

The most comprehensive guide to Incident Response ever published."

Marc J. Zwillinger, former trial attorney with the United States Department of Justice Computer Crime & Intellectual Property

INCIDENT RESPONSE

INVESTIGATING COMPUTER CRIME

Kevin Mandia & Chris Prosise

with contributing author Matt Pepe

Recognized security, forensics, and incident response experts

CASE NO. 21567

LEVEL ONE SECURITY CLEARANCE ONLY

HACKING CASE 21567 UNDER INVESTIGATION

INCIDENT RESPONSE

INVESTIGATING
COMPUTER CRIME

Learn secrets and strategies for recovering from computer crime incidents

Respond to security breaches and hacker attacks the right way with help from this insightful and practical guide. You'll get details on the entire computer forensic process and learn the importance of following specific procedures immediately after a computer crime incident occurs. Investigate various software including UNIX, Windows NT, Windows 2000, and application servers. Packed with technical examples and loads of how-to scenarios, this book will show you how to recognize unauthorized access, uncover unusual or hidden files, and monitor Web traffic. Detailed, authoritative, and up to date—**Incident Response** is the only book you need.

- Plan and prepare for all stages of an investigation—including detection, initial response, management interaction, and more
- Investigate Web server attacks, DNS attacks and router attacks
- Learn the importance of evidence handling and storage
- Perform a "trap and trace" and learn network protocols
- Monitor network traffic and detect illicit servers and covert channels

ABOUT THE AUTHORS:

Kevin Mandia is Director of Computer Forensics at Foundstone and is a well-recognized forensics and incident response expert. He has trained over 400 FBI agents as well as personnel from the State Department, the CIA, NASA, the Royal Canadian Mounted Police, the Air Force, and other government agencies. He has been hand picked to work on many of the most complex international intrusion cases, and currently assists law enforcement on several high-profile cases. He can be reached at kevin@incidentresponsebook.com.

Chris Prosis is Vice President of Professional Services at Foundstone, where he provides information system security consulting services. Chris has extensive experience in attack and penetration testing and incident response. He began his career in information security as an Air Force officer at the Information Warfare Center. He is a frequent speaker and author on computer security and incident response, and is a regular contributor to "Security Issues," a CNET column (builder.cnet.com). He can be reached at chris@incidentresponsebook.com.

OSBORNE 

REQUIRED READING for the Information Age

A Division of The McGraw-Hill Companies

\$39.99 USA

www.osborne.com

NETWORKING/SECURITY



ISBN 0-07-213182-9



INCIDENT RESPONSE: INVESTIGATING COMPUTER CRIME

CHRIS PROSISE
KEVIN MANDIA

Osborne/McGraw-Hill
New York Chicago San Francisco
Lisbon London Madrid Mexico City
Milan New Delhi San Juan
Seoul Singapore Sydney Toronto

Osborne/McGraw-Hill
2600 Tenth Street
Berkeley, California 94710
U.S.A.

To arrange bulk purchase discounts for sales promotions, premiums, or fund-raisers, please contact Osborne/McGraw-Hill at the above address. For information on translations or book distributors outside the U.S.A., please see the International Contact Information page immediately following the index of this book.

Incident Response: Investigating Computer Crime

Copyright © 2001 by The McGraw-Hill Companies. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

1234567890 CUS CUS 01987654321
ISBN 0-07-213182-9

Publisher

Brandon A. Nordin

Vice President & Associate Publisher

Scott Rogers

Senior Acquisitions Editor

Jane K. Brownlow

Senior Project Editors

Carolyn Welch

Lisa Theobald

Acquisitions Coordinators

Ross Doll

Emma Acker

Technical Editors

Clinton Mugge

Mike Shema

Development Editors

Mark Cierzniak

Marilyn Smith

Copy Editor

Marilyn Smith

Proofreader

Pat Mannion

Indexer

Jack Lewis

Computer Designers

Dick Schwartz

Kelly Stanton-Scott

Elizabeth Jang

Illustrators

Lyssa Sieben-Wald

Michael Mueller

Series Design

Dick Schwartz

Peter F. Hancik

Cover Design

Dodie Shoemaker

This book was composed with Corel VENTURA™ Publisher.

