)
- Incident response management process/procedures (forensic investigators are part of the response process and should be at least basically familiar with it)
- Communications systems (is there a central incident communication system/evidence repository?)
- Resources available (staff, equipment, funding, time)

1.5.2 Strategize

It is crucial that early on you take the time to accurately assess your resources and plan your investigation. While this is important for any investigation, it is especially important for network forensics because there are many potential sources of evidence, some of which are also very volatile. Investigators must work efficiently. You will want to regularly confer with others on the investigative/incident response team while planning and conducting the investigation to ensure that everyone is working in concordance and that important developments are communicated.

Source of Evidence	Likely Value	Effort	Volatility	Priority
Firewall logs	High	Medium	Low	2
Web proxy cache	High	Low	Medium	1
ARP tables	Low	Low	High	3

Figure 1–1. An example of evidence prioritization. In this example, we list potential sources of evidence, the likely value, the likely effort to obtain it, and the expected volatility. These values will be different for every investigation.

Here are some tips for developing an investigative strategy:

- Understand the goals and time frame of the investigation.
- List your resources, including personnel, time, and equipment.
- Identify likely sources of evidence.
- For each source of evidence, estimate the value and cost of obtaining it.
- Prioritize your evidence acquisition.
- Plan the initial acquisition/analysis.
- Decide upon method and times of regular communication/updates.
- Keep in mind that after conducting your initial analysis, you may decide to go back and acquire more evidence. Forensics is an iterative process.

Figure 1–1 shows an example of evidence prioritization. In this example, the organization collects firewall logs but stores them in a distributed manner on systems that are not easily accessed. The organization has a web proxy, which is centrally accessed by key security staff. ARP tables can be gathered from any system on the local LAN.

The table lists potential sources of evidence, the likely value for the investigation, the expected effort required to obtain the evidence, and the expected volatility. All of these values are unique to each investigation; every organization has different system configurations, data retention policies, and access procedures. Furthermore, the network equipment, investigative resources, and goals of each investigation vary widely.

Based on this information, we can create our evidence spreadsheet and prioritize accordingly. Next, we would develop a plan for evidence acquisition based on our available resources.

1.5.3 Collect Evidence

In the previous step, "Strategize," we prioritized our sources of evidence and came up with an acquisition plan. Based on this plan, we then collect evidence from each source. There are three components you must address every time you acquire evidence:

- **Document**—Make sure to keep a careful log of all systems accessed and all actions taken during evidence collection. Your notes must be stored securely and may be referenced in court. Even if the investigation does not go to court, your notes will still be very helpful during analysis. Be sure to record the date, time, source, method of acquisition, name of the investigator(s), and chain of custody.

- **Capture**—Capture the evidence itself. This may involve capturing packets and writing them to a hard drive, copying logs to hard drive or CD, or imaging hard drives of web proxies or logging servers.
- **Store/Transport**—Ensure that the evidence is stored securely and maintain the chain of custody. Keep an accurate, signed, verifiable log of the persons who have accessed or possessed the evidence.

Since the admissibility of evidence is dependent upon its relevance and reliability, investigators should carefully track the source, method of acquisition, and chain of custody. It's generally accepted that a bit-for-bit image of a hard drive is acceptable in court. For a lot of network-based evidence, the admissibility is not so clear-cut. When in doubt, take careful notes and consult legal counsel.

As with any evidence gathered in the course of an investigation, proper care must be taken to preserve evidence integrity and to document its use and disposition throughout its life cycle (from the initial acquisition to its return to its rightful owner). As we'll see, in some cases this may mean documenting and maintaining the physical chain of custody of a network device. However, in many cases the original incarnation of the evidence being acquired will never be taken into custody.

1.5.3.1 Tips for Evidence Collection

Best practices for evidence collection include:

- Acquire as soon as possible, and lawfully
- Make cryptographically verifiable copies
- Sequester the originals under restricted custody and access (or your earliest copy, when the originals are not available)
- Analyze only the copies
- Use tools that are reputable and reliable
- Document everything you do!

1.5.4 Analyze

Of course the analysis process is normally nonlinear, but certain elements should be considered essential:

- **Correlation** One of the hallmarks of network forensics is that it involves multiple sources of evidence. Much of this will be timestamped, and so the first consideration should be what data can be compiled, from which sources, and how it can be correlated. Correlation may be a manual process, or it may be possible to use tools to do it for you in an automated fashion. We'll look at such tools later on.

- **Timeline** Once the multiple data sources have been aggregated and correlated, it's time to build a timeline of activities. Understanding who did what, when, and how is the basis for any theory of the case. Recognize that you may have to adjust for time skew between sources!
- **Events of Interest** Certain events will stand out as potentially more relevant than others. You'll need to try to isolate the events that are of greatest interest, and seek to understand how they transpired.
- **Corroboration** Due to the relatively low fidelity of data that characterizes many sources of network logs, there is always the problem of "false positives." The best way to verify events in question is to attempt to corroborate them through multiple sources. This may mean seeking out data that had not previously been compiled, from sources not previously consulted.
- **Recovery of additional evidence** Often the efforts described above lead to a widening net of evidence acquisition and analysis. Be prepared for this, and be prepared to repeat the process until such time as the events of interest are well understood.
- **Interpretation** Throughout the analysis process, you may need to develop working theories of the case. These are educated assessments of the meaning of your evidence, designed to help you identify potential additional sources of evidence, and construct a theory of the events that likely transpired. It is of the utmost importance that you separate your interpretation of the evidence from fact. Your interpretation of the evidence is always a hypothesis, which may be proved or disproved.

1.5.5 Report

Nothing you'll have done to this point, from acquisition through analysis, will matter if you're unable to convey your results to others. From that perspective, reporting might be the most important aspect of the investigation. Most commercial forensic tools handle this aspect for the analyst, but usually not in a way that is maximally useful to a lay audience, which is generally necessary.

The report that you produce must be:

- Understandable by nontechnical laypeople, such as:
  - Legal teams
  - Managers
  - Human Resources personnel
  - Judges
  - Juries
- Defensible in detail
- Factual

In short, you need to be able to explain the results of your investigation in terms that will make sense for nontechnical people, while still maintaining scientific rigor. Executive summaries and high-level descriptions are key, but they must be backed by details that can easily be defended.

1.6 Conclusion

Network forensic investigations pose a myriad of challenges, from distributed evidence to internal politics to questions of evidence admissibility. To meet these challenges, investigators must carefully assess each investigation and develop a realistic strategy that takes into account both the investigative goals and the available resources.

We began this chapter with a series of case studies designed to illustrate how network forensic techniques are applied in real life. Subsequently, we reviewed the fundamental concepts in digital evidence, as employed in the United States common law system, and touched upon the challenges that relate specifically to network-based digital evidence. Finally, we provided you with a method for approaching network forensics investigations.

As Sun Tsu wrote 2,500 years ago: “A victorious army first wins and then seeks battle; a defeated army first battles and then seeks victory.” Strategize first; then collect your evidence and conduct your analysis. By considering the challenges unique to your investigation up front, you will meet your investigative goals most efficiently and effectively.

www.allitebooks.com

Index

Access control lists (ACLs), firewall, 359–60
Access logfile, Squid, 378
Active Directory domain controllers, 5
Active Directory events, 5
Activity pattern matching, 173, 175–77
  elements, 175
  patterns, 176–77
Address Resolution Protocol (ARP), 53–54, 338
Admissibility of evidence, 17
Adobe, 480, 483
Advanced Encryption Standard (AES), 211
Advanced persistent threat (APT), 480–84
  definition of, 481–82
  evolution of, 483–84
  examples of, early, 482–83
  term, early usage of, 481
AIM. *See* AOL Instant Messenger (AIM)
AirPcap USB adapter, 51
AirPort Express, 351
Alerts
  data, 265
  fidelity, 261
  "INFO Web Bug," 283–84
  NIDS/NIPS functionality, 260–61
  Snort, 269, 277
  "Tcp Window Scale Option," 284–85
Ann Tunnels Underground (case study), 441–59
  challenge questions, response to, 458
  DNS analysis, 443–46
  next steps, 459
  protocol statistics, 442–43
  theory of the case, 456–58
  timeline, 456
  tunneled IP packet analysis, 451–54
  tunneled IP packets, quest for, 446–50
  tunneled IP segment analysis, 454–56
Ann's Aurora (case study), 492–517
  challenge questions, response to, 515
  intrusion detection, 492–94
  next steps, 516–17
  overview of, 492
  TCP conversations, 495–513
  theory of the case, 514–15
  timeline, 513–14
Ann's Bad AIM scenario, 83–95, 100–101, 109, 131–33
Ann's Coffee Ring (case study), 356–68
  challenge questions, response to, 367
  DHCP server logs, 358–59
  DNS stimulus and response, 364
  explanations, potential, 366–67
  firewall ACLs, 359–60
  firewall diagnostic commands, 357–58
  firewall log analysis, 360–64
  next steps, 367–68
  overview of, 356–57
  prohibited connection attempts, 366–67
  rogue system, 366
  summary of events, 365–66
  theory of the case, 365–67
  timeline, 364–65
Ann's Rendezvous (case study), 135–57
  analysis (protocol summary), 135–36
  attachment, viewing, 147–49
  challenge questions, response to, 155–56
  DHCP traffic, 136–38
  email account monitoring, 157
  keyword search, 138–41
  overview of, 135
  packet capture, further analysis of, 157
  SMTP analysis, 141–46
  theory of the case, 155
  timeline, 154
Anonymizing proxy, 370, 371
Antivirus evasion, early, 464