Information has been obtained by Osborne/McGraw-Hill from sources believed to be reliable. However, because of the possibility of human or mechanical error by our sources, Osborne/McGraw-Hill, or others, Osborne/McGraw-Hill does not guarantee the accuracy, adequacy, or completeness of any information and is not responsible for any errors or omissions or the results obtained from use of such information.

About the Authors

Kevin Mandia

Kevin is the Director of Computer Forensics at Foundstone, Inc., an Internet security firm. Kevin is a well-recognized forensics and incident response expert. As a special agent, consultant, and instructor, Kevin has amassed a wealth of experience and expertise.

Kevin has a long history in teaching incident response, having developed a two-week computer intrusion response course and an advanced one-week network investigations course specifically designed for the FBI. Kevin taught at Quantico for over a year, and nearly 340 FBI agents specializing in computer intrusion cases have attended his course. The content of the course was tailored to meet the special needs of law enforcement, intelligence officers, and individuals who must understand the way computer networks operate and the methods hackers use to exploit networks.

Kevin has also provided computer intrusion training courses to other customers, including the State Department, the CIA, NASA, Prudential, SIAC, and the Air Force.

Kevin has assisted the FBI's National Infrastructure Protection Center, the Air Force Office of Special Investigations, corporate entities, and state law enforcement with investigative support. He has written court orders, affidavits, and developed specialized software to electronically track and catch computer hackers.

A frequent lecturer, Kevin speaks at dozens of conferences and events each year. He holds a B.S. in computer science from Lafayette College and an M.S. in forensic science from George Washington University, and is a reserve officer with the Air Force Office of Special Investigations.

Chris Prorise

Chris is Vice President of Professional Services at Foundstone, where he leads computer security consulting and training services. Chris has an extensive background in attack and penetration testing and incident response. He has led government and commercial security teams on missions worldwide, from sensitive incident response missions on top secret government networks to comprehensive security assessments of some of the world's largest corporations.

A co-founder of Foundstone, Chris has developed and taught courses for government and commercial firms in incident response, hacking, and network security. Chris has significant tool development experience, having developed automated scanning tools and real time intrusion detection and denial software for the Air Force, where he began his career in information security as an active duty officer at the Air Force Information Warfare Center.

Chris has been a featured speaker at conferences such as Networld Interop, SC Magazine's Securing the E-Business, and the Forum of Incident Response and Security Teams (FIRST). He writes frequently for magazines and has a regular security column—"Security Issues"—on CNET at <http://builder.cnet.com>.

Chris holds a B.S. in electrical engineering from Duke University and is a Certified Information Systems Security Professional (CISSP).

About the Contributing Author

Matt Pepe

Matt Pepe is one of the nation's most experienced incident response professionals, having performed forensic analysis in over 100 federal investigations for the Air Force Office of Special Investigations, FBI, and other government agencies. He is also a highly successful information security consultant, regularly leading network assessments and attack and penetration engagements. Matt Pepe can be reached at matt@incidentresponsebook.com.

About the Technical Editors

Clinton Mugge

Clinton Mugge has spent the past seven years dealing with intrusions and penetration testing in the IT security realm. He has served as a counter-intelligence agent performing IR and computer investigations involving classified and unclassified government networks. In the corporate world Mr Mugge has responded to incidents involving Fortune 500 companies and has developed comprehensive incident response programs. Mr. Mugge has spoken at various conferences and has served on forensics and incident response panels.

Mike Shema

Mike Shema is a principal consultant with Foundstone, where he co-developed and teaches incident response and security classes. He has performed dozens of security penetrations and incident response engagements for financial, IT, and government clients. A proficient coder, he has discovered zero-day exploits and pioneered Web-application testing methodologies. Mr. Shema holds a B.S. in electrical engineering from Penn State.

AT A GLANCE

Part I Learning the Ropes

- ▼ 1 Insiders and Outsiders: A Case Study 3
- ▼ 2 Introduction to Incident Response 15
- ▼ 3 Preparing for Incident Response 33

Part II Putting on the Gloves

- ▼ 4 Investigative Guidelines 75
- ▼ 5 The Computer Forensic Process 87
- ▼ 6 Learning Network Protocols and Performing
a Trap and Trace 131
- ▼ 7 Performing Network Surveillance 155
- ▼ 8 Advanced Network Surveillance 197

Part III Getting into the Ring: Investigating Systems

▼ 9	Initial Response to Windows NT/2000	225
▼ 10	Investigating Windows NT/2000	253
▼ 11	Initial Response to UNIX Systems	301
▼ 12	Investigating UNIX	325