Antivirus scan/scanner, 7, 8, 371, 496, 512, 516
Antivirus signatures, 287, 420
Antivirus software, 463–64, 515
Antivirus vendors, 481, 497
AOL Instant Messenger (AIM), 88
  Ann's Bad AIM scenario, 83–95, 100–101, 109, 131–33
  ICBM and, 88, 91, 102, 103
  OSCAR protocol and, 78, 88, 89
Apcupsd, 304
Apple
  AirPort Express, 216–17, 342, 346, 351
  Airport Extreme, 305
  iChat, 88
Application logs, 300–302
Application proxies, 345
Application servers, 29, 300–301
Arbor Networks, 478
Argus, 163, 171, 179
ARPANET, 76–77
ARP spoofing, 53–54
ASCII values associated with protocols, 83–84
Asleap tool, 213
Asymmetric warfare, 468
Attachment, viewing, 147–49
Attachment file carving in SMTP, 146–47
Attacks on wireless devices and networks, 224–28
  Evil Twin, 227–28
  rogue wireless access points, 225–27
  sniffing, 224–25
  WEP cracking, 228
Audit Record Generation and Utilization System. *See* Argus
Authentication
  AAA logging, 355
  EAP, 212–13
  failed, 319–22
  PAP, 213
  servers, 27
  SMTP, 127–28
  successful, 323–24
Authentication, authorization, and accounting (AAA) logging, 355
Authentication Header (AH), 427–28
Authentication logs, Linux, 299

BackOrifice, 463, 472
BackTrack Linux, 51
Bakos, George, 391
Baselining in flow record analysis, 173, 174
Base64, 464, 465
Basic Service Set Identification (BSSID), 204, 233
Bejtlich, Richard, 481
Bellovin, Steve, 49, 63
Berkeley Packet Filter (BPF), 55–59
  packet filtering with, 101
  primitives, 56–57
Berners-Lee, Tim, 120–21, 401
Best evidence, 11–12
Binary values associated with protocols, 83–84
Bitmasking, 58
Blacklisting, 373
Bless hex editor, 99
Blog spam, 485
Blue Coat Reporter, 381, 400
Bluetooth access point, 226–27
Border Gateway Protocol (BGP), 347
Botnets, 462–63
  distributed management, 462–63
  full-featured control, 463
  implications for network forensics, 463
  Storm, 462, 465, 469, 478, 479
  Waledac, 464, 470–71, 478, 479, 487, 489–90
BPF. *See* Berkeley Packet Filter (BPF)
Bradley, Brian, 42
Buffered local logging, 353
Business records, 14–15

Cables, 46–49
  coaxial, 46–47
  copper, 46–47
  intercepting traffic in, 47–49
  optical, 47
  twisted pair, 47
  undersea cable cuts, 49
Cabling, 24

Cache-control, 372
Caching, 371–73
  distributed, 374–75
  expiration, 372
  proxy, 370, 372
  Squid, 379–81
  validation, 372–73
Camouflaging Worm (C-Worm), 475
Capturing evidence, 20
Carnegie Mellon, 163
Carpenter, Shawn, 482
Carriage-return/linefeeds (CRLFs), 147, 386–87
Cascade virus, 463
Case studies. *See also individual case studies*
  Ann's Aurora, 492–517
  Ann's Coffee Ring, 356–68
  Ann's Rendezvous, 135–57
  Ann Tunnels Underground, 441–59
  Curious Mr. X, 184–97
  HackMe, Inc., 236–56
  Inter0ptic Saves the Planet, Part 1, 276–87
  Inter0ptic Saves the Planet, Part 2, 402–20
  L0ne Sh4rk's Revenge, 318–34
Catching a Corporate Pirate (real-world case), 6–7
  potential ramifications, 6
  questions, 6
  results, 7
  technical approach, 6–7
C&C. *See* Command-and-control channels (C&C)
Centralized C&C, 466
Centralized network log architecture, 307–8
Central log servers, 29
CERT, 170
Certificate authorities (CAs), 394–96
Challenge Handshake Authentication Protocol (CHAP), 213
Changing the channel, 225–26
China
  cybersecurity attack and defense capabilities, 482–83
  "Operation Aurora" and, 480–81
Circumstantial evidence, 12–13
Cisco, 166, 167
  ASA (*See* Cisco ASA)
  ASDM, 339, 342, 346, 351, 352
  Catalyst switches, 162
  CiscoWorks Management Center, 342, 351
  commercial enterprise tools, 233
  enterprise wireless access points (3600 AP), 216
  GRE, 423, 425, 427
  Inter-Switch Link (ISL), 424–25
  IOS, 349, 354
  ISL, 423, 424, 425
  Java-based cross-platform interfaces, 70
  Java-based proprietary interfaces, 351
  LEAP protocol, 213
  NetFlow, 162, 163, 166–68, 170–71, 177, 179, 184
  PEAP protocol, 213
  routers, 162, 229
  RSPAN, 53
  sensor software, 163
  SPAN, 53, 54, 184, 185
  trunking, 423, 424, 425
  Wireless Location Appliance (WLA), 233, 234
  WRT54G wireless router, 229
Cisco ASA
  5500, 54
  5505, 337, 338, 346, 349–50, 352, 354
  Ann's Coffee Ring (case study), 364, 366
  Curious Mr. X (case study), 184–85, 187, 193–94
  L0ne Sh4rk's Revenge (case study), 318, 319
  v8.3(2), 357
CLI. *See* Command-line interface (CLI)
Click fraud, 479
Coaxial cables, 46–47
Cole, Eric, 11
Collector, definition of, 161
Collector placement and architecture, 169–70
  capacity, 169–70

Collector placement and architecture (*cont.*)
  congestion, 169
  reliability, 169
  security, 169
  strategy for analysis, 170
Collector systems, 170–71
  Argus, 171
  flow-tools, 171
  nfdump, 171
  NfSen, 171
  SiLK, 170–71
Collision avoidance and detection, 201–2
Command-and-control channels (C&C)
  in blending network activity, 478–79
  centralized C&C, drawbacks of, 466
  communications in network behavior of malware, 487–90
  distributed, 466–69
  Downadup, 478–79
  hiding, in encryption and obfuscation, 464
  peer-to-peer, 469
  Storm/Waledec peer-to-peer C&C system, 478
Command-line interface (CLI), 266
  console, 349–50
  remote, 350–51
Commercial enterprise tools, 233
Commercial NIDS/NIPS, 262–63
Computer network operations (CNOs), 482–83
Conficker worm, 468, 472–73, 475–76, 478–79, 488, 489
Configuration, NIDS/NIPS
  evidence, 264
  Snort, 269
Configuration, Squid, 377–78
Console, 66–67
  CLI, 349–50
  local logging, 352–53
Consumer-class firewalls, 346
Consumer-class routers, 342
Consumer WAPs, 216–18
  Apple Airport Express, 216–17
  Linksys WRT54G, 217–18
Content-addressable memory (CAM), 25, 52–54, 69, 336–37
Content data, 265
Content filter/filtering, 370, 373–74, 400
Control frames, 204–5
  subtypes, 205
Conversations
  listing, in flow analysis, 109
  TCP (*See* TCP conversations (in case study))
  in tshark, 106–7
  in Wireshark, 106–7
Cookies, 122
Copper cables, 46–47
Correlation
Counter Mode with CBC-MAC Procotol (CCMP), 211
Covert network tunneling, 430–32
  detecting, 438–39
  DNS tunnels, 431–32
  strategies, 430
  TCP sequence numbers, 430–31
Crocker, Steve, 77
CSMA/CA, 202
CSMA/CD, 201–2
Curious Mr. X (case study), 184–97
  analysis (first steps), 185–86
  challenge questions, response to, 196
  DMZ victim, 189–93
  external attacker and port 22 traffic, 186–89
  internal victim (192.30.1.101), 193–94
  next step, 196–97
  overview of, 184–85
  theory of the case, 195–96
  timeline, 194–95

Daemon9, 423, 434, 439
Daemon ports, variable, 472–73
Data carving, 112–20
DATA command in SMTP, 127
Data frames, 205
Decryptors and decryption keys, 464
Department of Homeland Security, 13, 468, 474
Department of Justice (DoJ), 12–15