Part IV Investigating Nonplatform-Specific Technology

▼ 13	Investigating Routers	349
▼ 14	Investigating Web Attacks	369
▼ 15	Investigating Application Servers	381
▼ 16	Investigating Hacker Tools	399

Part V Appendixes

▼ A	Establishing Identity in Cyberspace	429
▼ B	Information Security Policies and Acceptable Use Policies	461
▼ C	Computer Crime Statutes	465
▼ D	Response Organizations	469
▼	Index	473

CONTENTS

Acknowledgments	xxiii
Foreword	xxv
Introduction	xxvii

Part I

Learning the Ropes

▼ 1	Insiders and Outsiders: A Case Study	3
	Deterring Attacks	4
	Real Insiders	5
	The Information Obtained Pursuant to a 18 U.S.C.	
	§ 2703 Court Order	12
	So What?	14
▼ 2	Introduction to Incident Response	15
	Goals of Incident Response	16
	Incident Response Methodology	16
	Pre-incident Preparation	17

Detection of Incidents	18
Initial Response	20
Response Strategy Formulation	21
Deciding on Your Response Strategy	21
Presenting Response Strategy Options to Management	22
Forensic Duplication	24
Investigation	25
Security Measure Implementation	26
Network Monitoring	27
Deciding Where and How to Monitor	28
Deciding What to Monitor	28
Recovery	29
Considering What Was Compromised	29
Choosing a Recovery Strategy	30
Reporting	31
So What?	32
▼ 3 Preparing for Incident Response	33
Identifying Your Vital Assets	34
Preparing Individual Hosts	35
Recording Cryptographic Checksums of Critical Files	36
Increasing or Enabling Secure Audit Logging	39
Building Up Your Host's Defenses	44
Backing Up Critical Data	46
Educating Your Users about Host-Based Security	47
Preparing a Network	48
Installing Firewalls and Intrusion Detection Systems	49
Using Access Control Lists on Your Routers	49
Creating a Network Topology Conducive to Monitoring	50
Encrypting Network Traffic	51
Requiring Authentication	52
Establishing Appropriate Policies and Procedures	52
Determining Your Response Stance	53
Understanding How Policies Can Aid Investigative Steps	56

18	Developing Acceptable Use Policies (AUPs)	62
20	Designing AUPs	64
21	Developing Incident Response Procedures	65
21	Creating a Response Toolkit	66
22	The Response Hardware	66
24	The Response Software	67
25	The Networking Monitoring Platform	68
26	Establishing an Incident Response Team	68
27	Deciding on the Team's Mission	69
28	Getting the Team Together	70
28	Incident Response Training and Professional	
29	Organizations	70
29	So What?	72

Part II**Putting on the Gloves**

33	▼ 4 Investigative Guidelines	75
34	Conducting an Initial Assessment	76
35	Asking Questions during Incident Notification	77
36	Incident Notification Checklist	77
39	Examining Network Topology	79
39	Verifying Policies	79
44	Investigating the Incident	80
46	Conducting Personnel Interviews	80
47	Taking Hands-On Actions	82
48	Formulating Response Strategy	83
49	Determining the Appropriate Type of Response	83
49	Determining the Type of Attack	84
50	Classifying the Victim System	85
51	Considering Other Influences	85
52	Obtaining Management Approval	86
52	So What?	86
53	▼ 5 The Computer Forensic Process	87
56	Learning to Handle Evidence	89
	Common Mistakes in Evidence Handling	89
	The Best Evidence Rule	90

The Chain of Custody	92
Performing an Initial Response	94
Volatile Data	95
"Live" System Review	96
Performing Forensic Duplication	98
Forensic Duplication Approaches	98
Verifying Low-Level System Configuration	100
Forensic Duplication Tools	102
Using Safeback	103
Creating a DOS Controlled Boot Floppy	104
Creating the Forensic Duplication with Safeback	107
Using UNIX Utilities for Forensic Duplication	112
Creating a UNIX-Controlled Boot Disk	112
Creating the Forensic Image with DD	113
Using EnCase	114
Creating Evidence Files with EnCase	116
Using EnCase to Preview Evidence Drives	118
Performing Forensic Analysis	119
Conducting Physical Analysis	120
Logical Analysis	121
Understanding Where Evidence Resides	122
So What?	130
▼ 6 Learning Network Protocols and Performing a Trap and Trace	131
Understanding TCP/IP	132
Encapsulation	132
The IP Header	136
The TCP Header	141
The UDP Header	145
Using Sniffers	147
Performing a Trap and Trace	149
Creating Output Files	152
So What?	154
▼ 7 Performing Network Surveillance	155
Why Perform Network Surveillance?	156
Network-Based Evidence	157

92	Maintaining the Chain of Custody	158
94	Network Forensics	159
95	The Challenges of Network Forensics	160
96	Setting up Your System	161
98	Determine Goals	162
98	Check with Legal Counsel	163
100	Choose Appropriate Hardware	163
102	Choose Appropriate Software	164
103	Location and Security of the Monitor	168
104	Performing Surveillance	169
107	Monitoring Telnet	169
112	Monitoring the File Transfer Protocol	179
112	Monitoring Web Traffic	188
113	Interpreting a Network Attack	190
114	So What?	195
116		
118	▼ 8 Advanced Network Surveillance	197
119	The Elite Attacker's Goals	198
120	Activities That Usually Are Not Monitored	199
121	Activities That Are Difficult to Detect	200
122	Activities That Are Difficult to Play Back	200
130	Attacks That Are Difficult to Trace Back to the Source IP Address	201
131	Making Evidence Collection as Difficult as Possible	201
132	Maintaining Plausible Deniability	202
132	ICMP Covert Channeling	203
136	Examining Ping Traffic	204
141	Recognizing Loki Covert Channeling	207
145	Recognizing the Next Generation of ICMP Covert Channeling	212
147	Stateless TCP Covert Channeling	214
149	Examining a Stateless TCP Session	215
152	Recognizing Stateless TCP Covert Channeling	215
154	HTTP Covert Channeling	217
155	Detecting Illicit Servers	221
156	So What?	222
157		

Part III

Getting into the Ring: Investigating Systems

▼ 9	Initial Response to Windows NT/2000	225
	Creating a Response Toolkit	226
	Toolkit Labels	227
	Toolkit Contents	227
	Storing Information Obtained During the Initial Response	229
	Obtaining Volatile Data Prior to Forensic Duplication	232
	Organizing and Documenting Your Investigation	233
	Executing a Trusted cmd.exe	234
	Determining Who Is Logged into the System	235
	Determining Open Ports and Listening Applications	236
	Listing All Running Processes	239
	Listing Current and Recent Connections	240
	Documenting the Commands Used During Initial Response	242
	Scripting your Initial Response	242
	Performing an In-Depth, Live Response	243
	Obtaining Event Logs During Live Response	244
	Reviewing the Registry During a Live Response	248
	Obtaining Modification, Creation, and Access Times of All Files	251
	Obtaining System Passwords	251
	Dumping System RAM	251
	Is Forensic Duplication Necessary?	252
	So What?	252
▼ 10	Investigating Windows NT/2000	253
	Where Evidence Resides on Windows NT/2000 Systems	254
	Setting Up a Forensic Workstation	255
	Reviewing Logical Files	256
	Handling Passwords	257
	Performing Initial Low-Level Analysis	259
	Conducting a Windows NT/2000 Investigation	259

225	Reviewing All Pertinent Logs	260
226	Performing Keyword Searches	267
227	Reviewing Relevant Files	268
227	Identifying Unauthorized User Accounts or Groups	286
229	Identifying Rogue Processes	286
232	Looking for Unusual or Hidden Files	287
233	Checking for Unauthorized Access Points	289
234	Examining Jobs Run by the Scheduler Service	292
235	Analyzing Trust Relationships	293
236	Reviewing the Security Identifiers (SIDs)	294
239	File Auditing and Theft of Information	295
240	Handling the Departing Employee	297
242	Reviewing Searches and Files Used	298
242	Conducting String Searches on Hard Drives	298
243	So What?	299
244	▼ 11 Initial Response to UNIX Systems	301
248	Creating a Response Toolkit	302
251	Storing Information Obtained During the Initial Response	304
251	Obtaining Volatile Data Prior to Forensic Duplication	304
252	Executing a Trusted Shell	306
252	Determining Who Is Logged on to the System	306
253	Determining the Running Processes	308
254	Detecting Loadable Kernel Module Rootkits	309
255	Determining Open Ports and Listening Applications	311
256	Reviewing the /Proc File System	316
257	Cleaning Up Your Tracks	320
259	Performing an In-Depth, Live Response	321
259	Obtaining Modification, Creation, and Access Times of All Files	321
	Obtaining the System Logs During Live Response	322
	Obtaining Important Configuration Files	323
	Dumping System RAM	324
	So What?	324