DHCP. *See* Dynamic Host Configuration Protocol (DHCP)
DHCP RFCs
  2131, 123, 124–25
  3315, 123
  3679 (Unused DHCP Option Codes), 105
Digital evidence. *See* Evidence
Digital Millennium Copyright Act (DMCA), 6
Direct evidence, 12
Directionality of flows, 175
Dirty values in flow record analysis, 173, 174
Dirty word list, 100
Disk cache, Squid, 379
Display filters
  in tshark, 96–97
  in Wireshark, 96–97, 101–3
Distributed caching, 374–75
  ICAP, 374–75
  ICP, 374
Distributed C&C, 465–69
  advantages of, 467–69
  centralized, drawbacks of, 466
  evolution toward, 466–67
  IRC, 465
  peer-to-peer C&C, 469
Distributed denial-of-service (DDoS) attacks, 462, 465
Distributed management, botnets and, 462–63
Distributed Management Task Force (DMTP), 294
Distributed scanning networks, 475
DNS. *See* Domain Name System (DNS)
Documenting evidence, 19–20
Docxtract, 151–52
Domain Name System (DNS), 26, 128–29. *See also* Ann Tunnels Underground (case study)
  covert network tunneling, 431–32
  fast-flux DNS, 479–80
  forensic value, 26
  higher-layer protocols, 128–29
  NULL record queries, 444–59
  queries, 129
  recursion, 129
  stimulus and response, 364
  tunnels, 431–32
  zone transfer, 128
Dow Chemical, 481
Downadup worm
  C&C, 478–79
  W32.Downadup.A, 468, 476, 488, 489
  W32.Downadup.B, 468, 476
  W32.Downadup.C, 468, 472–73, 475–76
Draft Internetwork Protocol Specification, 37
Dumpcap, 64–65
Dynamic Host Configuration Protocol (DHCP), 122–25
  exchange in, 124
  forensic value of, 26
  lease assignment logs, 5, 6
  MAC addresses in, 123–24
  purpose of messages, 124–25
  RFCs (*See* DHCP RFCs)
  server logs, 358–59
  servers, 26
  traffic, 136–38
Dynamic IP address, 122–23
Dynamic Random-Access Memory (DRAM), 336
Dynamic timing/volume, 475–77

EAP. *See* Extensible Authentication Protocol (EAP)
Eavesdropper, 209–10
802.11 protocol, 50, 51, 202–12
  AES, 211
  authentication, 213
  802.11n in Greenfield mode, 220, 226
  802.1X (*See* 802.1X)
  endianness, 208–9
  frame analysis, 205–6
  frame types, 203–5
  network-byte order, 207
  TKIP, 211
  WEP, 209–11
  WPA, 211
  WPA2, 211

802.1X, 212–13
  EAP, 212–13
  impact on wireless networks, 213
  implications for investigator, 213
Email account monitoring, 157
Email spam, 127, 462, 465, 471, 487
"Emerging Threats," 269
Encapsulating Security Payload (ESP), 428
Encrypted web traffic, 392–400
  access to (*See* Encrypted web traffic access)
  forensic investigators and, 394
  rise in, factors leading to, 393
  TLS/SSL (*See* TLS/SSL-encrypted traffic)
Encrypted web traffic access, 396–400
  intercepting proxy, 398–400
  server's private key, 396–98
Encryption and obfuscation, 463–65
  C&C channels, hiding, 464
  control, maintaining, 464–65
  IDS/antivirus evasion, early, 464
  modern, 464
Encryption keys, 50–51
Endianness, 208–9
  big-endian and, 206, 207
  frame analysis, 205–6
  Gulliver's Travels and, 205–6
  little-endian and, 206, 207
  mixed-endian, 208–9
End of file (EOF), 109
Enterasy, 167, 262
Enterprise-class firewalls, 345–46
Enterprise-class routers, 341–42
Entity Tag (ETag), 373
Environment, obtaining information on, 18
ESS capabilities, 223
EtherApe, 181–82
Etherleak, 120
European Organization for Nuclear Research (CERN), 120–21
Event logging. *See also* Logging
  Linux, 297–300
  Microsoft Windows, 292–96
Events of interest in analysis of evidence, 21
Evidence
  acquiring (*See* Evidence acquisition)
  admissibility of, 17
  analyzing (*See* Evidence analysis)
  best, 11–12
  business records, 14–15
  circumstantial, 12–13
  collecting (*See* Evidence collection)
  concepts in, 9–22
  content, 16
  definition of, 9
  direct, 12
  forensic value of (*See* Evidence sources)
  hearsay, 13–14
  intercepting (*See* Evidence interception)
  investigative strategies, 9–22
  network-based, 15–22
  off-system, 376
  prioritization of, 19
  privacy, 16
  real, 10–11
  reporting, 21–22
  seizure of, 17
  storage, 16
  volatile, 376
  WAP, 218–19
Evidence acquisition, 16, 45–72
  active, 45, 65–72
  conclusion, 72
  inspection without access, 70–71
  interactive, 45
  interfaces, 66–70
  NIDS/NIPS, 264–68
  passive, 45
  strategy, 71–72
  traffic acquisition software, 54–65
  wireless passive, 221–22
Evidence analysis, 20–21
  correlation, 20–21
  corroboration, 21
  events of interest, 21
  interpretation, 21
  recovery of additional evidence, 21
  timeline, 21
Evidence collection, 19–20
  capturing, 20
  documenting, 19–20

  storing/transporting, 20
  tips for, 20
Evidence interception, 46–54
  cables, 46–49
  hubs, 51–52
  radio frequency, 50–51
  switches, 52–54
Evidence sources, 23–29
  application servers, 29
  authentication servers, 27
  cabling, 24
  central log servers, 29
  DHCP servers, 26
  DNS servers, 26
  firewalls, 27–28
  routers, 25–26
  switches, 25
  web proxies, 28–29
  wireless networking, 24–25
Evil Bit set, 63
Evil systems, 277, 403, 441
Evil Twin, 227–28
Evolution-Data Optimized (EV-DO) wireless network, 213
Expiration in caching, 372
Expires header, 372
Exploitation, direct network-base, 485, 486, 487
Exporting fields, 92–95
Extensible Authentication Protocol (EAP), 212–13
  Lightweight Extensible Authentication Protocol (LEAP), 213
  Protected Extensible Authentication Protocol (PEAP), 213
  Transport Layer Security (EAP-TLS), 213

Facebook, 479
FBI, 13, 468, 474
Federal Communications Commission (FCC), 50, 200
Federal Rules of Evidence (FRE), 10
  best evidence, 11
  business records, 14
  hearsay, 13
Fiber optic taps, 48–49
Fidelity alerts, 261
File carving
  attachment, in SMTP, 146–47
  data, 112–20
  in TCP conversations, 495–98, 510–13
Filters/filtering. *See also* Packet filtering
  content, 373–74
  display, 96–97
  in flow record analysis, 173–74
  URI, 371, 373
Findsmtpinfo.py, 130–31, 152–54
Fingerprinting, 176–77
Firewalls, 27–28, 344–48. *See also* Ann's Coffee Ring (case study)
  ACLs, 359–60
  application proxies and, 345
  consumer-class, 346
  diagnostic commands, 357–58
  enterprise-class, 345–46
  forensic value of, 28
  investigating, reasons for, 344
  logs in L0ne Sh4rk's Revenge (case study), 325–28
  NAT-ing, 345–46
  network-based evidence, 27–28
  off-system, 348
  packet filters and, 344
  persistent, 347
  roll-your-own, 346
  session-layer proxies and, 345
  SO/HO, 346
  storage in, 336
  volatile, 347
Flags in flow record data, 175
Flow, definition of, 105
Flow analysis, 103–20
  definition of, 104
  record (*See* Flow record analysis)
Flow analysis techniques, 109–20
  export TCP flow, 110–12
  file and data carving, 112–20
  list conversations, 109
  list TCP flows, 110

Flow analysis tools, 105–9
  pcapcat, 107–8
  tcpflow, 107
  tcpxtract, 108–9
  tshark conversations, 106–7
  Wireshark, 105–7
Flow-dscan, 179
Flow export (transport-layer protocols), 168
Flow-nfilter, 179
Flow record
  analyzing (*See* Flow record analysis)
  data elements, 175
  definition of, 160
  flags, 175
  information, 265
  ports, 175
  processing (*See* Flow record processing system)
  protocols, 175 (*See also* Flow record export protocols)
Flow record analysis, 172–82
  goals and resources, 172
  starting indicators, 173
  techniques (*See* Flow record analysis techniques)
  tools (*See* Flow record analysis tools)
Flow record analysis techniques, 173–77
  activity pattern matching, 173, 175–77
  baselining, 173, 174
  dirty values, 173, 174
  filtering, 173–74
Flow record analysis tools, 177–82
  Argus, 179
  EtherApe, 181–82
  flow-tools, 178–79
  FlowTraq, 179–80
  nfdump, 180–81
  NfSen, 181
  SiLK, 177–78
Flow record export protocols, 166–68
  IPFIX, 167
  NetFlow, 166–67
  sFlow, 167–68
  transport-layer protocols and, 168
Flow record processing system, 161–82
  analysis, 172–82
  collectors and aggregators, 168–71
  flow record export protocols, 166–68
  sensors, 161–66
Flow sensing. *See* Sensors
Flow-tools suite, 171
FlowTraq, 179–80
"Follow TCP Stream" function in Wireshark, 105–6, 506, 507
Footers, 108
Footprints, 8–9
Forward proxy, 370
Frame analysis, 802.11, 205–6. *See also* Endianness
Frame types, 802.11, 203–5
  control frames, 204–5
  data frames, 205
  management frames, 203–4
FRE. *See* Federal Rules of Evidence (FRE)
Full-featured control, 463