▼ 12 Investigating UNIX	325
Preparing to Review a Restored Image	326
Booting into the Native Operating System	327
Performing Initial Low-Level Analysis	327
Conducting a UNIX Investigation	328
Reviewing Pertinent Logs	328
Performing Keyword Searches	334
Reviewing Relevant Files	336
Identifying Unauthorized User Accounts or Groups	342
Identifying Rogue Processes	343
Checking for Unauthorized Access Points	344
Analyzing Trust Relationships	345
So What?	346

Part IV

Investigating Nonplatform-Specific Technology

▼ 13 Investigating Routers	349
Obtaining Volatile Data Prior to Powering Down	350
Establishing a Router Connection	351
Recording System Time	351
Determining Who Is Logged On	351
Determining the Router's Uptime	352
Determining Listening Sockets	353
Saving the Router Configuration	354
Reviewing the Routing Table	354
Checking Interface Configurations	356
Viewing the ARP Cache	357
Finding the Proof	357
Handling Direct Compromise Incidents	357
Handling Routing Table Manipulation Incidents	359
Handling Theft of Information Incidents	360
Handling Denial-of-Service Attacks	360
Using Routers as Response Tools	362
Understanding Access Control Lists	362
Monitoring with Routers	365

325	Responding to DDoS Attacks	365
326	So What?	367
327	▼ 14 Investigating Web Attacks	369
327	Before You Power Down	370
328	Finding the Proof	370
328	Investigating Log Files	371
334	Investigating Web Site Defacement	376
336	Investigating Application-Level Attacks	378
342	Determining the Source of Attacks	379
343	So What?	380
344	▼ 15 Investigating Application Servers	381
345	Investigating Domain Name Server Incidents	382
346	Handling Direct Attacks	382
	Investigating Cache Corruption	384
	Investigating FTP Server Incidents	385
	Handling Direct Compromise Incidents	385
	Investigating File-Storage Abuse	388
349	Investigating RPC Service Incidents	390
350	Using Online Chat Program Records to Investigate	
351	Incidents	391
351	Handling Incidents Involving Microsoft Office	392
352	Finding Clues in Office Documents	392
353	Decrypting Office Documents	393
354	Determining the Source of Application Attacks	394
354	Recovering Compromised Application Servers	396
356	So What?	397
357	▼ 16 Investigating Hacker Tools	399
357	How Files Are Compiled	400
359	Statically Linked Programs	400
360	Dynamically Linked Programs	401
360	Programs Compiled with Debug Options	401
362	Stripped Programs	403
362	Programs Packed with UPX	403
365	Static Analysis of a Hacker Tool	404

Determining the Type of File	405
Reviewing the ASCII and UNICODE Strings	406
Performing Online Research	408
Performing Source Code Review	409
Dynamic Analysis of a Hacker Tool	409
Creating the Sandbox Environment	410
Dynamic Analysis on a UNIX System	411
Dynamic Analysis on a Windows System	421
So What?	425

Part V

Appendixes

▼ A Establishing Identity in Cyberspace	429
Investigating IP Addresses	430
Using nslookup	431
Using Traceroute or Tracert	433
Using the Whois Database	435
Investigating Dynamic IP Addresses	439
Investigating MAC Addresses	446
Tracing E-mail	449
Tracing a Fakemail	451
Tracing Back Web-based E-mail	453
Investigating E-mail Addresses, Nicknames, Usernames, and Hostnames	457
Piercing Anonymity through Legal Channels	458
Filing John Doe Lawsuits	458
Reporting an Incident to Law Enforcement	458
▼ B Information Security Policies and Acceptable Use Policies	461
Information Security Policy Areas	462
Acceptable Use Policies	463
Sample Acceptable Use Policy	463
▼ C Computer Crime Statutes	465
Federal Computer Intrusion Laws	466
Federal Intellectual Property Laws	466

Copyright Offenses	466
Copyright Management Offenses	467
Bootlegging Offenses	467
Trademark Offenses	467
Trade Secret Offenses	467
Offenses Relating to the Integrity of IP Systems	468
Offenses Relating to the Misuse of Dissemination Systems	468
Commerce and Trade Laws	468
▼ D Response Organizations	469
▼ Index	473

Links are used to associate a desktop shortcut or a Start menu item with an application or a document. Manually removing applications or documents does not remove the links that were created for them. Users may delete files but forget to delete the desktop icon on the system. The NTRK tool `chklnks.exe` is excellent for unearthing files that were once installed but now are nowhere to be found. As shown in Figure 10-9, `chklnks` finds dead (broken) links.

Links are also important when considering network connections and shortcuts. The average users have desktop shortcuts for their ISP dial-up connections and other network connections. Check out the user's `%systemroot%\Profiles\<user>\Desktop` directory and review all the links (*.lnk) for that user's desktop applications.

Reviewing Web Browser Files

Employees need access to the Internet at work, but many companies do not want their employees spending the majority of their work hours shopping, surfing, trading stocks, chatting, or downloading pornography on company systems. These activities require the use of Web browsers. Web browsers such as Netscape and Internet Explorer maintain log files. Both browsers record browsing history and track sites that were recently visited. They also maintain a cache that contains a certain amount of the actual files and Web pages recently viewed.

Viewing Netscape and Internet Explorer History Files The Netscape history file, `netscape.hst`, is normally located in the `\Program Files\Netscape\Users\<username>` directory. Netscape's `fat.db` file maintains an even longer history of browsing activity, and it is usually located in the `\Program Files\Netscape\Users\<username>\Cache` directory. Most people are aware of Netscape's history file, and individuals who wish to hide their cyber-shenanigans (using their browser to visit inappropriate sites) often erase this file or clear it via the Netscape Preferences settings. However, the `fat.db` file is often overlooked and is an excellent source for tracing browser use. For the initial response, you can simply use the `about:cache` URL to review the contents of the `fat.db`.

Internet Explorer maintains its temporary Internet files in the `\WINNT\Profiles\<UserId>\Local Settings\Temporary Internet Files\Content XX` directory, in the `index.dat` file. The actual HTML and files are stored in the Internet Explorer cache files, usually found in the `\WINNT\Temporary Internet Files` directory. Windows 2000 stores these files in a different directory, but they contain similar information. Windows 2000 maintains the Web browser cache in `\Documents and Settings\<User Account>\Local Settings\Temporary Internet Files`. The `index.dat` file in Windows 2000 that maps cached html pages to actual dates, times, and specific URLs is located in the `\Documents and Settings\<User Account>\Application Data\Microsoft\Internet Explorer\UserData` directory.

Netscape's fat.db and netscape.hst files and Internet Explorer's index.dat file are binary files. Therefore, you must use a special utility to view them. The Internet Explorer History Viewer, a tool discussed earlier in the "Reviewing Files in the Recycle Bin" section of this chapter, allows you to view most of the binary files maintained by both Netscape (fat.db and netscape.hst) and Internet Explorer (index.dat). The Internet Explorer History Viewer will parse and print the history of URLs visited using Internet Explorer versions 3.x, 4.x, and 5.x, as shown in Figure 10-10.

EnCase is another tool that provides phenomenal features that automate the process of outlining the Web activity of a system. Figure 10-11 shows the EnCase script that reports on Netscape and Internet Explorer Web use.

Reviewing Dialup Networking Another way to determine the browsing activities of a user is to review the Dial Up Networking (DUN) settings on the system. DUN has a feature called dial-on-demand, which many applications try to set automatically as a default. Dial-on-demand, or autodial, allows Windows systems to initiate a connection automatically whenever an application requires the use of the Internet.

Windows NT/2000 maintains a listing of the IP addresses that have been connected to via the autodial feature. To view the autodial database, use the following command:

```
rasautou -s
```

Figure 10-12 displays the output of the attempted connections made by the RAS dialer.