General rule options, 271
Generator ID (GID), 273
Generic Routing Encapsulation (GRE), 425
Google, 33–34, 301, 446, 480, 481, 488, 513
Grant, Rebecca, 481
Greenfield mode (GF), 220, 226
Gudjonsson, Kristinn, 107, 129
Guénichot, Franck, 91, 93, 130
GUI interfaces, 266
Gulliver's Travels (Swift), 205–6

Hacked Government Server (real-world case), 7–8
  potential ramifications, 7
  questions, 7
  results, 8
  technical approach, 7–8
Hacker, Alyssa P., 42
HackMe, Inc. (case study), 236–56
  associated stations, 241–42
  bad actor, possible, 250–51
  Beacon frames, 236–37
  challenge questions, response to, 253–55

  filter on WAP-announcing management frames, 237–38
  management frames, 248–50
  next steps, 255–56
  overview of, 236
  patterns and time frames, 245–47
  quick-and-dirty statistics, 242–48
  stimulus and response, 252–53
  theory of the case, 252–53
  timeline, 247–48, 251–52
  WAP, inspecting, 236–42
  WEP Cracking Attack, 253
  WEP-encrypted data frames, 242–44
  WLAN, inventory of stations on, 238–40
  WLAN encryption, 240–41
Ham, Jonathan, 425
Hard drive, 336
Headers, 108
Health Information Technology for Economic and Clinical Health (HITECH) Act, 4
Health Insurance Portability and Accountability Act (HIPAA), 4, 292, 393
Hearsay, 13–14
HELO command in SMTP, 127
Hexadecimal values associated with protocols, 83–84
Hex editors, 98–99
Hidden node, 202, 204
Higher-layer analysis tools, 129–31
  findsmtpinfo.py, 130–31
  multipurpose tools, 132–33
  NetworkMiner, 131
  oftcat, 129
  small specialized tools, 131–32
  smtpdump, 130
Higher-layer protocols, 120–29
  analyzing (*See* Higher-layer analysis tools)
  DHCP, 122–25
  DNS, 128–29
  HTTP, 120–22
  SMTP, 126–28
Higher-layer traffic analysis. *See* Higher-layer protocols
Higher-level protocol awareness, 259–60
  normalization, 260
  protocol reassembly, 259–60
Hjelmvik, Erik, 131, 397
Honeynet Project, 478–79, 480
Hospital Laptop Goes Missing (real-world case), 4–6
  potential ramifications, 4
  questions, 4
  results, 5–6
  technical approach, 4–5
Host baselines, 174
Host intrusion detection/prevention systems (HIDS/HIPS), 258
Hping3, 433–34
HTTP. *See* Hypertext Transfer Protocol (HTTP)
Hubs, 51–52
Huitema, C., 426
Hypertext, 121
HyperText Markup Language (HTML), 121
Hypertext Transfer Protocol (HTTP), 120–22
  analysis, in TCP conversations, 508–10
  messages, 121
  methods defined by RFC 2616, 121–22
  reason phrase, 122
  status code, 122
  TCP conversations (in case study), 508–10

ICMP. *See* Internet Control Message Protocol (ICMP)
ICMP tunneling, 432–39
  analyzing (*See* ICMP tunneling analysis)
  hping3, 433–34
  implications for the investigator, 438–39
  IP and, 39
  Loki, 434, 439
ICMP tunneling analysis, 434–38
  attack, 435–36
  packet capture analysis, 436–38
IDG News Service, 468
IEEE. *See* Institute of Electrical and Electronics Engineers (IEEE)

IETF. *See* Internet Engineering Task Force (IETF)
Incident, obtaining information on, 17–18
Induction coils, 48
Information, obtaining, 17–18
  on environment, 18
  on incident, 17–18
“INFO Web Bug” alert, 283–84
Initialization vectors (IVs), 228
Inline network taps, 47–48
InMon Corporation, 167–68
Inspection without access, 70–71
  port scanning, 71
  vulnerability scanning, 71
InSSIDer, 231
Institute of Electrical and Electronics Engineers (IEEE), 50. *See also* 802.11 protocol
  CSMA/CA, 202
  CSMA/CD, 201–2
  IEEE 802.1Q, 424
  LAN/MAN Standards Committee, 78
  Layer 2 protocol series, 201–12
  reasons for layers, 201
  Standards Association (IEEE-SA), 78
Inter0ptic Saves the Planet, Part 1 (case study), 276–87
  “INFO Web Bug” alert, 283–84
  next steps, 287
  overview of, 276–77
  packet analysis, initial, 278–79
  Snort alert analysis, 277
  Snort rule analysis, 279–81
  suspicious file from Snort capture, 281–82
  “Tcp Window Scale Option” alert, 284–85
  theory of the case, 286–87
  timeline, 285–86
Inter0ptic Saves the Planet, Part 2 (case study), 402–20
  challenge questions, response to, 418–19
  next steps, 419–20
  overview of, 402–3
  pwny.jpg analysis, 403–5
  Squid access.log file, 408–11
  Squid cache analysis, further, 411–15
  Squid cache page extraction, 405–8
  theory of the case, 417–18
  timeline, 415–17
Intercepting proxy, 398–400
Intercepting traffic in cables, 47–49
Inter Client Basic Messages (ICBM), 88, 91, 102, 103
Interfaces, 66–70, 348–51
  console, 66–67
  console CLI, 349–50
  proprietary, 70, 351
  remote CLI, 350–51
  SCP and SFTP, 67
  SNMP, 68–69, 351
  SSH, 67
  Telnet, 68
  TFTP, 70
  web, 70, 348–49
International Organization for Standardization (ISO), 31, 78
Internet Access Monitor, 381
Internet Architecture Board (IAB), 77
Internet Assigned Numbers Authority (IANA), 40, 77, 85
Internet Cache Protocol (ICP), 374
Internet Content Adaptation Protocol (ICAP), 374–75
Internet Control Message Protocol (ICMP), 39. *See also* ICMP tunneling
Internet Engineering Task Force (IETF), 31, 76–78, 201. *See also* Requests for comments (RFCs)
  EAP, 212–13
  GRE protocol, 425
  IPFIX standard, 166, 167
  ISO 8601 compliance standards, 298
  Teredo, 426
  TLS protocol, 394, 429
Internet Key Exchange (IKE), 427–28
Internet Message Access Protocol (IMAP), 141
Internet Protocol (IP), 37–41. *See also* IP addresses
  characteristics of, 39
  as connectionless protocol, 38

  header, 37
  ICMP and, 39
  packet, 37
  specification, 37
Internet Protocol Security (IPsec), 427–28
Internet Protocol Suite, 35–44
  history and development of, 36–37
  TCP and, 41–43
  UDP and, 43–44
Internet Relay Chat (IRC), 465
Internet Society (ISOC), 77
Internet Standards. *See* Requests for comments (RFCs)
Internetworking, principles of, 30–35
  OSI Model, 31–35
  protocols, 30–31
Inter-Switch Link (ISL), 424–25
Intrusion detection systems (IDSs), 257–58, 464
  reports, 6, 7, 14
Intrusion prevention systems (IPSs), 257–58
Investigative strategies, 3–22
  conclusion, 22
  evidence, 9–22
  footprints, 8–9
  real-world cases, 3–8
Iodine, 432
IP. *See* Internet Protocol (IP)
IP addresses. *See also* IPv4; IPv6
  dynamic, 122–23
  MAC-to-IP mappings, 338, 340, 343, 358
  source and destination, 175, 176
  static, 122
IP Flow Information Export (IPFIX), 162, 164, 166, 167–68, 170–71, 177, 179
IP packet analysis, tunneled, 451–54
  encapsulated protocol type, 453–54
  IP packet length, 452–53
  quest for, 446–50
  source and destination IPv4 addresses, 451–52
IP packets, 446–50
IPv4, 39, 40–41
  IPv6 over, with Teredo, 425–26
  NAT traffic, 426
  protocol identification, 84
  source and destination, in tunneled IP packet analysis, 451–52
IPv6, 39, 40–41
  in hexadecimals, 41
  over IPv4 with Teredo, 425–26
  protocol identification, 84

Javascript, 265, 464, 509–10, 513–15
JPEG
  cached, 389–90, 392
  suspicious, 281–83, 286–87
Juniper, 163, 167, 342, 429, 481, 483

Kang, B. B. H., 470
Keys, Squid cache, 380
KisMAC, 232–33
Kismet, 232
Koobface worm, 479

L0ne Sh4rk's Revenge (case study), 318–34
  activity following compromise, 324–25
  analysis, first steps in, 319
  authentication failure, 319–22
  challenge questions, response to, 332–33
  firewall logs, 325–28
  internal victim, 328–30
  next steps, 333–34
  overview of, 318–19
  successful logins, 323–24
  targeted accounts, 322–23
  theory of the case, 332
  timeline, 330–31
Laptop tracking software, 5
Last-Modified header, 373
Least-recently-used (LRU) algorithm, 379
Legacy equipment, 210
Libpcap, 55
Lightweight Extensible Authentication Procotol (LEAP), 213
Linksys WRT54G router, 23, 216, 217–18, 342, 346
Linux
  AirPcap USB adapter, 222
  AirPort utility and, 351

Linux (*cont.*)
  apcupsd, 304
  ARP cache, 338
  BackTrack Linux, 51
  command-line tools, 381, 383–84, 389, 403, 444
  console connection, example of, 349–50
  etc/passwd file on, 274
  event logging (*See* UNIX/Linux event logging)
  "file" command, 390
  iptables, 336, 342, 346
  Kismet and, 232
  MARS and, 310
  ROM, 336
  "root" account, 319, 332
  "screen" command, 67, 349
  shell commands, 383, 384, 385
  SNARE and, 310
  Snort files and directories, 269
  switch 802.11 interface into infrastructure mode, 228
  TCP/UDP port numbers, 85
  Ubuntu Linux server, 297, 299–300, 338, 361–62, 365, 455
  uniq tool, 238
  ZoneMinder, 303
Lisiecki, Philip, 303
Local area network (LAN), 6
Local logging, 352–53
  buffered, 353
  console, 352–53
  network log architecture, 306
  terminal, 353
Logging, 352–55. *See also* Logs
  AAA, 355
  DHCP, 358–59
  event (*See* Event logging)
  firewall log analysis, 360–64
  local (*See* Local logging)
  SNMP, 353–54
  syslog, 354
Logins
  failed, 319–22
  successful, 323–24
Logs, 291–334. *See also* Logging; Network log architecture
  aggregation and analysis tools, 309–10
  application, 300–302
  conclusion, 317
  forensics relating to, 311–17 (*See also* OSCAR methodology)
  L0ne Sh4rk's Revenge (case study), 318–34
  laundry event, 303
  lease assignment, 5, 6
  network equipment, 305
  operating system, 292–300
  physical device, 302–5
  server, DHCP, 358–59
  SSH, 8
  TLS, 307
  WAP, 5
  web proxy, 5
Lojack for Laptops, 5
Loki, 423, 434, 439
Lua plugin, 81, 91–92, 93

MAC addresses. *See also* Ann's Coffee Ring (case study); HackMe, Inc. (case study)
  ARP table, 338, 340, 343
  CAM table, 337, 340
  control frames, 205
  destination station, 229
  in DHCP, 123–24
  802.11 network adapters, 51
  flooding, 53–54
  locating, 229, 231, 232
  MAC-to-IP mappings, 338, 340, 343, 358
  management frames, 203–4
  randomized scanning, 473
  Skyhook to get GPS coordinates of, 234
  spoofed scanning, 474
  switches, 52–54, 337
  tracing, 6–7
  WAP, 215, 218, 219, 222
MAC OS X, 33, 163, 232, 297, 351
Magic numbers, 108
MAIL command in SMTP, 127
Mail delivery agent (MDA), in SMTP, 126

Mail eXchanger (MX), in SMTP, 126
Mail submission agent (MSA), in SMTP, 126
Mail transfer agent (MTA), in SMTP, 126
Mail user agent (MUA), in SMTP, 126
Malware forensics, 461–517
  Ann's Aurora (case study), 492–517
  APT, 480–84
  botnets, 462–63
  distributed C&C systems, 465–69
  encryption and obfuscation, 463–65
  fast-flux DNS, 479–80
  future of, 491
  goals of, 461
  metamorphic network behavior, 472–77
  network activity, blending, 477–79
  network behavior of malware, 484–90
  self-updates, automatic, 469–71
  social networking sites and, 479, 485, 487, 488
  trends in, 462–84
Managed switches, 339
Management frames, 203–4
  subtypes, 204
Management information base (MIB), 69
Many to many IP addresses, 176
Many to one IP addresses, 176
Mapping ports, 338, 340, 343, 358
Marlinspike, Moxie, 399–400
Maximum transmit unit (MTU), 228
McMillan, Bob, 468–69
Media access control addresses. *See* MAC addresses
Memory cache, Squid, 380–81
Message body, 121
Message header, 121
Metadata options, 271
Metamorphic network behavior, 472–77
  daemon ports, variable, 472–73
  propagation strategies, multiple, 472
  scanning for new targets, 473–77
Microsoft
  IE6, 509, 513
  ISA, 381, 382
  MS-CHAP, 213
  online library of technical specifications, 78
  Operation b49, 471
  Operation b49, 471
  Remote Desktop Protocol, 192
  SQL servers, 486
  WinRM, 294–95
  WS-Management, 294–95
Microsoft Windows
  AirPcap software, 221–22
  ARP cache, 338
  event logging (*See* Microsoft Windows event logging)
  MARS and, 310
  NetStumbler, 231
  Server 2008, 295, 455
  SNARE and, 310
  Windows 7, 231, 295, 296
  Windows executable files, 494, 501, 505, 514–15
  Windows NT, 292, 293, 509
  Windows Server 2003 R2, 295
  Windows Vista, 292, 293, 294, 295, 455
  Windows XP, 293, 294, 295, 509, 513, 514
  WinDump, 59
Microsoft Windows event logging, 292–96
  Event Log Service and Event Viewer, 293
  example of, 295–96
  Windows Eventing 6.0, 293–95
  workstations.log, 318, 330–31
Miller, Damien, 163
Mixed-endian, 208–9
Monitor mode, 51
Morgan Stanley, 481
Mozilla, 478
MyDoom self-mailer worm, 470
MySpace, 479

Name servers, 26
NAT. *See* Network Address Translation (NAT)
National Vulnerability Database, 513
Nazario, Joe, 478
Nelson, Ted, 121
NetBee library, 79

NetFlow, 166–67
Net-SNMP suite, 351
NetStumbler, 231
Network activity in malware, blending, 477–79
  Downadup C&C, 478–79
  social networking sites, 479
  Storm/Waledec C&C protocol evolution, 478
Network Address Translation (NAT)
  in Curious Mr. X case study, 185
  firewalls, 345–46
  IPv4 traffic, 426
  NAT traversal (NAT-T) techniques, 426
  routers, 341–42
  WAPs, 214, 216
Network-based evidence, 15–22. *See also* Evidence
  challenges relating to, 16–17
  definition of, 15
  OSCAR methodology in, 17–22
Network baselines, 174
Network behavior of malware, 484–90
  C&C communications, 487–90
  payload behavior, 490
  propagation, 485–87
Network-byte order, 207
Network devices and servers
  Ann's Coffee Ring (case study), 356–68
  conclusion, 355
  firewalls, 344–48
  interfaces, 348–51
  logging, 352–55
  logs, 291–334
  routers, 340–43
  storage media, 336
  switches, 336–40
  web proxies, 369–420
Network equipment logs, 305
Network forensics investigative methodology. *See* OSCAR methodology
Network intrusion detection/prevention systems. *See* NIDS/NIPS
Network log architecture, 306–11
  centralized, 307–8
  local, 306
  log aggregation and analysis tools, 309–10
  remote decentralized, 306–7
  remote logging pitfalls and strategies, 308–9
NetworkMiner, 131, 150–51
Network Situational Awareness (NetSA) group, 170
Network Time Protocol (NTP), 85
Network tunneling, 423–59
  Ann Tunnels Underground (case study), 441–59
  conclusion, 439–40
  confidentiality (*See* Network tunneling for confidentiality)
  covert, 430–32
  function of (*See* Network tunneling for functionality)
  ICMP tunnels, 432–39
  IP packets, 446–50 (*See also* IP packet analysis, tunneled)
  TCP segment analysis (*See* TCP segment analysis, tunneled)
Network tunneling for confidentiality, 427–30
  implications for the investigator, 430
  IPsec, 427–28
  TLS and SSL, 428–29
Network tunneling for functionality, 423–27
  GRE, 425
  implications for the investigator, 426–27
  IPv6 over IPv4 with Teredo, 425–26
  ISL, 424
  VLAN trunking, 424
Nfdump, 171, 180–81
NfSen, 171, 181
Ngrep, 97–98, 100–101
NIDS/NIPS, 27, 217, 258–87
  commercial, 262–63
  conclusion, 275
  detection modes, 261
  in encryption and obfuscation, 463–65
  evidence (*See* NIDS/NIPS evidence)
  function of (*See* NIDS/NIPS functionality)

  InterOptic Saves the Planet (case study), 276–87
  interfaces, 266
  investigating, reasons for, 258
  packet logging, 267–68
  roll-your-own, 263
  Snort, 268–75
  Snort rule language, 269–72
  types of, 262–63
NIDS/NIPS detection modes, 261
  behavioral analysis, 261
  protocol awareness, 261
  signature-based analysis, 261
NIDS/NIPS evidence
  acquisition, 264–66
  activities correlated across multiple sensors, 265
  alert data, 265
  available, 267–68
  configuration, 264
  content data, 265
  forensic value of, 27
  packet header and/or flow record information, 265
  types of, 264–65
NIDS/NIPS functionality, 258–61
  alerts, 260–61
  higher-level protocol awareness, 259–60
  sniffing, 259
NIDS/NIPS interfaces, 266
  CLI interfaces, 266
  GUI interfaces, 266
  off-system logging, 266
Nimda worm, 472
Nonpayload detection rule options, 271–72
Nonvolatile Random-Access Memory (NVRAM), 336
Normalization, 260
Northrup Grumman, 480–81
Nunnery, C., 470