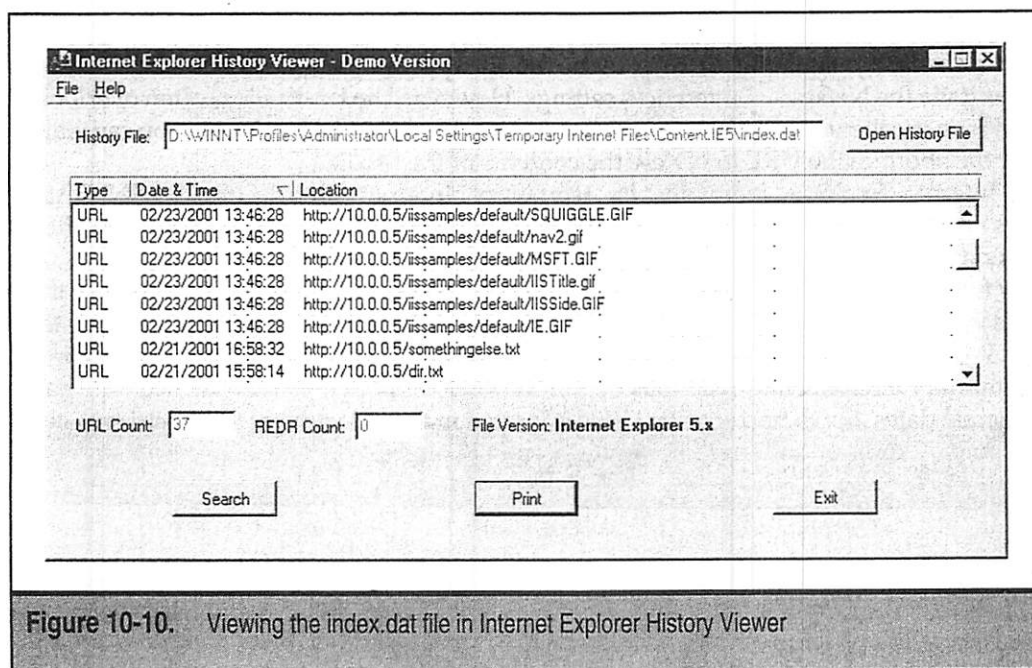


Figure 10-10. Viewing the index.dat file in Internet Explorer History Viewer

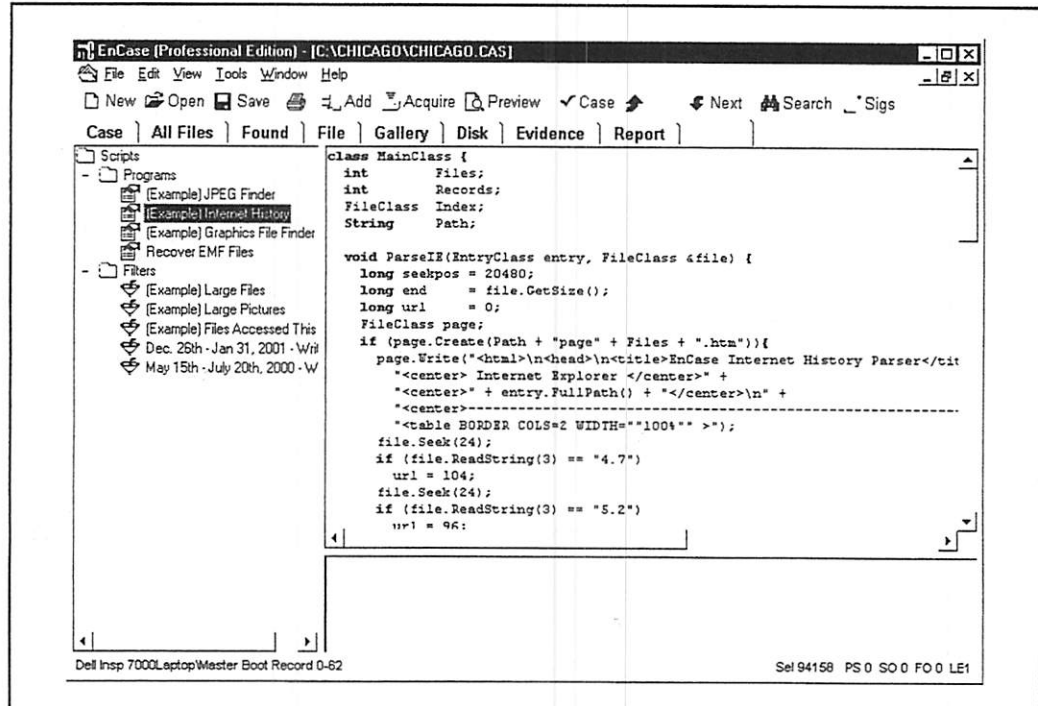


Figure 10-11. Using EnCase scripts to determine Web browsing activity

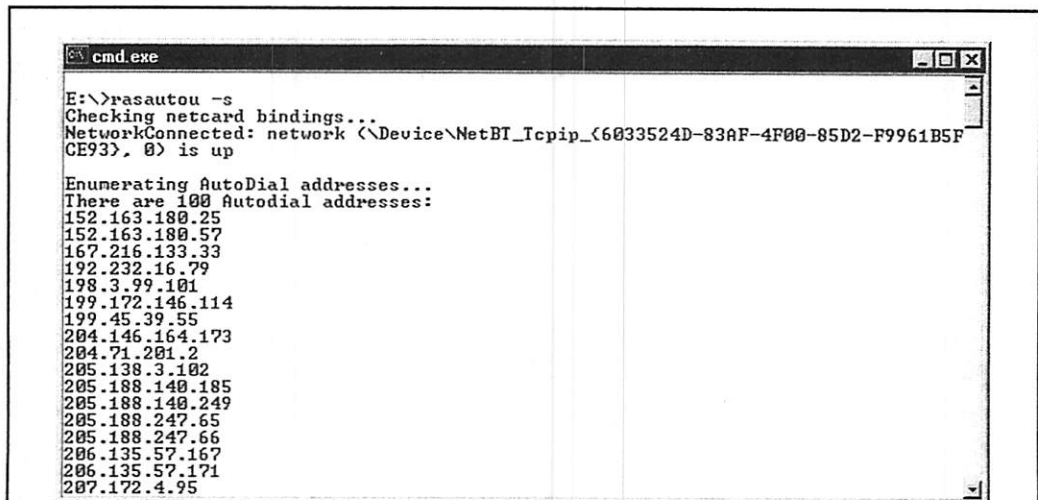
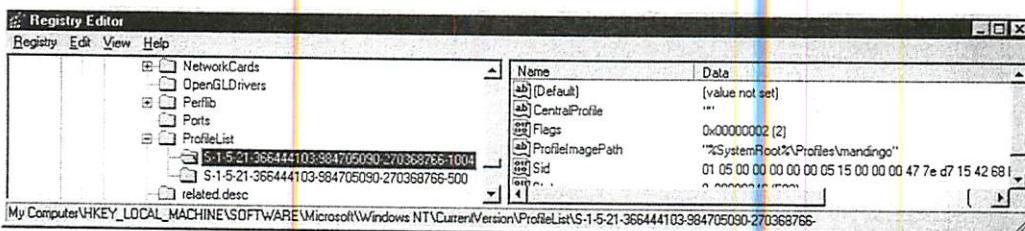


Figure 10-12. Reviewing attempted DUN connections using the rasautou command

Identifying Unauthorized User Accounts or Groups

A common ploy by evildoers is to start rogue accounts on a system or to elevate their privileges to an unauthorized level, where they can get to data that they should not be able to access. There are several ways to audit user accounts and user groups on a live system:

- ▼ Look in the User Manager for unauthorized user accounts (during a live system response).
- Use `usrstat` from the NTRK to view all domain accounts on a domain controller, looking for suspicious entries.
- Examine the Security log using Event Viewer, filtering for event ID 624 (addition of a new account), 626 (user account enabled), 636 (changing an account group), and 642 (user account changed).
- Check the `\WINNT\Profiles` directories on the system. If the user account exists, but there is no corresponding `\WINNT\Profiles\<useraccount>` directory, that user account has not been used to log in to the system yet. If that directory does exist, but the user account is no longer listed in the User Manager or Registry (at `HKLM\SAM\SAM\Domains\Account\Users\Names`), that user ID did exist at one time but no longer exists.
- ▲ Review the SIDs in the Registry, under `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\ProfileList`. When a user account is deleted, the corresponding Profile directory entry is not deleted, and the corresponding SID will remain in the Registry, as shown in the following illustration (in which the Registry shows a SID value exists for user ID mandingo, which no longer exists as a valid user account on the system). This allows you to trace which user IDs have been deleted over the course of a system's life.



Identifying Rogue Processes

Identifying rogue processes is much simpler when reviewing a live system. Since most rogue processes listen for network connections or sniff the network for clear-text user IDs and passwords, these processes are easier to find when they are executing. As you learned in Chapter 9, the `pslist` command will list the name of the running process, `listdlls`

NOTE

EnC

Other
changing
tem files
can fool