Obfuscation. *See* Encryption and obfuscation
Off-system evidence, 376
  firewalls, 348
  logging, 266
  routers, 343
  switches, 340
  WAPs, 219
  web proxies, 376
Oftcat, 129
Ohio State University, 475
One to many IP addresses, 176
One to one IP addresses, 176
Open Pluggable Edge Services (OPES), 370
Open System for Communication in Realtime (OSCAR) protocol, 78, 88, 89
  File Transfer (OFT), 88, 94
Open Systems Interconnection (OSI) Model, 31–35
  benefits of, 33
  layers in, 31–32, 38
  web surfing example using, 33–35
Operating system logs, 292–300
  Microsoft Windows event logging, 292–96
  UNIX/Linux event logging, 297–300
Operation Aurora, 480–81, 483, 513. *See also* Ann's Aurora (case study)
Operation b49, 471
"Operation: Bot Roast," 468
Optical cables, 47
Optical time-domain reflectometers (OTDRs), 48–49
Organizational Unique Identifier (OUI), 123–24, 136, 137
OSCAR File Transfer (OFT), 88, 94
OSCAR methodology, 17–22
  Obtain information, 17–18, 311–13
  Strategize, 18–19, 313–14
  Collect evidence, 19–20, 314–16
  Analyze, 20–21, 316–17
  Report, 21–22, 317
OSI Model. *See* Open Systems Interconnection (OSI) Model
Overnet/eDonkey protocol, 469

Packet, 37
Packet analysis, 95–103
  in Ann's Rendezvous (case study), 157
  capture, 436–38
  definition of, 96
  in ICMP tunnel analysis, 436–38
  techniques (*See* Packet analysis techniques)
  tools (*See* Packet analysis tools)
  tunneled (*See* Network tunneling)
Packet analysis techniques, 99–103
  packet filtering, 101–3
  parsing protocol fields, 101
  pattern matching, 99–101
Packet analysis tools, 96–99
  hex editors, 98–99
  ngrep, 97–98
  Wireshark/tshark display filters, 96–97
Packet Details Markup Language (PDML), 79
Packet filtering, 101–3
  by bit value, 58
  with BPF language, 101
  by byte value, 57–58
  firewalls and, 344
  with tcpdump, 61–63
  techniques, 101–3
  with Wireshark display filters, 101–3
Packet header, 265
Packet logging, NIDS/NIPS, 267–68
Packet Summary Markup Language (PSML), 79
Parsing protocol fields, 101
Passive evidence acquisition, 45
Password Authentication Protocol (PAP), 213
Passwords. *See* Logins
Pattern matching, 99–101
Payload behavior, 490
Payload detection rule options, 272
Pcapcat, 107–8
Peer-to-peer (P2P)
  C&C, 469
  filesharing, 6
Perl-compatible regular expressions (PCRE), 271, 272
Permutation scanning, 474
Persistent evidence, 375–76
  firewalls, 347
  routers, 343
  switches, 340
  WAPs, 219
  web proxies, 375–76
Phrack magazine, 434
Physical device logs, 302–5
  camera logs, 303–4
  uninterruptible power supply logs, 304
Pidgeon sniffing, 46
Pietrosemoli, Ermanno, 50
Point-to-Point Protocol (PPP), 212–13
Point-to-Point Protocol over Ethernet (PPPoE), 213
Politecnico di Torino, 79
Porras, Phillip, 488
Ports
  blocking, 472–73
  daemon, variable, 472–73
  in flow record data, 175
  MAC-to-IP mappings, 338, 340, 343, 358
  mapping, 338, 340, 343, 358
  mirroring, 53–54, 166
  scanning, 71
  TCP (*See* TCP ports)
  wireless port knocking, 227
Post-detection rule options, 272
Postel, John, 37
Premaster secret, 396
Pre-shared keys (PSKs), 211
Pretty Park worm, 465
Privacy, 16
Propagation
  identifying, 486–87
  in metamorphic network behavior, 472
  in network behavior of malware, 485–87
  vectors for, 485
Proprietary interfaces, 70, 351
ProQueSys, 179, 180
Protected Extensible Authentication Protocol (PEAP), 213
Protocol analysis, 76–95
  definition of, 76
  IEEE-SA, 78
  information on, 76

- ISO, 78
- researchers, 78–79
- RFCs, 76–77
- techniques (*See* Protocol analysis techniques)
- tools (*See* Protocol analysis tools)
- vendors, 78

Protocol analysis techniques, 82–95
- Ann's Bad AIM scenario, 83–95
- exporting fields, 92–95
- protocol decoding, 90–92
- protocol identification, 82–90

Protocol analysis tools, 79–82
- PDML, 79
- PSML, 79
- tshark, 81–82
- Wireshark, 79–81

Protocols, 30–31. *See also* Internet Protocol (IP); Internet Protocol Suite
- ASCII values associated with, 83–84
- binary values associated with, 83–84
- connectionless, 38, 39, 43, 105, 168, 169
- connection-oriented, 43, 122
- decoding, 90–92
- definition of, 30, 31
- 802.11 protocol suite, 202–12
- in flow record data, 175
- flow record export (*See* Flow record export protocols)
- hexadecimal values associated with, 83–84
- higher-layer (*See* Higher-layer protocols)
- higher-level protocol awareness, 259–60
- identification, 82–90
- IEEE Layer 2 protocol series, 201–12
- in internetworking, 30–31
- mismatch, 30
- reassembly in higher-level protocol awareness, 259–60
- transport-layer, 168

Pwny.jpg, 403–5

PySiLK, 178

QoSient, LLC, 163

Queries, DNS, 129

Qwest DSL modem/router, 342, 346

Ra, 179

Rackspace, 481

Racluster, 179

Radio frequency, 50–51

Ragraph, 179

Ragrep, 179

Rahisto, 179

Randomized scanning, 473–74

Rasort, 179

Raw traffic, 209

RCPT command in SMTP, 127

Read-Only Memory (ROM), 336

Real evidence, 10–11

Real-world cases, 3–8
- Catching a Corporate Pirate, 6–7
- Hacked Government Server, 7–8
- Hospital Laptop Goes Missing, 4–6

Reason phrase, HTTP, 122

Received Signal Strength Indication (RSSI), 231

Recursion, DNS, 129

Red Line Software, 381

Reed, David P., 44

Regional Internet Registries (RIRs), 40

Remote access Trojans (RATs), 463

Remote CLI, 350–51

Remote decentralized network log architecture, 306–7

Remote logging pitfalls and strategies, 308–9
- confidentiality, 309
- integrity, 309
- reliability, 308
- time skew, 309

Remote Switched Port Analyzer (RSPAN), 53

Reporting evidence, 21–22

Requests for comments (RFCs)
- 10 (Documentation Conventions), 77
- 527 (ARPAWOCKY), 75
- 675 (Specification of Internet Transmission Control Program), 36, 42
- 783 (TFTP Protocol (revision 2)), 70
- 791 (Internet Protocol), 37, 63
- 792 (Internet Control Message Protocol), 39, 129

Requests for comments (RFCs) (*cont.*)
  793 (Transmission Control Program), 37
  854 (Telnet Protocol Specifications), 68
  855 (Telnet Option Specifications), 68
  1035 (Domain names—implementation and specification), 128
  1149 (Standard for the transmission of IP datagrams on avian carriers), 40
  1350 (TFTP Protocol (revision 2)), 70
  1918 (Address Allocation for Private Internets), 40
  2026 (The Internet Standards Process – Revision 3), 77
  2616 (Hypertext Transfer Protocol—HTTP/1.1), 121–22
  2722 (Traffic Flow Measurement: Architecture), 159
  2784 (Generic Routing Encapsulation), 78
  3176 (InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks), 167–68
  3514 (The Security Flag in the IPv4 Header), 63
  3954 (Cisco Systems NetFlow Services Export Version 9), 167
  3955 (Evaluation of Candidate Protocols for IPFIX), 169
  4677 (The Tao of IETF: A Novice's Guide to the Internet Engineering Task Force), 77
  4954 (SMTP Service Extension for Authentication), 128
  4960 (Stream Control Transmission Protocol), 161
  5101 (Specification of the IPFIX Protocol for the Exchange of IP Traffic Flow Information), 167
  5103 (Bidirectional Flow Export Using IPFIX), 167
  5321 (Simple Mail Transfer Protocol), 126
  5473 (Reducing Redundancy in IPFIX), 167
  ARPANET and, 76–77
  canonical repository of, 77
  definition of, 77
  DHCP (*See* DHCP RFCs)
  HTTP methods defined by, 121–22
  IETF approval of, 76–77
  maturity levels, 77
Reserved bit, 63
Reverse proxy, 370, 470
Reverse proxy systems, 470–71
RFCs. *See* Requests for comments (RFCs)
Ritter, Jordan, 97, 98
Robust security network associations (RSNAs), 211
Robust security networks (RSNs), 211–12
Rogue system, 366
Rogue wireless access points, 225–27
  Bluetooth access point, 226–27
  changing the channel, 225–26
  802.11n in Greenfield mode, 220, 226
  wireless port knocking, 227
Roll-your-own firewalls, 346
Roll-your-own NIDS/NIPS, 263
Roll-your-own routers, 342
Rossi, Jeremy, 130–31
Rough consensus and running code, 77
Routers, 25–26, 340–43
  consumer-class, 342
  enterprise-class, 341–42
  investigating, reasons for, 341
  NAT-ing, 341–42
  off-system, 343
  persistent, 343
  roll-your-own, 342
  storage in, 336
  volatile, 343
RSA Security, 213, 396–97, 465, 470, 483–84
Rsyslogd, 298–99
Rule body, Snort, 271–72
Rule header, Snort, 270
Rule language, Snort, 269–72
Rule options, Snort, 271–72
Rwcount, 178
Rwcut, 178
Rwfilter, 177–78

Rwidsquery, 178
Rwpmatch, 178
Rwstats, 178
Rwuniq, 178

SANS Institute, 303
Santorelli, Steve, 477
Schneier, Bruce, 469
Secure Copy Protocol (SCP), 67
Secure Shell (SSH), 8, 67
Secure Socket Layer (SSL). *See also* TLS/SSL-encrypted traffic
  encrypted web interfaces and, 70
  function of, 394–96
  network tunneling for confidentiality, 428–29
  protocol identification and, 86
  remote logging and, 309
  rsyslog and, 298
  session-layer proxies and, 345
  stripping attacks, 228
  syslog and, 354
SecureWorks, 465, 469
Security Associations (SAs), 427–28
Seizure of evidence, 17
Self-updates, automatic, 469–71
  authenticated updates, 470
  early systems, 469
  success and failure, 471
  updating system, 470–71
Sensor placement, 164–65
  capacity, 165
  duplication, 164
  perimeter *vs.* internal traffic, 165
  resources, 165
  time synchronization, 164–65
Sensors, 161–66
  deploy additional sensors, 166
  environmental modification, 165–66
  leverage existing equipment, 165–66
  network equipment, 162
  placement of (*See* Sensor placement)
  software (*See* Sensor software)
  standalone appliances, 162
  types of, 162
  upgrade network equipment, 166
Sensor software, 163–64
  Argus, 163
  softflowd, 163–64
  yaf, 164
Server logs, DHCP, 358–59
Server's private key, 396–98
Service Set Identifiers (SSIDs), 204
Session-layer proxies, firewalls and, 345
SFlow, 167–68
Shell commands, Linux, 383, 384
Shutko, Alexandr, 79
Signal strength, 231–33
  KisMAC, 232–33
  Kismet, 232
  NetStumbler, 231
  RSSI, 231
SiLK
  flow record analysis, 177–78
  statistical flow analysis, 170–71
Simple Mail Transfer Protocol (SMTP), 126–28
  analyzing (*See* SMTP analysis)
  Ann's Rendezvous (case study), 135–57
  application logs, 301–2
  authentication, 127–28
  commands, 127
  mail transfer agent, 126
  mail user agent, 126
  RCPT command, 127
  terminology, 126
  transcript, 127
  use of, 126
Simple Network Management Protocol (SNMP), 68–69
  interfaces, 351
  logging, 353–54
  Net-SNMP suite, 351
  NIDS/NIPS traps, 260, 261
  Snort alerts, 269
Sinclair, G., 470
Single pre-shared key (PSK), 50–51
Sixth-byte offset, 57
Skyhook, 233–34
Slammer worm, 473–74, 486
Small office/home office (SO/HO)
  firewalls, 346
  unmanaged switches, 339

Smart switches, 339
Smith, Rick, 391
SMTP. *See* Simple Mail Transfer Protocol (SMTP)
SMTP analysis, 141–46
  attachment file carving, 146–47
  docxtract, 151–52
  findsmtpinfo.py, 152–54
  NetworkMiner, 150–51
  smtpdump, 151–52
  tcpflow, 143–46
  Wireshark, 141–43
Smtpdump, 130, 151–52
Snaplen, 60
Sniffing, 224–25. *See also* Evidence interception
  NIDS/NIPS functionality, 259
  pidgeon, 46
SNMP. *See* Simple Network Management Protocol (SNMP)
Snort ID (SID), 269, 273, 274, 275, 279, 283
Snort in NIDS/NIPS, 268–75. *See also* Inter0ptic Saves the Planet, Part 1 (case study)
  architecture of, 268–69
  configuration, 269
  examples, 273–75
  overview of, 268
  rule body, 271–72
  rule header, 270
  rule language, 269–72
  rule options, 271–72
Social networking sites, malware and, 479, 485, 487, 488
Softflowd, 163–64
SO/HO. *See* Small office/home office (SO/HO)
Solaris, 163, 179, 297, 310, 346
SolarWinds Network Management Software, 351
SonicWALL, 163
Spam, 127, 462, 465, 471, 485, 487
Spectrum analysis in capturing and analyzing wireless traffic, 220–21
Splunk, 310, 382, 383, 384
Spoofed scanning, 474
Squid, 377–81. *See also* Inter0ptic Saves the Planet, Part 2 (case study)
  access logfile, 378
  automated Squid cache extraction, 391–92
  configuration, 377–78
  disk cache, 379
  dissecting a disk cache in web proxy analysis, 384–92
  extracting a cached web object, 385–90
  keys, 380
  memory cache, 380–81
  swap.state, 379–80
Squid Analysis Report Generator (SARG), 382, 383
Squidview, 382
SSH File Transfer Protocol (SFTP), 67
SSL. *See* Secure Socket Layer (SSL)
Sslsniff, 400
Sslstrip, 399–400
Standards-track documents. *See* Requests for comments (RFCs)
Starting indicators in flow record analysis, 173
Static IP address, 122
Statistical flow analysis, 159–97
  collection and aggregation, 168–71
  conclusion, 183
  Curious Mr. X (case study), 184–97
  flow record, definition of, 160
  flow record analysis, 172–77
  flow record export protocols, 166–68
  flow record processing system, 161
  process overview, 160–61
  purposes of, 159–60
  sensors, 161–66
Statistics, definition of, 172
Status code, HTTP, 122
Stevens, Kathryn, 479
Stevens, W. Richard, 35
Stewart, Joe, 465, 469, 478
Storage media, 336
Storing/transporting evidence, 16, 20
Storm worm, 462, 465, 469, 478, 479

"Strategic Command" (STRATCOM), 481
Strategy, investigative. *See* Investigative strategies
Stream Control Transmission Protocol (SCTP), 168
Stream reassembly, 105
Stumbler malware, 474–75
Stuxnet worm, 466–67
Sub7, 463, 472, 491
Sun Tsu, 22
Swap.state, Squid cache, 379–80
Switched Port Analyzer (SPAN), 53, 54, 184–85
Switches, 25, 52–54, 336–40
  ARP, 338
  CAM table, 337
  investigating, reasons for, 337
  managed, 339
  off-system, 340
  persistent, 340
  smart, 339
  storage in, 336
  unmanaged, 339
  volatile, 340
Symantec, 466, 467, 471, 476–77, 478, 483, 486
Syslog, 297, 354
Syslog-ng, 297–98
System for Internet Level Knowledge. *See* SiLK

Targets, scanning for new, 473–77
  distributed scanning networks, 475
  dynamic timing/volume, 475–77
  permutation scanning, 474
  randomized scanning, 473–74
  spoofed scanning, 474
TCP. *See* Transmission Control Protocol (TCP)
TCP conversations (in case study), 495–513
  file carving, 495–98, 510–13
  HTTP analysis, 508–10
  traffic analysis, 502–5
Tcpdump, 59–63
  in capturing and analyzing wireless traffic, 222–24
  fidelity, 60–61
  filtering packets with, 61–63
Tcpflow, 107, 143–46
TCP in flow analysis
  exporting, 110–12
  listing, 110
TCP/IP Illustrated Volume 1 (Stevens), 35
TCP/IP Model, 32
TCP/IP protocol suite. *See* Internet Protocol Suite
TCP ports
  port 20, 196, 197
  port 21, 193, 196, 197, 197 196, 333
  port 22, 185–87, 195, 196, 454–55
  port 25, 126
  port 53, 129
  port 80, 61, 121, 178, 194, 196
  port 143, 141
  port 443, 86, 102, 194, 196
  port 445, 476
  port 514, 196
  port 587, 126, 141
  port 3389, 192, 195, 196
  port 4022, 67
  port 4444, 495
  port 4445, 495, 502
  port 5190, 102, 109, 111
  port 8080, 495, 510
  port 29008, 82
  leveraging port number in protocol identification, 84–86
  values for, possible, 42
TCP segment analysis, tunneled, 454–56
  TCP destination port, 455
  TCP flags, 456
  TCP source port, 454–55
"Tcp Window Scale Option" alert, 284–85
Tcpxtract, 108–9
Team Cymru, 477
Technical fundamentals, 23–44
  conclusion, 44
  Internet Protocol Suite, 35–44

Technical fundamentals (*cont.*)
  internetworking, principles of, 30–35
  network-based evidence, sources of, 23–29
Telnet, 68
Temporal Key Integrity Protocol (TKIP), 211
Tenebro, Gilou, 464, 478, 490
Teredo, IPv6 over IPv4 with, 425–26
Terminal local logging, 353
ThreatExpert, 498, 499
Three-way handshake, 43
Timeline in analysis of evidence, 21
Time magazine, 482
Time to live (TTL), 57, 179, 271, 272, 479–80
Titan Rain, 482–83
TLS. *See* Transport Layer Security (TLS)
TLS/SSL-encrypted traffic, 396–400
  commercial interception tools, 400
  intercepting, 398–400
  Wireshark for decrypting, 397–98
Tools in higher-layer traffic analysis
  multipurpose, 132–33
  small specialized, 131–32
Top-level domains (TLDs), 128, 277, 403, 441, 468
Traffic acquisition software, 54–65
  BPF language, 55–59
  dumpcap, 64–65
  libpcap, 55
  tcpdump, 59–63
  tshark, 64
  WinPcap, 55
  Wireshark, 64
Traffic analysis, 75–287
  Ann's Rendezvous (case study), 135–57
  conclusion, 133–34
  flow analysis, 103–20
  higher-level traffic analysis, 120–33
  NIDS/NIPS, 257–87
  packet analysis, 95–103
  protocol analysis, 76–95
  statistical flow analysis, 159–97
  in TCP conversations, 502–5
  wireless devices and networks, 199–256
Transmission Control Protocol (TCP), 41–43
  characteristics of, 43
  as connection-oriented protocol, 43
  in conversations (*See* TCP conversations (in case study))
  flow analysis, 110–12
  handshake, 31, 188, 499, 502, 504, 505–6, 514
  port values, 42 (*See also* TCP ports)
  segments, 41 (*See also* TCP segment analysis, tunneled)
  sequence numbers in covert network tunneling, 430–31
  TCP RST packets, 190, 502, 505, 506, 515
  TCP SYN ACK packets, 31, 38, 188, 190–92, 431, 499, 502, 505, 515
  TCP SYN packets, 190, 192, 431, 502, 505
  three-way handshake in, 43
  values for ports, 42
  Windows Size, 474
Transmit (Tx) Rate information, 231
Transport-layer protocols, 168
Transport Layer Security (TLS). *See also* TLS/SSL-encrypted traffic
  EAP and, 213
  encrypted web interfaces and, 70
  function of, 394–96
  implementing, 396
  logs and, 307
  network tunneling for confidentiality, 428–29
  protocol identification and, 86
  remote logging and, 309
  rsyslog and, 298
  session-layer proxies and, 345
  stripping attacks, 228
  syslog and, 354
  syslog-ng and, 297
  in web applications, purposes of, 394
  yaf and, 164
Transport mode, 428
Tribe Flood Network (TFN), 462–63
Tribe Flood Network 2000 (TFN2K), 463

Trinoo, 462
Trivial File Transfer Protocol (TFTP), 70
Tshark, 64
  capturing and analyzing wireless traffic, 222–24
  conversations in, 106–7
  display filters, 96–97
  protocol analysis, 81–82
TSL servers, 470–71
Tu, Alan, 391
Tunneling. *See* Network tunneling
Tunnel mode, 428, 429
Twisted pair (TP) cables, 47
Twitter, 479
Type-of-service (TOS), 271, 274

Ubuntu Linux server, 297, 299–300, 338, 361–62, 365, 455
UDP. *See* User Datagram Protocol (UDP)
UDP ports
  port 67, 123
  port 68, 123
Ullrich, Johannes, 303, 304
Undersea cable cuts, 49
Uniform Resource Identifier (URI)
  extract web object from Squid cache, 385–86
  filtering, 373
United States *v.* Simpson, 13
UNIX
  apcupsd, 304
  ARP cache, 338
  etc/passwd file on, 274
  event logging (*See* UNIX/Linux event logging)
  Kismet and, 232
  MARS and, 310
  "root" account, 319
  shell commands, 385
  TCP/UDP port numbers, 85
  timestamps, 382, 384
  Zebra, 342
UNIX/Linux event logging, 297–300
  authentication logs, 299
  auth.log, 318, 319–20, 323, 325, 330–31
  kernal logs, 299–300
  Linux kernal logs, 299–300
  rsyslogd, 298–99
  "sudo" command, 324–25
  syslog, 297, 354
  syslog-ng, 297–98
Unmanaged switches, 339
URI. *See* Uniform Resource Identifier (URI)
User Datagram Protocol (UDP)
  Internet Protocol Suite, 43–44
  port numbers, 84–86

Validation in caching, 372–73
Vampire taps, 48
Vendors, in protocol analysis, 78
Verisign, 394–95
"Victory in Cyberspace" report, 481
Virtual LAN. *See* VLAN
Virtual private networks (VPNs), 427, 429
VirusTotal, 497–98
VLAN
  consumer-class firewalls, 346, 357
  ID (VID), 424
  sensor placement, 165
  switches, 25, 339, 424
  tags, 424
  trunking, 424, 425
  tunneling over, challenge of, 425
Volatile evidence, 376
  firewalls, 347
  routers, 343
  switches, 340
  WAPs, 218–19
  web proxies, 376
Volume of data transferred, 175
Voo Doo (MIT magazine), 42
VPN concentrators, 5
Vulnerability Research Team (VRT), 269–70, 273
Vulnerability scanning, 71

Waledac worm, 464, 470–71, 478, 479, 487, 489–90
WAP evidence, 218–19
  off-system, 219
  persistent, 219
  volatile, 218–19

WAP inspection, 236–42
  associated stations, 241–42
  Beacon frames, 236–37
  filter on WAP-announcing management frames, 237–38
  WLAN, inventory of stations on, 238–40
  WLAN encryption, 240–41
WAPs. *See* Wireless access points (WAPs)
Web interfaces, 70, 348–49
Web proxies, 369–420. *See also* Encrypted web traffic; Squid
  analyzing, 381–92 (*See also* Web proxy log analysis tools)
  conclusion, 401
  evidence in (*See* Web proxy evidence)
  functionality of (*See* Web proxy functionality)
  Inter0ptic Saves the Planet, Part 2 (case study), 402–20
  investigating, reasons for, 369–71
  logs, 5
  types of, 370
Web proxy evidence, 375–76
  forensic value of, 28–29
  obtaining, 376
  off-system, 376
  persistent, 375–76
  volatile, 376
Web proxy functionality, 371–75
  caching, 371–73
  content filtering, 373–74
  distributed caching, 374–75
  URI filtering, 371, 373
Web proxy log analysis tools, 5, 381–84
  Blue Coat Reporter, 381
  Internet Access Monitor, 381
  SARG, 382, 383
  shell commands, Linux, 383, 384
  Splunk, 382, 383, 384
  Squidview, 382
Welchia worm, 470
WEP. *See* Wired Equivalent Privacy (WEP)
Whitelisting, 373
Wi-Fi, 50–51, 200–201. *See also* 802.11 protocol
  frequency ranges, 220
  hardware supporting WPA2, 210
  WPA and WPA2 and, 211
Wi-Fi Protected Access (WPA), 211
Wi-Fi Protected Access 2 (WPA2), 211
WinPcap, 55, 79
Wired Equivalent Privacy (WEP), 51, 209–11
  Cracking Attack, 228, 253
  encryption and, 210–11
  problems in, 209–10
  studying, reasons for, 210
  WEP Cracking, 204, 210, 224, 228, 244, 252–54
Wireless access points (WAPs), 214–19
  consumer, 216–18
  enterprise, 215–16
  evidence (*See* WAP evidence)
  identifying nearby, 229–31
  inspecting (*See* WAP inspection)
  investigating, reasons for, 214
  logs, 5
Wireless Control System (WCS), 233
Wireless devices and networks, 199–256
  attacks on, common, 224–28
  capturing and analyzing, 219–24
  collisions in, 202
  conclusion, 235
  802.11 protocol suite, 202–12
  802.1X, 212–13
  HackMe, Inc. (case study), 236–56
  investigating, reasons for, 200
  locating (*See* Wireless devices and networks, locating)
  types of, 199–200
  WAPs, 214–19
Wireless devices and networks, locating, 229–34
  commercial enterprise tools, 233
  nearby wireless access points, identifying, 229–31
  signal strength, 231–33
  Skyhook, 233–34
  station descriptors, gathering, 229
Wireless intrusion detection systems (WIDSs), 225, 226, 233

Wireless Local Area Network (WLAN), 50, 201
Wireless Location Appliance (WLA), 233, 234
Wireless networking, 24–25. *See also* Wireless devices and networks
Wireless passive evidence acquisition, 221–22
Wireless port knocking, 227
Wireless Positioning System (WPS), 233
Wireless traffic capture and analysis, 219–24
  spectrum analysis, 220–21
  tcpdump, 222–24
  tshark, 222–24
  wireless passive evidence acquisition, 221–22
Wireshark, 64, 79–81
  conversations in, 106–7
  decrypting TLS/SSL-encrypted traffic, 397–98
  display filters, 96–97
  "Follow TCP Stream" function, 105–6, 506, 507
  packet filtering, 101–3
  Protocol Hierarchy Statistics, 442–43, 499, 501
  in SMTP (Ann's Rendezvous case study), 141–43
W95/Babylonia self-mailer worm, 469
W95/Hybris worm, 470
Worms. *See* Botnets
Wright, Joshua, 51, 213, 220
W32/Blaster, 472
W32/Doomjuice, 470
W32.SQLExp., 473–74
W32.Stuxnet Dossier, 467
W32.Waledac, 478, 487
W32.Welchia, 473
W32/Witty, 472

XOR-ing, 464

Yaf (Yet Another Flowmeter), 164

Zero-byte offset, 57
Zombies, 462, 463, 464
ZoneMinder, 303
Zone transfer, DNS, 128

www.allitebooks.com